VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ
ÚSTAV BIOMEDICÍNSKÉHO INŽENÝRSTVÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION
DEPARTMENT OF BIOMEDICAL ENGINEERING

PROCESSING OF IMAGE SEQUENCES FROM FUNDUS CAMERA

ZPRACOVÁNÍ OBRAZOVÝCH SEKVENCÍ SÍTNICE Z FUNDUS KAMERY
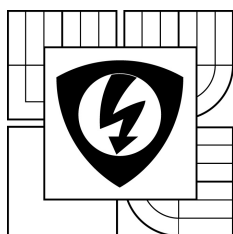
DIPLOMOVÁ PRÁCE
MASTER'S THESIS

AUTOR PRÁCE   Bc. FILIP KLIMEŠ
AUTHOR

VEDOUCÍ PRÁCE   doc. Ing. RADIM KOLÁŘ, Ph.D.
SUPERVISOR

BRNO 2015

BRNO UNIVERSITY
OF TECHNOLOGY

Faculty of Electrical Engineering and Communication

Department of Biomedical Engineering

Diploma thesis

Master's study field
Biomedical Engineering and Bioinformatics

Student: Bc. Filip Klimeš   *ID:* 137001
Year of study: 2   *Academic year:* 2014/15

TITLE OF THESIS:

Processing of image sequences from fundus camera

INSTRUCTION:

1) Study of the properties of image sequences from experimental retinal camera and the methodology for the frames registration. 2) Study of the methods for image quality assessment; design of the methodology for quality assessment of individual frames in retinal sequences. Consider also the possible artefacts in these sequences. 3) Implementation of the method, test on real sequences and evaluation. 4) Use this method in approach for the frame registration in order to increase the robustness of this method. 6) Modification of image assessment method for quality evaluation of registered sequences. 7) Evaluation of this approach and discussion of the results.

REFERENCE:

[1] KOLAR, Radim, Bernhard HOEHER, Jan ODSTRCILIK, Bernhard SCHMAUSS a Jiri JAN. Registration of Image Sequences from Experimental Low-Cost Fundus Camera. In: Proceedings of Workshop on Biomedical Image Registration, Lecture Notes in Computer Science 8454, 2014, s. 174-183.
[2] NIXON, M. a A. AGUADO. Feature Extraction and Image Processing, Elsevier, 2005

Assigment deadline: 9. 2. 2015   *Submission deadline:* 31.7.2015

Head of thesis: doc. Ing. Radim Kolář, Ph.D.
Consultant:

prof. Ing. Ivo Provazník, Ph.D.
Subject Council chairman

KLIMEŠ, F. *Processing of image sequences from fundus camera*. Master's thesis. Brno: Brno University of Technology. Faculty of Electrical Engineering and Communication. Department of Biomedical Engineering, 2015. 88 p. Thesis supervisor: doc. Ing. Radim Kolář, Ph.D.

PROHLÁŠENÍ

Prohlašuji, že svoji diplomovou práci na téma Zpracování obrazových sekvencí sítnice z fundus kamery jsem vypracoval samostatně pod vedením vedoucího diplomové práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené diplomové práce dále prohlašuji, že v souvislosti s vytvořením tohoto projektu jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a jsem si plně vědom následků porušení ustanovení § 11 a následujících zákona č. 121/2000 Sb., o právu autorském, včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.

V Brně dne ....

(Filip Klimeš)

PODĚKOVÁNÍ

Děkuji vedoucímu diplomové práce doc. Ing. Radimu Kolářovi, Ph.D a Dr. Ing. Ralfu-Peteru Tornowovi za cenné rady, připomínky a inspiraci při zpracování mé diplomové práce. Dále bych chtěl poděkovat svým nejbližším za podporu nejen při vypracování této práce, ale i po celou dobu mého studia.

V Brně dne ....

(Filip Klimeš)

ABSTRAKT

Cílem mé diplomové práce bylo navrhnout metodu analýzy retinálních sekvencí, která bude hodnotit kvalitu jednotlivých snímků. V teoretické části se také zabývám vlastnostmi retinálních sekvencí a způsobem registrace snímků z fundus kamery. V praktické části je implementována metoda hodnocení kvality snímků, která je otestována na reálných retinálních sekvencích a vyhodnocena její úspěšnost. Práce hodnotí i vliv této metody na registraci retinálních snímků.

KLÍČOVÁ SLOVA: oko, sítnice, fundus kamera, zpracování retinálních snímků, registrace, hodnocení kvality snímku, analýza hlavních komponent, diskriminační analýza, metoda podpůrných vektorů, metoda k-nejbližších sousedů, normalizovaná vzájemná korelace, suma čtvercových rozdílů

ABSTRACT

The aim of my master's thesis was to propose a method of retinal sequence analysis which will evaluate the quality of each frame. In the theoretical part, I will also deal with the properties of retinal sequences and the way of registering the images of the fundus camera. In the practical part the method of evaluating image quality is implemented. This algorithm is tested on real retinal sequences and its success is assessed. This work also evaluates the impact of proposed method on the registration of retinal images.

KEY WORDS: eye, retina, fundus camera, processing of retinal image, registration, image quality assessment, principal component analysis, discriminant function analysis, support vector machine, k-nearest neighbour algorithm, normalized cross-correlation, sum of squared differences

CONTENT

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS AND ABBREVIATIONS

BIQI	Blind Image Quality Index
BLIINDS-II	BLind Image Notator using DCT Statistics
BRISQUE	Blind/Referenceless Image Spatial QUality Evaluator
DIIVINE	Distortion Identification-based Image INtegrity and Verity Evaluation
DA	Discriminant function Analysis
DCT	Discrete Cosine Transform
FR IQA	Full-Reference Image Quality Assessment
GGD	Generalized Gaussian Distribution
GRNN	General Regression Neural Network
HVS	Human Visual System
IQA	Image Quality Assessment
kNN	k-Nearest Neighbour
LBIQ	Learning based Blind Image Quality measure
MSCN	Mean Subtracted Contrast Normalized coefficients
MSE	Mean Squared Error
MS-SSIM	Multi-Scale Structural SIMilarity
NR IQA	No-Reference Image Quality Assesment
PCA	Principal Component Analysis
PDF	Probability Density Function
PSNR	Peak Signal-to-Noise Ratio
ROI	Region Of Interest
RR IQA	Reduced-Reference Image Quality Index
SNR	Signal-to-Noise Ratio
SPC	Specificity
SSIM	Structural SIMilarity
SSD	Sum of Squared Differences
SVM	Support Vector Machine
TPR	Sensitivity
VIF	Visual Information Fidelity

1 INTRODUCTION

Retinal imaging has developed rapidly during 20th century and the photographs taken by *fundus camera* are widely used for detection of retinal diseases.

With the launch of advanced ophthalmic devices which can capture, store and display visual data there is a rising requirement on image quality. Recently the *image quality assessment (IQA)* has become an important issue in many applications such as image acquisition, restoration and enhancement. The aim of many scientists and also of this thesis is to invent an automatic metric that can provide an image score which will correspond to subjective judgment.

The most frequently used modality to collect information about the human eye is fundus imaging. Objective and quantitative assessment of acquired images is essential for automatic diagnosis. Basically the image quality can be evaluated in three different approaches:

- Reference image quality assessment
- Reduced-reference image quality assessment
- No-reference image quality assessment

See the *Chap. 4* for more detailed information about dissimilar image quality approaches.

In the last years Kohler et al. [17] suggested a first metric which is dealing with quality assessment of retinal colour images. The principle of proposed algorithm is to take the vessels of retina as guidance to determine an image quality score. They presented an improved no-reference image quality metric to quantify the amount of noise and blur in retinal fundus images. For reliable quality estimation, they employ the vessel tree, detected by the well known vesselness measure as guidance to determine a global quality score from local estimates in anisotropic patches. The proposed metric shows high agreement with the established full-reference metrics PSNR and SSIM [17].

Recently, no other method solving image quality judgment of retinal images has been introduced.

This master's thesis is in total separated into eight chapters. Second chapter deals with anatomical background, which is quite important to know in the case of ophthalmic data processing. Features of retinal sequences are discussed in detail in the third chapter. The core of this theoretical part is in *Chap. 4-6*. Fourth chapter is dedicated to image quality assessment methods and is divided into three subchapters which are discussing *full-reference image quality assessment*, *reduced-reference image quality assessment* and *no-reference image quality assessment*. In the fifth chapter a basics of registration process within *phase-correlation based alignment* and *blood vessels tracking* are introduced. Finally sixth chapter is devoted to proposal of method, which will successfully evaluate a quality of each retinal frame in captured sequence. The theoretical part is followed by *the Chap. 7*. This chapter is firstly dedicated to implementation of proposed algorithm. This task includes choosing the best set of features for

classification and selecting the best classifier for every group of sequences. Finally we discuss the influence of our algorithm on registration process. The success of registration is evaluated by means of *sum of squared differences algorithm* (*SSD*) and *normalized cross-correlation* (*NCC*). In each of these sub-chapters are the partial results evaluated. In the conclusion (*Chap.8*), we discuss contribution of this thesis, utilization and significance of achieved results.

2 ANATOMY OF RETINA

The retina is the inner layer of the eyeball and represents sensory organ of the eye. Its receptors collect light that passes through the lens and converts it into electrical impulses which extend further into the brain.

The retina is soft, semitransparent and consists of the following parts: pigment epithelium, photoreceptors (rods and cones), horizontal cells, amacrine cells, bipolar cells and ganglion cells [3].
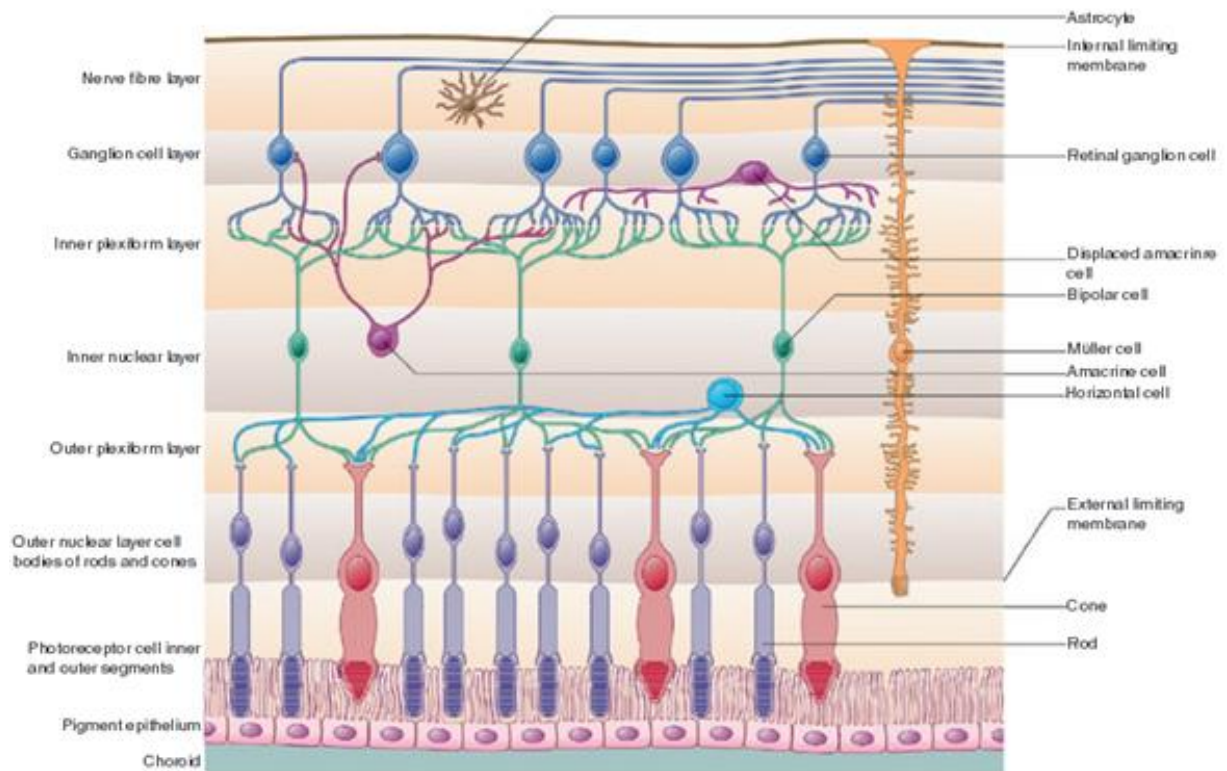
Fig. 1 - Retina - Adapted from [4].

Owing to the presence of a light-sensitive receptor protein named *rhodopsin* (visual purple) retina seems to have a purple tint. When is retina exposed to sunlight it becomes opaque and bleached. In the center of the posterior part of the retina and at a point in which the sense of vision is most perfect, is an oval yellowish area called the *macula lutea*. About 3 mm to the nasal side of the *macula lutae* is the entrance of the optic nerve (optic disk). This is the only part of retinal surface which is insensitive to light, and it is termed the *blind spot* [4].

3 FEATURES OF RETINAL IMAGES

The basis of this chapter is to obtain a certain awareness of the elementary properties of retinal images and their acquisition by fundus camera.

3.1 Fundamental features of obtained frames

When we look at the unfiltered image of retina (*Fig. 2*), we can clearly see an optic disc and vessels. It is obvious that all the nerve fibers come out directly from the optic disc, this feature could cause difficulties in some image analysis.

Fig. 2 - Frame no. 1 of 189 in a video-sequence.

After acquisition of these frames we are able to identify typical artefacts. In case of fast eye movements a blurred image (see *Fig. 3*) can occur. The image noise can be eliminated by using averaging filter and contrast limited adaptive histogram equalization [1, 10]. By these two steps we are able to increase the contrast of the blood vessels and equalize image illumination.

While no retinal image is seen we can assume that retina is out of focal plane (*Fig. 4*).

First objective of this thesis is to propose some metric, which will detect these *"bad"* frames automatically.
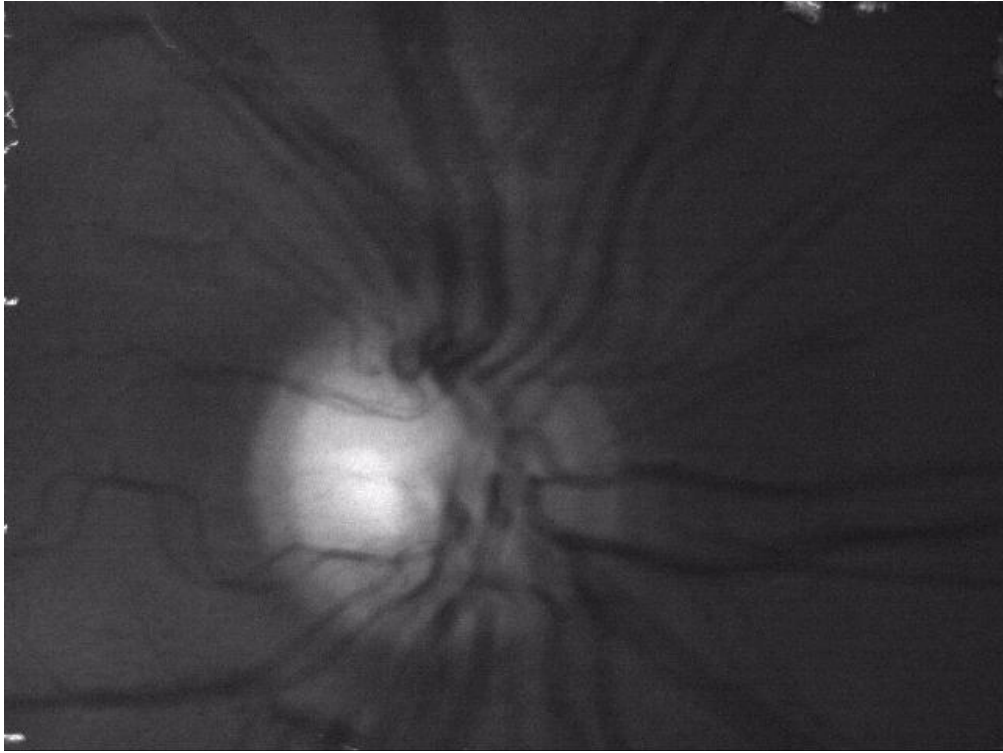
Fig. 3 - Example of blurred image.
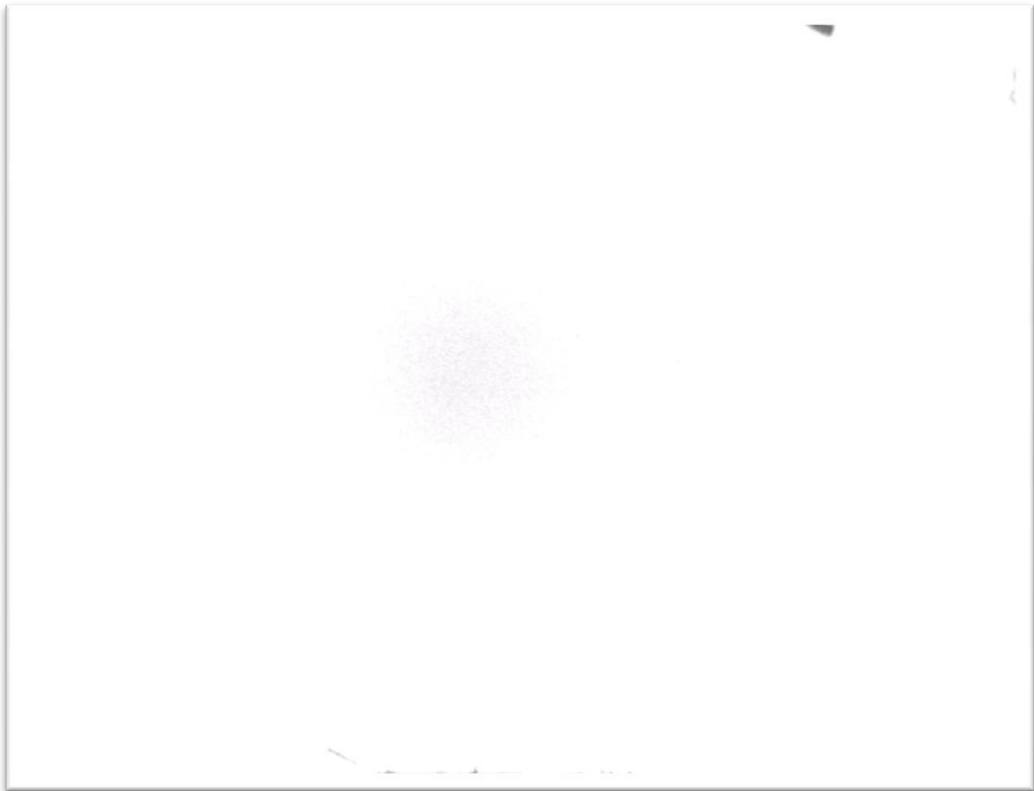
Fig. 4 - Example of image where retina is out of focal plane.

3.2 Description of acquisition parameters

A dataset of video-sequences has been acquired by experimental digital fundus camera which enables to take video sequences. Fundus camera is widely exploited in ophthalmology for the human retina assessment. Nevertheless, the fundus video imaging is not so often used. The construction of used fundus camera is currently in patent pending. The parameters of video-sequences acquisition are given in *Tab. 1*.

Tab. 1 - Acquisition parameters

Field of view (FOV) [°]	20 x 15
Frame rate [fps]	11 - 39
Count of frames in sequence [-]	189 - 964
Size of images [px]	640 x 480

Orange light ($\lambda = 575$ nm) was used for illumination of the retina and the data are stored in 8-bit grey-scale as standard uncompressed AVI format. An example of one frame in a sequence is shown in *Fig. 2* [9].

4 METHODS OF IMAGE QUALITY ASSESSMENT

The best way to assess the quality of image is subjective testing by a group of independent human observers. However this approach is extremely time-consuming and costly, the observers must be properly instructed and the conditions of measurement must be precise. Since we are unable to simulate *human visual system (HVS)*, we have to rely on objective quality metrics which is usually divided into three categories: *Full-reference quality metrics*, *Reduced-reference quality metrics* and *No-reference quality metrics* [5].

4.1 Full-reference image quality metrics (FR IQA)

The basic principle of full reference image quality metrics is to compare distorted images to their original. The original is considered as the best in terms of quality.

According to Krasula et al. [5] we can separate full reference image quality metrics into seven sections: *Pixel-based metrics, correlation measures, edge quality measures, HVS-based metrics, spectral distance measures, DCT-based metrics* and *metrics based on information fidelity*. The list of all metrics is shown in *Tab. 2* [5].

The substance of these metrics is different but in this chapter we will discuss only the three most interesting. For example the *pixel-based metrics* is searching for the differences between corresponding pixels. This process is happening regardless of the information in the image. *Pixel-based metrics* are quite simple however they do not correspond too much with subjective assessment. Other remarkable approaches are the *HVS based metrics*. These approaches are using psycho-visual models of *HVS* and have much better results than the previous metric. Unfortunately their handicap is that they can simulate just one aspect of *HVS* and fails in the others. The last one is fully different and is based on knowledge of information loss during the distortion process. This metrics is called *information fidelity metrics* [5].

Tab. 2 - Examples of full reference objective image quality metrics. Adapted and adjusted from [5].

Pixel-based metrics	MSE, SNR, PSNR, Universal Image Quality Index
Correlation measures	Correlation, Cross-Correlation, Czekanowski distance
Edge quality measures	Pratt Distance, Edge Stability
HVS-based metrics	Absolute HVS Error, HVS MSE, SSIM, MS-SSIM
Spectral distance measures	Spectral Magnitude Distortion, Spectral Phase Distortion, Weighted Spectral Phase-Magnitude Distortion, Block Spectral Magnitude Distortion, Block Spectral Phase Distortion, Block Weighted Spectral Phase-

	Magnitude Distortion
DCT-based metrics	DCT metric
Information fidelity metrics	VIF

4.2 Reduced-reference image quality metrics (RR IQA)

Reduced-reference IQA methods are predicting the quality degradation of an image. Limited information about the reference image is known in form of features. *Fig. 5* illustrates how an *RR IQA* system (method) may be constructed. The algorithm includes a feature extractor at the sender side and a feature extraction/quality analysis process at the receiver side. Original image is going through the distortion channel and the ancillary channel is used for the extracted image features or side information. After the information arrives to receiver side, the features extracted from the reference and distorted images are compared and then the image quality is evaluated [7].
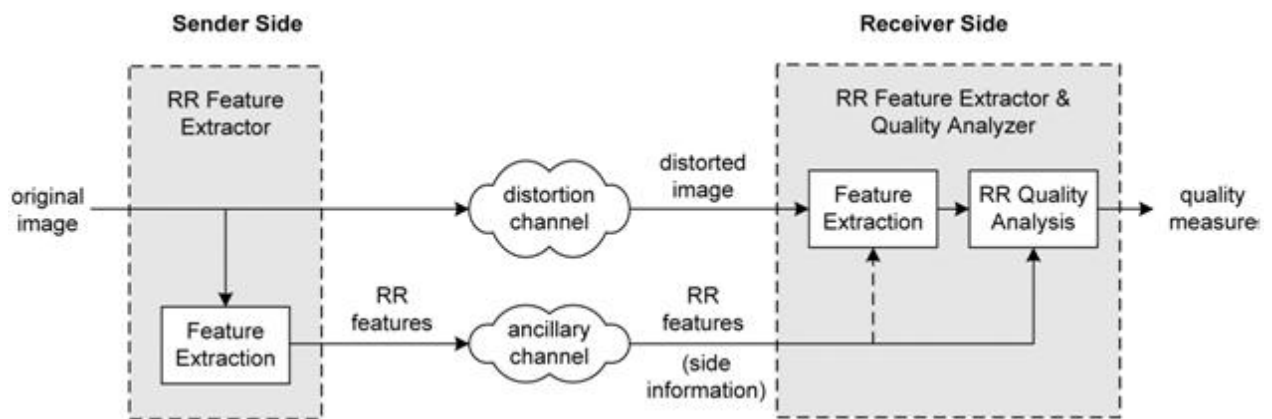
Fig. 5 - General framework of reduced-reference image quality metrics. Adapted from [7].

Reduced-reference image quality metrics provides useful and convenient tool in applications such as visual information communications over wired or wireless networks, where they can be employed to monitor image quality degradations or control the network streaming resources in the real time [7].

4.3 No-reference image quality metrics (IR IQA)

As the title suggests *no-reference algorithms* are dealing with *image quality assessment* without any reference image. Based on knowledge of distortion type *no-reference metrics* could be further classified into *distortion-specific* and *non-distortion-specific*. Currently the *distortion-specific* algorithms are more used and are dealing with distortion type, such as the white noise and blurring. Limitation of this metrics is also the assumption of above mentioned distortion type. On the other hand *non-distortion-specific* algorithms do not consider the prior knowledge of distortion type. Their output is quality score related to specific distortion. These methods often

extract features to train learning models and then these models evaluate the quality of tested image [8].

4.3.1 Distortion-specific IQA methods

Distortion specific methods assume that the distorting medium such as compression is known and they try to model these specific distortions. Modelling is followed by quality assessment. Three most common distortion categories are *JPEG compression*, *JPEG2000 compression* and *blurring* [15].

1) *JPEG compression*: The general principle of *JPEG compression image quality assessment* is to measure edge strength and relate this strength and possibly some measure of image activity to perceived quality. *JPEG IQA* algorithms include those that use a hermite transform based approach to model blurred edges, those that estimate first-order differences and activity in an image and those that use a threshold-based approach on computed gradients. Each of these approaches measures a blur, activity and computes perceptual quality [15].

2) *JPEG2000 compression*: For *JPEG2000* is typical *ringing artefacts*. They are often modelled by measuring edge-spread using an edge-detection based approach. Intensity of edge spread is highly related to quality. Other approaches include those that compute simple features in the spatial domain, or those that utilize natural scene statistics and assume that distortion will modify the dependency between a wavelet coefficient and its neighbours [15].

3) *Blurring*: *Blur IQA* methods also model edge spreads and relate these spreads to image quality. These approaches are very similar to the *JPEG2000 image quality assessment* algorithms. Edge strengths are quantified using a variety of techniques, including *DCT* coefficients, iterative thresholding of a gradient image or measuring the probability of blur detection.

Each of these distortion specific *image quality assessment* algorithms attempt to model indicators of quality for the distortion but should be unsuitable for use in a many cases [15].

4.3.2 Non-distortion-specific IQA methods

The *non-distortion-specific IQA* algorithms are quite difficult to develop, so that not a great number of scientist is working in this area. Nevertheless a few approaches are known nowadays.

For example, Tang et al. [11] developed an image quality measure (*LBIQ*) that combines numerous low-level image quality features (e.g. blur/noise estimation). This algorithm is able to capture variations of distortion type and highly correlate with perceptual image quality. Then Li et al. [12] proposed a general regression neural network based method (*GRNN*), which evaluate the quality of image quality by estimating a function relationship between features (e.g. the entropy of the distorted image) and subjective mean opinion scores using *GRNN*. Results show that this new method accords closely with human subjective judgment [12].

Recently, there were developed algorithms that are extracting different image features within application of training methods. Those methods are described below. *BRISQUE* [13] is *NR IQA* method which operates in the spatial domain. This model does not compute distortion specific features like blurring, but uses locally normalized luminance to quantify possible losses in the image which are caused by distortions. The advantage of this method is that no transform (e.g. wavelet, *DCT*) is required. Despite its simplicity, this algorithm has shown a much higher performance than the algorithms mentioned above [8, 13]. Other algorithm introduced by Moorthy and Bovik [14] is called *DIIVINE* (Distortion Identification-based Image Verity and INtegrity Evaluation). *DIIVINE* is based on a two-stage framework involving distortion identification followed by *distortion-specific quality assessment*. *DIIVINE* is capable of assessing the quality of a distorted image across multiple distortion categories because of its assumption that certain statistical properties will be modified in presence of distortion [8, 14]. *BIQI* [15], the blind image quality index, is a wavelet-based *NR IQA* method which extracts image features by modeling the wavelet subband coefficients based on *GGD* (generalized Gaussian distribution). For each image 18 features are extracted. These extracted features characterize the type of distortion which is present in the image. Then a classifier is used to classify elements of feature vector into five categories [8, 15]. The last described algorithm *BLIINDS-II* [16] measures image quality based on the discrete cosine transform (*DCT*) coefficients of image. This approach also relies on a Bayesian inference model that predicts image quality scores of extracted features which are indicative of perceptual quality [8, 16].

5 REGISTRATION METHODS

5.1 Main principle of registration

Image *registration* is the process of aligning two or more images of the same scene which can change during the acquisition. Firstly we have to set one image as the reference and then geometric transformations to the other images of sequence are applied, so that they align with the reference one. Misalignment of images can be caused by a variety of reasons. The most common reason is that the frames are captured under unequal conditions. Images can be misaligned as a result of sensor and lens distortions or differences among applied capture devices. Estimating the correct geometrical transformation parameters is a key to the image registration process. Inaccurate *registration* can lead to artefacts, loss of resolution or misinterpretation of processed image. Geometrical transform may be divided into two groups: *rigid transforms* and *flexible transforms*. *Rigid* transformations are used when no geometrical distortion present is. We define reference image *I*, its spatial coordinates $r = (x, y)$ and the spatial coordinates of corrected image *I'* are defined as $r' = (x', y')$. *The plain shift* is then determined by equation *5.1* [21].

$$r' = r + \Delta r, \tag{5.1}$$

where $\Delta r = [\Delta x, y\Delta]^T$ means changes of position between spatial coordinates of reference image *I* and transformed image *I'*. *Rotation* displacements are defined by the transform matrix *B* (*Equation 5.3*) and angle θ.

$$r' = B \cdot r. \tag{5.2}$$

Transform matrix *B* is then defined as follows:

$$B = \begin{bmatrix} cos\theta & -sin\theta \\ sin\theta & cos\theta \end{bmatrix}. \tag{5.3}$$

The whole *rigid* transformation, consisting of *plain shift* and *rotation*, can be expressed by single equation:

$$r' = B \cdot r + \Delta r. \tag{5.4}$$

Flexible transformation enables particular deformation of transformed image *I'* to cover image distortion. For such purposes a *plain scaling* is most common option. Equation of plain scaling is below.

$$r' = S \cdot r, \tag{5.5}$$

where *S* is scaling matrix (see *Equation 5.6*) and by changing values s_x and s_y we are able to modify the image proportions.

$$\mathbf{S} = \begin{bmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{5.6}$$

There is also the possibility of linking *flexible* with *rigid* transform that means all previous transformations are combined into one and the following equation is formed:

$$\boldsymbol{r'} = \boldsymbol{S} \cdot \boldsymbol{R} \cdot \boldsymbol{r}, \tag{5.7}$$

where the result of multiplication between *S* and *R* is:

$$\boldsymbol{S} \cdot \boldsymbol{R} = \begin{bmatrix} s_x \cdot cos\theta & -s_x \cdot sin\theta & s_x \cdot \Delta x \\ s_y \cdot sin\theta & s_y \cdot cos\theta & s_y \cdot \Delta y \\ 0 & 0 & 1 \end{bmatrix}. \tag{5.8}$$

This transformation is recognized as *scaling transformation*. Other transform related to *flexible* is *skewing* also known as image shearing. The transformation is defined as:

$$\boldsymbol{G} = \boldsymbol{G_x} \cdot \boldsymbol{G_y}, \boldsymbol{G_x} = \begin{bmatrix} 1 & g_{xy} & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \boldsymbol{G_y} = \begin{bmatrix} 1 & 0 & 0 \\ g_{yx} & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{5.9}$$

The last of linear *flexible* transform is *affine transform* which is combining *skewing* with above mentioned *scaling transformation*.

$$\boldsymbol{r'} = \boldsymbol{G} \cdot \boldsymbol{S} \cdot \boldsymbol{R} \cdot \boldsymbol{r} = \boldsymbol{A} \cdot \boldsymbol{r} \tag{5.10}$$

There are also non-linear methods of geometrical flexible transformation (e.g. projective transformations) however we will not focus on them [21].

Registration of images is often used as a preliminary step in other image processing applications (e.g. alignment of medical multimodal images).

In this chapter we will further deal with *registration* methods which are or were applied on retinal images. Majority of existing registration approaches utilize identification and extraction of features (*feature-based registration approaches*). These features are mainly derived from retinal vasculature and this extraction of features in distorted images represents quite difficult task. The most considerable advantage of *feature-based* methods is the robustness against changes in illumination during the acquisition. *Feature-based* algorithms for retinal image registration usually rely on the assumption that network of blood vessels is able to be extracted. Images acquired with a small overlap are often aligned using these *feature-based* algorithms, while for frames captured with greater overlaps are used *intensity-based approaches* (also known as *area-based approaches*) [18, 19].

The *area-based approaches* compare and match the intensity differences of an image pair and then apply an optimization technique to maximize the similarity between image pairs. The

optimum of similarity is reached when two frames are perfectly matched. As we mentioned before the *intensity-based approaches* are inconvenient in low overlapped areas. This problem is can be overcome by assigning a region of interest (*ROI*) in one or in both frames and then the similarity metric is used [18, 19].

In conclusion we can say that f*eature-based approaches* are more suitable for retinal image registration. A *hybrid approach* that effectively combines both *area-based* and *feature-based* approaches has also been proposed by Ronkova et. al [20]. In this paper iterative gradient method is supplemented as an additional method to the genetic algorithm.

Finally in [1] is described two way registration of retinal image sequence captured by experimental low-cost fundus camera. Before these two steps a pre-processing of each frame is needed due to dissimilar illumination and presence of noise. Pre-processing step contains using 3x3 median filter and contrast limited adaptive histogram equalization to remove noise respectively to emphasize a contrast of the blood vessels [1]. Registration process of this paper is discussed in *Chap. 5.2* and *5.3*.

5.2 Phase-Correlation based alignment

The first registration step introduced in [1] is based on *phase correlation*. It uses a fast frequency-domain approach to estimate an offset of two images. Consider two functions $f_1(x, y)$ and $f_2(x, y) = f_1(x - x_0, y - y_0)$, where x_0, y_0 is shift between their spatial coordinates. Cross-power spectrum is then given by:

$$\frac{F_2(u, v) \cdot F_1(u, v)^*}{|F_2(u, v) \cdot F_1(u, v)^*|} = e^{-j \cdot (u \cdot x_0 + v \cdot y_0)} \tag{5.11}$$

$F_1(u, v)$ and $F_2(u, v)$ are Fourier spectra of defined functions (f_1, f_2) and the asterisk sign (*) means complex conjugate. By applying an inverse Fourier transform on computed cross spectrum, we are able to determine spatial shift [1, 22]. Phase correlation is usually used for large shift compensation and is able to cope with spatial shift, rotation and scaling.

5.3 Image registration based on optical flow

Tracking of blood vessels is implemented by *Lucas-Kanade tracking*. This method has following assumptions:

a) The two images are captured during short time increment Δt and objects in images do not move or are not noticeably dislocated.
b) Natural scene is depicted in acquired images and the intensity changes, represented by shades of grey do not vary too much.

Let I_i and I_{i+1} be two consecutive images and $X_j = (x_j, y_j)$ is point in first image I_i. The principle of this method is to find appropriate point $X'_j = (x'_j, y'_j) = X_j + D_j$ in second image I_{i+1}.

D_j is a vector with two spatial coordinates *(dx_j, dy_j)* and is termed as optical flow in point X_j. The task is to minimize the function below by utilizing vector D_j.

$$\epsilon(D_j) = \sum_{x = x_j - w_{x_j}}^{x_j + w_{x_j}} \sum_{y = y_j - w_{y_j}}^{y_j + w_{y_j}} \left[I_i(x, y) - I_{i+1}\left(x + d_{x_j}, y + d_{y_j} \right) \right]^2, \qquad (5.12)$$

where index $j = 1,2 \ldots N$. N is total number of points in image I_i. Variable w_x respectively w_y specifies size of summation window (usually $w_x = w_y$). Window size is very important parameter because it enables to capture small and large movements in images. Smaller window increases the ability of detecting smaller shifts. It is also essential to set the window size proportional to the size of objects (blood vessels). A great disadvantage of this method is dependency on successful vascular tree segmentation [1, 23]. This approach is followed by another key step: *detection of tracking points*.

This process is performed on one representative image. In the chosen pre-processed image a position of blood vessels is detected through Hessian matrix (*Equation 5.13*).

$$H_{ij} = \begin{pmatrix} \partial_{xx} \cdot I(x, y) & \partial_{xy} \cdot I(x, y) \\ \partial_{yx} \cdot I(x, y) & \partial_{yy} \cdot I(x, y) \end{pmatrix}. \qquad (5.13)$$

Hessian matrix and its eigenvalues are figured out for every element (pixel) of image. For mentioned step Sobel operator is used. Sobel operator is typical local operator approximating first derivative with differences. It is also important to realize that Sobel masks are directional. For this reason mask for horizontal and vertical direction is used. Definition of Sobel masks:

$$h_0 = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}, h_1 = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}. \qquad (5.14)$$

After Hessian matrices are obtained the highest positive value of eigenvalue for every pixel is chosen, by this selection a parametric image is created. The parametric image is already a first outcome of this detection. Usually thresholding is applied for improving the current results - elimination of false positive tracking points [1, 21].

For detected tracking points the *spatial transformation* (mentioned previously in *Chapter 5.1*) is done.

6 DESCRIPTION OF SELECTED METHOD

Our selected algorithm is based on previous works described in [8] and [13]. It expects that some statistical features of an image will be changed in the presence of artefacts, respectively distortions in general. *Image quality assessment* is done by extracting image statistic features (image *entropy*, *point-wised based features* and *pair-wised log-derivative features*) in a spatial and frequency domain. These features are then used for image quality estimation. All steps of algorithm are shown in *Fig. 6* and described below.
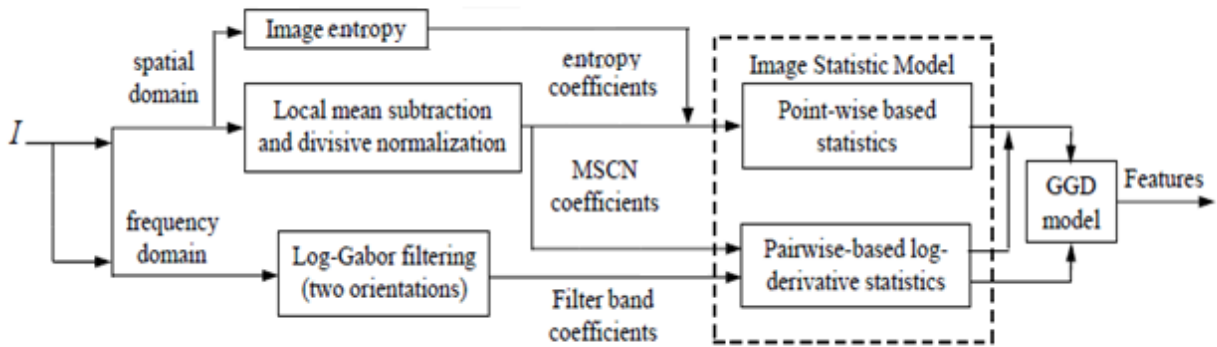
Fig. 6 - A block diagram of selected method. Adapted from [8].

6.1 Image statistics in a spatial domain

The spatial approach consists of computing single frame *entropy* (*Chap. 6.1.1*) , *point-wised based features* (*Chap. 6.1.2*) and *pair-wised based log-derivative features* (*Chap. 6.1.3*). These steps are described in detail in following chapters.

6.1.1 Entropy of an image

The main principle of *entropy* is firstly introduced in Shannon's paper, entitled "A Mathematical Theory of Communication" [31]. He considers *entropy* as a measure of uncertainty or amount of information which contains the image. Therefore the *entropy H(X)* of random variable *X* with mass probability function $p_x = (p_1,...p_n)$ is defined as [31]:

$$H(X) = \sum_{i=1}^{n} p_i \cdot \log\left(1/p_i\right), \qquad (6.1)$$

where n indicates a number of elements that belong to random variable *X* and *log (1/p_i)* corresponds to quantity of information expressed by image.

The equation *6.1* has following mathematical properties [31, 32]:

1. $H(X) \geq 0$ and only depends on probability function p_x.

2. *H(X)* is greater when all elements of variable *X* have equal probabilities.
3. *H(X,Y) ≤ H(X) + H(Y)*, the entropy (uncertainty) of joint event *H(X,Y)* is less or equal to the sum of separate entropies - *H(X)* and *X(Y)*.

For our retinal images we have to consider two-dimensional variable $X = (X_1, X_2)$. Probability of the pixel intensities can be estimated by applying normalized histogram:

$$p(i) = h_x(i)/N \,, \tag{6.2}$$

where *p(i)* represents pixel intensities probability, *$h_x(i)$* corresponds to intensity value *i* that enter the histogram function *h_x* of image *X* and *N* is total amount of pixel in image *X*. Thus we are able to extend formula *6.1* to this form [31, 32]:

$$H(X) = \sum_i h_x(i) \cdot \log N/h_x(i). \tag{6.3}$$

We obtained a corresponding value of *entropy* for each frame by this command:

```
E = entropy(I),
```

where *I* is greyscale retinal input image and *E* belong to the calculated *entropy*. This function is already in Matlab implemented. The *entropy* represents first statistical feature extracted by proposed method.

6.1.2 Point-wised based features

Point-wised based approach is based on application of local mean subtraction and normalization. This operation may be described by following equation:

$$\hat{I} = \frac{I(i,j) - \mu(i,j)}{\sigma(i,j) + C}, \tag{6.4}$$

where $i \in 1, 2 \ldots M, j \in 1, 2 \ldots N$ are spatial indices, *M* and *N* also represents the image *I* height and width, *C* = 1 is a constant that prevents the situation when a zero occurs in denominator, *μ(i, j)* and *σ(i, j)* are defined in *equation 6.5*, respectively *equation 6.6*.

$$\mu(i,j) = \sum_{k=-K}^{K} \sum_{l=-L}^{L} \omega_{k,l} \cdot I_{k,l}(i,j), \tag{6.5}$$

$$\sigma(i,j) = \sqrt{\sum_{k=-K}^{K} \sum_{l=-L}^{L} \omega_{k,l} \cdot (I_{k,l}(i,j) - \mu(i,j))^2} \,, \tag{6.6}$$

where $\omega = \{\omega_{k,l} / k = -K, \ldots, K, l = -L, \ldots L\}$ is a two dimensional circularly-symmetric Gaussian weighting function. The resulting values are known as *local mean subtracted coefficients (MSCN)*. In next images we show how window size of weighting function influences normalization of retinal sequence (*Fig. 7 – Fig. 9*). In all of these images four sub-pictures are

shown: original retinal image, *local mean field, local variance field* and *MSCN coefficients*. The most important image is in the bottom right corner, it is representing a feature which is further evaluated. We decided to set K and L to size of 13 because captured images contain large amounts of noise that is not completely removed by *averaging*. In *Fig. 7* we see that almost no feature was extracted from original image, however in next two images we can quite clearly see the vasculatory tree of retina. In the last picture (*Fig. 9*) the local mean field and the local variance field are too much smoothed, that is why we choose value of window size equals to 13.
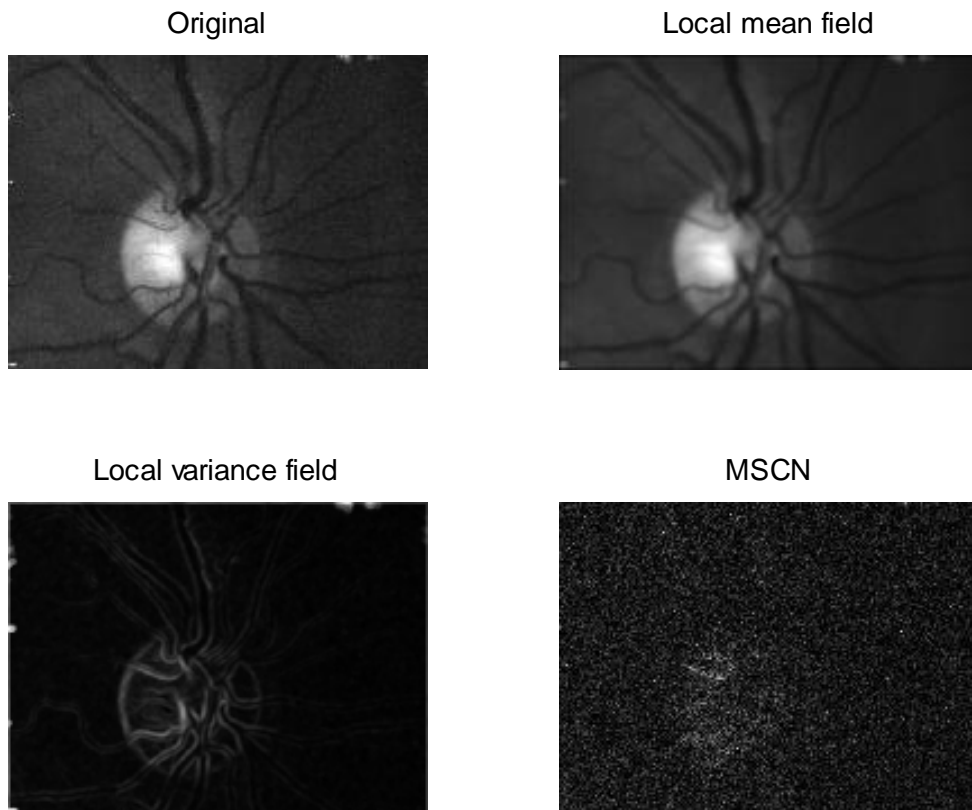
Fig. 7 - Effect of normalization on retinal image with size of weighting function 5x5.
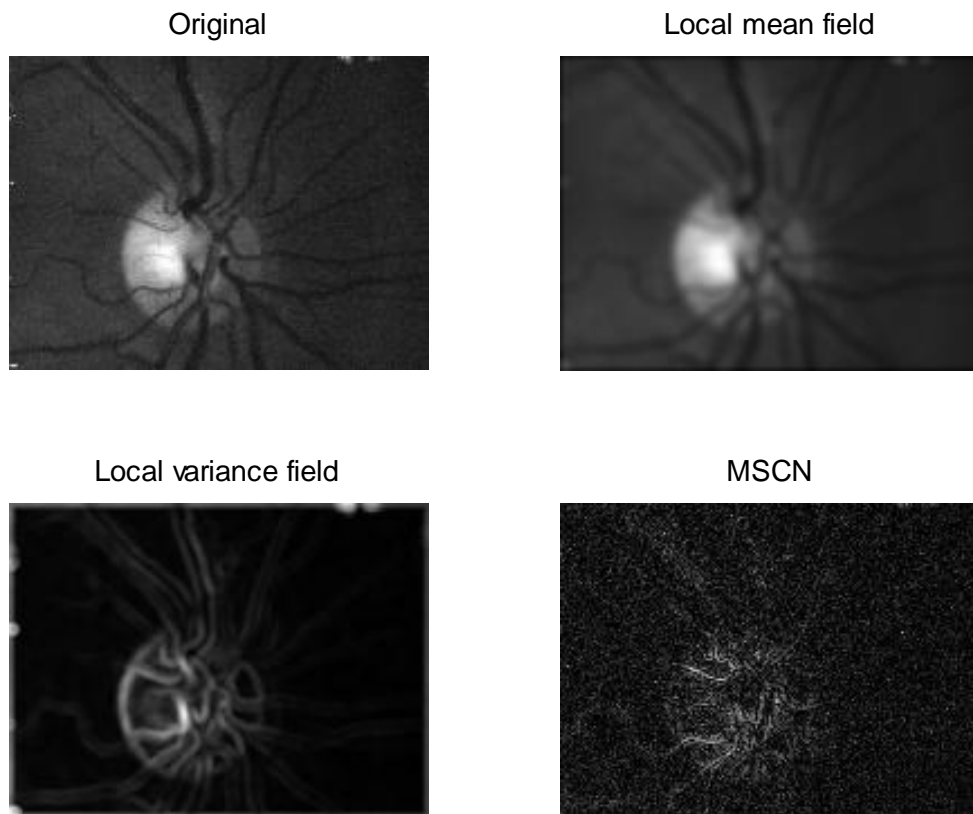
Fig. 8 - Effect of normalization on retinal image with size of weighting function 10x10.
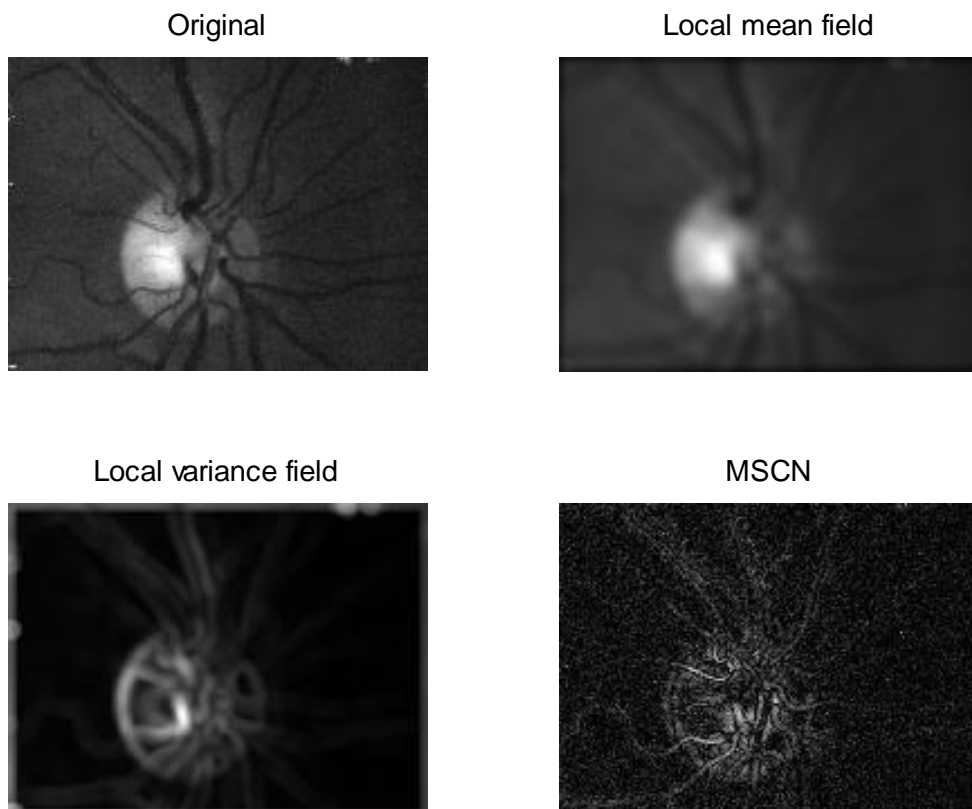
Fig. 9 - Effect of normalization on retinal image with size of weighting function 15x15.

For capturing the image statistics changes we assume that image pixel values have *Gaussian distribution.* That is why we are able to compute mean and standard deviation of this distribution. The mean and the standard deviation of every frame pixel distribution form the first set of features vector. The calculation of these two values is reached by performing Matlab function *normfit* with the following command:

```
[muhat,sigmahat] = normfit(data),
```

as a input we insert our pixel distribution (*data*) of the given frame and we get a mean value (*muhat*) and a standard deviation (*sigmahat*) of input distribution.

These computed features forms with entropy the first set of features that reveal the distortion and quality of each frame.

6.1.3 Pair-wised based log-derivative features

Statistical relations between adjacent pixels of images are computed through log-derivative statistics. Firstly logarithm of each pixel is computed:

$$J(i, j) = \log [I(i, j) + K] \tag{6.7}$$

where constant K excludes the possibility of original image pixels *I(i,j)* to be zero. Differences between greyscale values of neighbouring pixels are considered as its derivatives. Distance between adjacent pixels is equals to one. Each pair of pixels may have five orientation: horizontal, vertical, main-diagonal, secondary-diagonal and combined-diagonal. According to *Fig. 10* we can define the first four out of five types named above.
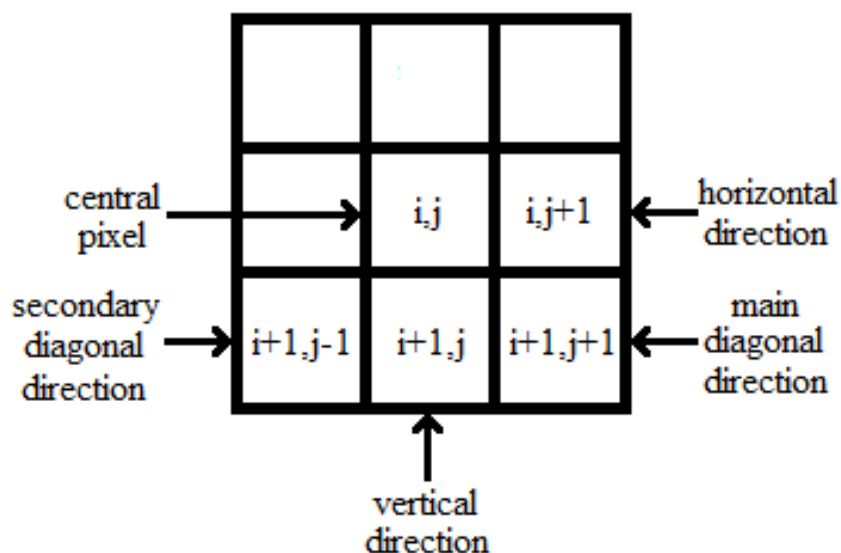
Fig. 10 - Illustration of pixel pairs orientations. Adopted from [13].

Mathematical notation of mentioned orientations is following:

$$H: \nabla_x J(i,j) = J(i, j+1) - J(i,j) \tag{6.8}$$

$$V: \nabla_y J(i,j) = J(i+1, j) - J(i,j) \tag{6.9}$$

$$D1: \nabla_{xy} J(i,j) = J(i+1, j+1) - J(i,j) \tag{6.10}$$

$$D2: \nabla_{yx} J(i,j) = J(i+1, j-1) - J(i,j) \tag{6.11}$$

$$D3: \nabla_x \nabla_y J(i,j) = J(i,j) + J(i+1, j+1) - J(i, j+1) - J(i+1, j) \tag{6.12}$$

The calculation of these directional differences represents one of the most effective ways of capturing an image distortion according to [8, 13].

We also use this log-derivative technique on *MSCN* coefficients $\hat{I}(i, j)$ which were mentioned in *Chap. 6.1.2. Equation 6.4* is therefore updated to:

$$J(i,j) = \ln\left[\left|\hat{I}(i,j)\right| + K\right]. \tag{6.13}$$

After this step all five orientations of neigboring pixels are computed and modelled in the same way as previous features in spatial domain, but with one exception. Estimation of the features (mean and standard deviation) is also accomplished by introducing Matlab function by this command:

```
parmhat = lognfit(data),
```

the input of function is formed by pixel distribution (*data*) and as a result we get variable *parmhat*. The first element of *parmhat* is mean of given lognormal distribution and the second one is standard deviation. In the following illustration (*Fig.11*) relations between adjacent pixels are depicted.
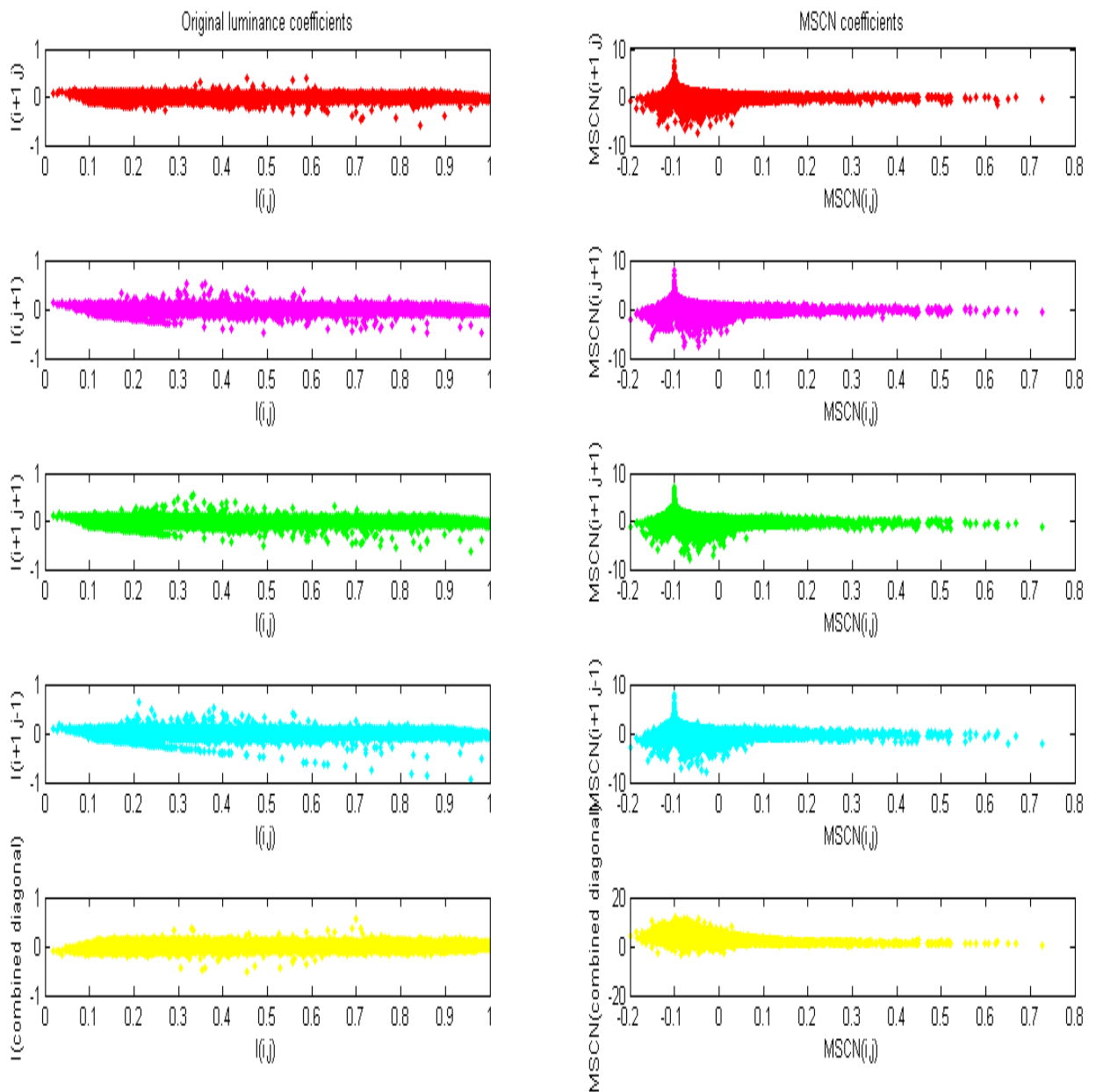
Fig. 11 - Scatter plot between neighbouring values of original luminance coefficients (left column) and MSCN coefficients (right column). Rows from top to bottom illustrate horizontal, vertical, main diagonal, secondary diagonal and combined diagonal neighbours.

We can see that there is high correlation between adjacent pixels in horizontal direction in original luminance coefficients. These dependencies are corrected by normalization. Process of normalization is evidently shown in the *MSCN* scatter plots (in the right part of the *Fig.11*).

6.2 Image statistics in a frequency domain

For quality estimation in frequency domain Log-Gabor filters are applied [25], thereby we achieve that the whole image is separated in horizontal and vertical direction. Convolution between input image and Log-Gabor filters is done via function implemented by Peter Kovesi in the year 2001 [33]. Following equation represents the header of mentioned function.

EO = gaborconvolve(im,   nscale,   norient,   minWaveLength,   mult,   sigmaOnf,   dThetaOnSigma,
feedback)

Input arguments are explained in *Tab. 3*. Majority of parameters is related to the filter

specification in the frequency plane.

Tab. 3 - Description of input parameters. Adjusted from [33].

im	Image to be convolved
nscale	Number of wavelet scales
norient	Number of filter orientations
minWaveLength	Wavelength of smallest scale
mult	Scaling factor between successive filters
sigmaOnf	Ratio of the standard deviation of the Gaussian describing the log Gabor filter's transfer function in the frequency domain to the filter central frequency.
dThetaOnSigma	Ratio of angular interval between filter orientations and the standard deviation of the angular Gaussian function used to construct filters in the frequency plane.
feedback	Optional parameter. If set to 1 a message indicating which orientation is being processed is printed on the screen.

The function returns 2D cell array (*EO*) which represents convolution over the image at given

scale and orientation. A typical example of results is depicted in *Fig. 12*.

Wavelet scale n. 1 and horizontal direction   Wavelet scale n. 1 and vertical direction

Wavelet scale n. 2 and horizontal direction   Wavelet scale n. 2 and vertical direction

Wavelet scale n. 3 and horizontal direction   Wavelet scale n. 3 and vertical direction

Wavelet scale n. 4 and horizontal direction   Wavelet scale n. 4 and vertical direction
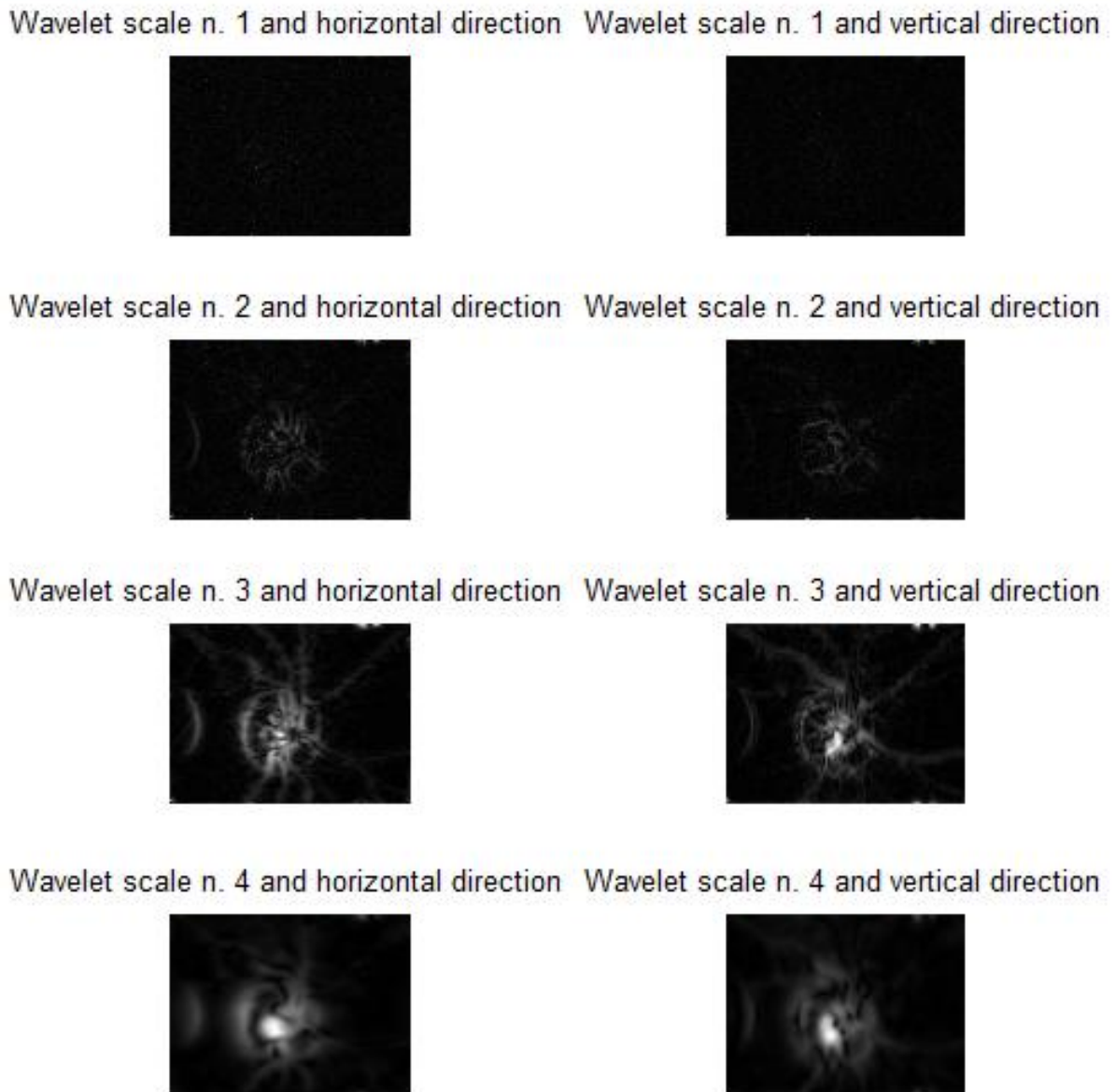
Fig. 12 - Magnitude of the convolution over the original image at four scales and two directions (horizontal, vertical).

We can clearly see that the most important information is hidden in bottom figures (wavelet scale n.3, n.4 in horizontal and vertical directions). Those figures represent high frequencies with information about details. Other input parameters were carefully empirically tested and we consider as the best to set *minWaveLenght to 3, mult to 3, sigmaOnf to 0.5 and dThetaOnSigma to 1.5.* The figures of wavelet scale n. 4 seem to be blurred by some high frequency noise, so they do not contain so much vesselness information. For this reason we decided to use just the third wavelet scale with two filter's orientation in horizontal and vertical direction. We also rejected the other low frequency bands (scale n.1 and n.2) and compute log-derivative features in five directions only for the third scale images, as in spatial domain. *Equation 6.7* is for frequency domain adapted to the following form:

$$J(i,j) = ln\ [|g(i,j)| + K], \tag{6.14}$$

where $g(i,j)$ represents the magnitude of the log-Gabor filter subband coefficient and K is small constant (typically $K = 1$) preventing to be zero in argument of logarithm. The *general Gaussian distribution* is used for each of computed coefficients to determine frequency domain set of features (*see Tab. 4*).

7 PRACTICAL PART

An essential part of this thesis is also dedicated to practical tasks. At first we discuss the implementation of proposed method. To find the best results of our method we evaluate features by means of statistics (*Chap. 7.1*) and select the proper classifier (*Chap. 7.2*). This two steps play very important role to achieve the best possible results. Then the calculation of quality is introduced (*Chap. 7.3*) and finally the *Chap. 7.4* deals with evaluation of implemented algorithm and its effect on quality of registrated sequences.

The selection of proper attributes for description of retinal image represents quite difficult task. In this work the features (for better reference see *Tab.4*), introduced in *Chap. 6* , are used for classification of retinal images into two classes. All 33 features belong to statistical parameters. *Tab. 4* describes serial number, name and provide basic information about each feature.

Tab. 4 - Table of statistical features extracted from every single frame

N.	Feature	Description
1	σ_{sin}	standard deviation of MSCN coefficients within single pixel
2	E	entropy
3	σ_{adj_hor}	standard deviation of MSCN coefficients within horizontal orientation
4	σ_{adj_ver}	standard deviation of MSCN coefficients within vertical orientation
5	σ_{adj_main}	standard deviation of MSCN coefficients within main diagonal orientation
6	σ_{adj_sec}	standard deviation of MSCN coefficients within secondary diagonal orientation
7	σ_{adj_comb}	standard deviation of MSCN coefficients within combined diagonal orientation
8	σ_{freq1_hor}	standard deviation in frequency domain n.1 within horizontal orientation
9	σ_{freq1_ver}	standard deviation in frequency domain n.1 within vertical orientation
10	σ_{freq1_main}	standard deviation in frequency domain n.1 within main diagonal orientation
11	σ_{freq1_sec}	standard deviation in frequency domain n.1 within secondary diagonal orientation
12	σ_{freq1_comb}	standard deviation in frequency domain n.1 within combined diagonal orientation
13	σ_{freq2_hor}	standard deviation in frequency domain n.2 within horizontal orientation
14	σ_{freq2_ver}	standard deviation in frequency domain n.2 within vertical orientation
15	σ_{freq2_main}	standard deviation in frequency domain n.2 within main diagonal orientation
16	σ_{freq2_sec}	standard deviation in frequency domain n.2 within secondary diagonal orientation
17	σ_{freq2_comb}	standard deviation in frequency domain n.2 within combined diagonal orientation
18	μ_{sin}	mean of MSCN coefficients within single pixel
19	μ_{adj_hor}	mean of MSCN coefficients within horizontal orientation
20	μ_{adj_ver}	mean of MSCN coefficients within vertical orientation
21	μ_{adj_main}	mean of MSCN coefficients within main diagonal orientation

22	μ_{adj_sec}	mean of MSCN coefficients within secondary diagonal orientation
23	μ_{adj_comb}	mean of MSCN coefficients within combined diagonal orientation
24	μ_{freq1_hor}	mean of magnitude in frequency domain n.1 within horizontal orientation
25	μ_{freq1_ver}	mean of magnitude in frequency domain n.1 within vertical orientation
26	μ_{freq1_main}	mean of magnitude in frequency domain n.1 within main diagonal orientation
27	μ_{freq1_sec}	mean of magnitude in frequency domain n.1 within secondary diagonal orientation
28	μ_{freq1_comb}	mean of magnitude in frequency domain n.1 within combined diagonal orientation
29	μ_{freq2_hor}	mean of magnitude in frequency domain n.2 within horizontal orientation
30	μ_{freq2_ver}	mean of magnitude in frequency domain n.2 within vertical orientation
31	μ_{freq2_main}	mean of magnitude in frequency domain n.2 within main diagonal orientation
32	μ_{freq2_sec}	mean of magnitude in frequency domain n.2 within secondary diagonal orientation
33	μ_{freq2_comb}	mean of magnitude in frequency domain n.2 within combined diagonal orientation

Essentially just mean and standard deviation of every single feature distribution are used. These features should represent the relations between pixels in spatial and frequency domain. We must not forget about the second feature which contains information about amount of information in every single image. This feature is called *entropy* (introduced in *Chap. 6.1*).

Due to the different values of the extracted features, we decided to divide the sequence into groups according to their acquisition parameters. The *Group1* consists of five sequences from different subjects. Each of them have about 190 frames (altogether 947 frames), which were captured by fundus camera with frame rate approximately equal to 12.5 fps (frames per second). The second *Group2* contains of four sequences that make up group of 1533 frames. The frame frequency is in this case about 25 fps. The largest *Group3* is formed by three sequences which have together 2569 frames. Frequency of capturing these frames is ca. 40 fps. For this reason, the further steps as features selection (*Chap. 7.1*), classification (*Chap. 7.2*) and evaluation of registration process (*Chap. 7.4*) are accomplished separately for this three groups.

Very important role also played subjective judgment whether given frame should belong to group of distorted or undistorted frames. This part represent very difficult task and is conditional to personal view. It means that this subjective determination may have some impact on further results.

Before we even start think about classifying our data we have to consider selecting features which should be "the best". This step will lead to reduction of the features vector in a proper manner. We also introduce *principal component analysis* (*PCA*) for extracting new features that may be more reliable than the original vector of features . The following chapter is dedicated to above mentioned issues.

For better orientation we introduce a workflow (see *Fig. 13*) which shows how we proceeded during practical realization of this master's thesis.
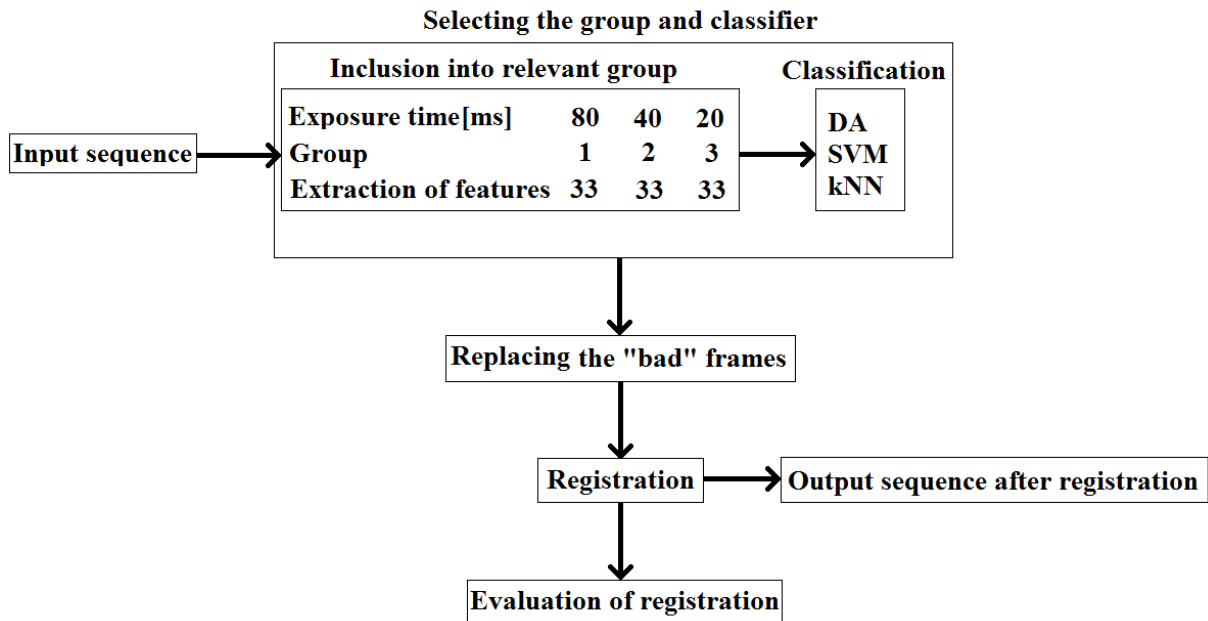
Fig. 13 - Workflow of practical part

An input retinal sequence is subjected to a decision-making process which assign the sequence into a relevant group according to the exposure time (frame rate) of given input sentence. This process involves the inclusion of input sequence into one of mentioned groups, then selection of the most robust features according to statistical analysis or *PCA*, and the last part of this process is formed by a binary classification of frames whether their quality is sufficient to registration or not. For testing the classification three classifiers were used: *Discriminant analysis function* (*DA*), *support vector machine* (*SVM*) and *k-nearest neighbour algorithm* (*kNN*) - described in Chap. 7.2. As we can see from *Fig. 13* we distinguish three groups (*Group1*, *Group2*, *Group3*) and for each of these groups the most suitable classifier was selected by testing the *success rate* (*Acc*), *specificity* (*SPC*) and *sensitivity* (*TPR*) of mentioned classifiers.

After this pre-processing step of determination the best classifier, the frames selected as "bad" are replaced by the closest neighbour frame which is regarded as good frame. Next step involves the registration process introduced in *Chap. 5*. As a result we get an output retinal sequence which frames should be aligned. The last step of the workflow is evaluation of registration by means of *normalized cross-correlation* (*NCC*) and *sum of squared differences metric* (*SSD*) which are introduced in *Chap. 7.4*.

7.1 Feature selection

For proper function of classifier it is needed to give him sufficient information on the analyzed data. Therefore, the features used to represent the data should differ. We also want to get rid of the features, which have poor discrimination quality and could negatively affect our classification accuracy. Thus this chapter deals with the analysis of image features (which are potentially most suitable for classification) by means of statistical tests.

7.1.1 Test of normality

Before choosing any procedure for the statistical evaluation of differences between good and distorted images, it is first necessary to test a data normality because the selection of a particular procedure is driven mainly by information whether the tested data are normally distributed or not. To test the normality values of the features calculated from fundus images was introduced Shapiro-Wilk test which is performed by Matlab function *swtest*. Null hypothesis states as follows: "X is normal distribution with unspecified mean and variance.". The function is called by command:

$$[H, \text{pValue}, \text{SWstatistic}] = \text{SWTEST}(X, \text{ALPHA}),$$

where *X* represents sample of data, *ALPHA* is the significance level. As a result we get three outputs. The most important is logical output *H* which determines whether we accept or decline the null hypothesis, *pValue* then represents p-value or probability that the null hypothesis is true. Finally *SWstatistic* variable give more specific information about the test statistic. To sum it up this test determines if the null hypothesis of normality is a reasonable assumption regarding the population distribution of a random sample *X* at the significance level *ALPHA* = 0.05. Results of the normality test are evidently shown is *Tab. 5*.

Tab. 5 - Test of normality. Table contains logical value of variable H and p-values, which are describing if we accept or decline null hypothesis. If H = 1, then we reject null hypothesis at the significance level 0.05, it means that given feature has not normal distribution. On the other hand H = 0 means that we do not reject null hypothesis at significance level 0.05.

Group / Feature n.	Group1 (fps = 12,5)		Group2 (fps = 25)		Group3 (fps = 40)	
	H	p-value	H	p-value	H	p-value
1	1	0.0000	1	0.0000	1	0.0000
2	1	0.0000	1	0.0000	1	0.0000
3	1	0.0000	1	0.0000	1	0.0000
4	1	0.0000	1	0.0000	1	0.0087
5	1	0.0000	1	0.0000	1	0.0000
6	1	0.0000	1	0.0000	1	0.0008
7	1	0.0000	1	0.0000	1	0.0000
8	1	0.0000	1	0.0000	1	0.0000
9	1	0.0000	1	0.0000	1	0.0000

10	1	0.0000	1	0.0000	1	0.0000
11	1	0.0000	1	0.0000	1	0.0000
12	1	0.0000	1	0.0000	1	0.0000
13	1	0.0000	1	0.0000	1	0.0000
14	1	0.0000	1	0.0000	1	0.0000
15	1	0.0000	1	0.0000	1	0.0000
16	1	0.0000	1	0.0000	1	0.0000
17	1	0.0000	1	0.0000	1	0.0000
18	1	0.0000	1	0.0000	1	0.0000
19	1	0.0000	1	0.0000	1	0.0359
20	1	0.0000	1	0.0000	1	0.0000
21	1	0.0000	1	0.0000	1	0.0000
22	1	0.0000	1	0.0000	1	0.0000
23	1	0.0001	1	0.0000	1	0.0000
24	1	0.0000	1	0.0000	1	0.0000
25	1	0.0000	1	0.0000	1	0.0000
26	1	0.0000	1	0.0000	1	0.0000
27	1	0.0000	1	0.0000	1	0.0000
28	1	0.0000	1	0.0000	1	0.0000
29	1	0.0000	1	0.0000	1	0.0000
30	1	0.0000	1	0.0000	1	0.0000
31	1	0.0000	1	0.0000	1	0.0000
32	1	0.0000	1	0.0000	1	0.0000
33	1	0.0000	1	0.0000	1	0.0000

We can clearly see that all of features do not have normal distribution of values. This finding leads to application of non-parametric test that will test the difference between two groups (undistorted and distorted images). Examples of how the features look like can we see from the following histograms (*Fig.14*).

The distribution of values demonstrated in *Fig.14* apparently cannot be considered as normal one. We can only assume that with this feature we will potentially be able to separate our set of retinal image data into two groups.
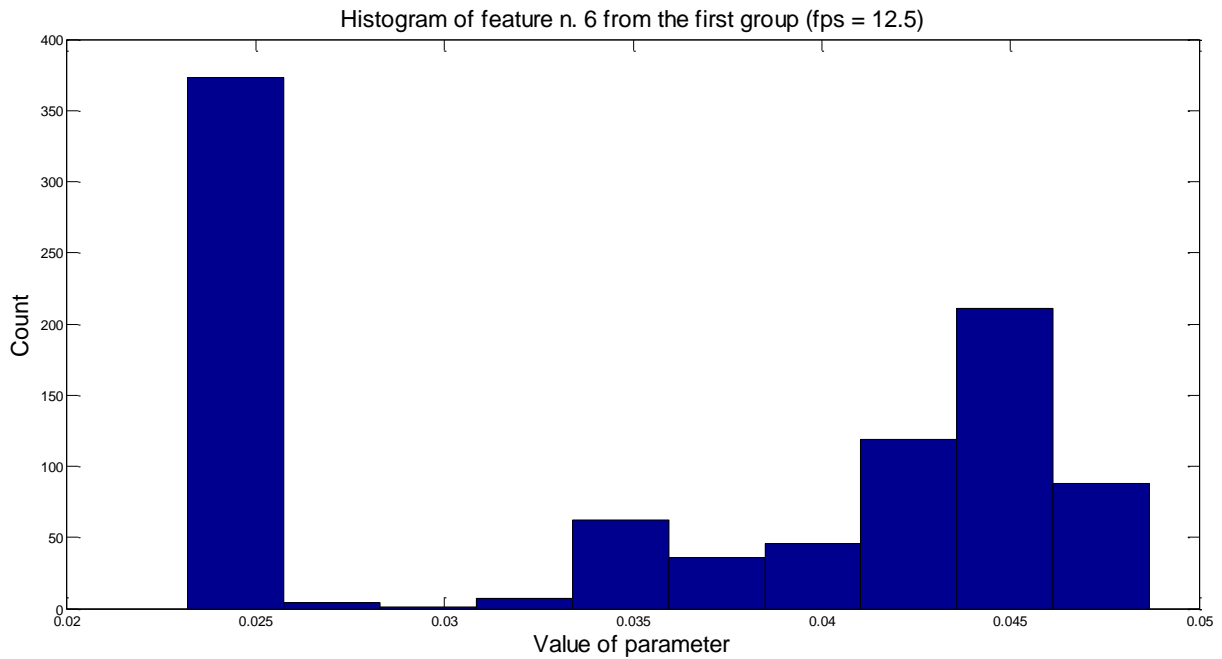
Fig. 14 - Histogram of features n.6 from the Group1 (fps = 12.5).

7.1.2 Testing difference between two datasets

For testing the differences between two classes (distorted and undistorted images) were introduced non-parametric Wilcoxon rank sum tests, which null hypothesis assumes that data in groups are samples from continuous distributions with equal medians. This test is in Matlab realized through function *ranksum*. The function is called by command:

```
[p,h] = ranksum(x,y),
```

where *x* and *y* are the input data, which are independent and can have different lengths. The first output is *p*, which means p-value of test. The function also returns logical value *h* that indicates the decision of the test. If *h = 1* then we reject null hypothesis, the other case (*h = 0*) means the opposite.

Tab. 6 - Results of Wilcoxon rank sum test. Table contains p-values, which decides if the given feature has differences between groups (p < 0,05) or not (p > 0,05). This decision is made by the value in h. Boldface numbers indicates statistically significant result.

	Group1 (fps = 12,5)		Group2 (fps = 25)		Group3 (fps = 40)	
Group / Feature n.	h	p-value	h	p-value	h	p-value
1	0	0.3132	**1**	0.0000	0	0.1536
2	0	0.9991	**1**	0.0000	0	0.0792
3	0	0.9847	0	0.1058	0	0.8990
4	0	0.0597	0	0.4399	**1**	0.0339
5	0	0.4754	0	0.1390	**1**	0.0468
6	0	0.3112	0	0.0552	**1**	0.0141

7	0	0.7613	0	0.3689	0	0.8342
8	1	0.0000	1	0.0000	1	0.0000
9	1	0.0000	1	0.0000	1	0.0001
10	1	0.0000	1	0.0000	1	0.0002
11	1	0.0000	1	0.0000	1	0.0000
12	1	0.0000	1	0.0000	1	0.0000
13	1	0.0000	1	0.0000	1	0.0000
14	1	0.0000	1	0.0000	1	0.0400
15	1	0.0000	1	0.0000	1	0.0064
16	1	0.0000	1	0.0000	1	0.0007
17	1	0.0000	1	0.0000	1	0.0000
18	1	0.0006	1	0.0007	0	0.0796
19	0	0.6596	1	0.0002	0	0.6846
20	1	0.0083	0	0.4938	0	0.2726
21	0	0.5523	0	0.9774	0	0.7875
22	1	0.0227	0	0.0718	0	0.3968
23	1	0.0125	0	0.1063	0	0.2125
24	1	0.0005	1	0.0000	1	0.0000
25	1	0.0006	1	0.0000	1	0.0000
26	1	0.0021	1	0.0000	1	0.0000
27	1	0.0008	1	0.0000	1	0.0000
28	1	0.0043	1	0.0000	1	0.0000
29	1	0.0002	1	0.0000	1	0.0000
30	1	0.0081	1	0.0000	1	0.0000
31	1	0.0073	1	0.0000	1	0.0000
32	1	0.0020	1	0.0000	1	0.0000
33	1	0.0047	1	0.0000	1	0.0003

From this test we cannot deduce the discriminatory strength of features but we can see that there are some differences also between groups, which differ in acquisition parameters. For further analysis and classification we pick only features with positive result, that means 24 features out of 33 for the *Group1* and *Group2* and for *Group3* this test delivered 23 features. According to this test we can assume that features from spatial domain are not very convenient, on the other hand features from frequency domain seems to be more reliable.

7.1.3 Box whisker plots

To visualize previous results are box-whisker diagrams applied, which graphically represent the distribution of values in groups and their variability. On every box there is horizontal central line that represents the median. The inner part of the box expresses 50 % of all values, therefore the edges of the box are the 25th and 75th percentiles. The data between the edges of the box are also known as interquartile range (IQR). The whiskers extend to the most extreme data that are belonging to the interval of $\pm 1.5 \times$ IQR. Finally, outliers are depicted individually as red cross.

In this chapter we present only the typical representatives of box-plots . From each group we illustrate one diagram that represents the most common boxplot in the given group. The rest of box-whisker plots are attached in *Appendix*.
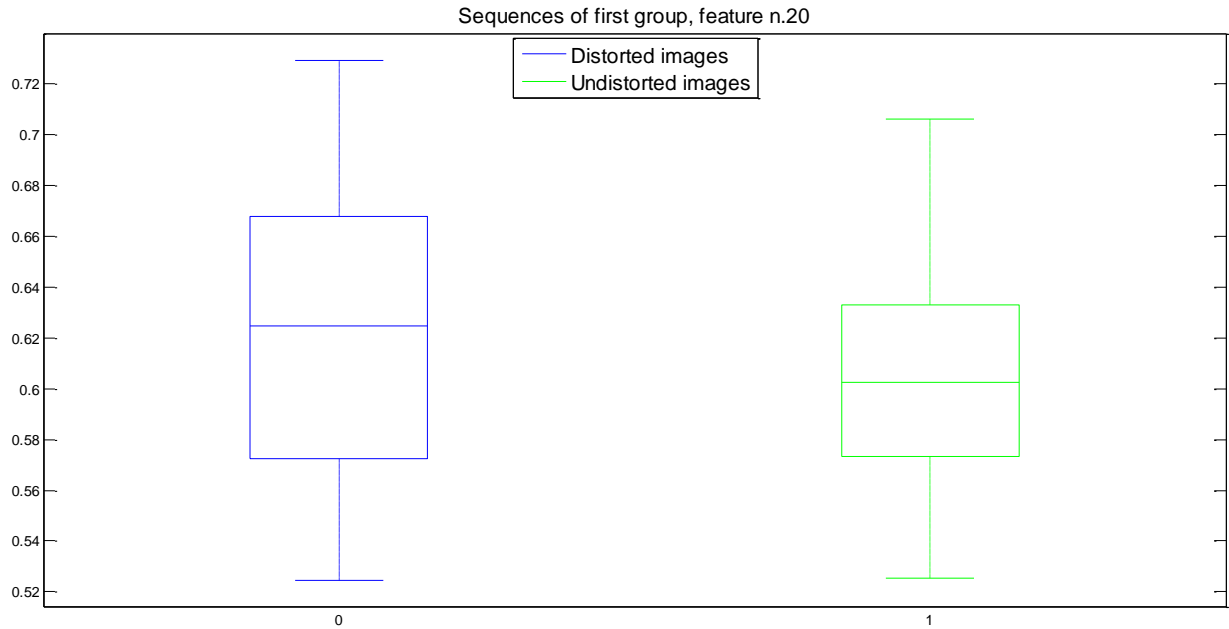
Fig. 15 - Box-whisker plot of feature n. 20 from the Group1 (fps = 12.5) frames.
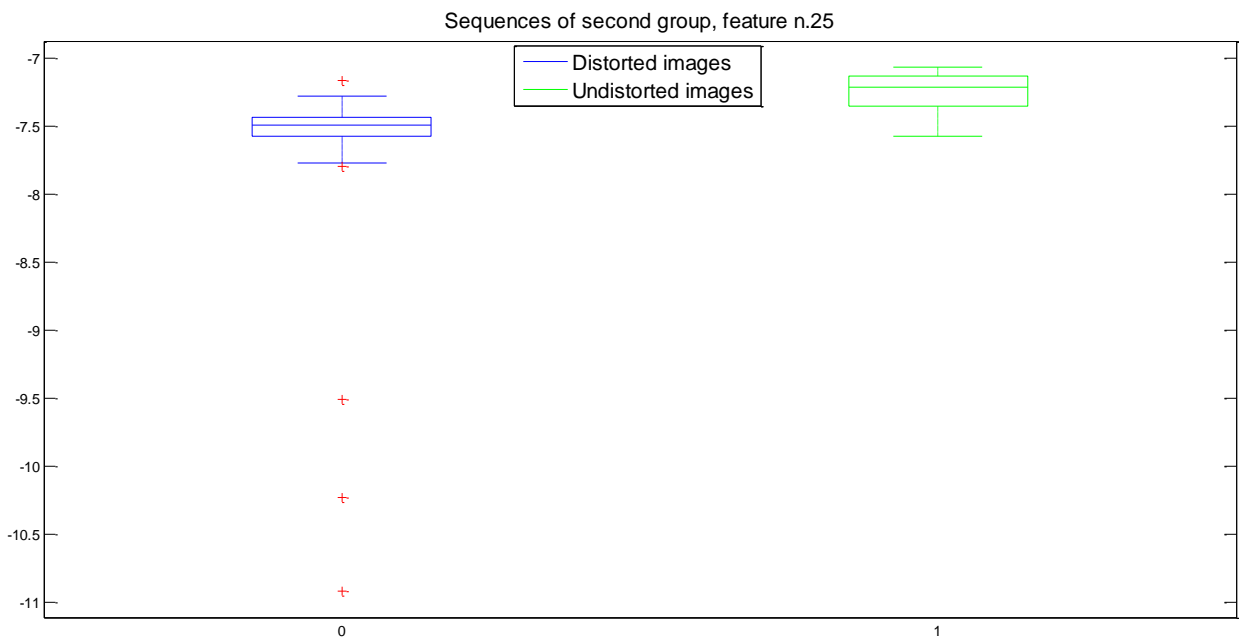
Fig. 16 - Box-whisker plot of feature n. 25 from the Group2 (fps = 25) frames.
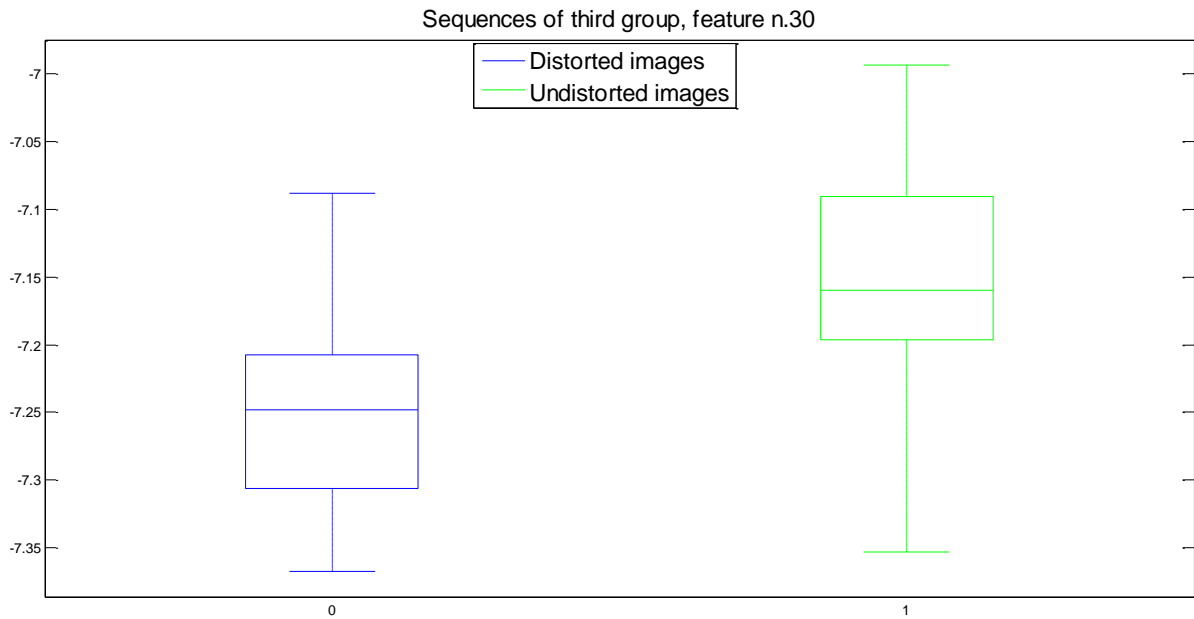
Fig. 17 - Box-whisker plot of feature n. 30 from the Group3 (fps = 40) frames.

From previous whisker-box diagrams (*Fig. 15-17*) we can observe that there are differences among groups, which confirms the results of Wilcoxon rank tests. We can also see that the medians are quite distinguished, however in the majority of diagrams there is an overlap of the values between undistorted and distorted images. This is probably due to the different anatomical proportions of every subject retina in groups and also the fundus camera adjustment can cause some issues.

7.1.4 Extraction of new features

Closely related to features selection is extraction of new features. For this task we introduce *Principal component analysis (PCA). Principal component analysis* is a statistical method used to an *ordination analysis*. By *ordination analysis* is understood the dimensionality reduction. This reduction proceeds by merging of correlated variables into a so-called factor variables. Merging is always accompanied by a loss of information. The aim is to find such factor axes in that the information loss is as small as possible. The assumptions of *PCA* is the absence of outliers and the lack of several object groups [26].

It is necessary to work with association matrices when we are finding the factor axes. Elements of association matrices are indicating differences between individual variables against each other. Typically *correlation* or *covariance matrix* are used. The *correlation matrix* is used when the compared variables are not in comparable units. Otherwise we use the *covariance matrix*. In the *correlation matrix* elements of the diagonal are equal to one, which corresponds to each variable compliance. On the other hand the diagonal elements of *covariance matrix* characterized the variance [26].

An important step of *PCA* analysis is the calculation of *eigenvalues* of an used association matrix. The characteristic value (*eigenvalue*) of each variable indicates, how many times is the

factor axis exhausting the variability more, than in normal distribution case. Furthermore the *eigenvectors* are also usually calculated. The *eigenvectors* show the direction of factor axes [26].

Difficult and not always completely clear objective is to identify the optimal number of factor axes for further analysis. In this case we use the so-called *Kaiser-Guttman criterion*, which says that for further analysis, we use only the factorial axes that have the characteristic value greater than one (in the case of the *correlation matrix*) or that have the characteristic value greater than average characteristic value (in the case of *covariance matrix*) [26].

PCA is in Matlab performed by function *princomp*. The function is called by command:

```
[COEFF,SCORE,latent,tsquare] = princomp(X),
```

where the input *X* is formed by the matrix of features and as output we get matrix *COEFF* with principal component coefficients, where each column describes one principal component. Variable *SCORE* returns the principal component score, *latent* corresponds to eigenvalues of the covariance matrix of our input data matrix *X* and *tsquare* represents some additional statistics for each data point.

We perform *PCA* on two different groups of features. First group contains all extracted features (33 features), the second group contains only reduced number of features. This number of features matches with results from Wilcoxon rank test in *Chap. 7.1.2*. A necessary step before the *PCA* itself is a normalization of feature values into the interval from 0 to 1. *PCA* can be introduced to create new features or to select reduced set of features. By applying the *Kaiser-Guttman criterion* we reached following results (see Fig. 18-20). The following graphs expresses how much of total variability is explained by those components whose *eigenvalue* is greater than 1. On the left side are always depicted results for whole set of features, extracted by used method. On the right side of every image are results of features, which were reduced by Wilcoxon rank sum test (see *Chap. 7.1.2*). Each column represents the percentage of explained variability by the corresponding component and the blue curve shows the cumulative sum of previously explained total variability.
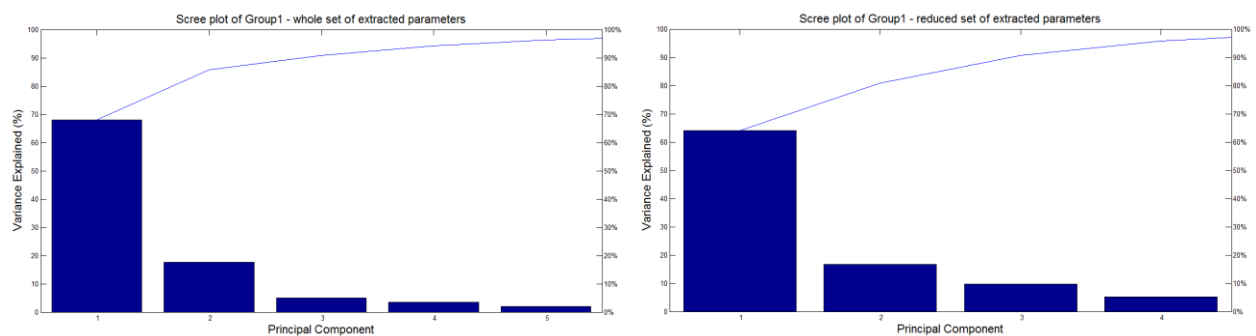
Fig. 18 - Scree plot of the percent variability explained by each principal component for Group1. On the left is result for whole set of extracted features and on the right side is result for reduced set of extracted features (24).
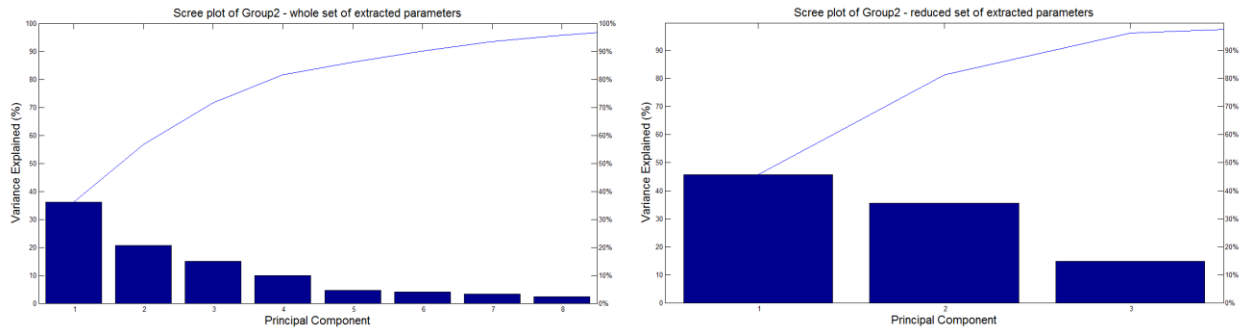
Fig. 19 - Scree plot of the percent variability explained by each principal component for Group2. On the left is result for whole set of extracted features and on the right side is result for reduced set of extracted features (24).
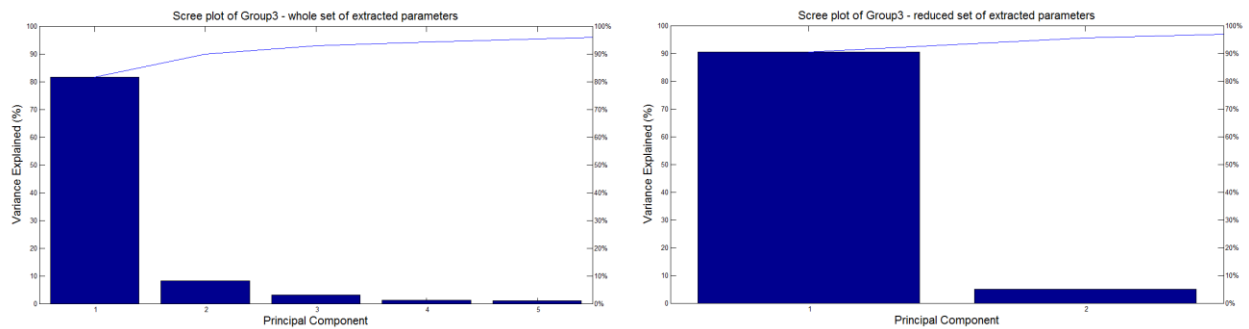
Fig. 20 - Scree plot of the percent variability explained by each principal component for Group1. On the left is result for whole set of extracted features and on the right side is result for reduced set of extracted features (23).

For classification of *Group1* sequences will thus be used five main components for the first case and four main components for the second case. Subsequently by classification of sequences belonging to *Group2* we will used eight main components for the first case and three main components for second case. Finally in the *Group3* will be used just five main components for the first case and only two main components for the second case. The results of *PCA* are summarized in *Tab.7*.

Tab. 7 - Results of PCA. Table of main components used for further classification with appropriate number of explained variability.

Group	Number of components - whole set of features [-]	Explained variability - whole set of features [%]	Number of components - reduced set of features [-]	Explained variability - reduced set of features [%]
Group1	5	96.25	4	95.78
Group2	8	95.76	3	96.31
Group3	5	95.43	2	95.64

7.2 Classification

After we have successfully chosen a set of appropriate features we can move to automatic frame classification. There are numerous different methods that allow user to classify his data. In particular we will concentrate on *Discriminant function analysis* (*Chap. 7.2.1*), *Support vector machine* (*Chap. 7.2.2*) approach and *k-nearest neighbours algorithm (Chap. 7.2.3)*. A substantial part is formed by evaluating previous mentioned approaches and selection of the best approach. For the classification are used only features, which have appeared to be statistically significant (*Chap. 7.1.2*) and features obtained from *Principal component analysis* (*Chap. 7.1.4*).

7.2.1 Discriminant function analysis

The main aim of *Discriminant function analysis (DA)* is the identification of variables. By these variables it is possible to distinguish a predetermined groups of objects. If there is a linear relationship between the variables, then this analysis is called *Linear discriminant analysis* (*LDA*). The requirement of *DA* is the possibility of separating the objects along a line which goes through multidimensional space. The second assumption is the absence of outliers and redundant variables. The last assumption is multidimensional normal distribution of predictors in each group of features. *DA*'s output is called *Wilks lambda*, which indicates the difference of the group centroid position. These positions are defined by given variables. Alternatively *Wilks lambda* shows how much are we able to distinguish group of objects by using proposed axes. In contrast to the *PCA* there is not assessment of exhausted variability, but we examine the ability of distinguishing the groups in the set of data. Every model of *DA* must be verified whether its sufficient quality and versatility. The verification can be done in different ways. In our case *cross-validation* will be applied. *Cross-validation* performs random partition of examined data into two groups. On the first group is the model created and then the results are tested on the remaining second group [26].

For classification by means of *discriminant analysis* Matlab offers function *classify*. With this function we are also able to estimate posterior probability that the training group was source of the tested sample. This property allows us to estimate a quality of sequence and also of single frames. Function *classify* is called as follows:

```
[class,err,POSTERIOR] = classify(sample,training,group,'type'),
```

where in *sample* are tested data, variable *training* contains training features which are grouped by variable *group* and *type* specify the discriminant function. The output include the estimated group of frames - variable *class*, in variable *err* returns the misclassification error and *POSTERIOR* is a matrix of the posterior probabilities.

For data classification, we use consecutive discriminant functions: linear, diaglinear, quadratic, diagquadratic and mahalanobis. See *Chap. 7.2.4* for the results of above mentioned discriminant functions.

7.2.2 SVM (Support vector machines)

SVM was introduced in early 90's of the 20th century by Boser et al. [27]. It is algorithm based on statistical learning theory and its application is quite wide (e.g. pattern recognition or bioinformatics). The aim of the method is to find a hyperplane that linearly separates the classification groups. Such a separation hyperplane should simultaneously provide the widest range between the hyperplane and positive cases (cases of type 1) and the widest range between the hyper plane and negative cases (cases of type 2). The robustness of the classification is directly proportional to the width of these ranges. The closest points from the separating hyperplane on both sides are called *support vectors*. These points actually determine support for the proposed separating hyperplane, and have significant affect on classification robustness. Usually it is not possible to find a linear separating hyperplane in the original data and therefore need to transform the original data into a space with more dimensions. Basically, *SVM* is trying to achieve linear separation of non-linearly separable data by transferring data from n-dimensional original space into an m-dimensional characteristic feature space, where $m > n$. In this characteristic space it is already possible to perform linear separation of data [27, 28, 29, 30]. An exemplary case of using *SVM* is depicted in *Fig. 21* and *Fig. 22*.
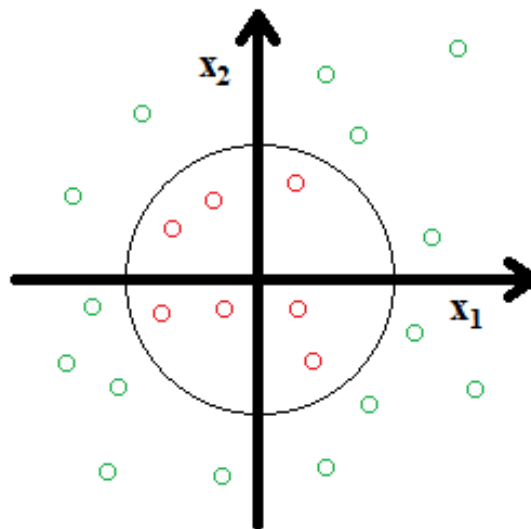
Fig. 21 - Two-dimensional space with positive (marked with red colour) and negative (marked with green colour) samples used for training. The actual boundary is represent by the circular function. Adapted from [27].
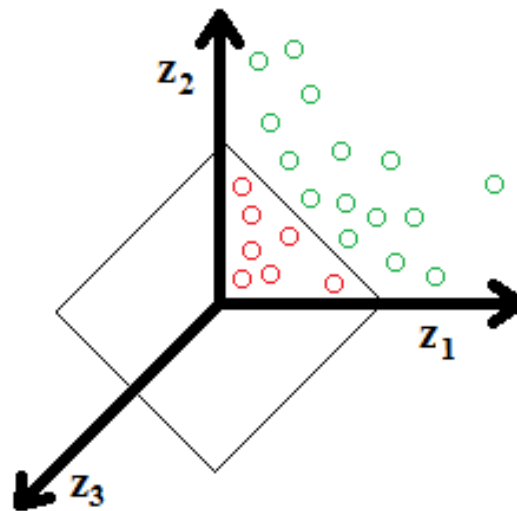
Fig. 22 - The same data mapped to a three-dimensional space. The circular boundary from Fig. 20 become linear. Adapted from [27].

By this approach will be the retinal frames also separated into two classes. First class concerns frames regarded as not-distorted, on the other hand in the second class are frames considered as blurred or damaged by any type of distortion. To verify the reliability is again performed *cross-validation*, which was introduced in *Chap. 7.2.1*.

Support vector machine classifier is in Matlab divided into two stages. Firstly the training is done by this command:

```
SVMStruct = svmtrain(Training,Group,Name,Value),
```

the function returns a structure *SVMStruct* which contains information about the trained support vector machine classifier. As input we enter training data (*Training*) , their grouping variable (*Group*) and as optional parameter we specify four different kernel functions (linear, quadratic, polynomial or rbf). Afterwards we simply determine the belonging of frames into groups through subsequent function:

```
Group = svmclassify(SVMStruct,Sample),
```

where each row of variable *Sample* is classified with information from *SVMStruct* into output variable *Group*. The uncertainty of wrong classification is calculated by Matlab function *svmdecision.m*. This measure allows us to estimate the quality of sequences respectively single frame quality. Results of SVM classifiers are illustrated and described in *Chap. 7.2.4*.

7.2.3 k-nearest neighbour algorithm (kNN)

The *k-nearest neighbour* algorithm is often mentioned in association to so-called instance-based methods. The principle is quite simple. The aim is to classify an instance (in our case frame) into a group through its similarity, respectively distance to nearest neighbours. For a distance

definition we need to apply some kind of metric. The nearest neighbour of an sample is usually determined by Euclidean metric [34]:

$$d\left(x_i, x_j\right) = \sqrt{\sum_{r=1}^{n}\left(a_r(x_i) - a_r(x_j)\right)^2}, \qquad (7.1)$$

where $a_r(x)$ represents the value of rth feature of instance x and x_i, x_j are two independent samples. For the special case when k = 1, is the sample classified to group of nearest neighbour from testing set of data. Otherwise the algorithm assigns to examined sample the most common value among k nearest neighbour training set of instances [34].

The big advantages of this method is its very simple construction and low computational requirements in comparison to other types of classification method like *SVM*. Reliability of this method is also verified by means of *cross-validation.*

Classification of data using nearest neighbour method is in Matlab implemented by *knnclassify* function, which is called by this command:

```
Class = knnclassify(Sample, Training, Group, k, distance),
```

where *Sample* is tested set of data, *Training* are data intended for training, variable *Group* define grouping of *Training* matrix, k is the number of nearest neighbour used in the classification and finally *distance* specify the distance metric. As a output we get a vector *Class* which indicates an assigned group for the *Sample* matrix.

We tested to change the number of nearest neighbour k with the following order 1, 3, 5, 8, 10. As a distance metric was used just Euclidean metric. The results are described in *Chap. 7.2.4*.

7.2.4 Implementation of classification

Three classification methods were tested on three datasets (*Group1*, *Group2* and *Group3*) - these methods were in detail discussed in *Chap. 7*. Training of classifiers and their testing was performed by tenfold cross validation. This implies that at the beginning the whole data were randomly divided into ten sets. Nine of them were used for training an classification model and the remaining one was applied for testing.

Each classification method has been tested on several different combinations of features, which represents the given class (*0*-distorted frame, *1*-undistorted frame). To represent the frames were used all 33 extracted features (afterwards labelled as *All*), then features which were statistically significant according to *Wilcoxon rank test* (later labelled as *Chosen*) and finally the features extracted by *PCA* (afterward labelled as *All-PCA* or *All-Chosen*). In addition to the original set of values we tested classifiers also on normalized features from the first two groups of features. Therefore we posses six following groups of features:

1. *All* - 33 features for all three groups (*Group1*, *Group2*, *Group3*)

2. *All-norm* - 33 normalized features for all three groups (*Group1*, *Group2*, *Group3*)

3. *Chosen* - 24 features for *Group1*, *Group2* and 23 features for *Group3*

4. *Chosen-norm* - 24 normalized features for *Group1*, *Group2* and 23 normalized features for *Group3*

5. *All-PCA* - 5 *PCA* features for *Group1*, 8 *PCA* features for *Group2*, 5 *PCA* features for *Group3*

6. *Chosen-PCA* - 4 *PCA* features for *Group1*, 3 *PCA* features for *Group2*, 2 *PCA* features for *Group3*

Conclusively the average success rate of classification (*Acc*), specificity (*SPC)* and sensitivity (*TPR*) for each classification method are computed. The classification success rate is determined as the percentage of correctly classified frames out of the test set of frames. *Acc* indicates the average success rate of classification (in %) during training and testing by tenfold cross-validation. *SPC* measures the proportion (in %) of negatives (undistorted frames) which are correctly identified and *TPR* express the rate (in %) of positive (distorted frames) which are correctly determined.

The following tables (*Tab.8-13*) show results of classification and computational demands (elapsed time on a single training and performed classification by using given set of features).

1 *Group1*

Tab. 8 - The results of the classification success rate (Acc), the sensitivity (SPC) and specificity (TPR) of the classification performed by using the given set of features on Group1 sequences. The best result is indicated in boldface.

Selected features / classifier	All			All-norm			Chosen			Chosen-norm			All-PCA			Chosen-PCA		
	Acc [%]	SPC [%]	TPR [%]	Acc [%]	SPC [%]	TPR [%]	Acc [%]	SPC [%]	TPR [%]	Acc [%]	SPC [%]	TPR [%]	Acc [%]	SPC [%]	TPR [%]	Acc [%]	SPC [%]	TPR [%]
DA - linear	97.7	98.8	86.9	84.6	95.0	55.0	95.6	97.2	80.1	16.4	55.0	95.0	73.2	78.7	88.8	63.7	73.2	84.0
DA - diaglinear	72.7	79.8	66.2	84.6	95.0	55.0	68.0	76.7	66.7	16.3	55.0	93.9	74.1	79.4	87.9	64.0	73.4	84.0
DA - quadratic	97.5	98.4	89.0	7.1	50.0	100.0	**97.4**	**97.9**	**93.9**	7.1	50.0	100.0	90.7	92.2	89.8	73.0	79.4	77.3
DA - diagquadratic	94.9	99.9	60.2	7.1	50.0	100.0	94.8	99.8	60.2	7.1	50.0	100.0	71.9	78.5	86.5	44.8	65.1	81.0
DA - mahalanobis	95.8	96.3	93.0	7.1	50.0	100.0	89.8	90.3	99.0	7.1	50.0	100.0	58.3	70.0	99.0	35.5	59.6	97.9
SVM - linear	97.2	99.2	78.8	84.6	95.0	55.0	96.8	99.7	71.2	16.4	55.0	95.0	95.7	98.0	75.6	93.2	96.9	68.4
SVM - quadratic	97.1	99.0	81.2	7.1	50.0	100.0	97.5	99.7	79.0	7.1	50.0	100.0	90.3	92.0	84.7	73.1	79.4	79.4
SVM - rbf	95.2	99.8	62.7	92.9	100.0	50.0	96.9	99.8	73.7	92.9	100.0	50.0	92.2	94.2	79.6	85.5	89.7	74.7
SVM - polynomial	97.0	98.7	82.8	84.6	95.0	55.0	96.6	98.5	79.0	16.4	55.0	95.0	73.4	79.8	70.4	69.2	77.3	68.1
kNN - k=1	96.5	98.7	78.4	15.5	55.0	95.3	96.7	99.0	77.4	15.5	55.0	95.3	73.4	79.8	70.4	69.2	77.3	68.1
kNN - k=3	96.9	99.1	77.6	15.4	55.0	95.0	96.9	99.3	75.1	15.4	55.0	95.0	74.8	80.9	68.4	69.6	77.5	68.2
kNN - k=5	97.3	99.4	76.7	15.4	55.0	95.0	96.9	99.3	74.8	15.4	55.0	95.0	74.4	80.8	67.2	70.4	78.1	67.1
kNN - k=8	97.2	99.6	74.4	15.4	55.0	95.0	96.4	99.6	69.7	15.4	55.0	95.0	74.9	81.2	66.1	70.2	78.2	64.2
kNN - k=10	96.9	99.6	73.0	15.4	55.0	95.0	96.4	99.6	69.2	15.4	55.0	95.0	75.1	81.4	66.3	70.7	78.5	64.2

Tab. 9 - Duration of each classifier setting for the given features of Group1 sequences.

Selected features/ classifier	All - t[s]	All-norm -t[s]	Chosen - t[s]	Chosen-norm - t[s]	All-PCA - t[s]	Chosen - PCA - t[s]
DA - linear	0.010	0.011	0.007	0.006	0.007	0.006
DA - diaglinear	0.004	0.004	0.004	0.003	0.003	0.002
DA - quadratic	0.003	0.003	0.002	0.002	0.002	0.001
DA - diagquadratic	0.002	0.002	0.002	0.002	0.001	0.001
DA - mahalanobis	0.003	0.003	0.002	0.002	0.001	0.001
SVM - linear	95.730	89.946	79.787	79.340	77.967	76.558
SVM - quadratic	110.960	107.280	90.310	89.440	91.690	82.023
SVM - rbf	71.196	68.475	70.500	70.130	67.898	76.217
SVM - polynomial	110.980	107.070	101.900	104.906	102.356	109.022
kNN - k=1	0.016	0.015	0.015	0.017	0.013	0.014
kNN - k=3	0.006	0.006	0.005	0.005	0.005	0.004
kNN - k=5	0.006	0.006	0.006	0.005	0.004	0.004
kNN - k=8	0.006	0.005	0.005	0.004	0.005	0.005
kNN - k=10	0.006	0.005	0.005	0.005	0.004	0.004

Discussion of achieved results - *Group1*

The most stable results for all groups of features are achieved by *SVM* method. The average success rate of *SVM* for all groups of features is 82.8 % with specificity even to 88.9% and sensitivity reaches just 75.7 %. The best *SVM* result was by using *polynomial kernel* on the group of *All* extracted features. In this case the Acc come up to 97 %, specificity to 98.7 % and sensitivity to 82.8%. At this point we have to emphasize an essential role of sensitivity which is not ideal. Other thing which would not favour *SVM* is its quite high time consumption in comparison to *DA* or *kNN* classifiers. The average time of *SVM* classifier learning is 88.8 seconds.

The second place regarding the average success rate of classification occupy the *kNN* classifier (*Acc* = 61.5 %). The average specificity and sensitivity were for this case also the second best (*SPC* = 77.9 % and *TPR* = 78.9 %). The most reliable outcome is by applying *kNN* classifier on set of *All* features with 1 (k = 1) nearest neighbour used in classification (*Acc* = 96.5 %, *SPC* = 98.7 % and *TPR* = 78.4 %). In principle the results are getting worse with an increasing number of nearest neighbours (k). The computational demands of this method are significantly lower than by *SVM*. The duration of one whole classification is on average 0.007 second.

The worst results of average success rate of classification showed the *DA* method (*Acc* = 59.9 %). The big advantage of *DA* is its high average sensitivity which is even to 85.7 %. No other classifier has so great sensitivity. *DA* also posses the lowest average specificity even to 76.2 %. However the finest result appears also in *DA* with *quadratic discriminant function* for group of *Chosen* features according to Wilcoxon rank sum test. The average success rate of

classification is in this case even to 97.4 %, specificity even to 97.9 % and sensitivity even to 93.9 %. The worst results of *DA* occurs for features that were normalized before classification - group *All-norm* and *Chosen-norm*. These two groups of features we can label as very weak features.

To conclude the best result for *Group1* comes from *DA* with *quadratic discriminant function*. *SVM* seems to be most reliable, but its sensitivity we consider as rather too low beside *DA* and *kNN*, nevertheless these two methods seem to be, especially for groups of normalized features, inappropriate. For this reason we will only use *DA* with the *quadratic discriminant function* for classifying sequences with frequency about 12.5 fps (*Group1*).

2 Group2

Tab. 10 - The results of the classification success rate (Acc), the sensitivity (SPC) and specificity (TPR) of the classification performed by using the given set of features on Group2 sequences. The best result is indicated in boldface.

Selected features / classifier	All			All-norm			Chosen			Chosen-norm			All-PCA			Chosen-PCA		
	Acc [%]	*SPC [%]*	*TPR [%]*	*Acc [%]*	*SPC [%]*	*TPR [%]*	*Acc [%]*	*SPC [%]*	*TPR [%]*	*Acc [%]*	*SPC [%]*	*TPR [%]*	*Acc [%]*	*SPC [%]*	*TPR [%]*	*Acc [%]*	*SPC [%]*	*TPR [%]*
DA - linear	98.5	99.7	77.0	87.3	95.0	55.0	98.6	99.8	77.0	87.3	95.0	55.0	95.2	96.5	79.7	84.7	87.3	85.5
DA - diaglinear	97.9	99.9	64.7	87.4	95.0	55.0	97.9	99.9	64.7	87.4	95.0	55.0	95.7	97.0	79.7	86.8	89.0	83.5
DA - quadratic	97.8	99.7	62.9	3,1	50.0	100.0	98.6	99.6	79.7	3,1	50.0	100.0	90.9	92.4	83.8	95.0	96.0	83.2
DA - diagquadratic	97.7	99.8	59.9	3,1	50.0	100.0	97.7	99.8	60.3	3,1	50.0	100.0	93.2	94.3	88.1	93.5	94.8	84.0
DA - mahalanobis	97.8	99.3	74.8	3,1	50.0	100.0	93.2	93.7	96.3	3,1	50.0	100.0	45.6	64.2	100.0	44.8	63.7	100.0
SVM - linear	98.6	99.8	79.7	12.7	55.0	95.7	98.7	99.9	81.2	12.7	55.0	95.7	98.2	99.6	75.9	98.5	99.7	75.6
SVM - quadratic	98.0	99.4	71.0	3.1	50.0	100.0	**99.2**	**99.8**	**88.1**	40.1	70.0	79.0	92.0	93.2	86.8	90.6	92.1	85.9
SVM - rbf	96.9	100.0	50.7	98.7	100.0	50.0	98.5	99.9	70.3	96.9	100.0	50.0	96.9	99.9	51.3	96.9	97.6	87.5
SVM - polynomial	97.5	98.9	67.9	22.0	60.0	90.0	98.8	99.7	84.2	78.0	90.0	60.0	93.9	95.2	81.5	95.9	96.6	88.3
kNN - k=1	99.1	99.7	83.0	12.7	55.0	95.7	99.0	99.7	83.1	12.7	55.0	95.7	89.2	91.2	82.4	92.5	93.6	88.0
kNN - k=3	98.8	99.8	80.5	12.7	55.0	95.7	98.8	99.8	80.5	12.7	55.0	95.7	89.4	91.6	78.6	92.6	94.2	78.0
kNN - k=5	98.8	99.8	80.3	12.7	55.0	95.7	98.8	99.8	80.3	12.7	55.0	95.7	91.7	93.2	82.0	93.7	94.8	86.6
kNN - k=8	98.7	99.7	80.4	12.7	55.0	95.7	98.6	99.7	79.8	12.7	55.0	95.7	92.5	94.0	82.0	93.9	94.9	87.3
kNN - k=10	98.8	99.7	80.9	12.7	55.0	95.7	98.8	99.7	80.9	12.7	55.0	95.7	92.7	94.1	84.7	94.0	95.1	86.6

Tab. 11 - Duration of each classifier setting for the given features of Group2 sequences.

Selected features/ classifier	All - t[s]	All-norm - t[s]	Chosen - t[s]	Chosen-norm - t[s]	All-PCA - t[s]	Chosen - PCA - t[s]
DA - linear	0.009	0.008	0.008	0.009	0.005	0.006
DA - diaglinear	0.004	0.004	0.004	0.004	0.003	0.003
DA - quadratic	0.005	0.004	0.003	0.003	0.002	0.001
DA - diagquadratic	0.003	0.002	0.002	0.002	0.002	0.001
DA - mahalanobis	0.005	0.004	0.003	0.003	0.001	0.001
SVM - linear	384.353	386.063	392.858	387.101	364.283	371.226
SVM - quadratic	492.427	496.692	407.346	403.022	415.001	391.410
SVM - rbf	252.693	252.391	380.581	380.308	318.488	362.666
SVM - polynomial	468.581	471.158	420.928	424.113	499.026	443.139
kNN - k=1	0.022	0.021	0.019	0.019	0.016	0.016
kNN - k=3	0.012	0.011	0.009	0.009	0.008	0.006
kNN - k=5	0.012	0.012	0.009	0.009	0.008	0.006
kNN - k=8	0.013	0.012	0.010	0.009	0.009	0.007
kNN - k=10	0.011	0.012	0.009	0.009	0.008	0.005

Discussion of achieved results - *Group2*

The best results for all groups of features are also achieved by *SVM* method. The average success rate of *SVM* for all groups of features is 79.7 % with specificity even to 89.6% and the sensitivity reaches only 76.9 %. The best *SVM* and also overall result was achieved by using *quadratic kernel* on the group of *Chosen* features. In this case the Acc come up to 99.2 %, specificity to 99.8 % and sensitivity to 88.1%. One thing we have to consider before using *SVM* is its duration of training process. In comparison to *DA* or *kNN* classifiers the time consumption is many times higher. The average time of learning *SVM* classifier is 398.6 seconds.

The second place regarding the average success rate of classification occupy the *DA* classifier with *Acc* = 71.7 %. The average specificity and sensitivity were for this case also the second best (*SPC* = 84.9 % and *TPR* = 80.2 %). The most reliable outcome is by applying *DA* classifier on set of *All* features with *mahalanobis discriminant function* (*Acc* = 93.2 %, *SPC* = 93.7 % and *TPR* = 96.3 %). The computational demands of this method are significantly lower than by *SVM* and two times smaller than by *kNN*. The duration of one whole classification is on average 0.004 second.

The worst results of average success rate of classification showed the *kNN* method (*Acc* = 67.9 %). The big advantage of *kNN* is its high average sensitivity which is even to 86.8 %. No other classifier has so great sensitivity. However *kNN* also posses the lowest average specificity even to 82.2 %. The worst results of *kNN* occurs for features that were normalized before classification - group *All-norm* and *Chosen-norm*. These two groups of features strongly affect average values of success rate, specificity, sensitivity.

To conclude the best result for *Group2* comes from *SVM* with *quadratic kernel*. *SVM* seems to be most accurate, however its sensitivity is often too low. On the other side *DA* and *kNN* do not classify properly especially for groups of normalized features and their accuracy and specificity do not reach so great results as SVM. Despite the higher duration of training we pick *SVM* with the *quadratic kernel* for classifying sequences with frequency about 25 fps (*Group2*).

3 *Group3*

Tab. 12 - The results of the classification success rate (Acc), the sensitivity (SPC) and specificity (TPR) of the classification performed by using the given set of features on Group3 sequences. The best result is indicated in boldface.

Selected features / classifier	All			All-norm			Chosen			Chosen-norm			All-PCA			Chosen-PCA		
	Acc [%]	SPC [%]	TPR [%]	Acc [%]	SPC [%]	TPR [%]	Acc [%]	SPC [%]	TPR [%]	Acc [%]	SPC [%]	TPR [%]	Acc [%]	SPC [%]	TPR [%]	Acc [%]	SPC [%]	TPR [%]
DA - linear	99.3	99.9	73.8	20.8	60.0	94.2	99.5	99.9	78.8	99.5	99.9	78.4	80.9	84.0	82.7	76.2	80.9	81.0
DA - diaglinear	85.2	87.7	71.7	19.5	58.9	96.7	73.2	78.8	88.4	75.3	80.3	81.0	80.8	84.0	82.7	76.2	80.8	81.0
DA - quadratic	-	-	-	99.0	100.0	50.0	-	-	-	-	-	-	88.4	89.8	82.9	73.1	78.9	81.0
DA - diagquadratic	75.1	80.1	81.0	5.6	51.5	100.0	68.5	76.0	88.4	68.0	75.7	88.4	81.4	84.4	82.0	74.6	79.8	81.0
DA - mahalanobis	-	-	-	99.0	100.0	50.0	-	-	-	-	-	-	93.8	94.5	76.5	76.8	81.3	76.0
SVM - linear	99.4	100.0	69.2	20.7	60.0	92.5	99.3	99.9	68.4	99.2	100.0	62.2	99.4	100.0	69.2	99.0	100.0	50.0
SVM - quadratic	99.3	99.8	69.2	23.9	61.0	89.2	99.5	100.0	78.9	99.1	99.9	62.0	99.3	98.8	69.2	99.0	100.0	55.0
SVM - rbf	99.0	100.0	50.0	99.0	100.0	50.0	99.0	100.0	50.0	99.0	100.0	50.0	99.0	100.0	50.0	99.0	100.0	50.0
SVM - polynomial	98.8	99.4	65.7	28.4	63.1	89.2	99.1	99.6	69.3	95.8	96.7	69.7	98.8	99.4	65.7	99.1	99.9	63.4
kNN - k=1	99.3	99.8	77.2	82.6	86.4	71.4	**99.3**	**99.7**	**82.7**	99.2	99.7	77.7	99.1	99.6	79.3	98.9	99.6	57.7
kNN - k=3	99.4	99.9	69.2	90.1	91.9	67.2	99.3	99.9	61.7	99.2	99.8	62.2	99.3	100.0	66.5	99.0	99.9	52.7
kNN - k=5	99.3	100.0	66.8	92.8	94.3	65.2	99.1	100.0	55.3	99.1	100.0	54.4	99.1	100.0	52.7	99.1	100.0	52.7
kNN - k=8	99.2	100.0	56.9	95.6	96.6	62.7	99.0	100.0	51.7	99.0	100.0	51.7	99.0	100.0	50.0	99.0	100.0	50.0
kNN - k=10	99.0	100.0	51.7	99.1	100.0	58.4	99.0	100.0	50.0	99.0	100.0	50.0	99.0	100.0	50.0	99.0	100.0	50.0

Tab. 13 - Duration of each classifier setting for the given features of Group3 sequences.

Selected features/ classifier	All - t[s]	All-norm - t[s]	Chosen - t[s]	Chosen-norm - t[s]	All-PCA - t[s]	Chosen - PCA - t[s]
DA - linear	0.010	0.010	0.008	0.008	0.008	0.007
DA - diaglinear	0.006	0.005	0.004	0.004	0.003	0.004
DA - quadratic	-	0.007	-	-	0.002	0.003
DA - diagquadratic	0.009	0.003	0.006	0.006	0.002	0.002
DA - mahalanobis	-	0.006	-	-	0.002	0.002
SVM - linear	2423.800	2543.900	2485.300	2506.900	2566.900	4003.200
SVM - quadratic	3258.400	3412.000	2763.300	2812.300	3422.100	2497.600
SVM - rbf	1352.000	1419.900	2313.000	2357.400	1420.500	2438.500
SVM - polynomial	3109.700	3264.400	2982.200	3049.300	3269.800	3309.900
kNN - k=1	0.036	0.038	0.041	0.033	0.016	0.016
kNN - k=3	0.026	0.028	0.026	0.022	0.009	0.007
kNN - k=5	0.026	0.028	0.028	0.021	0.009	0.007
kNN - k=8	0.027	0.031	0.028	0.022	0.009	0.007
kNN - k=10	0.027	0.028	0.028	0.021	0.010	0.006

Discussion of achieved results - *Group3*

The best results for all groups of features are in this case achieved by *kNN* classifier. The average success rate of *kNN* for all groups of features is 97.9 % with specificity even to 98.9 % and the sensitivity reaches only 60.2 % (the worse result in comparison to other classifiers). The best overall result was achieved by using *k = 1 nearest neighbours* on the group of *Chosen* features. In this case the Acc come up to 99.3 %, specificity to 99.7 % and sensitivity to 82.7 %. The average time of learning *kNN* classifier is 0.022 second.

The second place regarding the average success rate of classification occupy the *SVM* classifier with *Acc* = 89.6 %. The average specificity and sensitivity were for this case also the second best (*SPC* = 95.0 % and *TPR* = 64.7 %). The most reliable outcome is by applying *SVM* classifier on set of *Chosen* features with *quadratic kernel* (*Acc* = 99.5 %, *SPC* = 100 % and *TPR* = 78.9 %). The computational demands of this method are significantly higher than by *kNN* and *DA*. The duration of one whole classification is on average 2714.26 seconds.

The worst results of average success rate of classification showed the *DA* method (*Acc* = 74.6 %). The big advantage of *DA* is its highest average sensitivity which is even to 80.3 % and the lowest computational time. However *DA* also posses the lowest average specificity even to 82.8 %. Best results comes up for *Chosen* features by applying *linear discriminant function* (*Acc* = 99.5 %, *SPC* = 99.9 % and *TPR* = 78.8 %), on the other hand the worst results of *DA* occurs for features that were normalized before classification - group *All-norm* and *Chosen-norm*. These two groups of features strongly affect average values of success rate, specificity, sensitivity. For some groups of features (*All*, *Chosen*, *Chosen-norm*), it was not possible to carry out a *DA* because we did not

fulfil the positive definite condition of covariance matrix. That means that the number of features was too big.

To conclude the best result for *Group3* comes from *knn* with *k = 1 nearest neighbours* on *Chosen* group of features. Classification by *kNN* seems to be most accurate, however there is no possibility to calculate posterior probabilities, second best classification was achieved by time consuming *SVM*. With respect to these circumstances we decided to apply the *DA* classifier with *linear discriminant function*.

7.2.5 Conclusion of achieved results

As we already mentioned the best results for *Group1* sequences were reached by *linear discriminant analysis* with *quadratic discriminant function*. The highest success rate was achieved in features, which were statistically tested and chosen by Wilcoxon rank sum test. A slightly weaker results came out for features groups - *All*, *All-PCA* and *Chosen-PCA*. At this point we have to say that features provided by *PCA* did not fulfil our expectations. Lastly the worst results reach the feature groups which were normalized - *All-norm*, *Chosen-norm*.

Very similar results are also for the remaining groups of sequences (*Group2*, *Group3*). They differ only in the best result. For the *Group2* we consider as a best the result coming from *SVM* with the *quadratic kernel* and for the *Group3* the best result was proved by *kNN* classifier with *k = 1* nearest neighbour. In mentioned *Group2* and *Group3* it is worth to take in account the computational requirements of *SVM,* which are quite high in comparison to *DA* or *kNN*.

It is important to be noted that when using the same methods for the classification of another retinal sequence we cannot confidently expect the same results, since the features can be slightly different, because of dissimilar anatomic properties of subjects retinas and different acquisition parameters.

7.3 Quality estimation

Quality of single frames as well as the whole sequences is discussed already in the previous chapter (*Chap. 7.2.4*). Every group of sequences is assessed separately. The advantage of some Matlab functions is among others also the calculation of posterior probabilities (e.g. *knnclassify* offers this option). For *SVM* method we gained function *svmdecision.m*, which calculates these posterior probabilities as well. The reciprocal values of posterior probabilities are then normalized into a vector in range from 0 to 1. Therefore we get for every frame one value, which represent its quality.

7.4 Evaluation of registration

Another essential task of this master's thesis is to incorporate suggested method of sequence quality evaluation into a current approach for sequence registration. The registration process is made in two steps, which are described in *Chap. 5*. In this part we firstly discuss two similarity measures for registration of images (*Chap. 7.4.1*), then we deal with choosing the most suitable frame for registration process (*Chap. 7.4.2*), subsequently the effect on the registration quality according to a selected reference image (*Chap. 7.4.3*) is examined.

7.4.1 Similarity measures

An ordinary task of image analysis is to compare how well aligned two images might be. Similarity measures produce a quantitative assessment of resemblance between reference and examined frame. There are a lot of metrics which evaluate the image similarity, but there is not one general image similarity metric which can be used for all images, respectively modalities. An overview of the most usual metrics used in medical image processing is presented in [36]. The common examples of image similarity measures contain *cross-correlation*, *sum of squared differences, mutual information* and *ratio image uniformity*. *Cross-correlation*, *sum of squared differences* and *ratio image uniformity* are mainly used for registration of images taken by the same modality. *Mutual information* and its normalized version are the most favour among image similarity measures that are applied for registration of multimodal images [35, 36, 37].

A proper measure should satisfy some requirements as [35, 36]:

1. **Reflexivity** - the measure output of two identical image may be equal to 1
2. **Symmetry** - independency on the order from which input frames are considered
3. **Reaction on noise** - the metric should be so less dependent on noise as it is possible
4. **Reaction to enlightening and darkening** - the measure output should be high in case of that one image is enlightened or darkened by a constant value

The similarity measures introduced throughout the last years is diverse, each of them has its own advantages and disadvantages. In the following part two similarity measures, which are also implemented, are described: *sum of squared differences* and *normalized cross-correlation*.

1 *SSD (Sum of Squared Differences) measure*

Sum of squared differences measure belongs to intensity based measures of similarity. *SSD* is calculated as squared difference between each pixel in the reference image *X* and the corresponding pixel in the examined image *Y*. For our case of retinal frames we can define *SSD* as [38, 39]:

$$S_{SSD}(X,Y) = \sum_{i=1}^{N} \sum_{j=1}^{M} (X_{i,j} - Y_{i,j})^2, \qquad\qquad (7.2)$$

where S_{SSD} is single value of *SSD* measure, *X* represents reference frame, *Y* corresponds to frame which similarity is examined, then $i \in 1, 2 \ldots N, j \in 1, 2 \ldots M$ are spatial indices, that means that *N* and *M* also represents the height and width of our retinal frames.

The prerequisites for *SSD* accuracy are that the intensity values of the two frames are in the same range, the frames are acquired from the same modality and the difference between frames *X*, *Y* has the Gaussian distribution [38]. The smaller value of *SSD* metrics means the higher similarity between the reference and examined frame. *SSD* provides simple, effective and fast way to measure the similarity of two images, on the other side it should be inconvenient when the changes of lighting or shape changes (of region of interest) appear in the scene [36, 38, 39].

2 *Normalized cross-correlation measure (NCC)*

Another common metric to evaluate the similarity degree of two signals (in our case sequences, respectively single frames) is *cross-correlation. Cross-correlation* has same properties as *SSD*. That means it is quite simple to compute, it represents reliable measure for pixel intensities which are linearly related, but there is negative effect when lighting changes appears [38, 40, 41]. For overcoming the most of disadvantages of *cross-correlation* the normalization of image can be used. The *normalized cross-correlation* coefficient γ of two images is thus defined as [41, 42]:

$$\gamma = \frac{\sum_{x,y}\left(f(x,y)-\bar{f}_{u,v}\right)\cdot\left(t(x-u,y-v)-\bar{t}\right)}{\sqrt{\sum_{x,y}\left(f(x,y)-\bar{f}_{u,v}\right)^2\cdot\sum_{x,y}\left(t(x-u,y-v)-\bar{t}\right)^2}},\text{''}\qquad(7.3)$$

where *x*, *y* are spatial indices of the image *f* which size is $M_x \cdot M_y$. The reference image is represent as *t*, its shift in *x* direction is *u* and shift in *y* direction is *v*. The mean value of *f*(*x*, *y*) is calculated [42]:

$$\bar{f}_{u,v} = \frac{1}{M_x\cdot N_y}\sum_{x=u}^{u+M_x-1}\sum_{y=v}^{v+N_y}f(x,y).\qquad(7.4)$$

The notation for the mean value of reference image is calculated analogously. The result can range in value from -1 to 1. When γ is even to 1, then the two time-series or sequences are identical, $\gamma = 0$ means that the sequences are entirely uncorrelated and $\gamma = -1$ represent the option when sequence frames pixels have opposite signs. The greater coefficient means good similarity match between sequences. Misalignment between the sequence frames results into small *normalized cross-correlation* measure values. The greatest advantage of this metric is that it is less sensitive on illumination changes in the two compared images than the standard *cross-correlation* [40].

Normalized cross-correlation is in Matlab implemented by the *normxcorr2* function. It computes 2D version of *normalized cross-correlation* that perfectly suits for our retinal sequences. Following command executes the computation:

```
C = normxcorr2(template, A),
```

where *template* represents the selected reference frame, variable *A* is examined frame from our retinal sequence and a result we get correlation coefficient *C* which evaluates the similarity of two sequences.

7.4.2 Selection of appropriate images for registration

The process of establishing the common reference frame represents quite demanding task. Based on previously introduced measures (*Chap. 7.4.1*) we are trying to find out the most suitable reference image in order to have the smallest misalignment with other images from retinal sequence.

For every sequence five reference images for registration process were tested in the following order:

1. Best quality image,
2. Median quality image,
3. Mean quality image,
4. Random image - twice.

The sequence's quality vectors are estimated through principals introduced in *Chap. 7.3* so that we are able to determine these first three reference (fixed) images. The results are divided into relevant groups according to given sequence frame rate. To assess the appropriateness of five investigated frames to be reference image for registration we need to consider both measures (*SSD* and *normalized cross-correlation*). The examined frames, mentioned above, are in *Tab. 14-16* marked subsequently: *Best* - best quality image of given sequence, *Med* - median quality frame, *Mean* - mean quality frame and *Rnd1*, *Rnd2* - randomly chosen images. The best results for every sequence are marked in boldface. The final numbers presented in tables (*Tab. 14-19*) are calculated as a mean value of given metric of tested sequence. For this reason, the numbers are even provided with standard deviations to see how far the values are spread above and below the mean.

1 *Group1*

Tab. 14 - Results of SSD and normalized cross-correlation measure after applying different frames for Group1 sequences registration.

Subject	N1				
Ref. frame	Best	Med	Mean	Rnd1	Rnd2
SSD [-]	239.8 ± 92.8	217.1 ± 73.1	194.8 ± 93.7	**173.3 ± 84.5**	241.3 ± 92.0
NCC [-]	0.984 ± 0.004	0.987 ± 0.004	0.988 ± 0.004	**0.990 ± 0.003**	0.984 ± 0.004
Subject	N2				
Ref. frame	Best	Med	Mean	Rnd1	Rnd2
SSD [-]	1187.4 ± 579.0	1357.5 ± 572.4	**928.3 ± 383.3**	1324.5 ± 547.4	1284.9 ± 716.0
NCC [-]	0.952 ± 0.01	0.954 ± 0.008	**0.958 ± 0.008**	0.952 ± 0.009	0.948 ± 0.013
Subject	N3				

Ref. frame	Best	Med	Mean	Rnd1	Rnd2
SSD [-]	222.1 ± 77.8	**191.9 ± 75.3**	241.5 ± 80.9	210.2 ± 69.6	272.6 ± 65.9
NCC [-]	0.955 ± 0.008	0.958 ± 0.009	0.946 ± 0.009	**0.959 ± 0.007**	0.941 ± 0.008
Subject	N4				
Ref. frame	Best	Med	Mean	Rnd1	Rnd2
SSD [-]	2822.8 ± 1378.0	2280.1 ± 1212.8	**1771.7 ± 927.8**	1947.2 ± 1086.9	3704.9 ± 1642.0
NCC [-]	0.955 ± 0.021	0.953 ± 0.029	0.960 ± 0.012	**0.965 ± 0.014**	0.954 ± 0.018
Subject	N5				
Ref. frame	Best	Med	Mean	Rnd1	Rnd2
SSD [-]	275.7 ± 117.1	374.2 ± 119.6	302.8 ± 137.4	**221.7 ± 101.8**	257.7 ± 84.7
NCC [-]	0.989 ± 0.002	0.989 ± 0.001	0.987 ± 0.002	**0.990 ± 0.001**	0.987 ± 0.001

To evaluate the result we have to know that the lowest value of *SSD* represents the better result of registration, on the other hand with applying *normalized cross-correlation* we are looking for the highest value. According to this rule, we see that for *SSD* the randomly chosen images (*Rnd1* or *Rnd2*) delivered the best results in three sequences. However mean (*Mean*) quality images represent in our opinion the most reliable reference frames for the registration process, because their results are the most stable and sometimes the other tested reference images (*Best*, *Rnd1* and *Rnd2*) lead to degradation of sequence registration quality. The same result is also proved by *normalized cross-correlation*. The worst outcome of registration is when using the best quality image (*Best*) as a reference frame. Examples of used reference frames shown in *Fig. 23*.
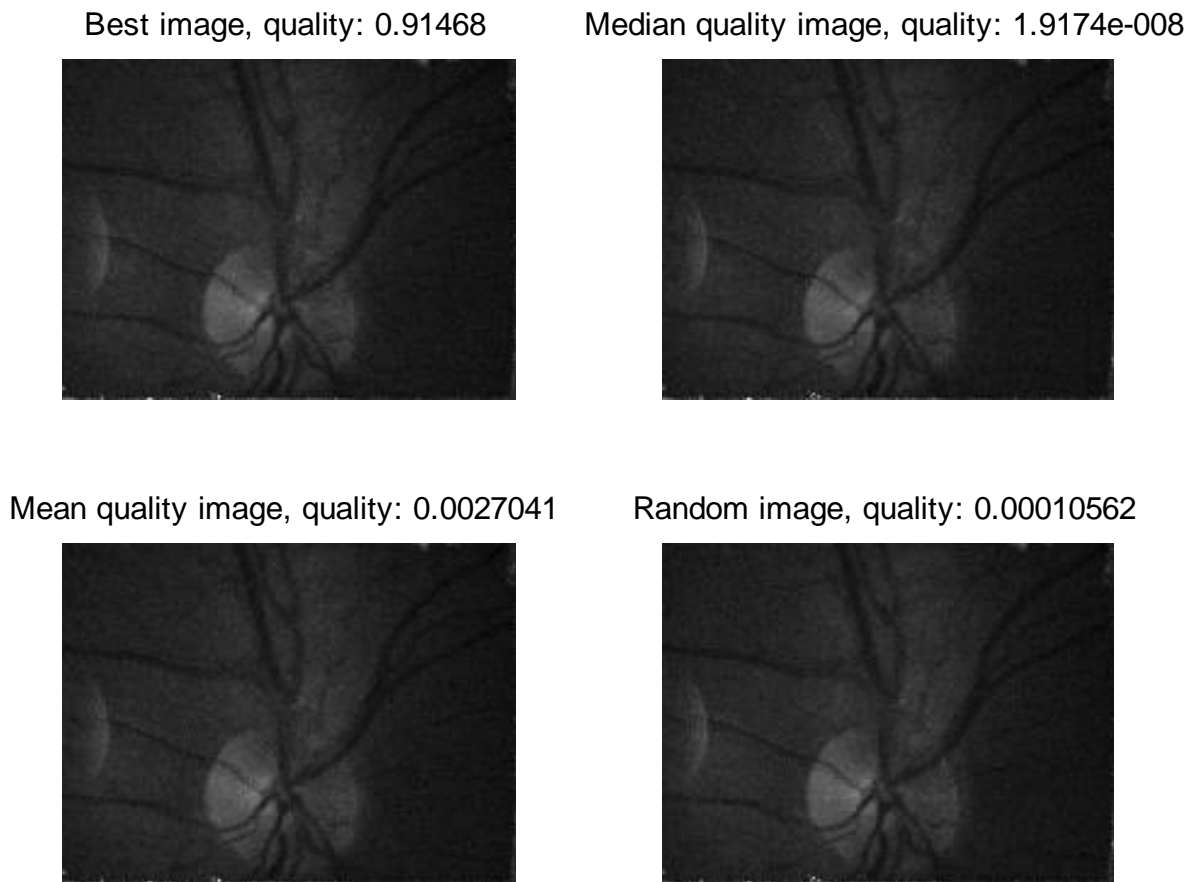
Best image, quality: 0.91468

Median quality image, quality: 1.9174e-008

Mean quality image, quality: 0.0027041

Random image, quality: 0.00010562

Fig. 23 - Illustration of used references images with corresponding quality for subject N3 - Group1.
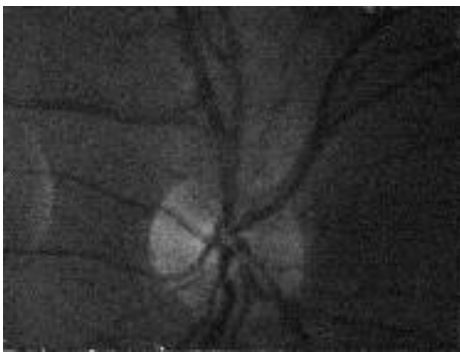
2 Group2

Tab. 15 - Results of SSD and normalized cross-correlation measure after applying different frames for Group2 sequences registration.

Subject	N1				
Ref. frame	Best	Med	Mean	Rnd1	Rnd2
SSD [-]	368.6 ± 54.7	391.1 ± 63.0	322.3 ± 41.7	339.6 ± 64.1	**286.1 ± 47.8**
NCC [-]	0.975 ± 0.002	0.973 ± 0.002	0.976 ± 0.002	0.976 ± 0.002	**0.979 ± 0.002**
Subject	N2				
Ref. frame	Best	Med	Mean	Rnd1	Rnd2
SSD [-]	473.0 ± 142.7	**396.3 ± 120.8**	**396.3 ± 120.8**	456.2 ± 147.3	403.5 ± 104.8
NCC [-]	0.917 ± 0.014	**0.930 ± 0.009**	**0.930 ± 0.009**	0.916 ± 0.012	0.926 ± 0.008
Subject	N3				
Ref. frame	Best	Med	Mean	Rnd1	Rnd2
SSD [-]	504.1 ± 148.2	577.3 ± 128.9	588.7 ± 157.7	532.0 ± 168.9	**472.9 ± 133.1**
NCC [-]	0.922 ± 0.009	0.909 ± 0.009	0.918 ± 0.008	0.920 ± 0.013	**0.935 ± 0.010**
Subject	N4				
Ref. frame	Best	Med	Mean	Rnd1	Rnd2

SSD [-]	859.6 ± 540.6	553.4 ± 420.6	549.9 ± 445.4	**437.8 ± 428.9**	597.5 ± 563.2
NCC [-]	0.851 ± 0.048	0.881 ± 0.044	0.848 ± 0.043	**0.891 ± 0.044**	0.886 ± 0.049

For *Group2* sequences it seems according to *SSD* and *NCC* that the *Med* and *Mean* reference frames return the most trustworthy results, however the best results come out from randomly chosen images (*Rnd1* or *Rnd2*). The poorest quality or registration alignment is here also for the *Best* quality image. Samples of the reference frames applied for registration process are depicted in *Fig. 24*.

Best image, quality: 0.97606   Median quality image, quality: 0.96293
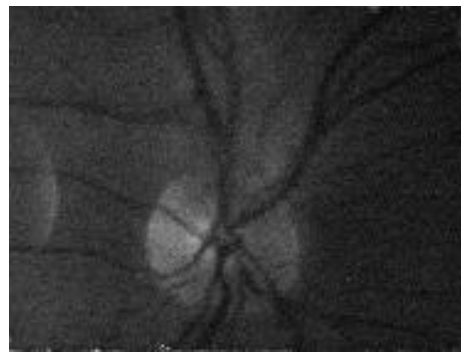
Mean quality image, quality: 0.95356   Random image, quality: 0.96065
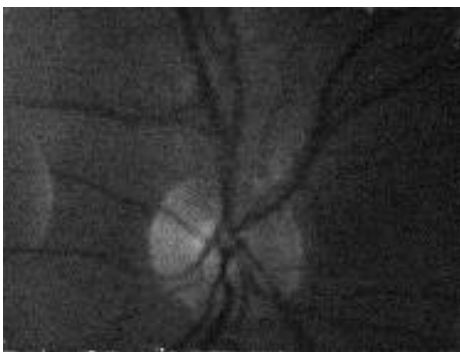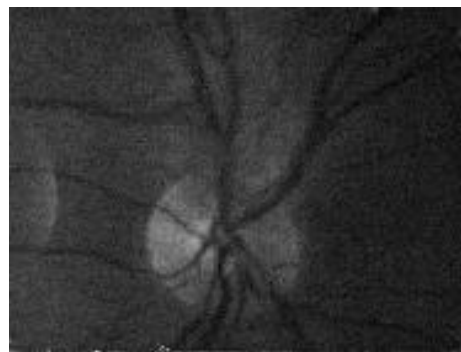
Fig. 24 - Illustration of used references images with corresponding quality for subject N3 -
Group2.

3 *Group3*

Tab. 16 - Results of SSD and normalized cross-correlation measure after applying different frames for Group2 sequences registration.

Subject	N1				
Ref. frame	Best	Med	Mean	Rnd1	Rnd2
SSD [-]	233.1 ± 50.3	289.5 ± 85.1	259.3 ± 52.2	**226.0 ± 62.3**	267.8 ± 40.6
NCC [-]	0.940 ± 0.008	0.947 ± 0.006	0.932 ± 0.008	**0.954 ± 0.007**	0.936 ± 0.007
Subject	N2				
Ref. frame	Best	Med	Mean	Rnd1	Rnd2
SSD [-]	282.2 ± 36.7	-	**232.1 ± 35.4**	252.4 ± 33.4	254.9 ± 36.4
NCC [-]	0.586 ± 0.025	-	**0.637 ± 0.030**	0.609 ± 0.027	0.614 ± 0.027
Subject	N3				
Ref. frame	Best	Med	Mean	Rnd1	Rnd2
SSD [-]	273.1 ± 39.6	**239.9 ± 29.3**	258.7 ± 33.5	320.9 ± 31.2	261.8 ± 29.8
NCC [-]	0.767 ± 0.014	**0.801 ± 0.015**	0.772 ± 0.014	0.747 ± 0.015	0.781 ± 0.015

In *Group3* we have only three representative sequences. The best results are different for every subject. For the first subject the randomly selected image (*Rnd1*) for registration represents the most excellent outcome, however for subject N2 sequence comes out the *Mean* image as the best one and finally for subject N3 the median (*Med*) provides the best result. The results in this group are the most variable nevertheless the most balanced results within the whole group shows the *Mean* image.

While registration of subject N2 sequence with a median (*Med*) frame a problem in registration problem occurred (marked with "-" in the *Tab. 16*), so that the metrics was not calculated.
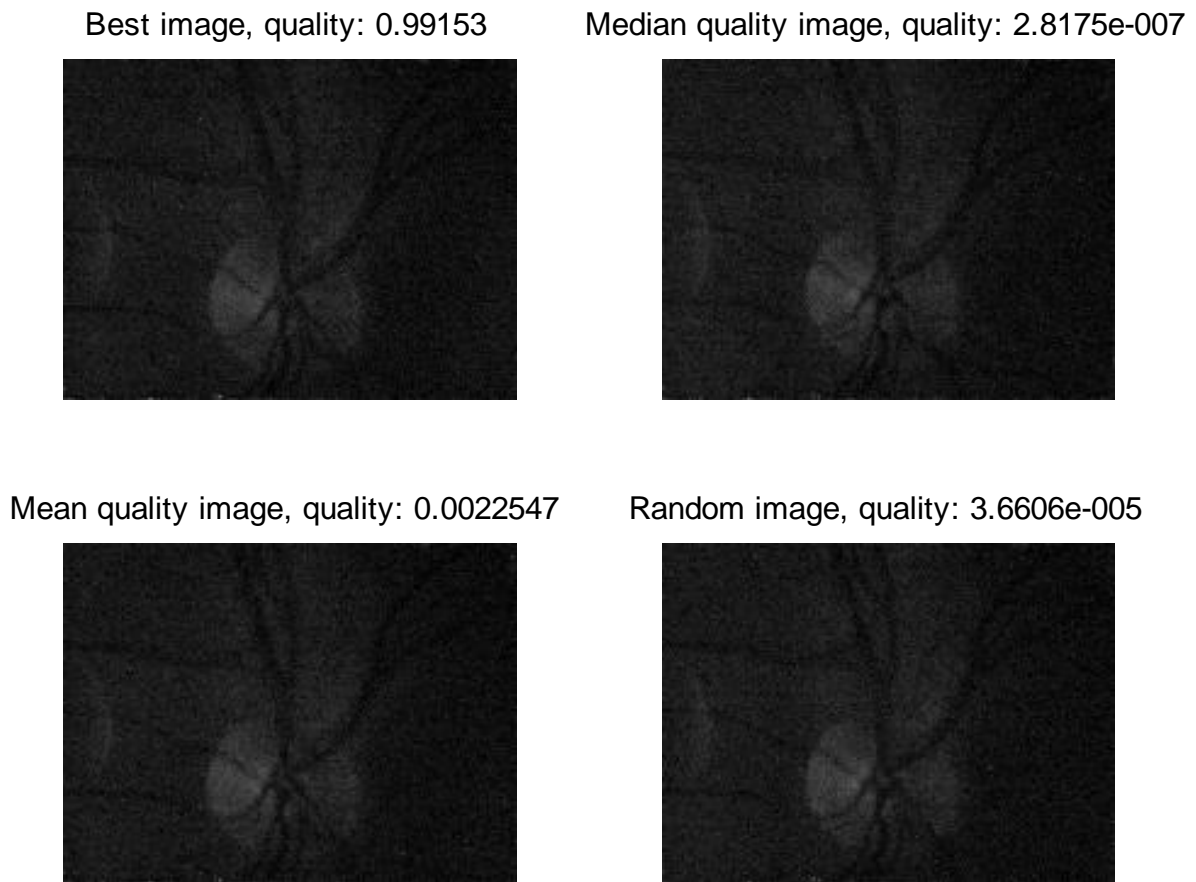
Fig. 25 - *Illustration of used references images with corresponding quality for subject N3 - Group3.*

7.4.3 Evaluation of registration quality

After we select appropriate reference image for the process of registration, we are able to evaluate what is the influence of a frame classification, removing the bad images and chosen reference image on registration quality. Due to the stability of results we choose the *mean quality image* as the most appropriate one to registration process of every sequence. However the user is able to select another one as reference image in our software. The quality of registration is also assessed by means of *SSD* and *normalized cross-correlation*. We compare the quality before using our software and after using software. These two occurrences are in *Tab.17-19* labelled as *Before* and *After*. Repeatedly we discuss every group of sequence (*Group1*, *Group2* and *Group3*) separately.

1 *Group1*

Tab. 17 - *Registration results before and after introduction of our classifying software - Group1*

Subject	N1		N2	
Ref. frame	Before	After	Before	After
SSD [-]	235.6 ± 82.2	194.8 ± 93.7	1550.4 ± 688.3	928.3 ± 383.3
NCC [-]	0.986 ± 0.004	0.988 ± 0.004	0.947 ± 0.012	0.958 ± 0.008

Subject	N3		N4	
Ref. frame	Before	After	Before	After
SSD [-]	2017 ± 17607	241.5 ± 80.9	2324.9 ± 2570.9	1771.7 ± 927.9
NCC [-]	0.955 ± 0.035	0.946 ± 0.009	0.956 ± 0.027	0.960 ± 0.012
Subject	N5			
Ref. frame	Before	After		
SSD [-]	321.9 ± 94.0	302.8 ± 137.4		
NCC [-]	0.985 ± 0.005	0.987 ± 0.002		

Within *Group1* sequences we can clearly see that almost for every sequence our software proved an improvement of registration quality except subject N3, where the *NCC* coefficient decreased by 0.91 %. On average the value of *SSD* was reduced by 35.30 % and the value of *normalized cross-correlation* measure increases on average by 0.25 %.

2 Group2

Tab. 18 - Registration results before and after introduction of our classifying software - Group2

Subject	N1		N2	
Ref. frame	Before	After	Before	After
SSD [-]	406.8 ± 67.6	322.3 ± 41.7	448.5 ± 118.9	396.3 ± 120.8
NCC [-]	0.968 ± 0.003	0.976 ± 0.002	0.930 ± 0.010	0.930 ± 0.008
Subject	N3		N4	
Ref. frame	Before	After	Before	After
SSD [-]	1873.6 ± 14070.0	588.7 ± 157.7	950.0 ± 4272.1	549.9 ± 445.4
NCC [-]	0.888 ± 0.031	0.918 ± 0.008	0.875 ± 0.049	0.848 ± 0.043

Regarding the *Group2* sequences, there also occurs the same effect as in previous group of sequences. We also observe the numerical improvement of registration quality apart from subject N4 where the *NCC* coefficient was reduces by 3.11 %. On average the *SSD* measure decreases by 35.78 % and the *normalized cross-correlation* raised only by 0.27 % on average.

3 Group3

Tab. 19 - Registration results before and after introduction of our classifying software - Group3

Subject	N1		N2	
Ref. frame	Before	After	Before	After
SSD [-]	353.5 ± 911.0	259.3 ± 52.2	302.9 ± 33.4	232.1 ± 35.4
NCC [-]	0.927 ± 0.048	0.932 ± 0.008	0.595 ± 0.028	0.637 ± 0.030
Subject	N3			
Ref. frame	Before	After		
SSD [-]	218.8 ± 30.4	258.7 ± 33.5		
NCC [-]	0.800 ± 0.014	0.772 ± 0.014		

In two out of three sequences from *Group3* better result was achieved. *SSD* measure fell on average by 10.57 % and *normalized cross-correlation* improved the registration results by 1.37 %. For N3 subject's sequence was not twice our software helpful, on the contrary it led to deterioration. Same effect occurred once also for subject N4.

4 *Discussion of achieved results*

To compare registration quality we employ two suitable metrics - *sum of squared differences measure* and the *normalized cross-correlation measure*. The success of our software is dependent on the given sequence quality (e.g. its contrast), on the sequence quality distribution, quantity of distorted frames and finally on the quality of frame which is replacing the distorted one. All these aspects influence the quality of the registration process. More detailed results for each group of sequences (*Group1*, *Group2*, *Group3*) are listed in relevant subchapters.

After the inclusion of our software into existing one, the improvement of registration results, occurred in the most sequences (83.33 %). In the rest (subject N3 in *Group1*, *Group3* and subject N4 in *Group3*) we achieved slightly worse outcomes especially in *normalized cross-correlation* coefficient. This phenomenon is probably caused by inappropriate choosing of image which is replacing the distorted one. That means that there is certain possibility there is more suitable reference frame and that the quality measure fails to reveal "something" what is important when choosing an appropriate reference frame for registration. Other issue which can cause inaccuracies are the black rims of frames. These rims are added to frames after registration process because of shift compensation. This problem we tried to eliminate by counting the metrics (*SSD* and *NCC*) without these peripheral pixels.

On average (for all sequences together) we managed to decrease the values of *SSD* by 25.58 % and correlation coefficient of *NCC* was increased by 1.80 %. In 66.67 % of introducing our software, we achieved also a reduction of standard deviations, this is very likely caused by replacement of distorted, blurred images. From this results we can see that the SSD measure is more appropriate to evaluate the registration quality of retinal sequences.

8 CONCLUSION

This master's thesis deals with processing of retinal image captured by fundus camera. Theoretical part includes description of retinal sequence features, acquisition parameters and registration approaches. The main section consists of image quality assessment metrics involving *full-reference image quality metrics*, *reduced-reference image quality metrics* and finally *no-reference image quality metrics*.

The proposed method of *image quality assessment* is based on evaluating an image statistics, which are changing in presence of distortion. *Entropy* of each frame is extracted in spatial domain, m*ean subtracted contrast normalized (MSCN)* coefficients and *log-derivative features* are extracted in spatial and frequency domain.

In the practical part the examined retinal sequences were firstly divided into three groups - *Group1*, *Group2*, *Group3* according to their acquisition parameters (exposure time, respectively frame rate). Then we created a function which realizes the proposed method and extract 33 features from every input sequence. These extracted features were statistically tested to verify their strength of distinguishing the distorted images from undistorted. Also *principal component analysis* (*PCA*) was employed to get another set of features. In total we had 6 sets of features for every group of sequences on which we tested three classifiers - *Discriminant function analysis* (*DA*), *Support vector machine* (*SVM*) and *k-nearest neighbour algorithm* (*kNN*). For each set of features we computed the average success rate of classification (*Acc*), *specificity* (*SPC*) and *sensitivity* (*TPR*) of used classifier. For the *Group1* the best result delivered *DA* with q*uadratic discriminant function* on *Chosen* features by statistical analysis (Wilcoxon rank sum test). For the *Group2* the best result was achieved also on *Chosen* features but by applying the *SVM* classifier with *quadratic kernel*. Finally for *Group3* sequences the best outcome appears by using *kNN* classifier with $k = 1$ neighbours, however we were not able to calculate a posterior probability of belonging to distorted images or not. Therefore a *DA* classifier with *linear discriminant function* was selected to *Group3* sequences. For verification of classifiers was for every group of sequences used a *cross-validation* approach. After every single frame is binary classified to a class, the quality of frames is calculated from the posterior probabilities (see *Chap. 7.3*). If the classification of sequence is not trustful (e.g. there are a lot frames classified as distorted), then a function to modify the results of classification is automatically launched. This function allows user to influence the results by making a decision from which quality the frames are distorted. The features of examined sequence are added to trained data, this step makes the classifier more robust to classify upcoming sequences. Implementation of this function was needed because of different contrast in captured sequences that is leading to slightly dissimilar values of features. This dissimilarities are resulting into lower success rate of classification.

Our goal was also to remove (replace) the distorted images of given sequence. In this work the distorted images are replaced by the closest image from sequence which is considered as a good, undistorted image.

The very last chapter is dedicated to registration of retinal sequences. We introduced two similarity metrics to evaluate the quality of registration process. First of them is called *sum of squared differences* metric (*SSD*) and the second one is *normalized cross-correlation* (*NCC*). We also determined which frames of sequence are and are not to suitable to be a reference image for registration. For this purpose five different frames were examined: Best quality image, median quality image, mean quality image and two randomly chosen images. This method of evaluation image sequences was incorporated into an existing registration approach. It was found out that the most stable results proves the mean quality image, on the other hand the best quality image shows most frequently the worst results. Nevertheless many times happened that a randomly chosen image proved a better result of registration quality. This phenomenon is probably caused by inaccurate measure of quality which does not reveal everything essential to determine the correct reference image or by the fact that a lot of images in sequence have good quality (approximately 95 %) so that if one of these good quality frames is selected as a reference image then the results of metrics will differ a little. The effect of choosing the reference frame on the quality of registration was also tested. Thus we compare the values of *SSD* and *NCC* before and after introducing our software of choosing the appropriate reference frame. Before incorporation of new software the reference image was the first from the sequence, in our case the mean quality image was selected to be most suitable for the registration. In 83.33 % of sequences the registration results were improved. For all sequences together the values of *SSD* metric were decreased by 25.58 % and the *normalized cross-correlation* coefficient was expanded by 1.80 %.

In conclusion we successfully managed to choose appropriate set of features and the most robust classifier to binary classify retinal data to group of distorted and undistorted images. We were able to calculate the quality of every frame in input sequence. Based on the frame quality, the most appropriate frame for image registration is chosen and finally the registration quality is estimated by *SSD* and *NCC* metrics.

APPENDIX

Box-whisker plots

Box-whisker plots for Group1 sequences
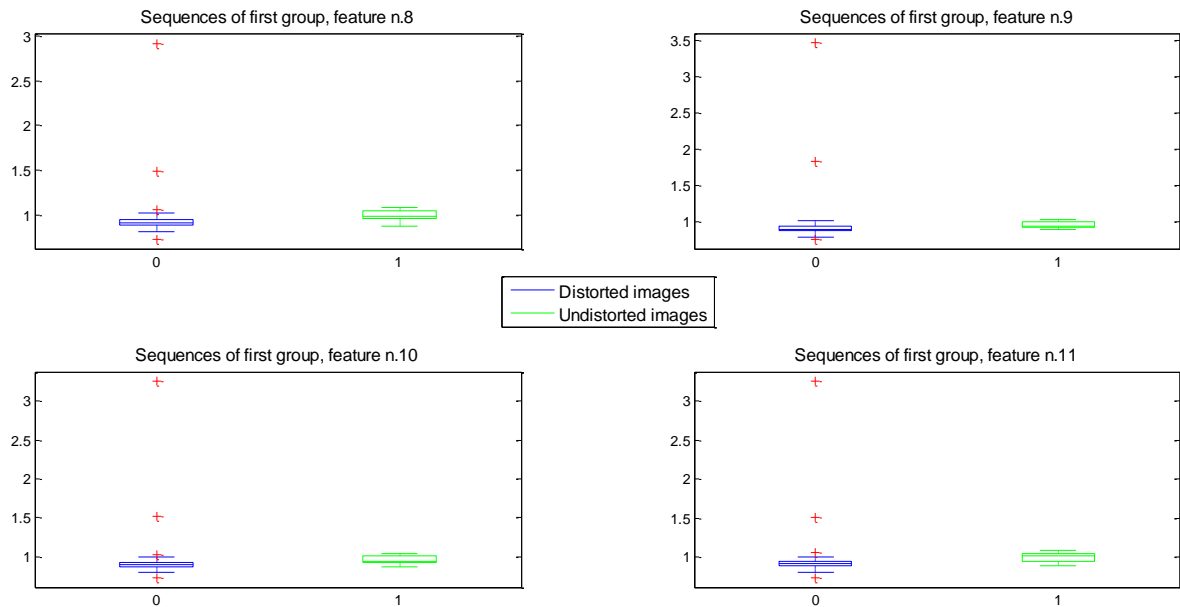
Fig. 26 - Box-whisker plots for Group1 sequences, feature n. 8-11.
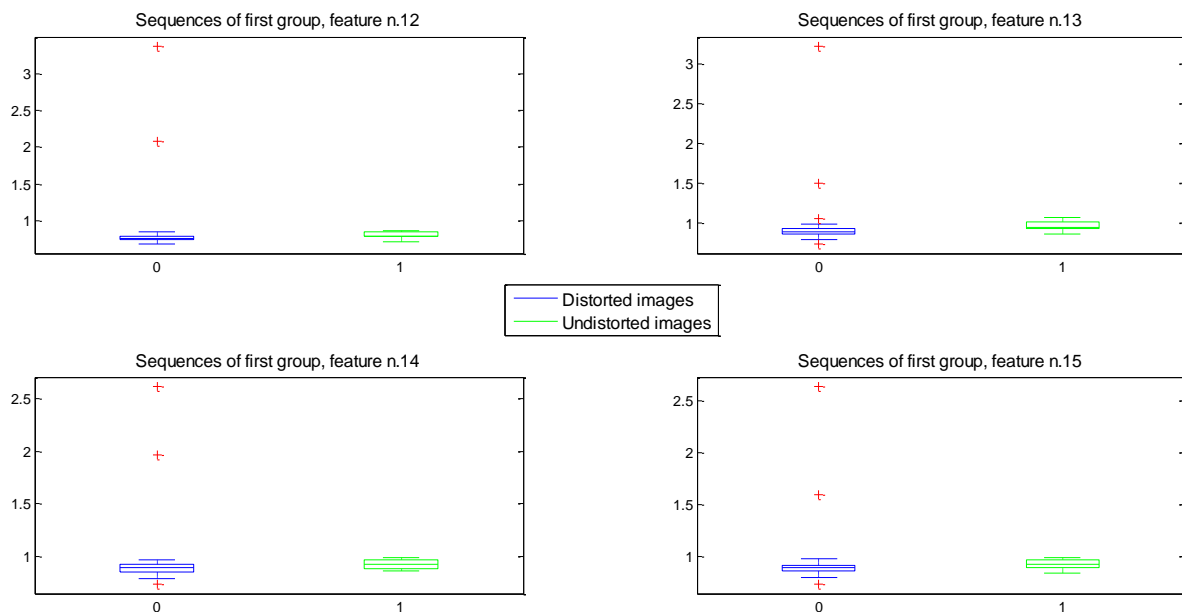
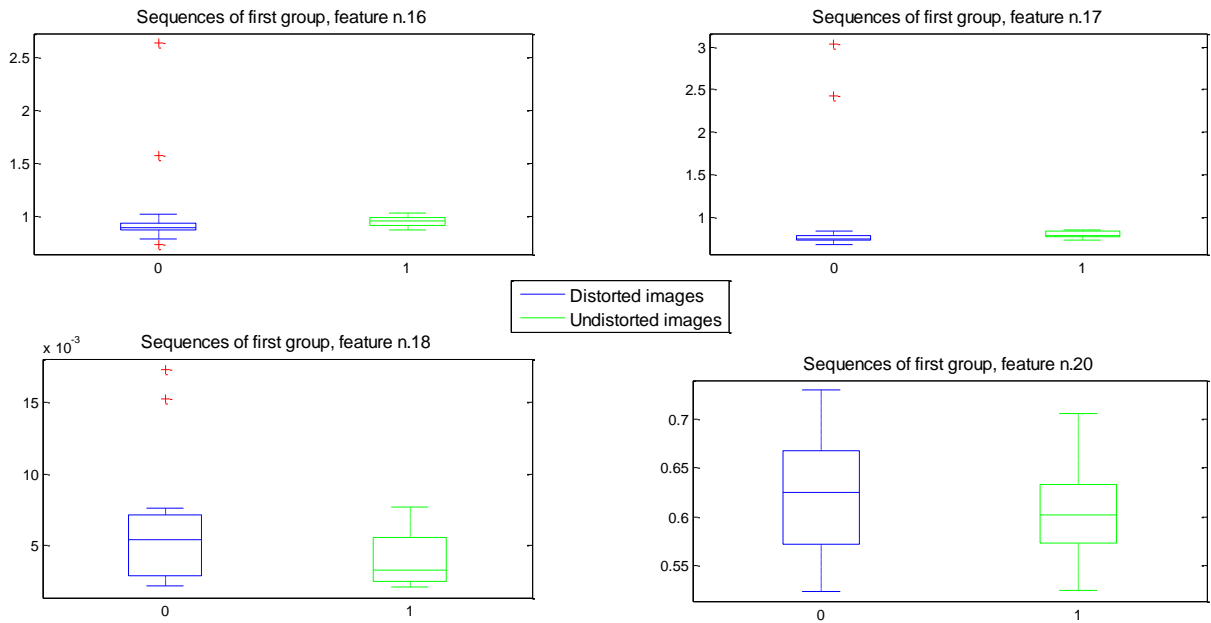*Fig. 27 - Box-whisker plots for Group1 sequences, feature n. 12-15.*

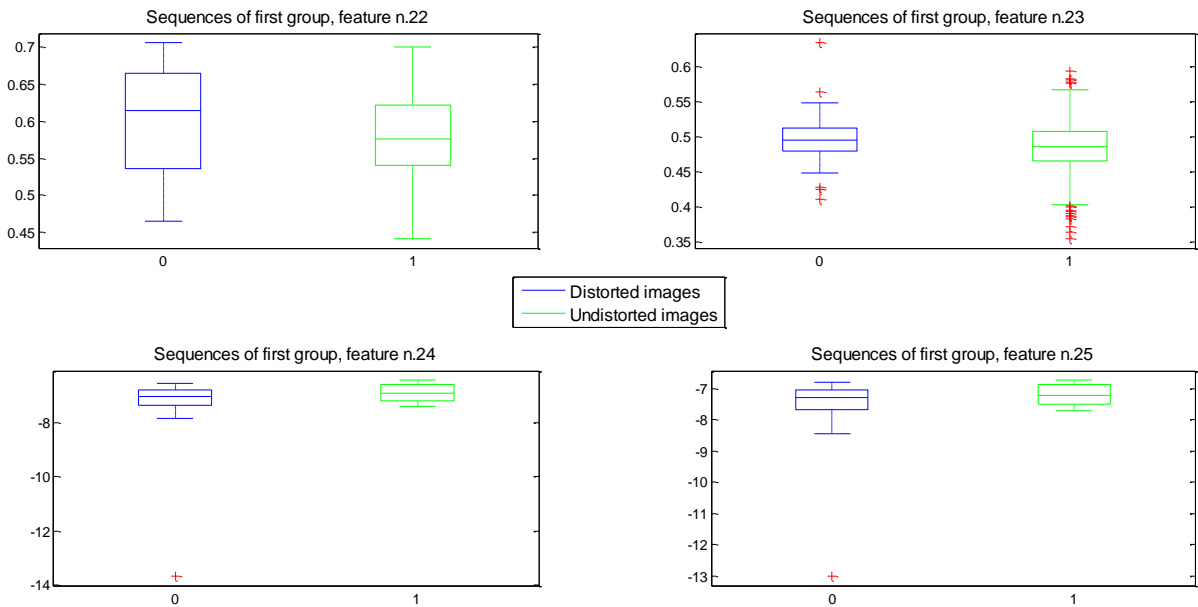*Fig. 28 - Box-whisker plots for Group1 sequences, feature n. 16-18 and 20.*

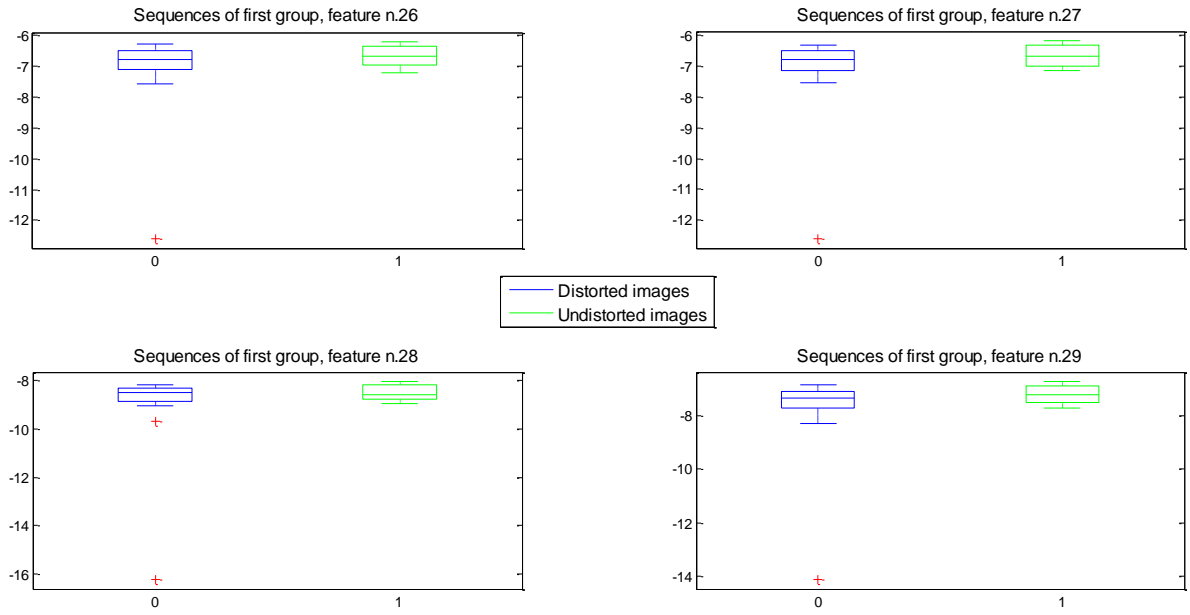*Fig. 29 - Box-whisker plots for Group1 sequences, feature n. 22-25.*

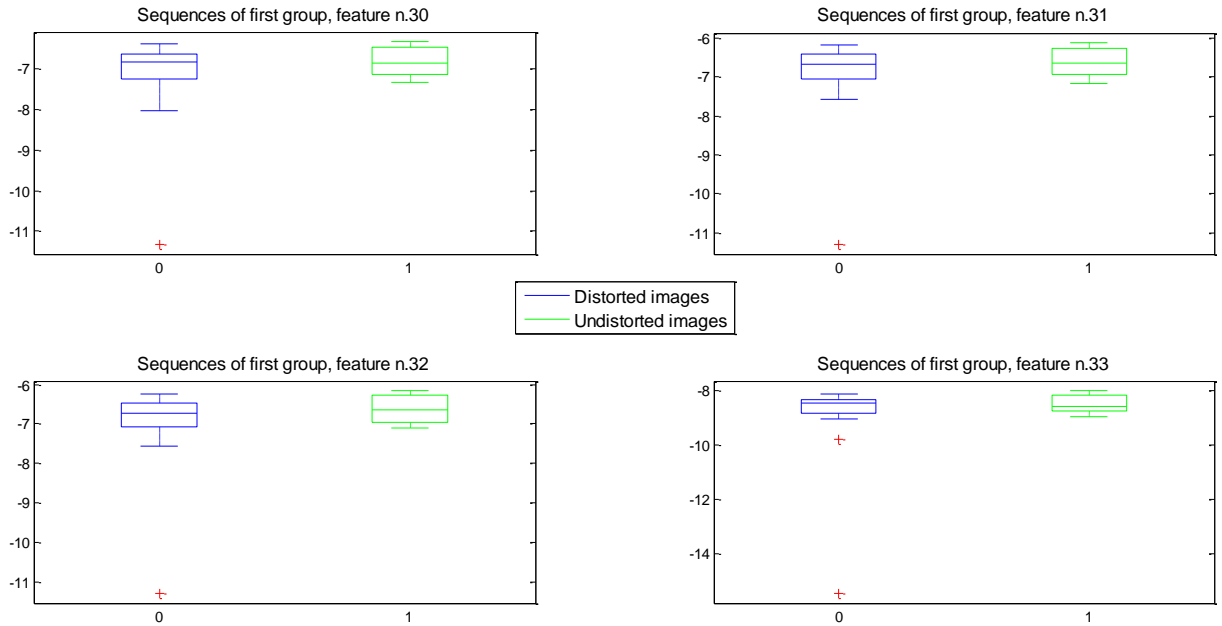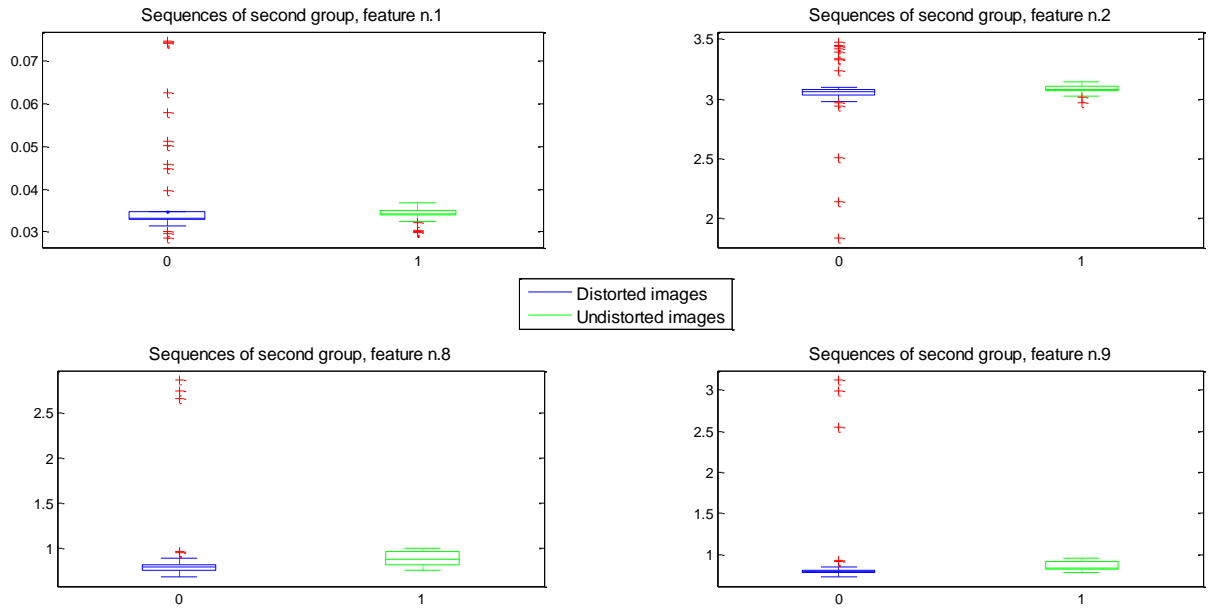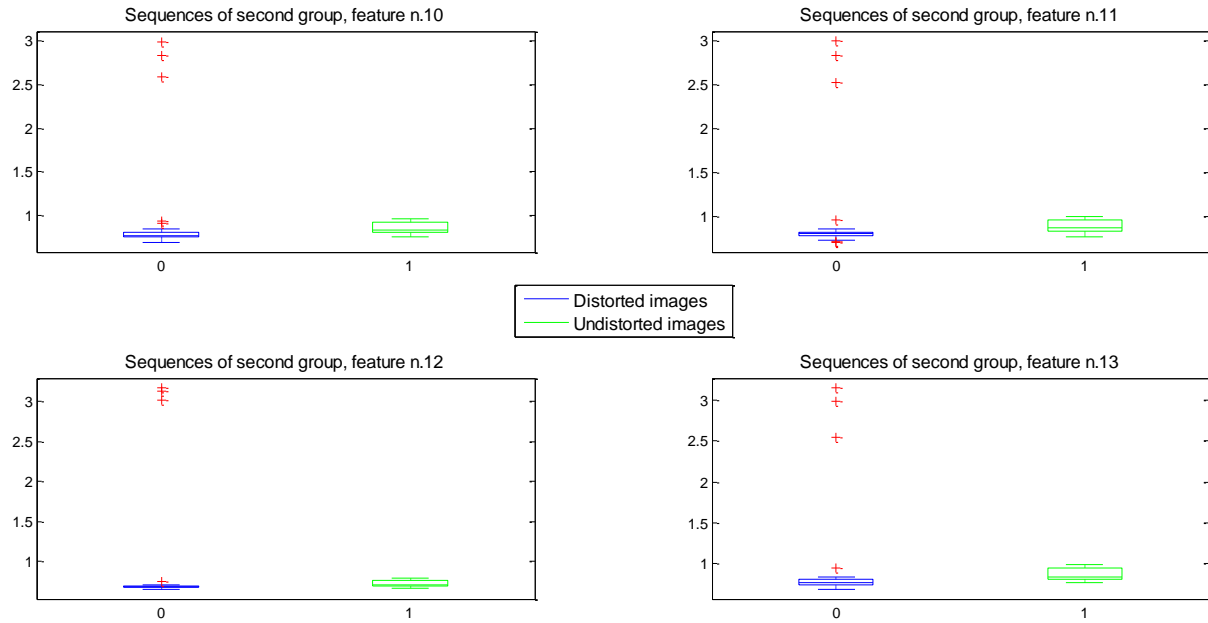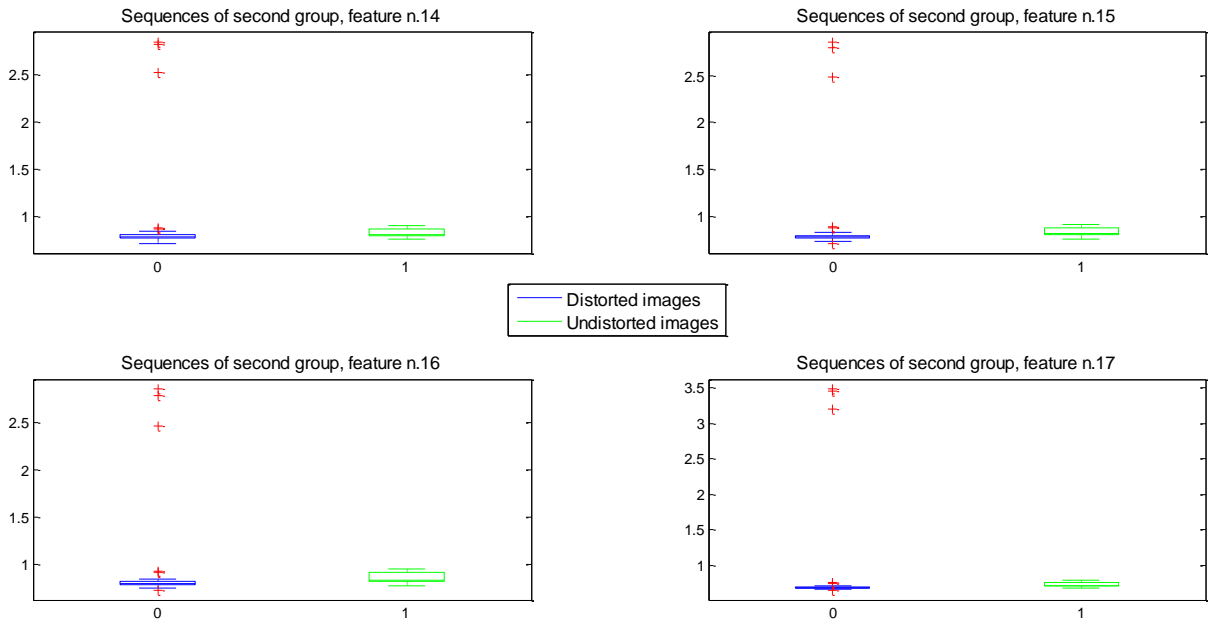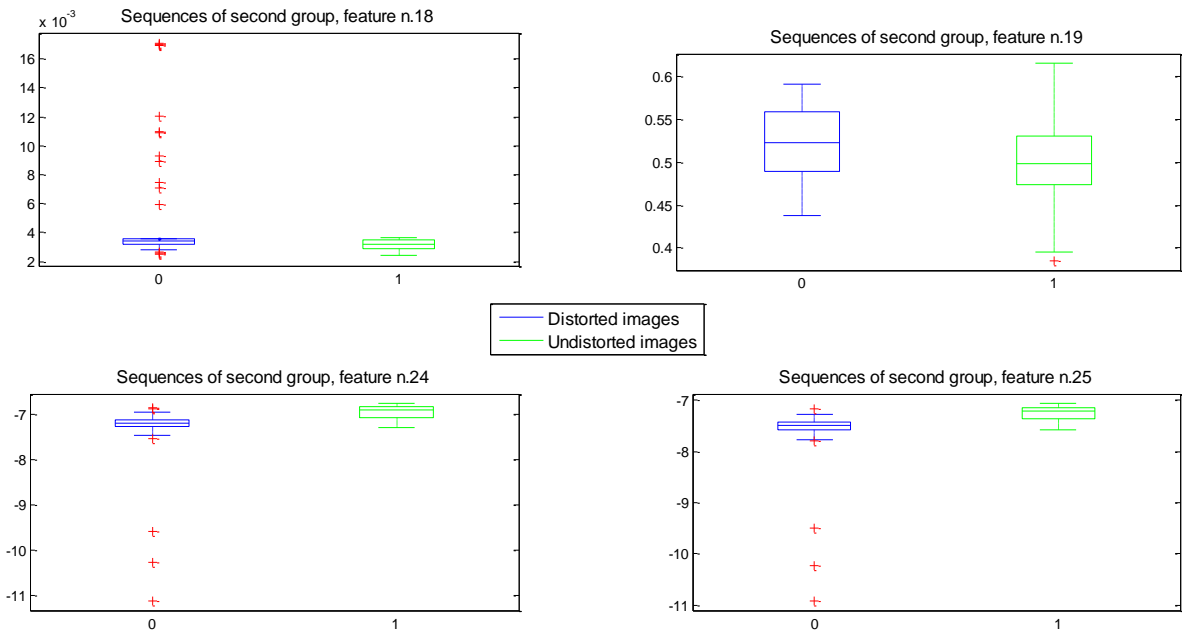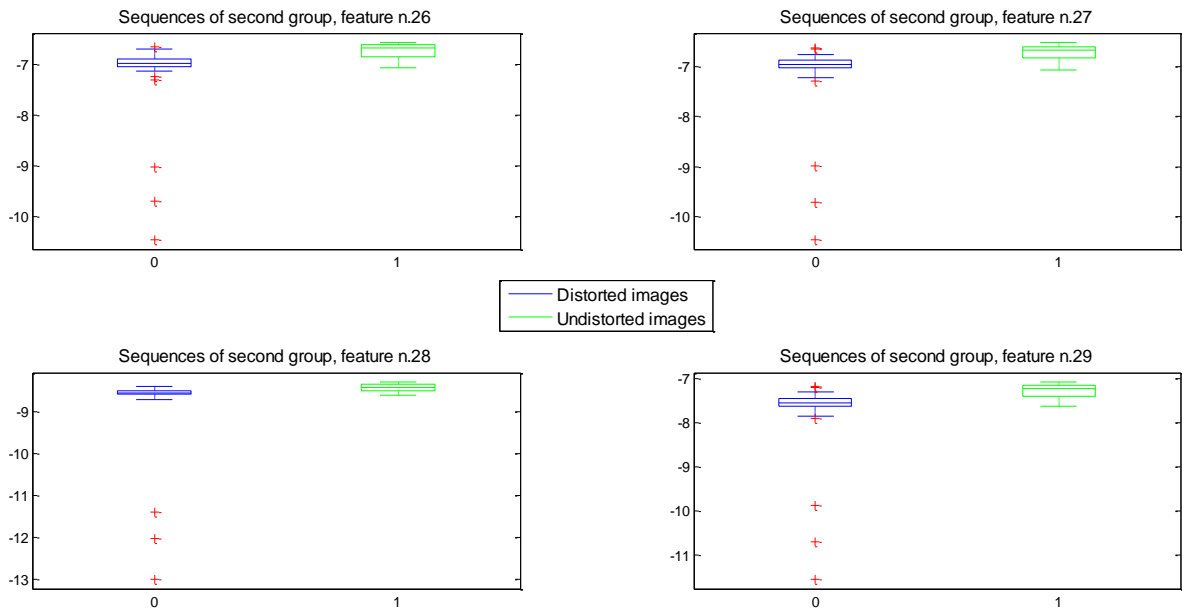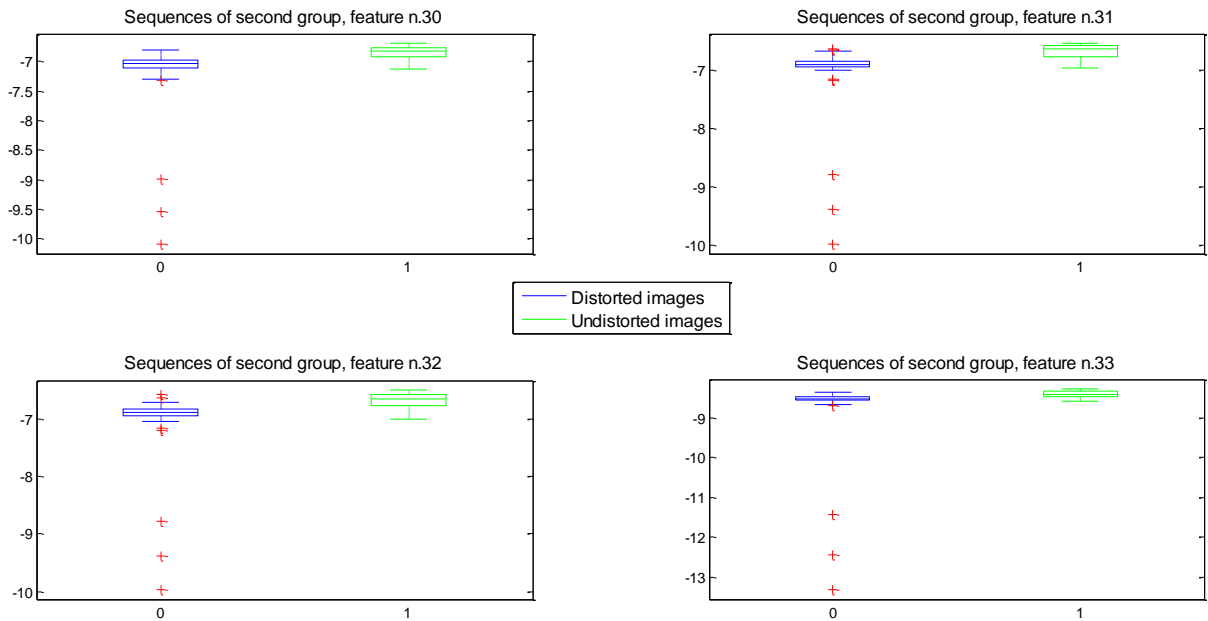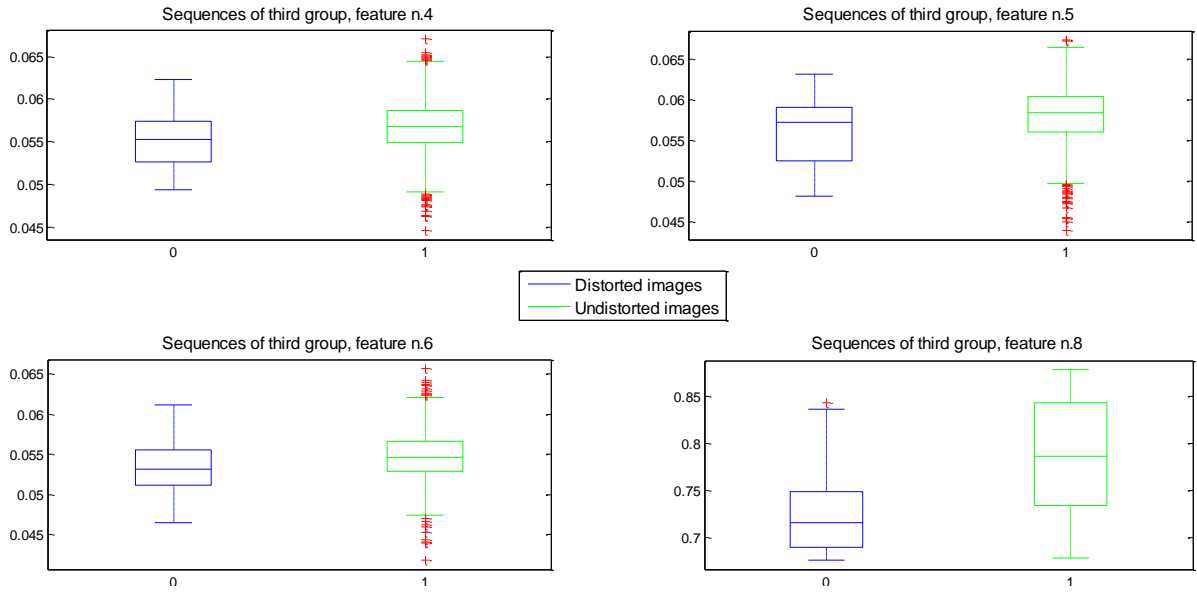*Fig. 30 - Box-whisker plots for Group1 sequences, feature n. 26-29.*

Fig. 31 - Box-whisker plots for Group1 sequences, feature n. 30-33.

Box-whisker plots for Group2 sequences

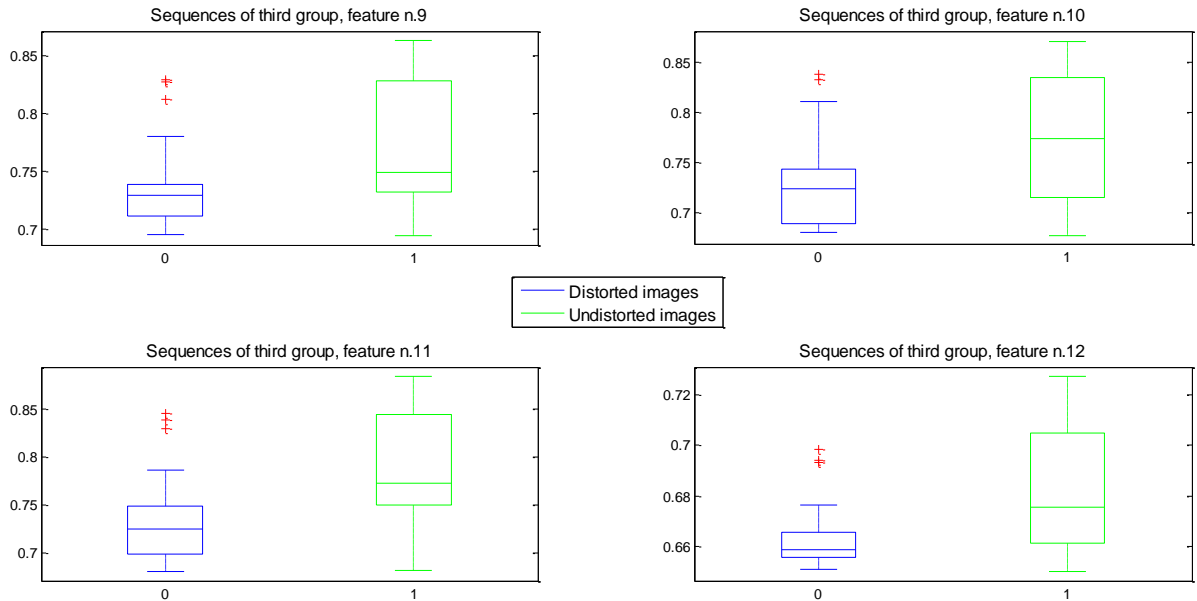*Fig. 32 - Box-whisker plots for Group2 sequences, feature n. 1, 2, 8, 9.*

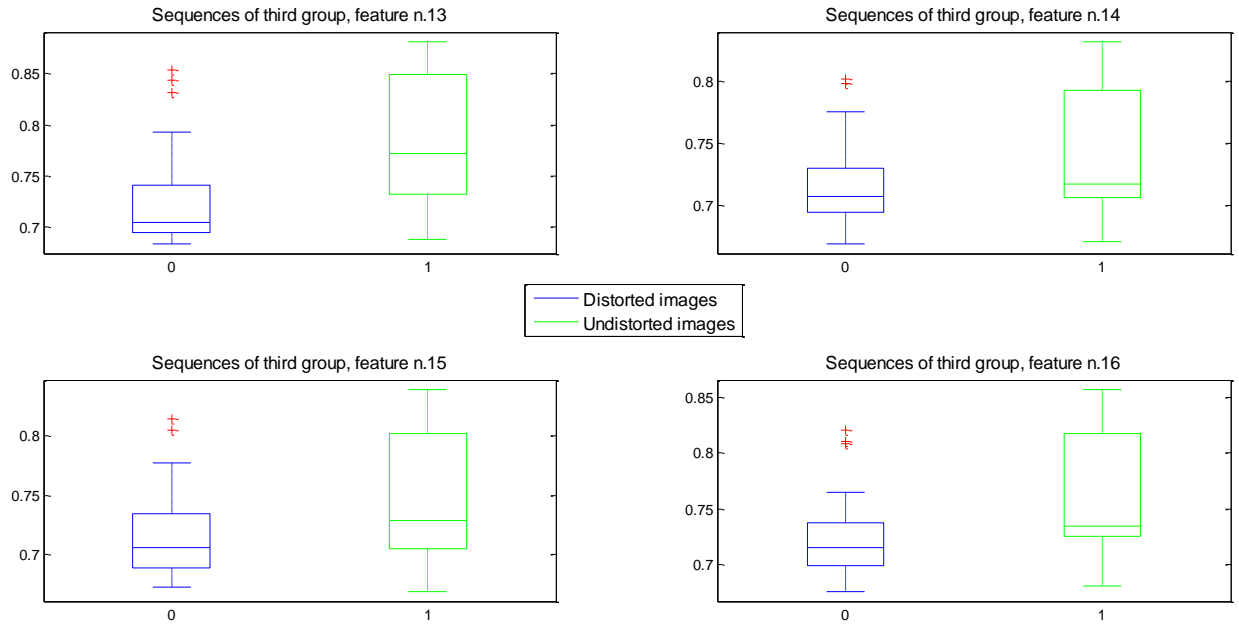*Fig. 33 - Box-whisker plots for Group2 sequences, feature n. 10-13.*

Fig. 34 - Box-whisker plots for Group2 sequences, feature n. 14-17.

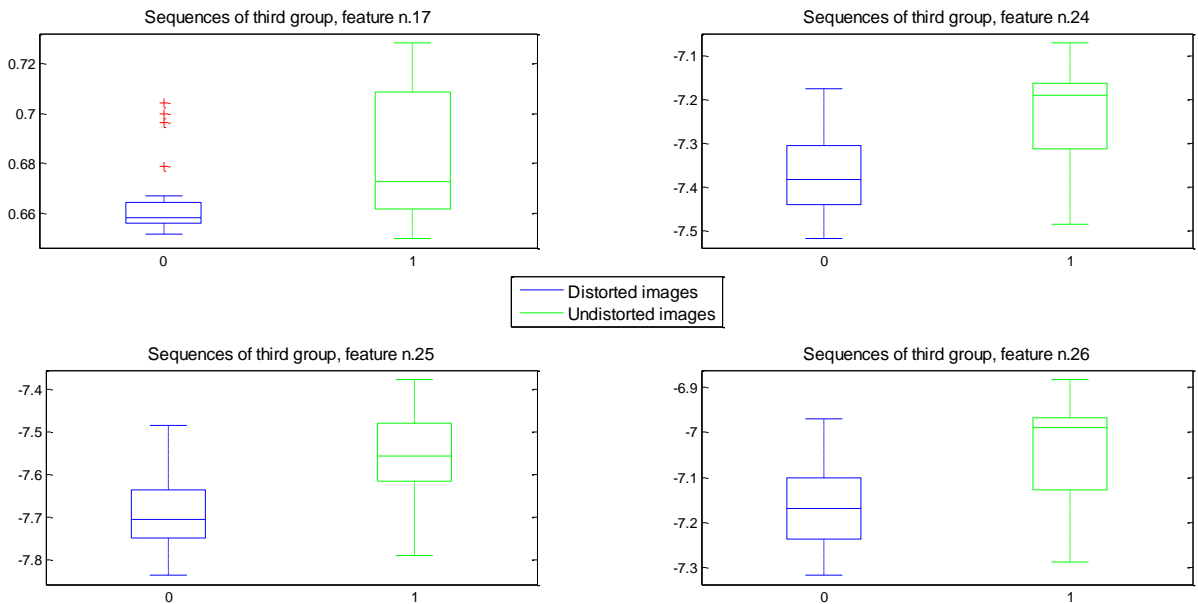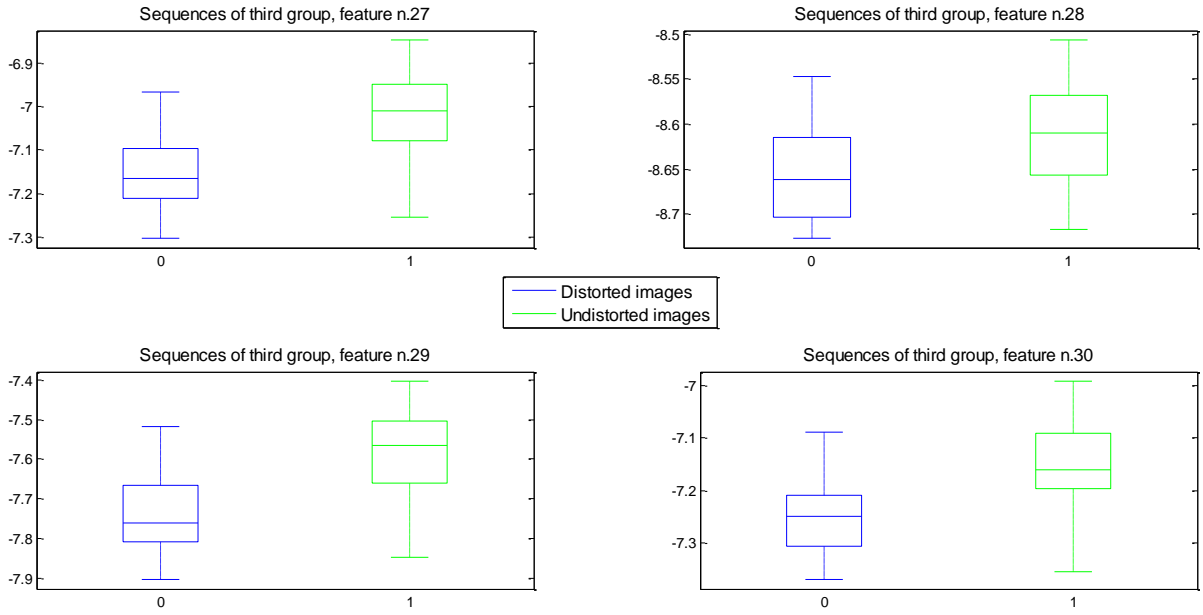*Fig. 35 - Box-whisker plots for Group2 sequences, feature n. 18,19, 24, 25.*

Fig. 36 - Box-whisker plots for Group2 sequences, feature n. 26-29.

Fig. 37 - Box-whisker plots for Group2 sequences, feature n. 30-33.

Box-whisker plots for Group3 sequences

Fig. 38 - Box-whisker plots for Group3 sequences, feature n. 4-6, 8.

Fig. 39 - Box-whisker plots for Group3 sequences, feature n. 9-12.

Fig. 40 - Box-whisker plots for Group3 sequences, feature n. 13-16.

Fig. 41 - Box-whisker plots for Group3 sequences, feature n. 17, 24-26.

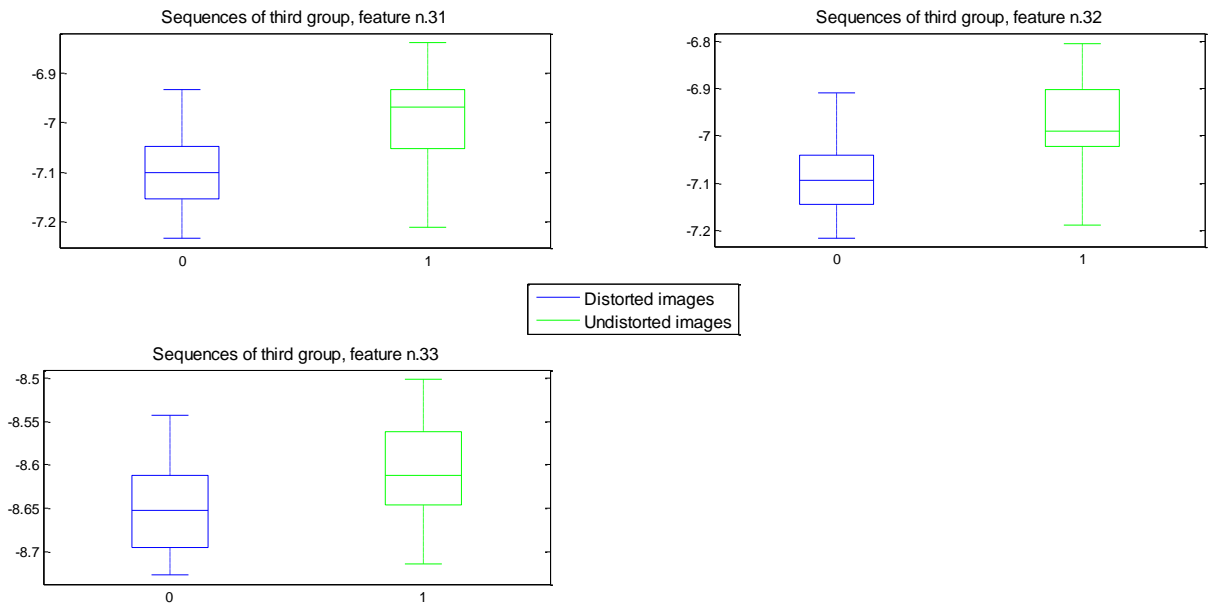*Fig. 42- Box-whisker plots for Group3 sequences, feature n. 27-30.*

Fig. 43 - Box-whisker plots for Group3 sequences, feature n. 31-33.

BIBLIOGRAPHY

[1]   KOLAR, R., HOEHER, B., ODSTRCILIK, J., SCHMAUSS, B. and JAN, J. *Registration of Image Sequences from Experimental Low-Cost Fundus Camera*. In: Proceedings of Workshop on Biomedical Image Registration, Lecture Notes in Computer Science 8454, 2014, s. 174-183.

[2]   NIXON, M. and A. AGUADO. *Feature extraction and image processing*. 1st ed. Boston: Newnes, 2002, xii, 350 p. ISBN 0750650788.

[3]   SILBERNAGL, S. and A. DESPOPOULOS. *Atlas fyziologie člověka*. 6. vyd., zcela přeprac. a rozš., Vyd. 3. české. Praha: Grada, 2004, xiii, 435 s. ISBN 80-247-0630-x.

[4]   GRAY, H. *Anatomy of the human body*. 30th American ed. Philadelphia: Lea & Febiger, 1985, xvii, 1676 p. ISBN 081210644x.

[5]   KRASULA, L., KLIMA, M., ROGARD, E. and E. JEANBLANC. *MATLAB-based Applications for Image Processing and Image Quality Assessment – Part I: Software Description.* [Online]. Available at:
http://www.radioeng.cz/fulltexts/2011/11_04_1009_1015.pdf

[6]   WANG, Z. and A. C. BOVIK. *A Universal Image Quality Index.* [Online]. Available at:
https://ece.uwaterloo.ca/~z70wang/publications/quality_2c.pdf

[7]   LI, Q. and Z. WANG. *Reduced-reference image quality assessment using divisive normalization-based image representation*. IEEE Journal of Selected Topics in Signal Processing 3, 202–211 (April 2009). [Online]. Available at:
https://ece.uwaterloo.ca/~z70wang/publications/JSTSP09.pdf

[8]   ZHANG, Y. and M. D. CHANDLER. *An Algorithm for No-Reference Image Quality Assessment Based on Log-Derivative Statistics of Natural Scenes.* Proceedings of the SPIE Conference on Image Quality and System Performance X, Volume 8653, February 3-7, 2013, Burlingame, CA., USA. [Online]. Available at:
http://vision.okstate.edu/pubs/yi_conf.pdf

[9]   ODSTRCILIK, J., KOLAR, R., TORNOW, R.P., BUDAI, A. and J. JAN. *Blood Vessel Segmentation in Video-Sequences from Experimental Low-Cost Fundus Camera.* 36th Annual International IEEE EMBS Conference. Received April 7, 2014. [Online]. Available at:
http://ieeexplore.ieee.org/xpl/articleDetails.jsp?reload=true&arnumber=6958459

[10]   PIZER, S.M., AMBURN, E.P., AUSTIN, J.D., CROMARTIE, R., GESELOWITZ, A., GREER, T., HAAR ROMENY, B. ter, ZIMMERMAN, J.B. and K. ZUIDERVELD. *Adaptive histogram equalization and its variations*. Computer Vision, Graphics, and Image Processing 39, 355-368 (1987). [Online]. Available at:
http://www.cs.unc.edu/Research/MIDAG/pubs/papers/Adaptive Histogram Equalization and Its Variations.pdf

[11]   TANG, H., JOSHI, N. and A. KAPOOR. *Learning a blind measure of perceptual image quality*. International Conference on Computer Vision and Pattern Recognition, (2011). [Online]. Available at: http://research.microsoft.com:8082/pubs/155043/576_iqa.pdf

[12]   LI, C., BOVIK, A. C. and X. Wu. *Blind image quality assessment using a general regression neural network.* IEEE Transaction on Neural Networks 22, 793–799 (May 2011). [Online]. Available at: http://www.researchgate.net/profile/Chaofeng_Li4/publication/224228818_Blind_Image_Quality_Assessment_Using_a_General_Regression_Neural_Network/links/0f317539e5baf1b7b9000000.pdf

[13]   MITTAL, A., MOORTHY, A. K. and A. C. BOVIK. *No-reference image quality assessment in the spatial domain.* IEEE Transations on Image Processing (2012). [Online]. Available at: http://live.ece.utexas.edu/publications/2012/TIP BRISQUE.pdf

[14]   MOORTHY, A. K. and A. C. BOVIK. *Blind image quality assessment: From natural scene statistics to perceptual quality.* IEEE Transactions on Image Processing 20(12), 3350–3364 (2011). [Online]. Available at: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5756237

[15]   MOORTHY, A. K. and A. C. BOVIK. *A two-step framework for constructing blind image quality indices.* IEEE Signal Processing Letters 17, 513–516 (May 2010). [Online]. Available at: http://live.ece.utexas.edu/publications/2010/akm_spl_may10.pdf

[16]   SAAD, M. A. and A. C. BOVIK. *Blind image quality assessment: A natural scene statistics approach in the DCT domain.* IEEE Transactions on Image Processing 21, 3339–3352 (2012). [Online]. Available at: http://live.ece.utexas.edu/publications/2012/saad_2012_tip.pdf

[17]   KOHLER, T., BUDAI, A., KRAUS, M.F., ODSTRCILIK, J., MICHELSON, G. and J. HORNEGGER. *Automatic no-reference quality assessment for retinal fundus images using vessel segmentation.* In: Computer-Based Medical Systems (CBMS), 2013 IEEE 26th International Symposium on. IEEE, 2013. p. 95-100. [Online]. Available at: http://www5.informatik.uni-erlangen.de/Forschung/Publikationen/2013/Koehler13-ANQ-talk.pdf

[18]   ABRÀMOFF, M. D., GARVIN, M. K. and M. SONKA. *Retinal Imaging and Image Analysis*. IEEE Reviews in Biomedical Engineering. 2010, vol. 3, s. 169-208 [cit. 2015-05-16]. DOI: 10.1109/rbme.2010.2084567. [Online]. Available at: http://www.ncbi.nlm.nih.gov/pmc/articles/PMC3131209/

[19]   CHEN, J., TIAN, J., LEE, N., ZHENG, J., SMITH, R.T. and A.F. LAINE. *A partial intensity invariant feature descriptor for multimodal retinal image registration.* IEEE Trans Biomed Eng. Jul; 2010 57(7):1707–1718. [PubMed: 20176538]. [Online]. Available at: http://www.ncbi.nlm.nih.gov/pmc/articles/PMC3030813/

[20] RONKOVÁ, P. and J. KOLÁŘOVÁ. *Retinal image registration based on hybrid genetic algorithm*. In Proceedings of the 19th Conference STUDENT EEICT 2013 Volume 3. LITERA. Brno: 2013, s.196-200, ISBN: 978-80-214-4695- 3.

[21] JAN, J. *Medical image processing, reconstruction, and restoration: concepts and methods*. Boca Raton: Dekker/CRC Press, 2005, xxiii, 730 s. Signal processing and communications, 24. ISBN 0824758498.

[22] MURAT, B. and F. HASSAN. *Subpixel registration directly from the phase difference*. EURASIP Journal on Applied Signal Processing 2006 (2006) Article ID 60796 11 pages. [Online]. Available at: http://asp.eurasipjournals.com/content/pdf/1687-6180-2006-060796.pdf

[23] LUCAS, B.D. and T. KANADE. *An Iterative Image Registration Technique with an Application to Stereo Vision*. In: Proceedings of Imaging Understanding Workshop. Volume 130. (1981) 121-130. [Online]. Available at: https://www.ri.cmu.edu/pub_files/pub3/lucas_bruce_d_1981_1/lucas_bruce_d_1981_1.pdf

[24] YU, S., ZHANG, A. and H. LI. *A Review of Estimating the Shape Parameter of Generalized Gaussian Distribution*. Journal of Computational Information Systems 8: 21 (2012) 9055–9064. [Online]. Available at: http://www.jofcis.com/publishedpapers/2012_8_21_9055_9064.pdf

[25] FISCHER, S., REDONDO, R. and G. CRISTÓBAL. *How to construct log-Gabor Filters?* Open Access Digital CSIC Document, 2009. [Online]. Available at: http://ftp.optica.csic.es/resources/Software-deployment/Log-Gabor-filtering/LogGaborresumen.pdf

[26] HARUSTIAKOVA, D. *Vícerozměrné statistické metody v biologii*. Vyd. 1. Brno: Akademické nakladatelství CERM, 2012, 110 s. ISBN 978-80-7204-791-8.

[27] BOSER, B. E., GUYON I. and V. VAPNIK. *A training algorithm for optimal margin classiers*. In Proceedings of the Fifth Annual Workshop on Computational Learning Theory , pages 144{152. ACM Press, 1992. . [Online]. Available at: http://w.svms.org/training/BOGV92.pdf

[28] JAKKULA, V. *Tutorial on support vector machine (svm)*. School of EECS, Washington State University, 2006. [Online]. Available at: www.ccs.neu.edu/course/cs5100f11/resources/jakkula.pdf

[29] *Support Vector Machines (SVM)*. [online]. Available at: http://is.muni.cz/el/1433/podzim2006/PA034/09_SVM.pdf

[30] WESTON, J. *Support Vector Machine*. In: Tutorial, 4 Independence Way, Princeton. [Online]. Available at: http://www.cs.columbia.edu/~kathy/cs4701/documents/jason_svm_tutorial.pdf

[31] SHANNON, C. E. *A mathematical theory of communication*. ACM SIGMOBILE Mobile Computing and Communications Review, 2001, 5.1: 3-55. [Online]. Available at: http://www.lanethames.com/dataStore/ECE/InfoTheory/shannon.pdf

[32] COVER, T.M. and J.A. THOMAS. *Elements of Information Theory (second edition)*. 2006. [Online]. Available at: http://coltech.vnu.edu.vn/~thainp/books/Wiley_-_2006_-_Elements_of_Information_Theory_2nd_Ed.pdf

[33] KOVESI, P. *gaborconvolve.m* [software]. May 2011. School of Computer Science & Software Engineering, The University of Western Australia. Available at: http://www.peterkovesi.com/matlabfns/PhaseCongruency/gaborconvolve.m

[34] MITCHEL, T. *Machine learning*. WCB. 1997. [Online]. Available at: http://personal.disco.unimib.it/Vanneschi/McGrawHill_-_Machine_Learning_-_Tom_Mitchell.pdf

[35] VAN DER WEKEN, D., NACHTEGAEL, M. and E. E. KERRE. *An overview of similarity measures for images*. In: Acoustics, Speech, and Signal Processing (ICASSP), 2002 IEEE International Conference on. IEEE, 2002. p. IV-3317-IV-3320.[Online]. Available at: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5745363&tag=1

[36] GOSHTASBY, A. A. *Similarity and dissimilarity measures*. In: Image Registration, Advances in Computer Vision and Pattern Recognition. Springer London, 2012. p. 7-66. [Online]. Available at: http://www.researchgate.net/publictopics.PublicPostFileLoader.html?id=54aaad9ad4c118e f048b4584&key=c5eb8739-70ee-442f-bf33-810d16e20c1c

[37] CHEN, C. C. and H. T. CHU. *Similarity measurement between images*. Department of Computer Science, National Tsing Hua University. In: null. IEEE, 2005. p. 41-42. [Online]. Available at: http://www.computer.org/csdl/proceedings/compsac/2005/2413/02/241320041.pdf

[38] WANG, B., PAN B. and G. LUBINEAU. *Some practical considerations in finite element-based digital image correlation*. Optics and Lasers in Engineering, 2015, 73: 22-32. [Online]. Available at: http://www.sciencedirect.com/science/article/pii/S0143816615000494#

[39] GOLKAR, E., RAHNI, A. and R. SULAIMAN. *Comparison of image registration similarity measures for an abdominal organ segmentation framework*. In: Biomedical Engineering and Sciences (IECBES), 2014 IEEE Conference on. IEEE, 2014. p. 442-445. [Online]. Available at: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=7047538

[40] RAO, Y. R., PRATHAPANI, N. and E. NAGABHOOSHANAM. *Application of normalized cross correlation to image registration*. IJRET: International Journal of Research in Engineering and Technology. May 2014. [Online]. Available at: http://doi.org/10.15623/ijret.2014.0317003

[41] LEWIS, J. P. *Fast template matching*. In: Vision interface. Canadian Image Processing and Pattern Recognition Society, Quebec City, Canada. 1995. p. 15-19. [Online]. Available at: http://scribblethink.org/Work/nvisionInterface/vi95_lewis.pdf

[42] LEWIS, J. P. *Fast normalized cross-correlation*. In: Vision interface. 1995. p. 120-123. [Online]. Available at: http://comupter-vision-summary.googlecode.com/svn/trunk/papers/FastNormalizedCrossCorrelation.pdf