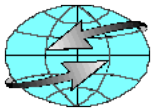
Slovak University of Technology Bratislava

FACULTY OF ELECTRICAL ENGINEERING AND INFORMATION TECHNOLOGY



Department of Telecommunications

Spectral Domain Image Segment Restoration

Diploma Thesis

Author: Ing. Andrej Varchola

Supervisor: Ing. Anton Březina

Image Reconstruction

- Filling in the missing areas or modifying the damaged ones in a **non-detectable** way by an observer not familiar with the original image
- Quite new method in digital image processing (1999)
- Visual system of human contains mechanisms actively connecting image regions that are physically disconnected on the retina – **visual completion**
- Application: data loss during transmission, compression

Existing Methods

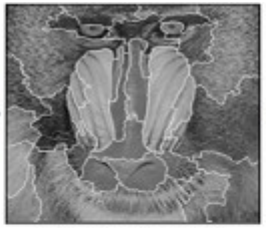
- Inpainting – image interpolation
  - Bertalmio, TV, CDD,...
  - Commonly applicable when small area is damaged
  - Not able to fix larger regions
- Texture synthesis
  - Resynthesizer
  - Heavy on computational time
  - Working with whole image
- Combination of both methods

Analysis and Design of Solution

Main Scheme

Original img.

Segmentation



Segmented img.

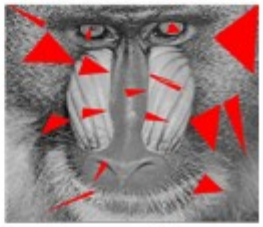
Triangulation

Triangle map

Pixel values

Triangles

Transmission of packets - triangles in a noisy channel



Damaged img.

Segmentation map

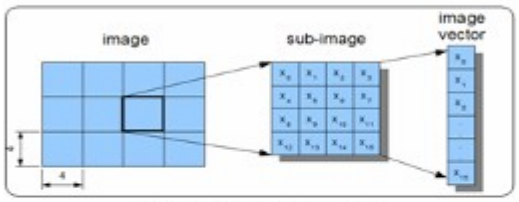
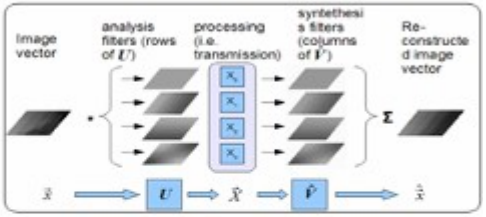


Image decomposition

KLT procedure



Transform coding

Reconstructed img.

Image Segmentation

- Connecting basic elements to the whole
- JSEG approach was used for this work – based on testing homogeneity of a given texture pattern
- Assumptions:
  - Each image contains homogeneous regions
  - Each image region can be represented by a set of few colors
  - The colors between two neighboring regions are distinguishable

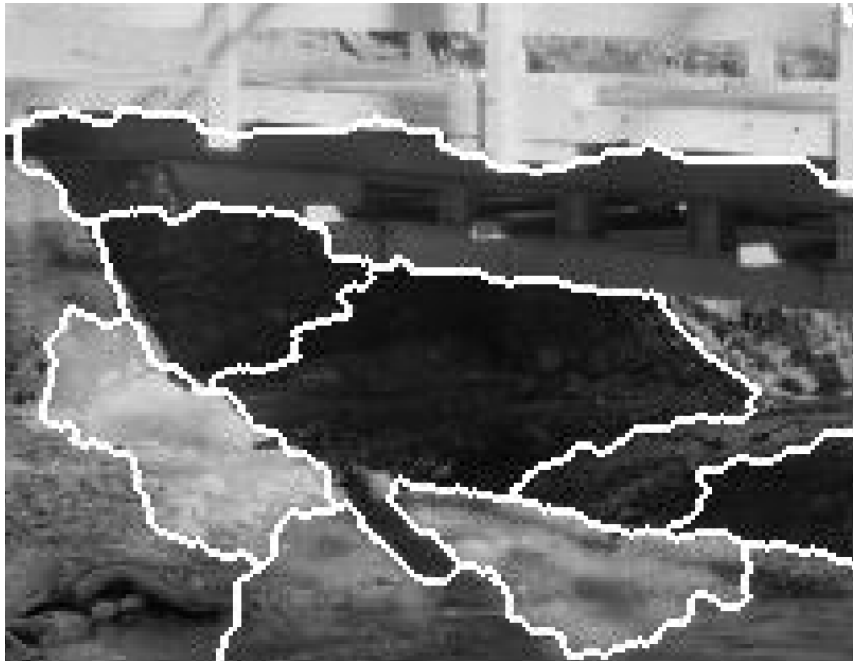
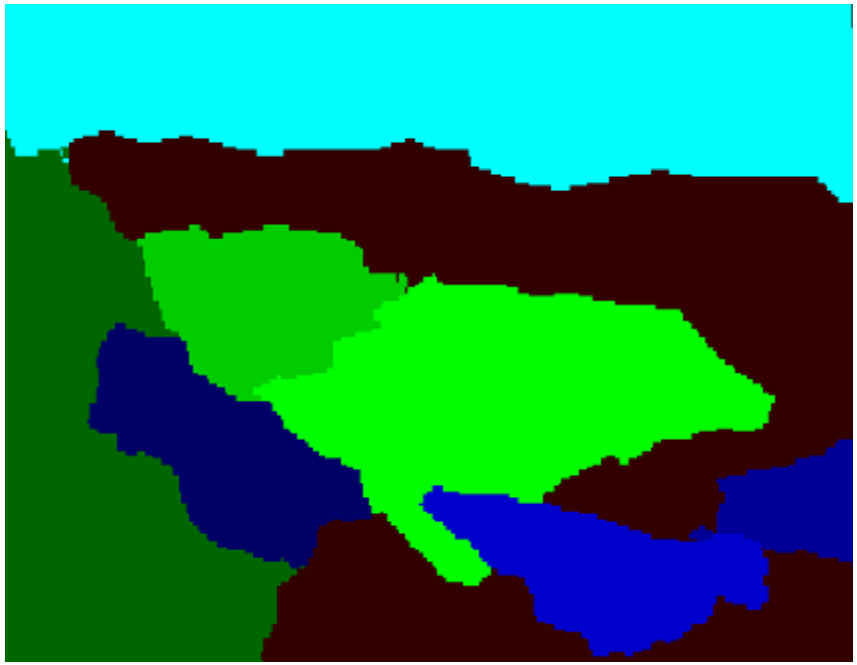


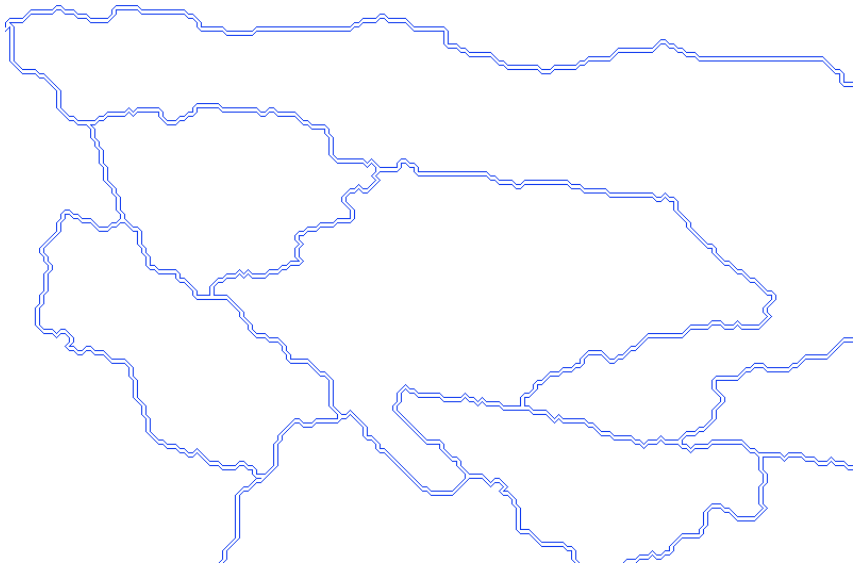
Image Segmentation

- Segmentation map – each image region is defined by unique label



Polygonal Approximation of the Region Boundaries

- Region boundary – value of at least one of neighboring pixels differs from the region value
- There are cases of two parallel boundaries between regions

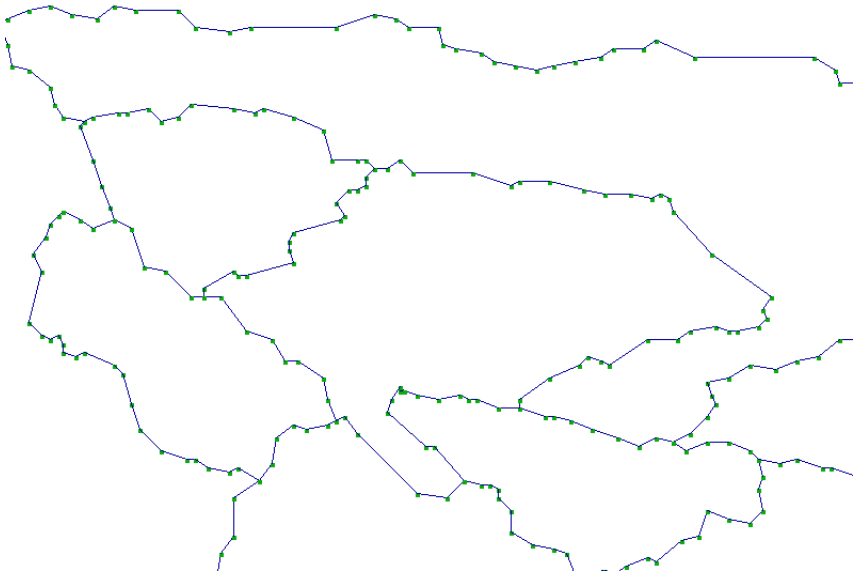


Polygonal Approximation of the Region Boundaries

- Doubled boundaries have to be reduced to be only one pixel wide
- Removing of very small segments by attaching to their larger neighbors has significant impact on the resulting code efficiency

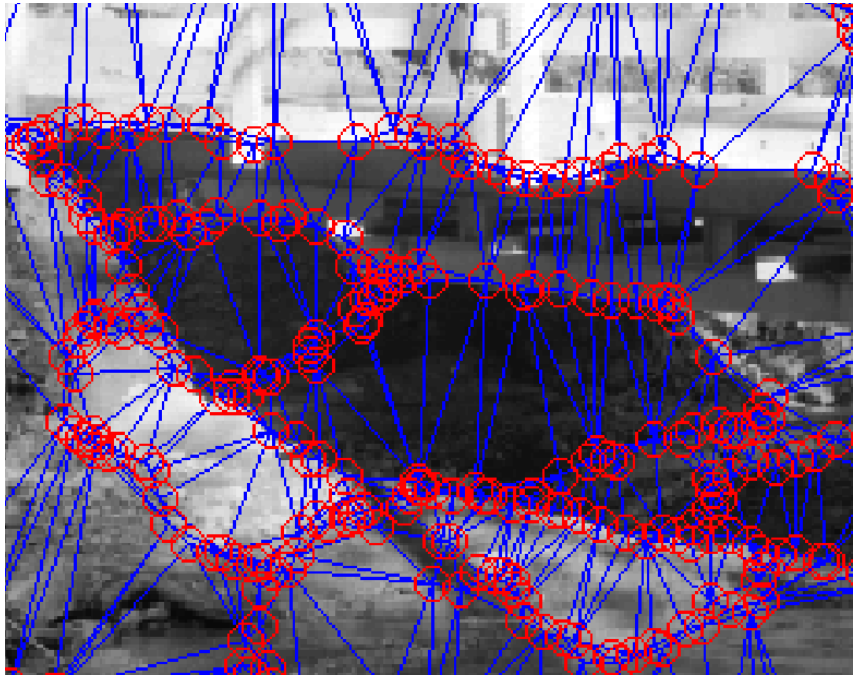
Polygonal Approximation of the Region Boundaries

- The boundaries of the segments are simply approximated with polygons
- The code is lossy – small influence on the quality of the resulting image



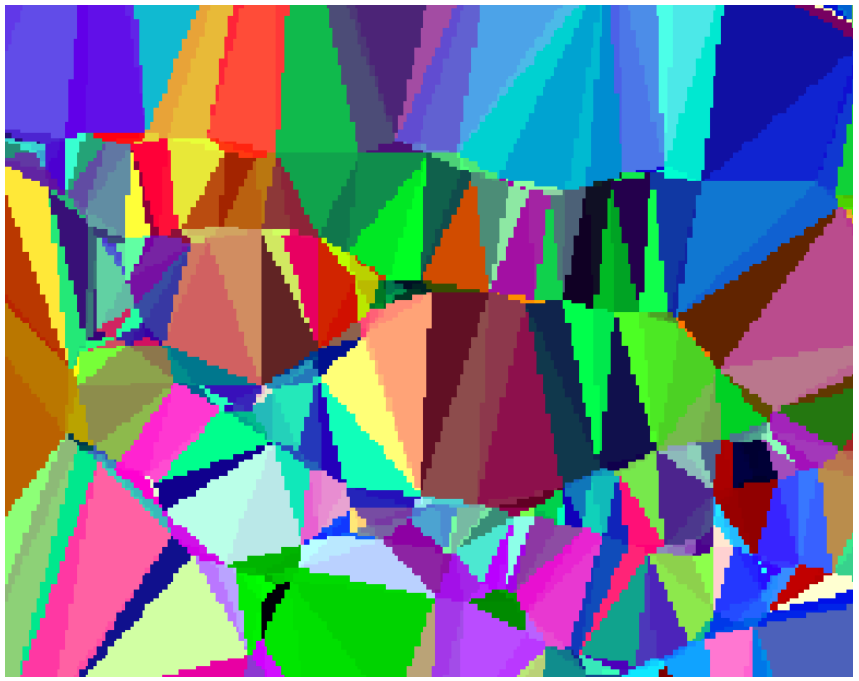
Triangulation of the Image Objects

- Image objects are divided into set of non-overlapping triangles using Delaunay triangulation that preserves original vertices of polygons



Triangulation of the Image Objects

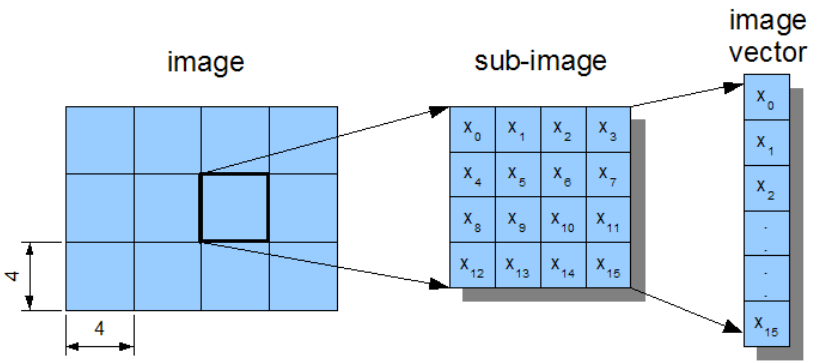
- Triangles are separated objects in the context of given image segment
- We assume loss or damage of these regions



Transform Coding

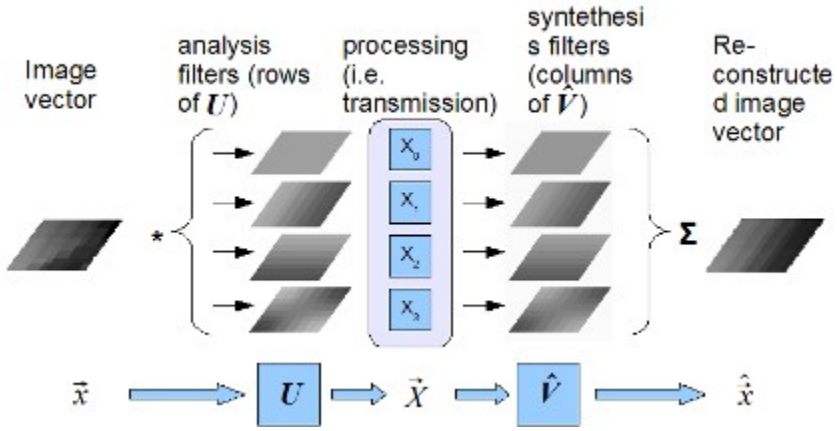
- Forward transformation

$$\vec{X} = \frac{1}{N} \cdot U \cdot \vec{x}$$



- Inverse (backward) transformation

$$\vec{x} = U^{¿T} \cdot \vec{X}$$



Karhunen-Loève Transform (KLT)

- The transformation matrix satisfies relation

$$UC = \Lambda U$$

- C is the covariance matrix and Λ is a diagonal matrix with eigenvalues

$$\Lambda = \begin{pmatrix} \lambda_1 & 0 & \cdots & 0 \\ 0 & \lambda_2 & & \vdots \\ \vdots & & \ddots & 0 \\ 0 & \cdots & 0 & \lambda_M \end{pmatrix}$$

Karhunen-Loève Transform (KLT)

$$\vec{x}_D = U_D^T \vec{X}$$

- If we constrain approximation with D eigenvectors from the C matrix, than **approximated** vector is reconstructed
- KLT is **optimal** at mean square error
- The basis is created by **adaptive** way, according to available vectors
- Optimal value of **D is dependent to data**

KLT for Gappy Data

- **Only** damaged data are available !
- Image vector with missing values is artificially created with **mask vector m**
- The goal is to restore complete image vectors from damaged vectors of the **segment**
- **Error minimisation** leads to the system of linear equations
- The solution is vector with **spectral coefficients**, which we wanted

$$M\,\vec{\beta} = \vec{f}$$

Construction of the KLT Basis

- Missing values are filled in with **average value** available for the pixel
- This new set is used for the first approximation of the basis
- KLT procedure is than used for reconstruction of each damaged vector
- This process can be iteratively repeated

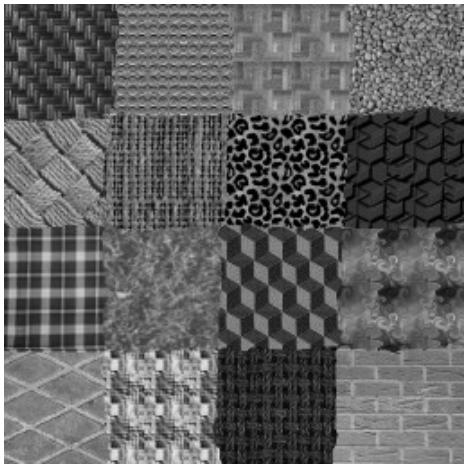
$$\begin{cases} \gamma_i^{(k)} & \text{if } m_i^{(k)} = 1 \\ \left(\gamma_D^{(k)}(0)\right)_i & \text{if } m_i^{(k)} = 0 \end{cases}$$

$$\gamma_i^{(k)}(1) = \{$$

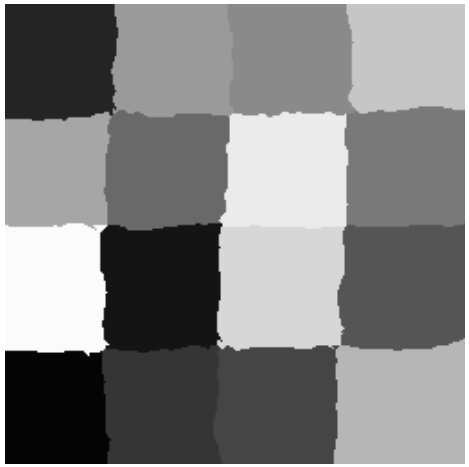
Measures of Quality

- Objective criterion:
  - mean square error
  - peak signal – noise ratio
- distance between two histograms

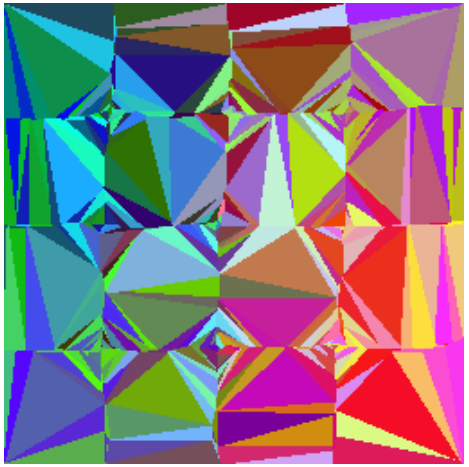
Results – Texture Restoration



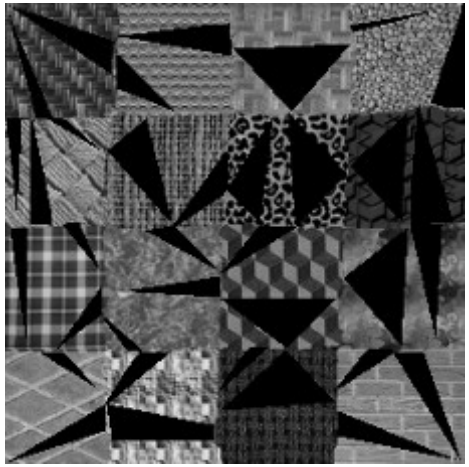
Original image 256x256 pixels



Segmentation map

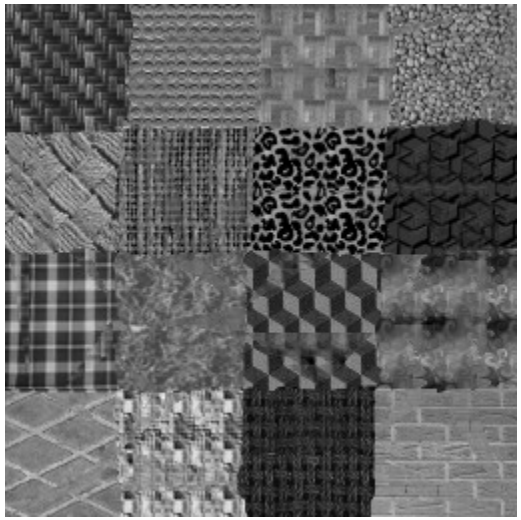


Triangle map

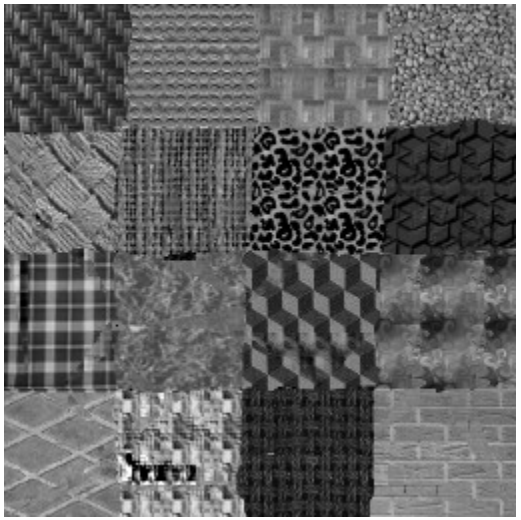


Damaged image – masked

Results – Texture Restoration

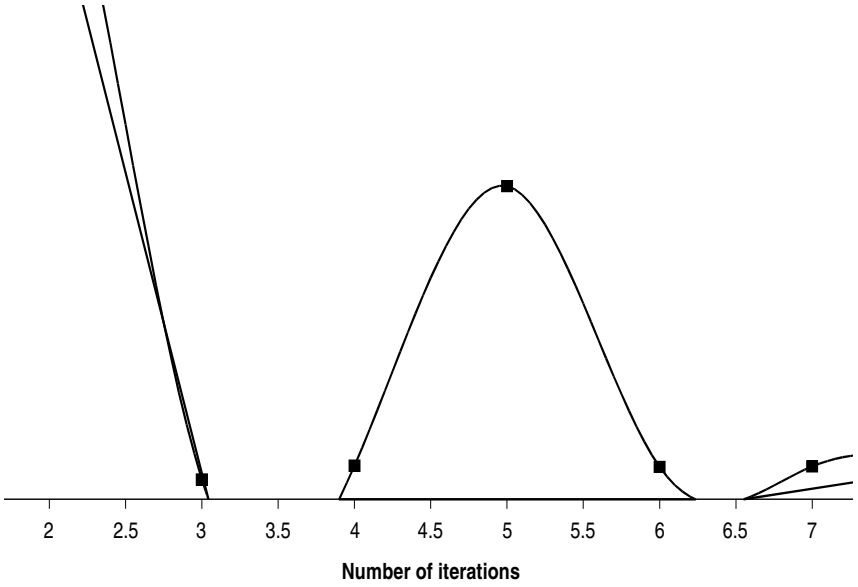


Restored image: KLT (D=2, 1 iteration)

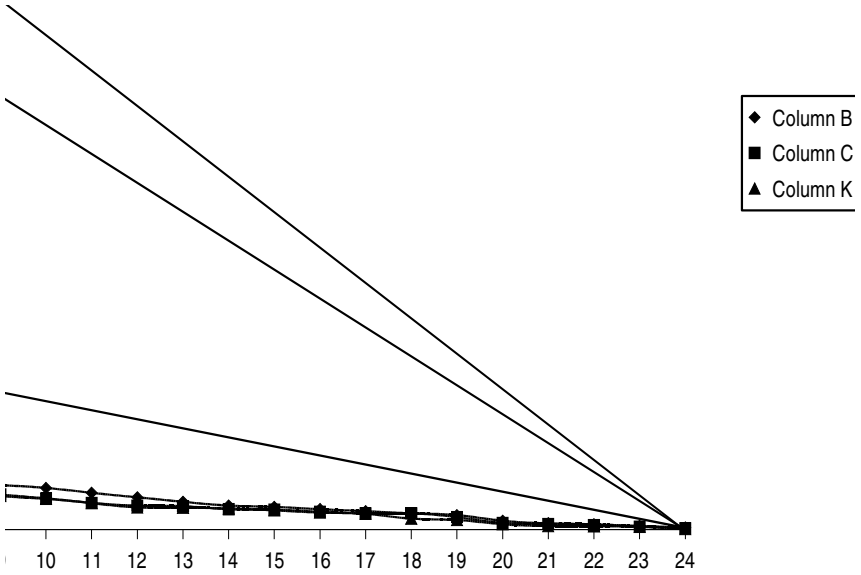


KLT (D=2, 10 iterations)

Graphs – Texture Restoration

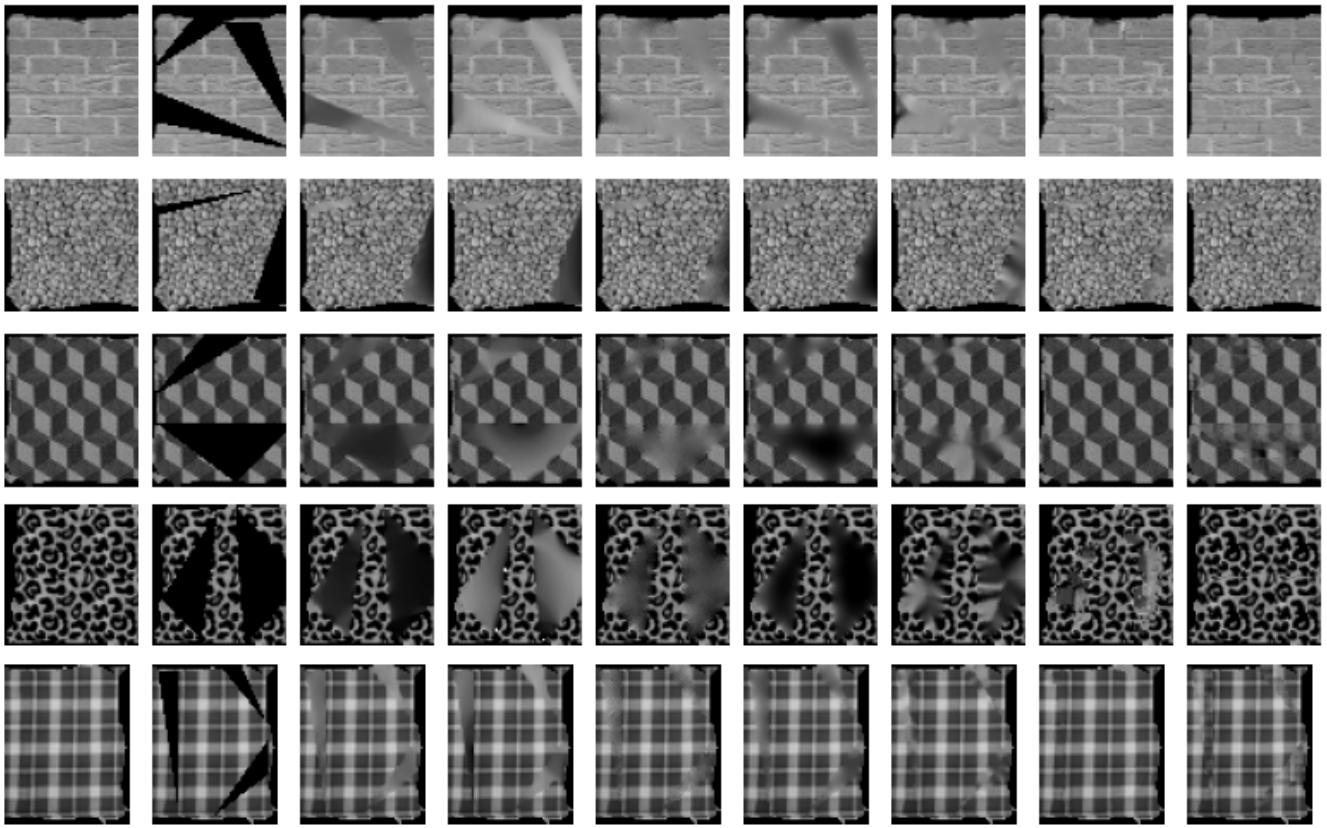


PSNR for segment no.2 – processing image by segments and as whole



Convergence of eigenvalues spectrum KLT (D=2) for segment no.8 – 1,2 a 10 iterations

Methods Comparison



a) b) c) d) e) f) g) h) i)

Methods comparison: a) original, b) damaged, c) Bertalmio, d) TV, e) CDD, f) Oliveira, g) Mean, h) Resynthesizer, i) KLT (D=2, 1 iteration)

Tables – Methods Comparison

Segment	Bertalmio	TV	CDD	Oliveira	Mean	Resynthesizer	KLT
12	16,508	19,614	21,377	18,420	20,868	21,123	**23,148**
13	9,774	9,949	11,360	9,428	16,016	16,707	**17,031**
14	15,040	16,065	17,684	14,189	16,151	**26,436**	19,450
15	10,668	10,748	12,326	10,656	11,623	12,324	**14,368**
16	17,353	16,193	18,658	18,769	18,585	**24,471**	17,709

Table with PSNR values in dB

Tables – Methods Comparison

Segment	Bertalmio	TV	CDD	Oliveira	Mean	Resynthesizer	KLT
12	1130	1018	680	1118	630	**484**	936
13	850	942	690	868	498	**398**	464
14	1032	1400	1370	1190	742	**358**	1280
15	2260	1714	2154	1634	1464	952	**414**
16	740	616	598	708	466	**438**	556

Histogram distance

Metóda	Čas CPU
Bertalmio	3 min 26,219 s
TV	1 min 19,828 s
CDD	1 min 38,906 s
Oliveira	1453 ms
Mean	**171 ms**
Resynthesizer	2 min 41 s
KLT	2.75 s

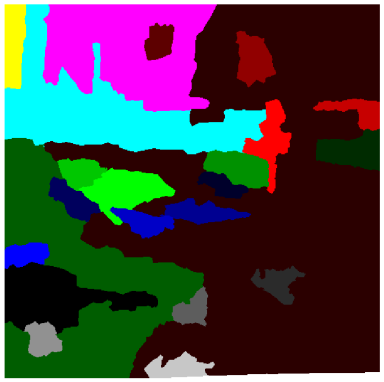
Computational time
(AMD Turion 64M, 1.8GHz)

Segment	Poškodenie oblati (%)
12	26,23
13	18,56
14	26,33
15	38,80
16	16,87

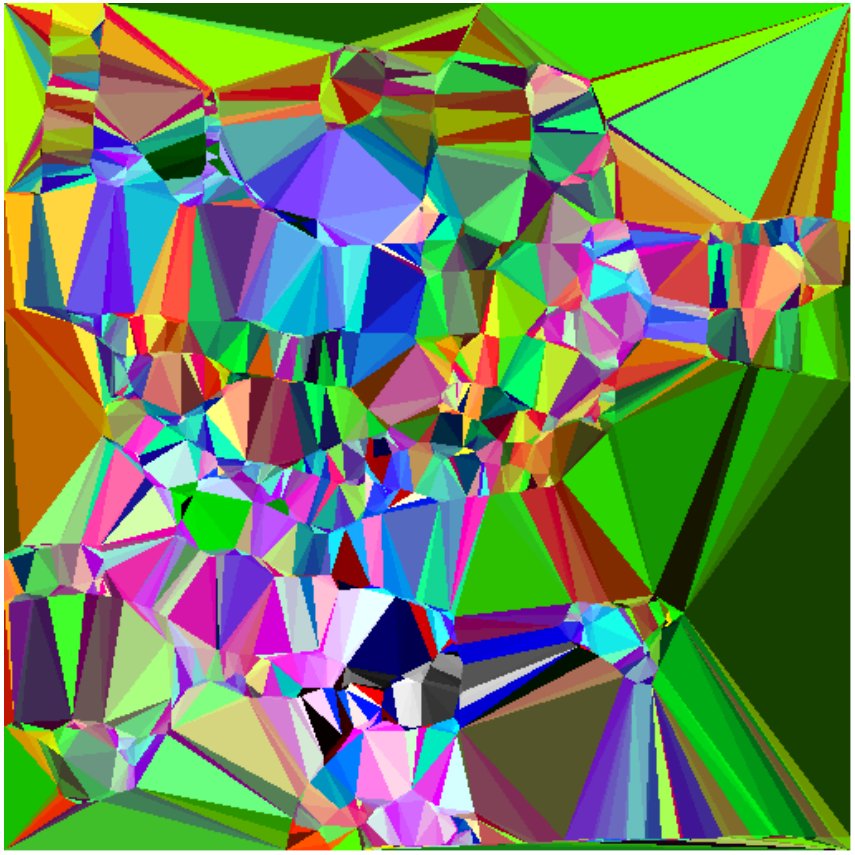
Damaged area in %

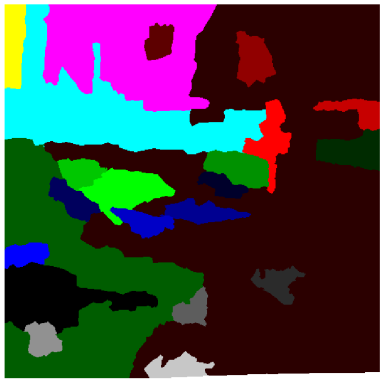
Image – „Stream&Bridge“

Segmentation – Segmentation Map

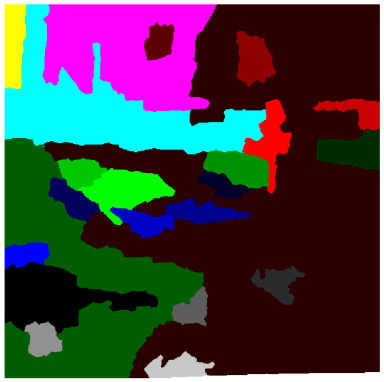


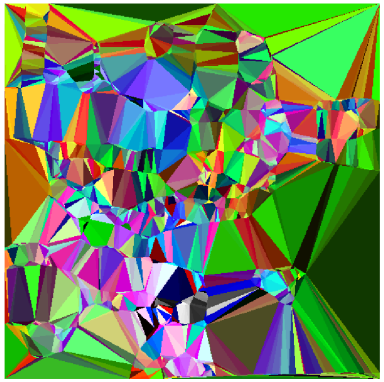
Triangulation



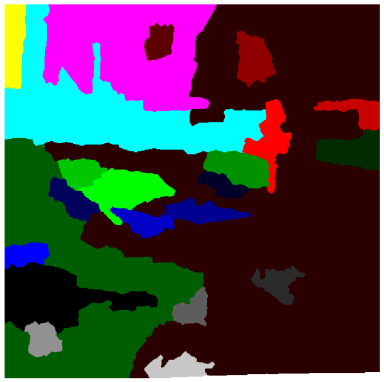


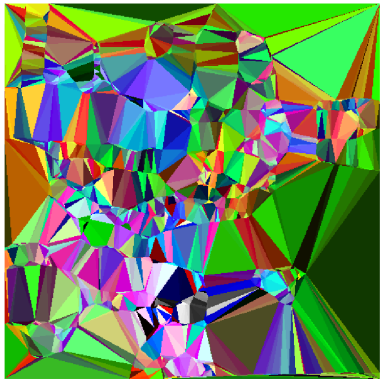
Map of Segments and Triangles



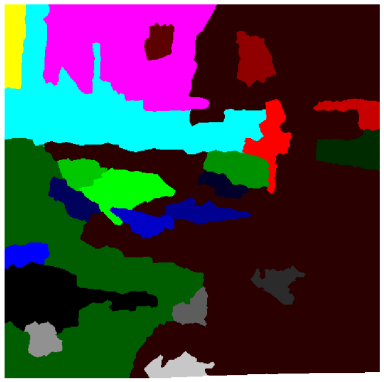


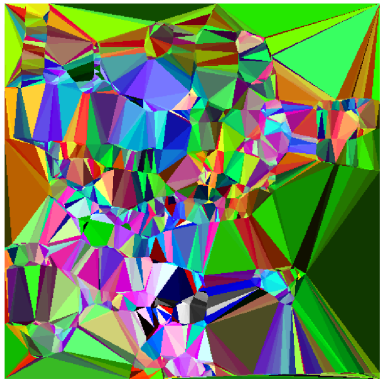
Masking – Damaged Image





KLT – D=2, 3 iterations





KLT vs. Bertalmio

KLT vs. TV

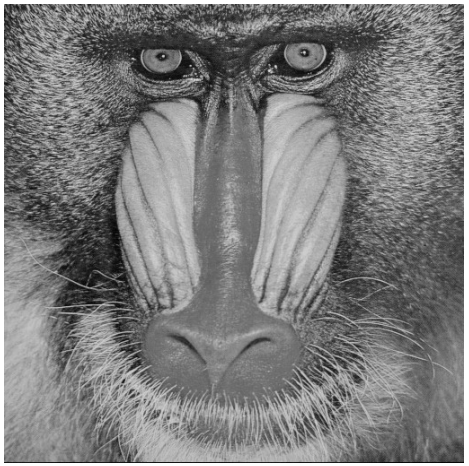
KLT vs. CDD

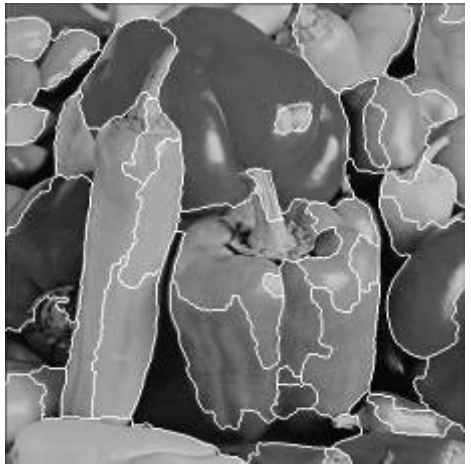
KLT vs. Oliveira

KLT vs. Mean

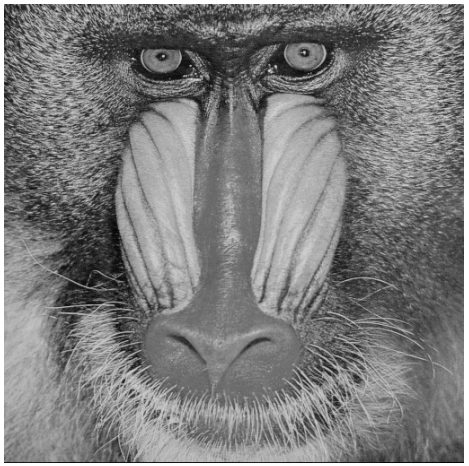
KLT vs. Resynthesizer

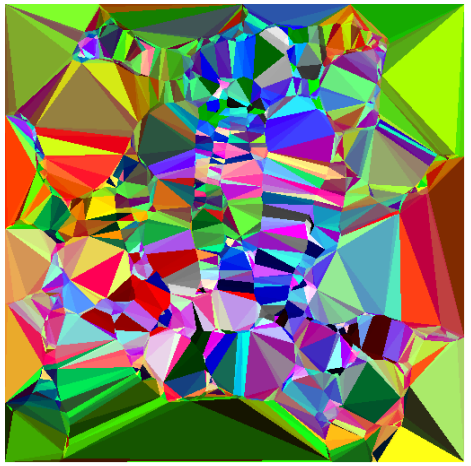
Baboon and Peppers - Segmentation



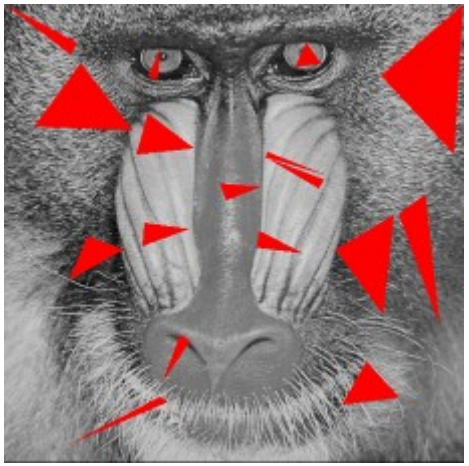


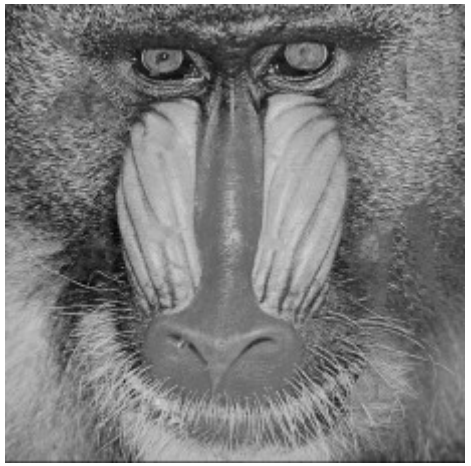
Baboon and Peppers - Triangulation





Baboon and Peppers – Restoration – KLT (D=2, 1 iteration)





Tables

Baboon

	Bertalmio	TV	CDD	Oliveira	Mean	Resynthesizer	KLT (D=2 iter=1)
PSNR [dB]	39.9215	44.3659	**49.5500**	39.7296	48.3711	48.0540	48.0446
CPU time	18 min 13 s	1 min 53 s	2 min 19 s	1.8 s	**250 ms**	42 min	18 s

Stream and Bridge

	Bertalmio	TV	CDD	Oliveira	Mean	Resynthesizer	KLT (D=2 iter=3)
PSNR [dB]	42.5476	41.0333	**49.7810**	41.8134	48.7467	47.1177	48.4396
CPU time	19 min 34 s	2 min 5 s	2 min 34 s	2 s	**280 ms**	17 min	25 s

Peppers

	Bertalmio	TV	CDD	Oliveira	Mean	Resynthesizer	KLT (D=2 iter=1)
PSNR [dB]	47.7050	45.7515	**56.4806**	47.2177	56.2860	57.3806	53.2132
CPU time	17 min 1 s	1 min 7 s	1 min 19 s	1 s	**171 ms**	7 min	16 s

Conclusion

- Comparable results with existing methods
- Faster algorithm than texture synthesis, we are working with segments, not whole image
- Two eigenvectors are enough for approximation
- Appropriate output of segmentation process is required
- Sometimes there are missing spatial information, average value cannot be estimated