Improving Quality of Image Magnification Based on Data-Dependent Triangulation

Michal Červeňanský

Motivation

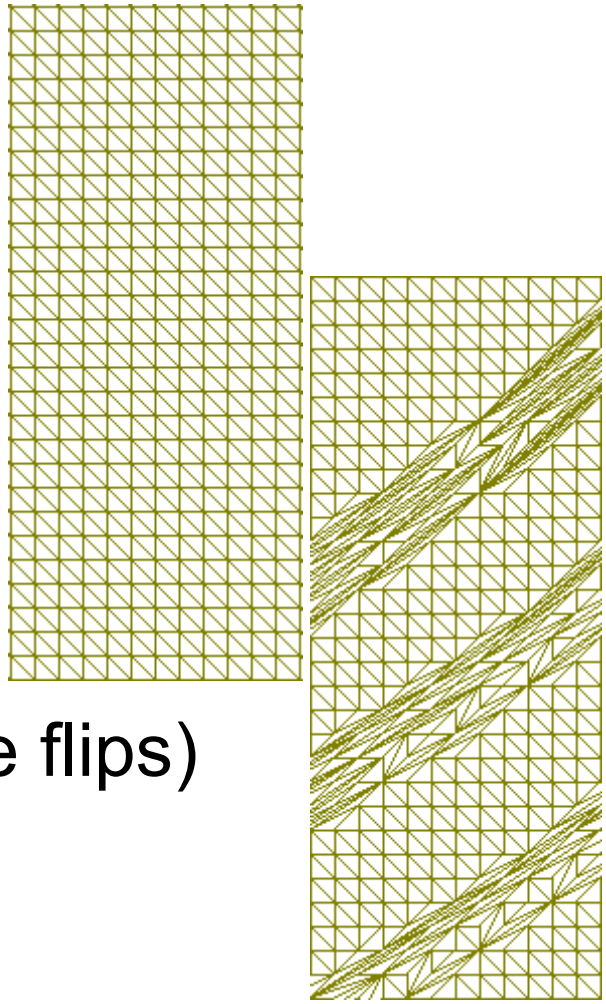
- Image Magnification
  - Convolution
  - Data-Dependent Triangulation
- New parallel algorithm
  - Fast (GPU)
  - Some artefacts
- Improve Quality
  - Various methods

Overview

- Motivation
- Previous Work
  - Data-Dependent Triangulation
  - Algorithm
  - Results
- Improving Quality
- Summary
- Future Work

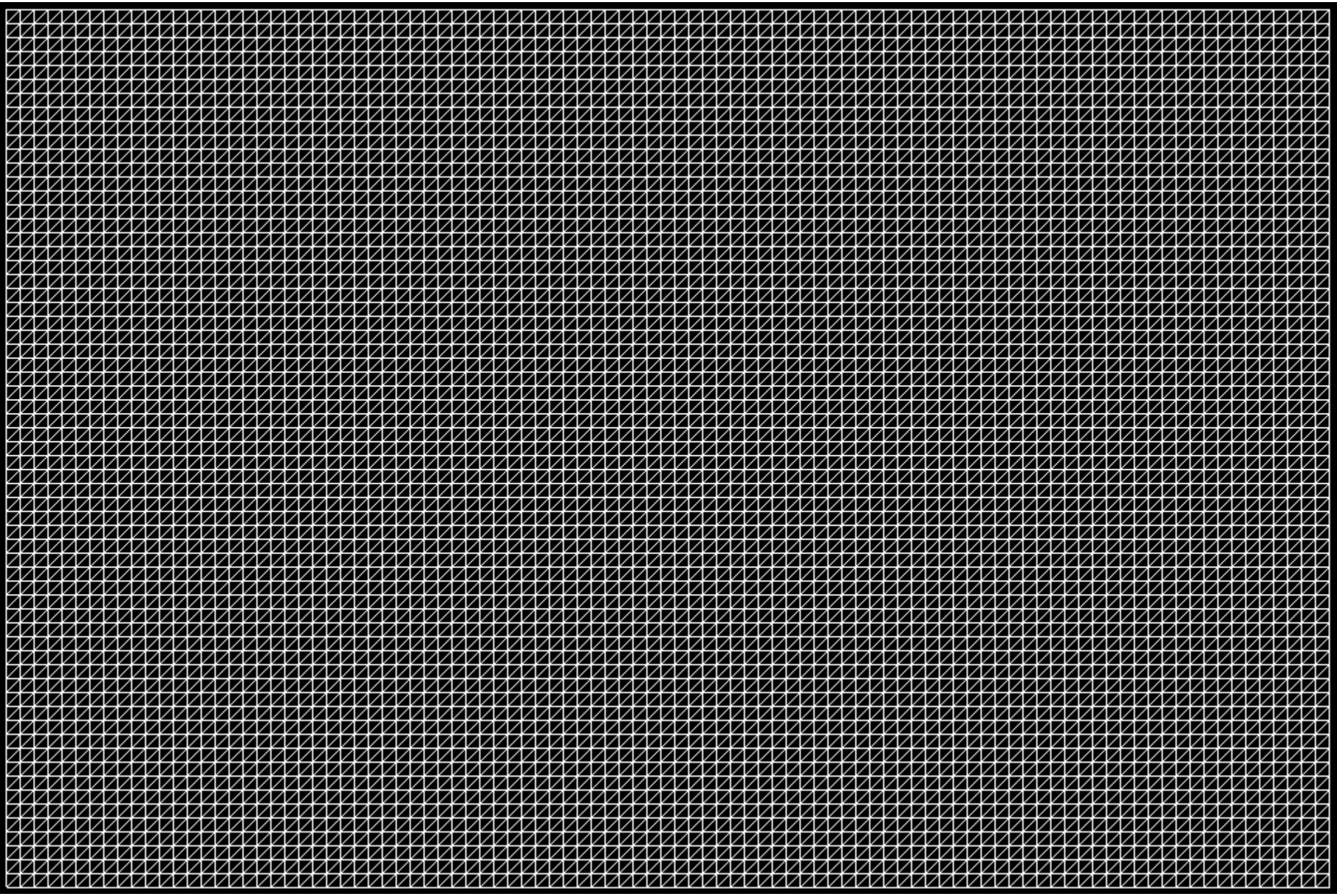
Data Dependent Triangulation DDT

- Triangulation
  - Position (Delanuay, ...)
  - Data component (MWT, …)
  - Locally optimal triangulation
- Lawson optimization process
  - Iterative
  - Topological operations (edge flips)
  - Cost function (optimization)
  - Sederberg's cost function



Algorithm

- Iterative process
  - Creation of candidates
    - based on cost function
  - Acceptance and rejection of candidates
    - handling collisions (iterative step)
  - Edge flipping
- Results processing
  - Save to disk
  - Visualize

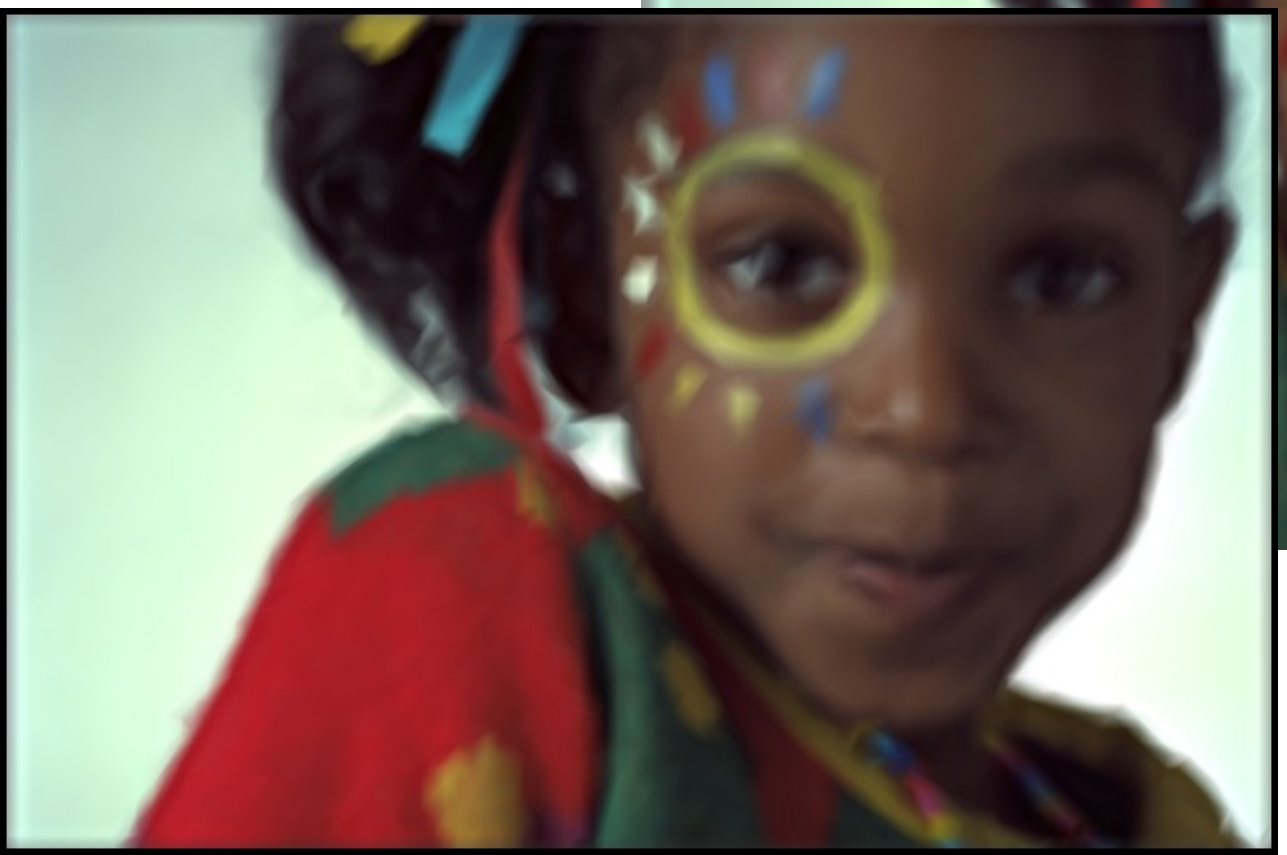


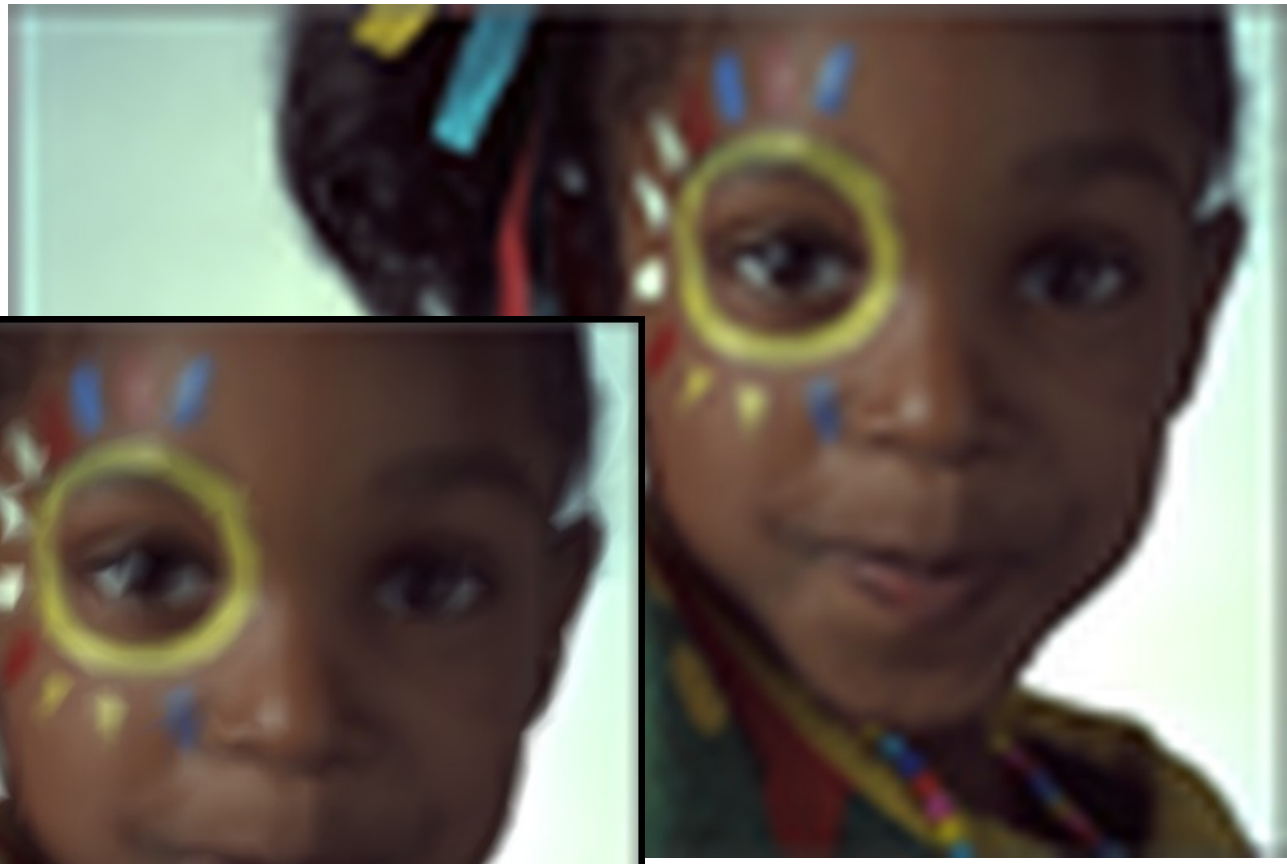
Results

- Various modifications (cost, quality)
- Reconstruction techniques
  - CPU DDT, Lanczos, bilinear, b-spline
- Perceptual metrics
  - Quality measurements
- Dataset
  - 12 real, 6 artificial images
- Results
  - Speed-up 6-10x over CPU DDT
  - Competitive to convolution techniques
    - Lanczos → CPU DDT → GPU DDT

Results (2)

DDT

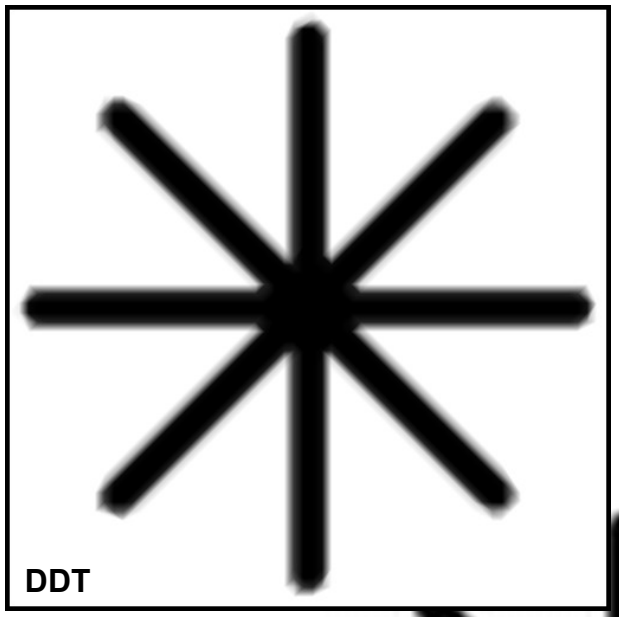


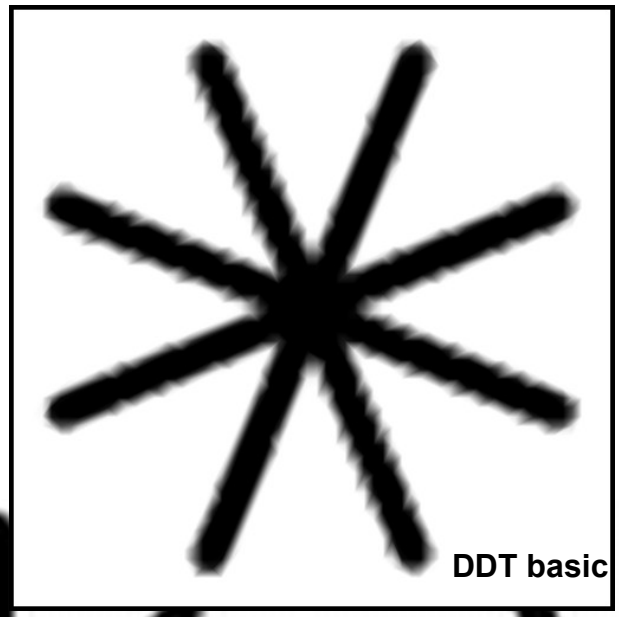


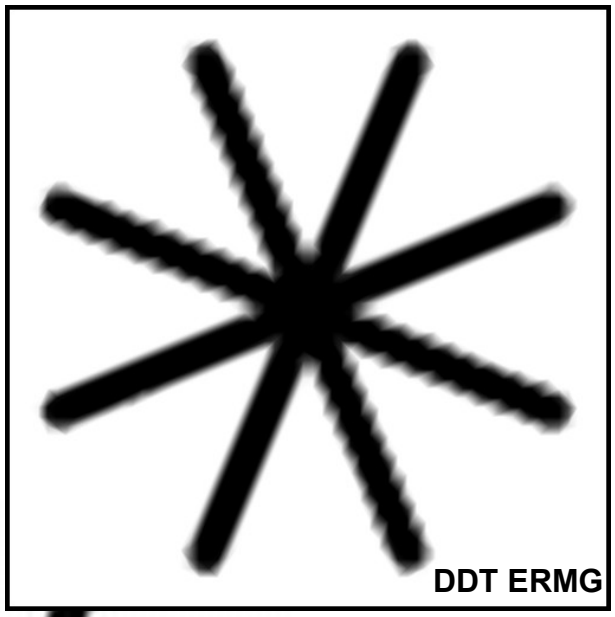
Lanczos

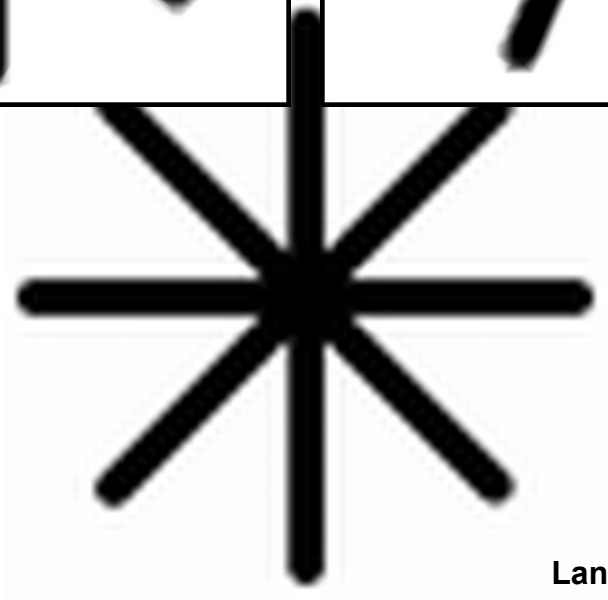
Results - artefacts

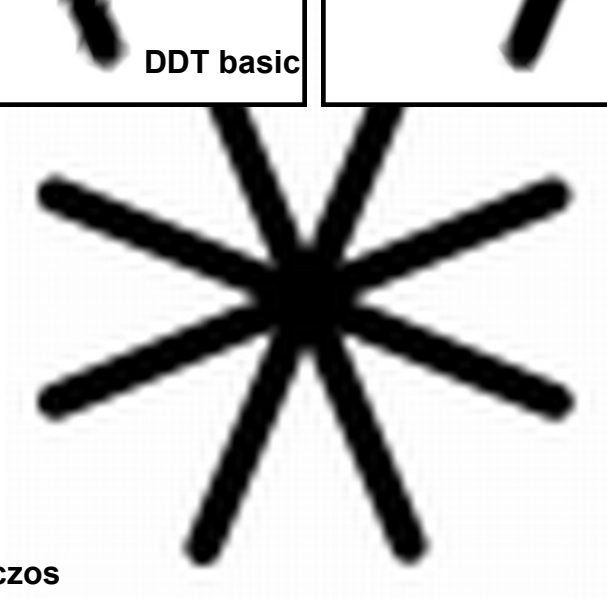
Jaggy artefacts on some edges (800% magnification)











Overview

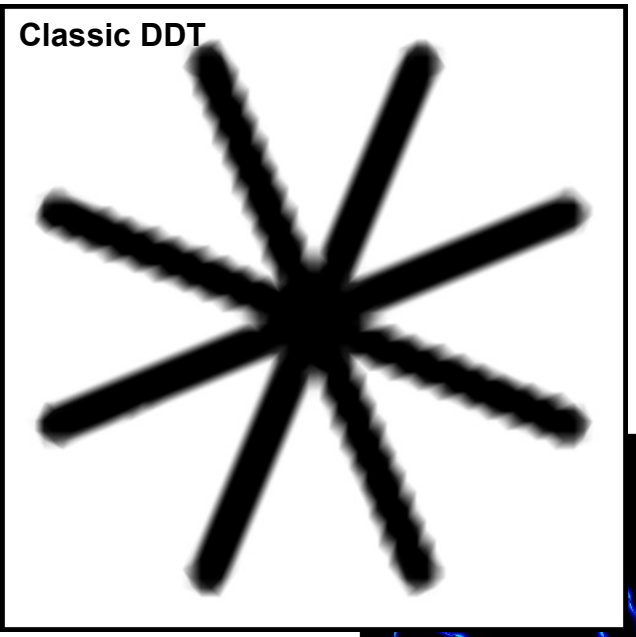
- …
- Previous Work
- Improving Quality
  - Rotation
  - Sequential magnification
  - Mesh preprocessing
  - Combination
- Summary
- …

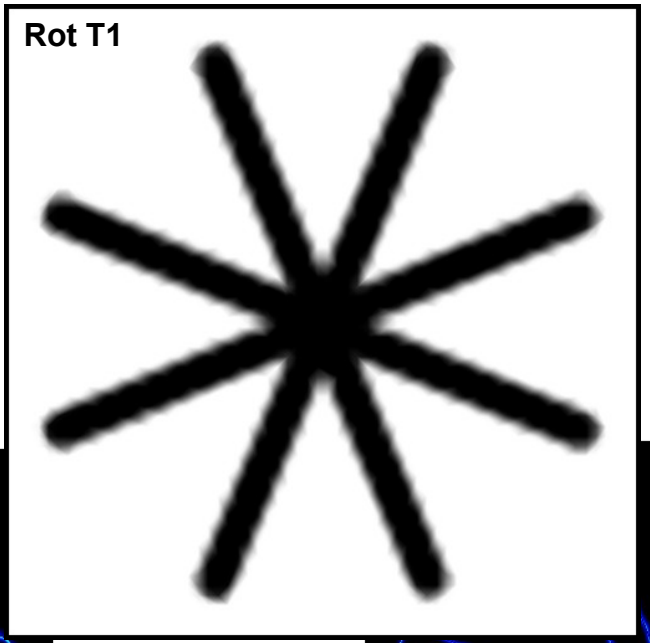
Rotation

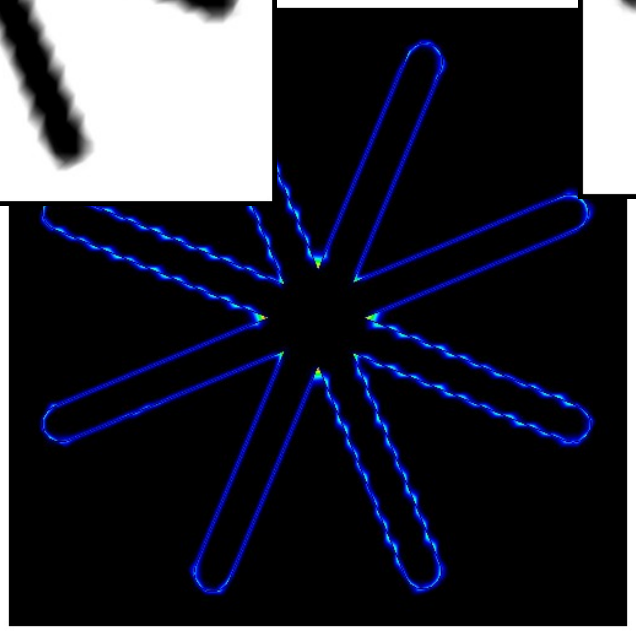
- Reconstruction based on
  - Rotated images (0,90,180,270 dg.)
  - Flipped images (horizontal/vertical flip)
- Subresults → blended

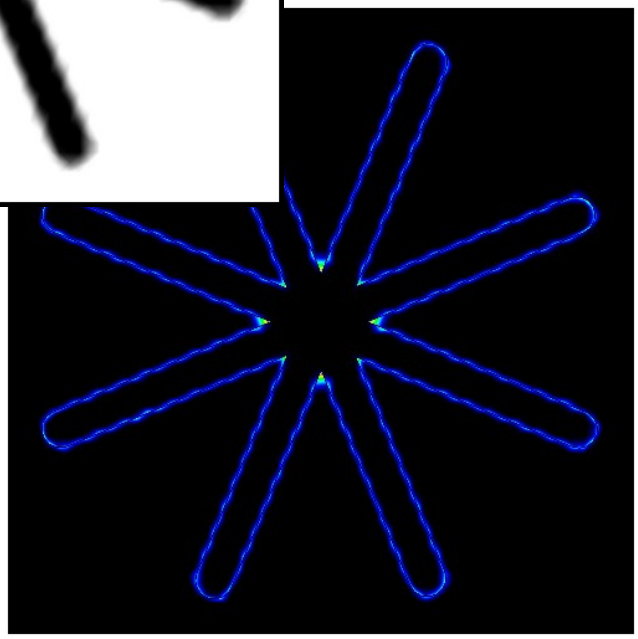
Rotation - results

Artefacts – soften on „bad edges“, appear on „good edges“









Rotation – results (2)

- T1 – 0+90+270+360 dg.
- T2 – 0+90 dg.
- T3 – 0+Horizontal Flip
- T4 – 0+Vertical Flip

- Irrelevant improvement
  - Other type of blending (analysis)

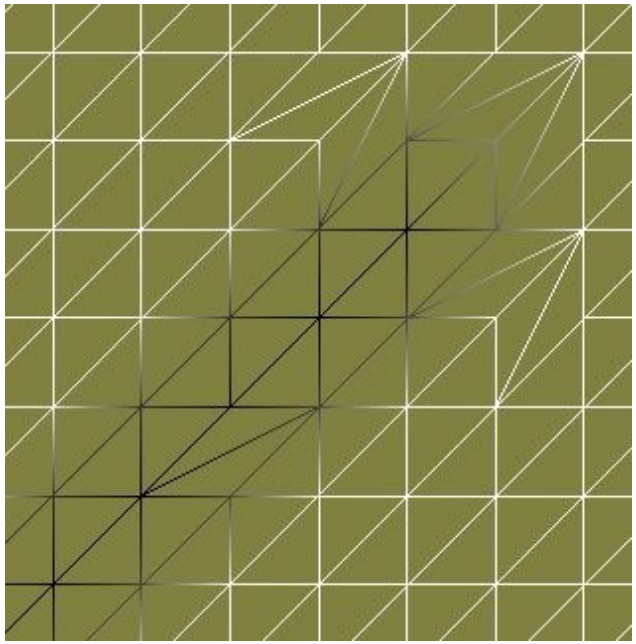
ExpRoiMaxGain					
Name	**DDT**	**Rot T1**	**Rot T2**	**Rot T3**	**Rot T4**
UIQI	0,480	0,481	0,481	0,481	0,481
CrossCorr	0,960	0,961	0,961	0,961	0,961
MSE	274,896	270,197	270,200	270,202	270,202
SNR	117,590	119,338	119,337	119,338	119,337
PSNR	338,278	343,271	343,270	343,271	343,270
Correlation	96,820	96,835	96,835	96,835	96,835

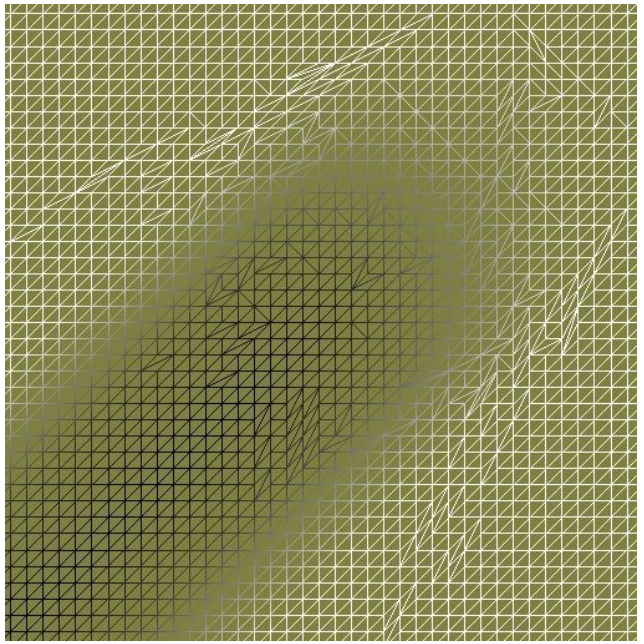
Sequential magnification

- Sequential magnification by 2x
  - Suppress “toothed triangles”
- More pixels → better approximation

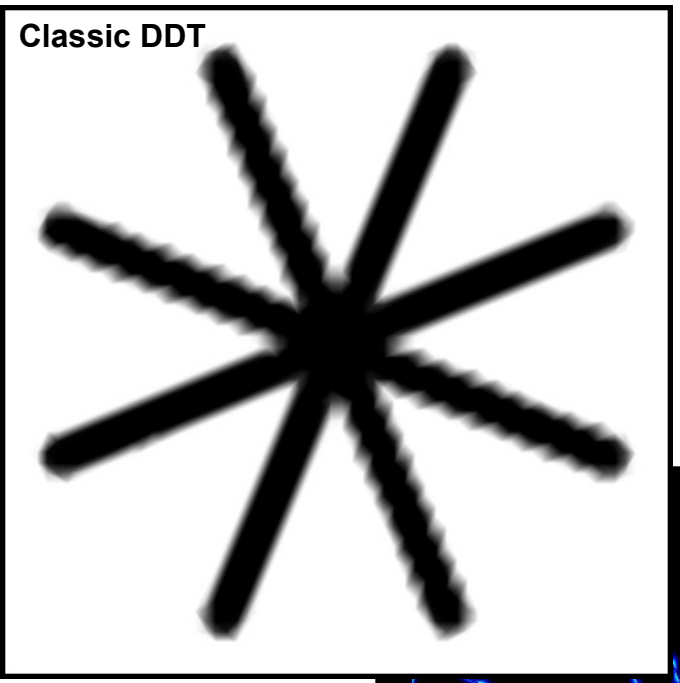
Sequential magnification results

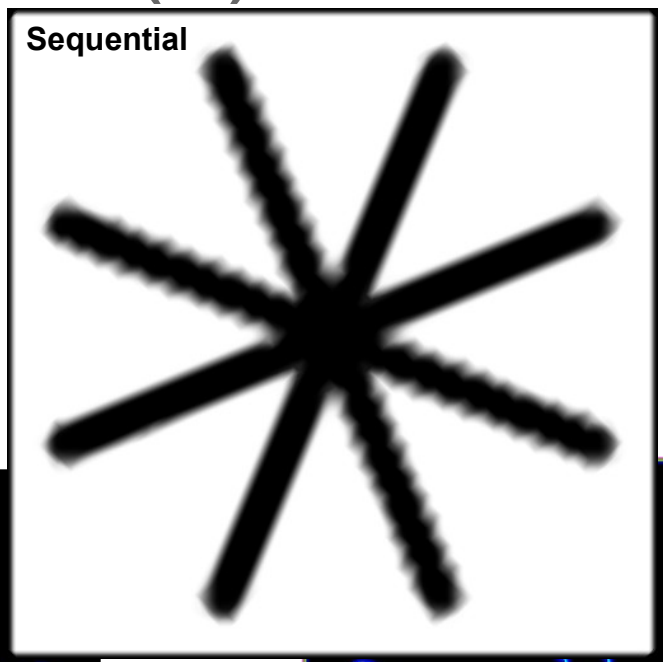
- Smoother edges
  - Convex, concave areas
  - Higher magnification

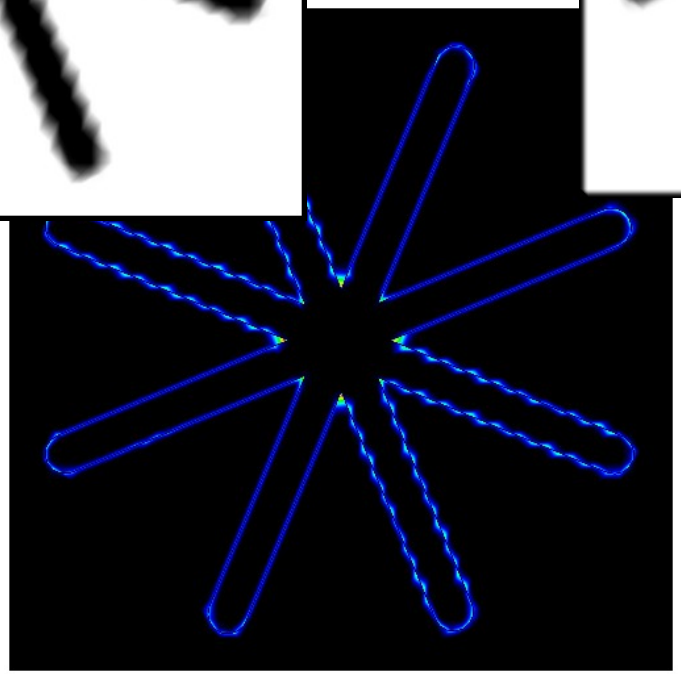


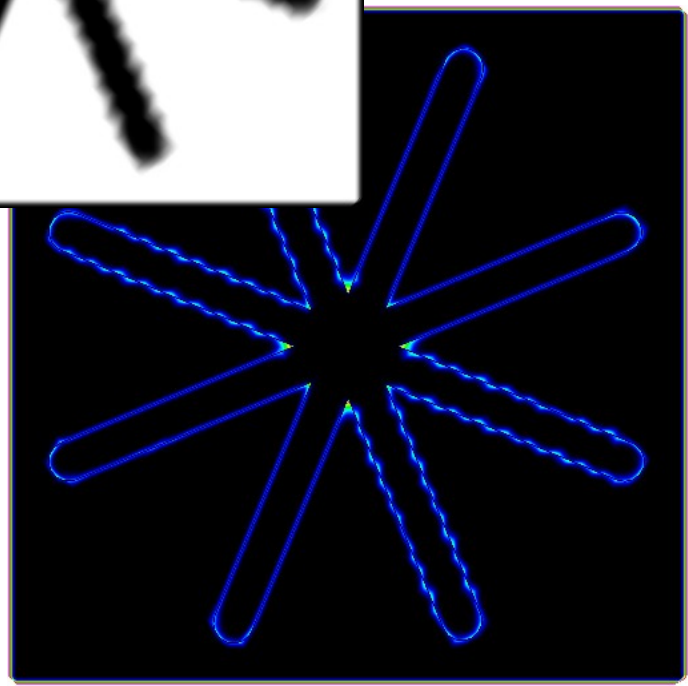


Sequential magnification results (2)









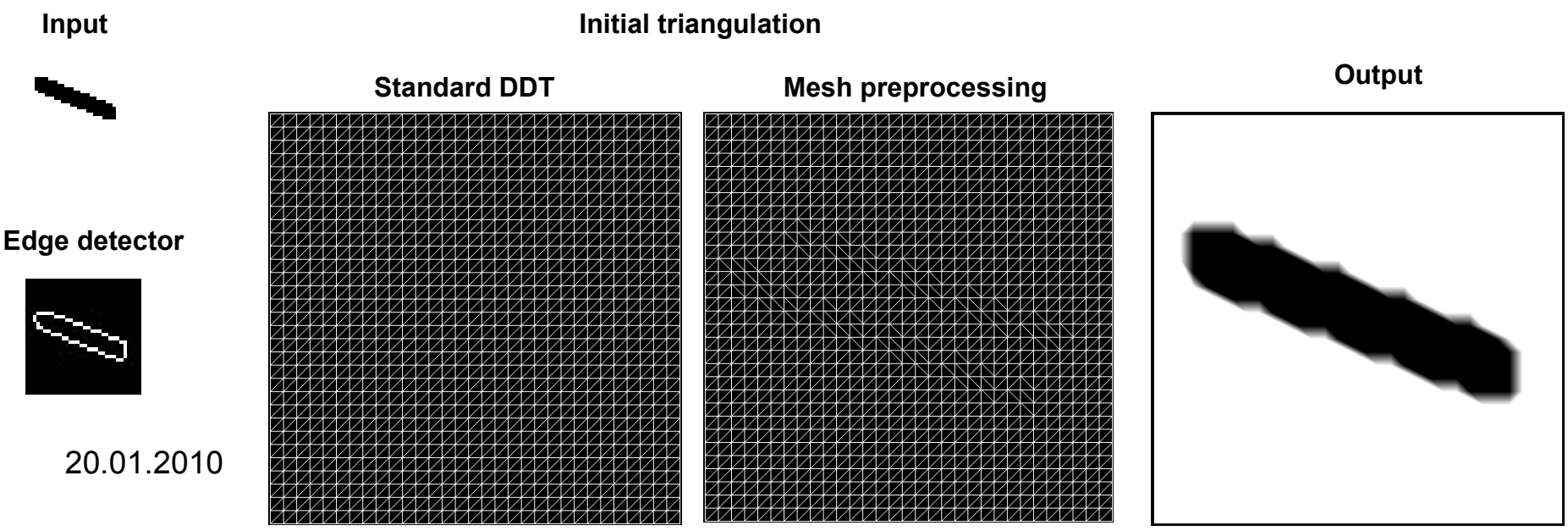
Sequential magnification results (3)

- Worse results

ExpRoiMaxGain		
Name	**DDT**	**Zvacsenie**
UIQI	0,480	0,466
CrossCorrelation	0,960	0,958
MSE	274,896	287,908
SNR	117,590	112,054
PSNR	338,278	322,637
Correlation	96,820	96,714

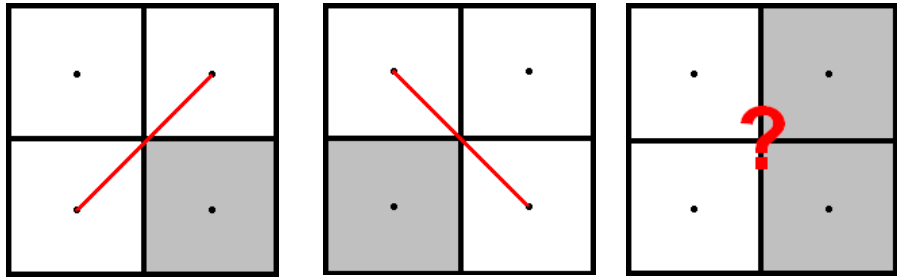
Mesh preprocessing

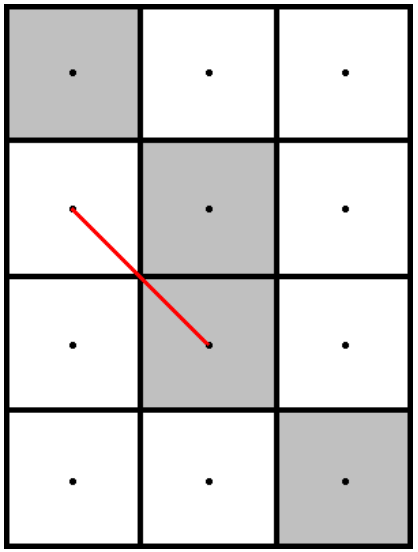
- Initial mesh
  - Homogeneous
  - Flip diagonals (edge detector)
  - Local / Global optimum



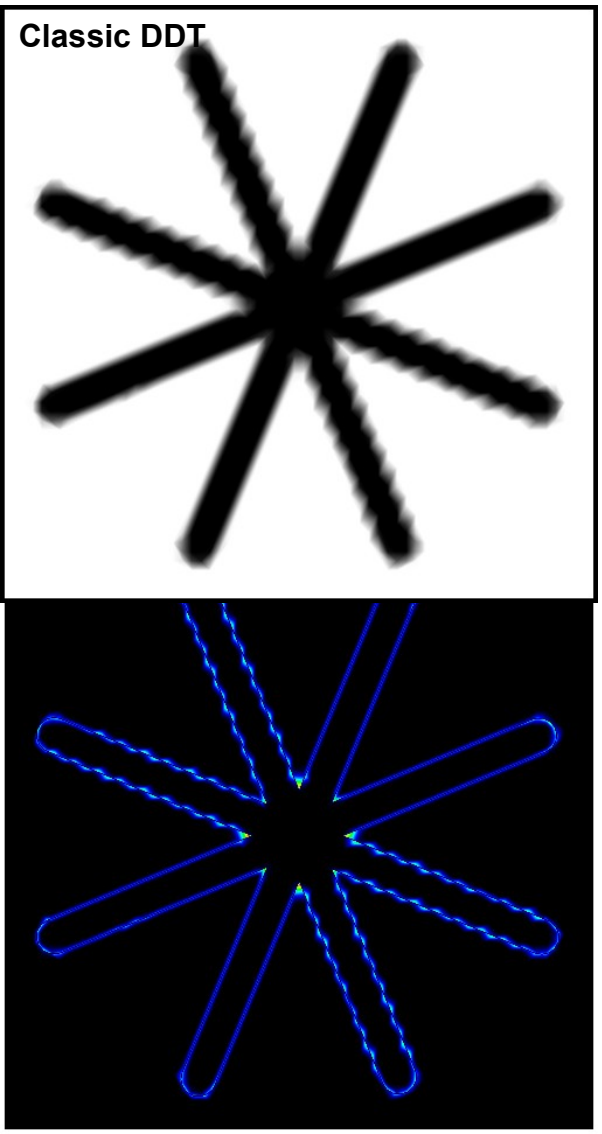
Mesh preprocessing (2)

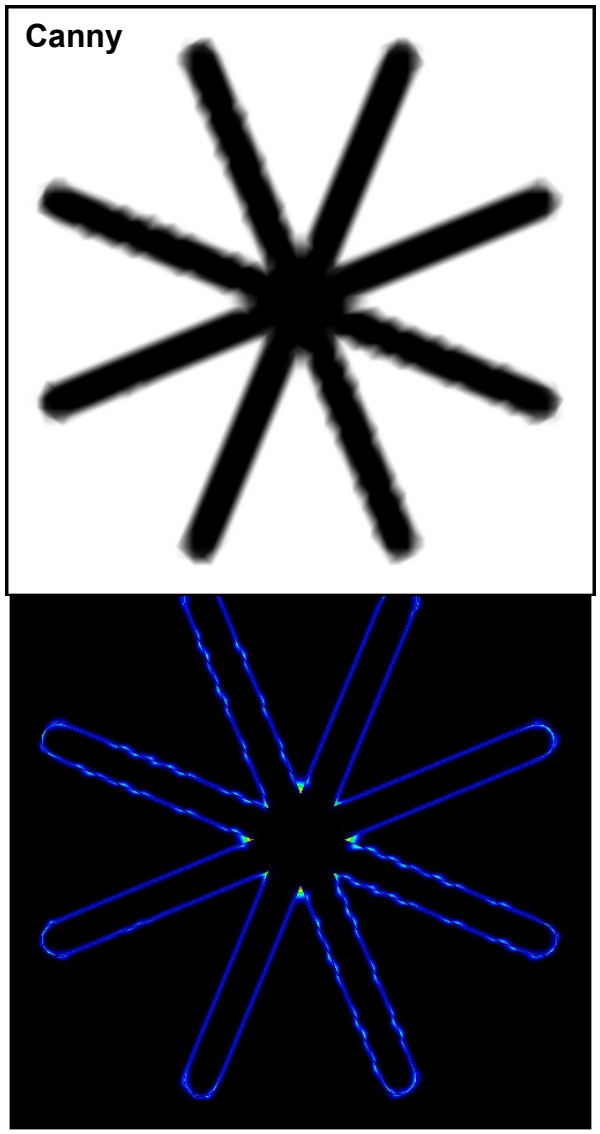
- Canny edge detector
  - 4 neighbours
  - Ambiguity
- Sobel edge detector
  - Maximal vector
  - Average vector

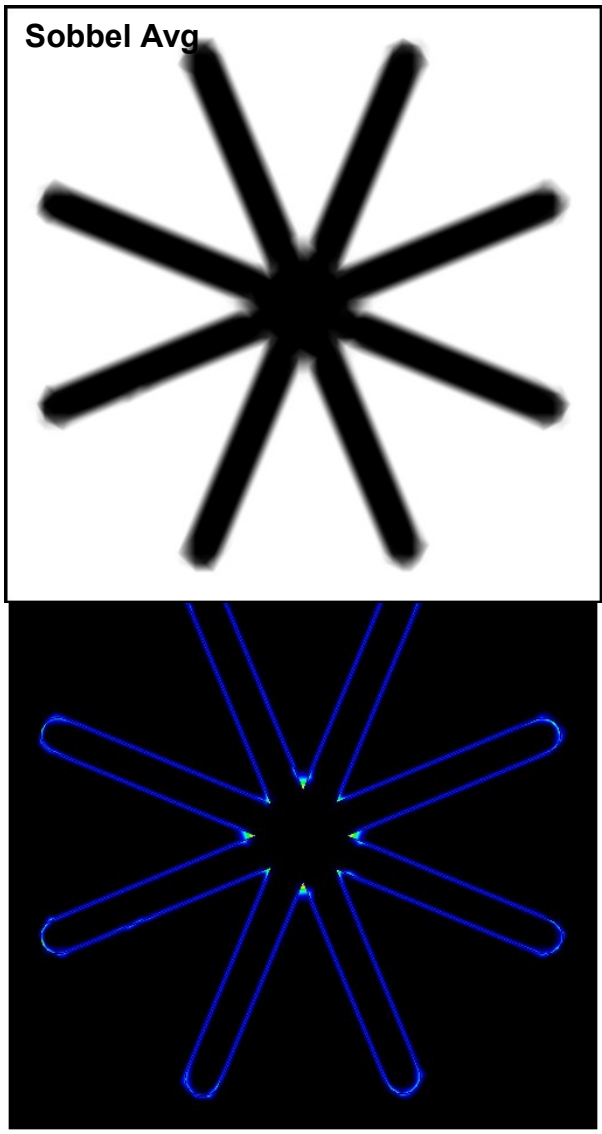




Mesh preprocessing results







Mesh preprocessing results (2)

- Better quality (visible)
- Lower cost
- Less iterations

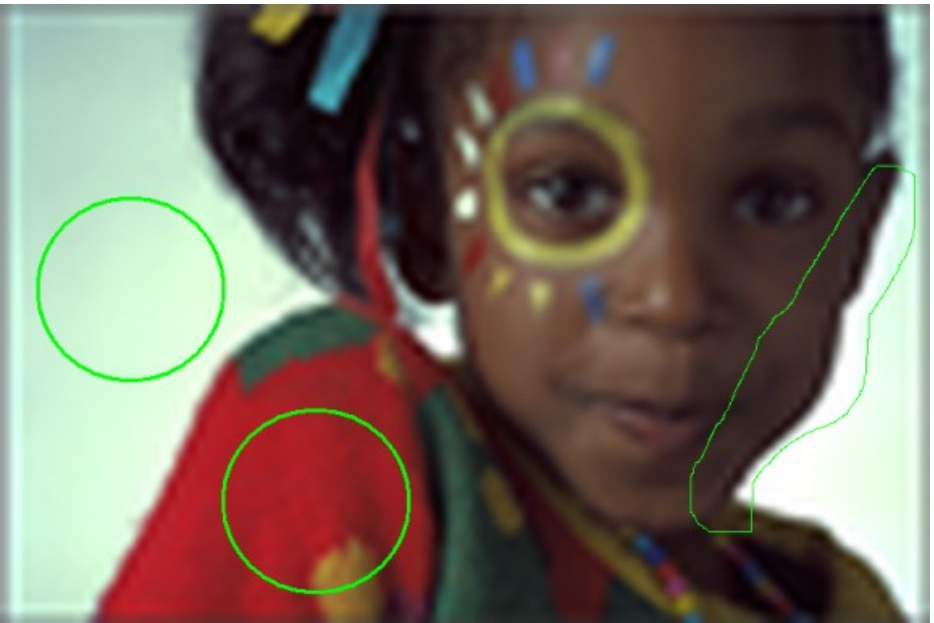
ExpRoiMaxGain				
Name	**DDT**	**Canny**	**SobMax**	**SobAvg**
UIQI	0,480	0,480	0,482	0,481
CrossCorrelation	0,960	0,960	0,961	0,961
MSE	274,896	272,788	268,451	269,742
SNR	117,590	118,450	120,579	119,946
PSNR	338,278	340,363	346,012	344,111
Correlation	96,820	96,825	96,847	96,842

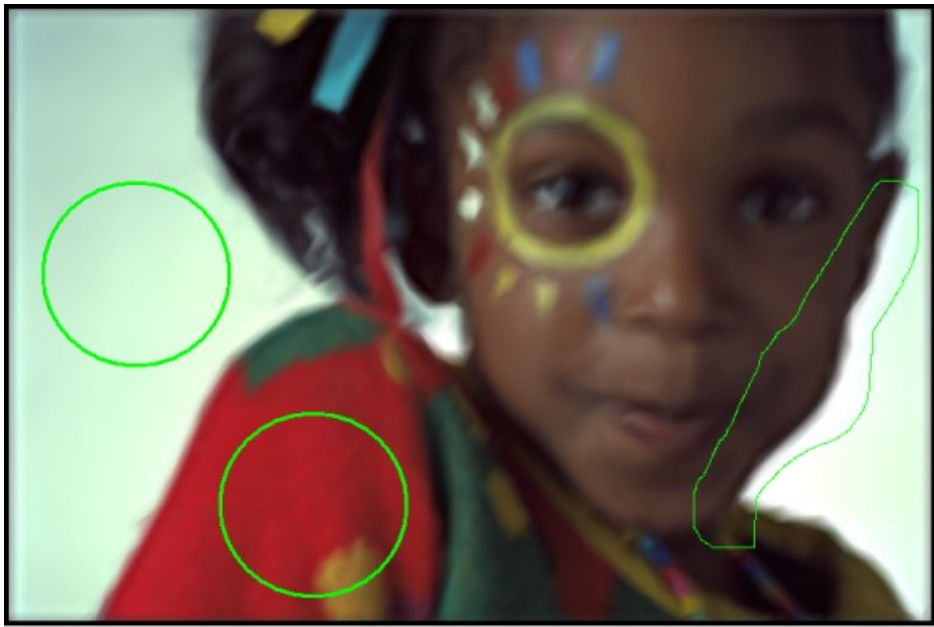
Overview

- …
- Improving Quality
  - Combination
    - Scheme
    - Implementation
    - Results
- Summary
- …

Combination

- DDT and Convolution techniques
  - DDT (slow) → non-homogeneous areas
  - Convolution (fast) → homogeneous areas





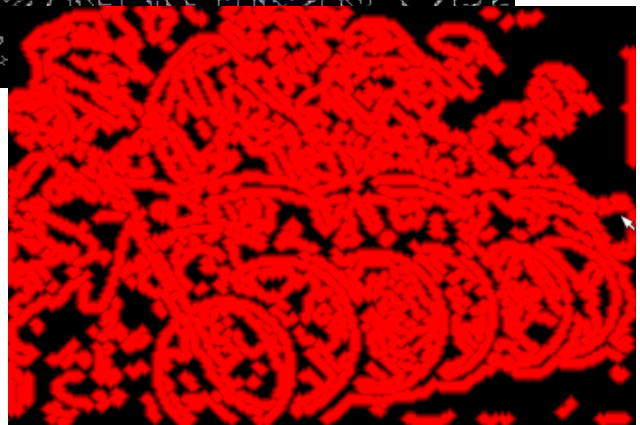
Scheme

Input

Edge detector

Vectorized

Mask



DDT

Convolution

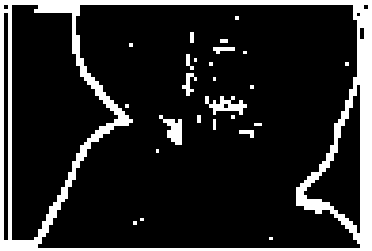
Output

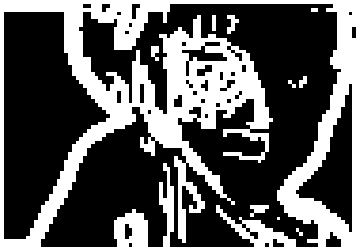
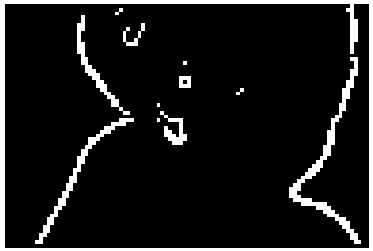
Implementation convolution

- Box, Linear, Cubic B-Spline, Catmul-Rom, Lanczos

Implementation edge detector

- Canny (Fung), Sobbel, Prewit, Roberts, Laplace
  - Gaussian blur
  - Various thresholds



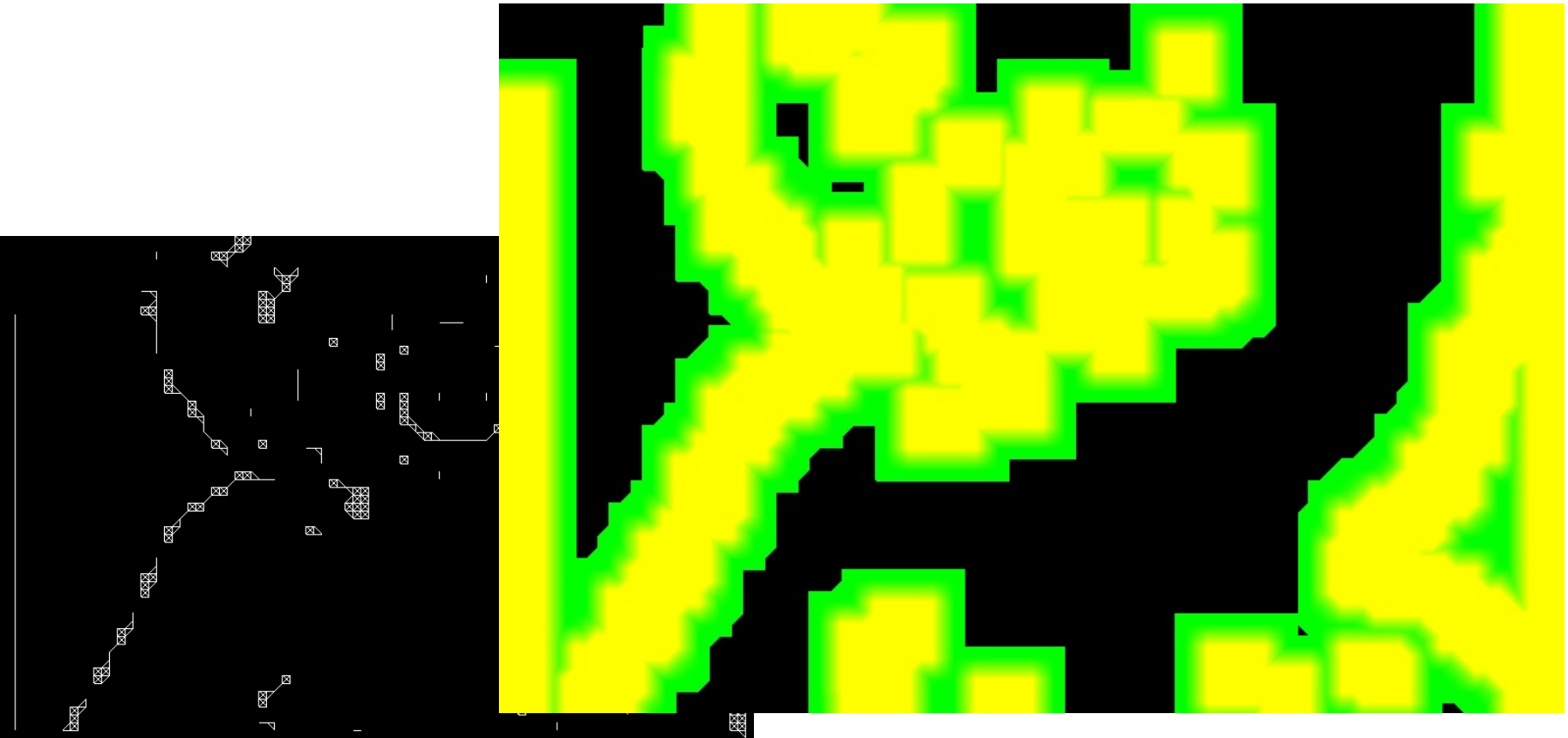




Implementation vectorization

- Edge detector → vectorization → mask (dilatation)
- Edge detector → mask (dilatation) → magnify
- Edge detector → magnify → mask

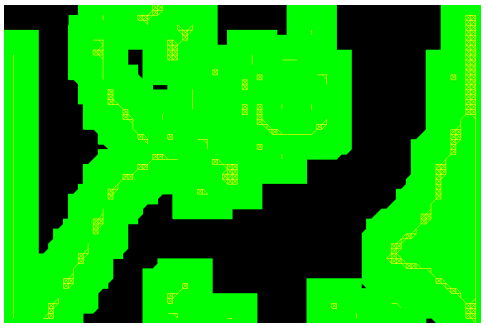
Implementation mask

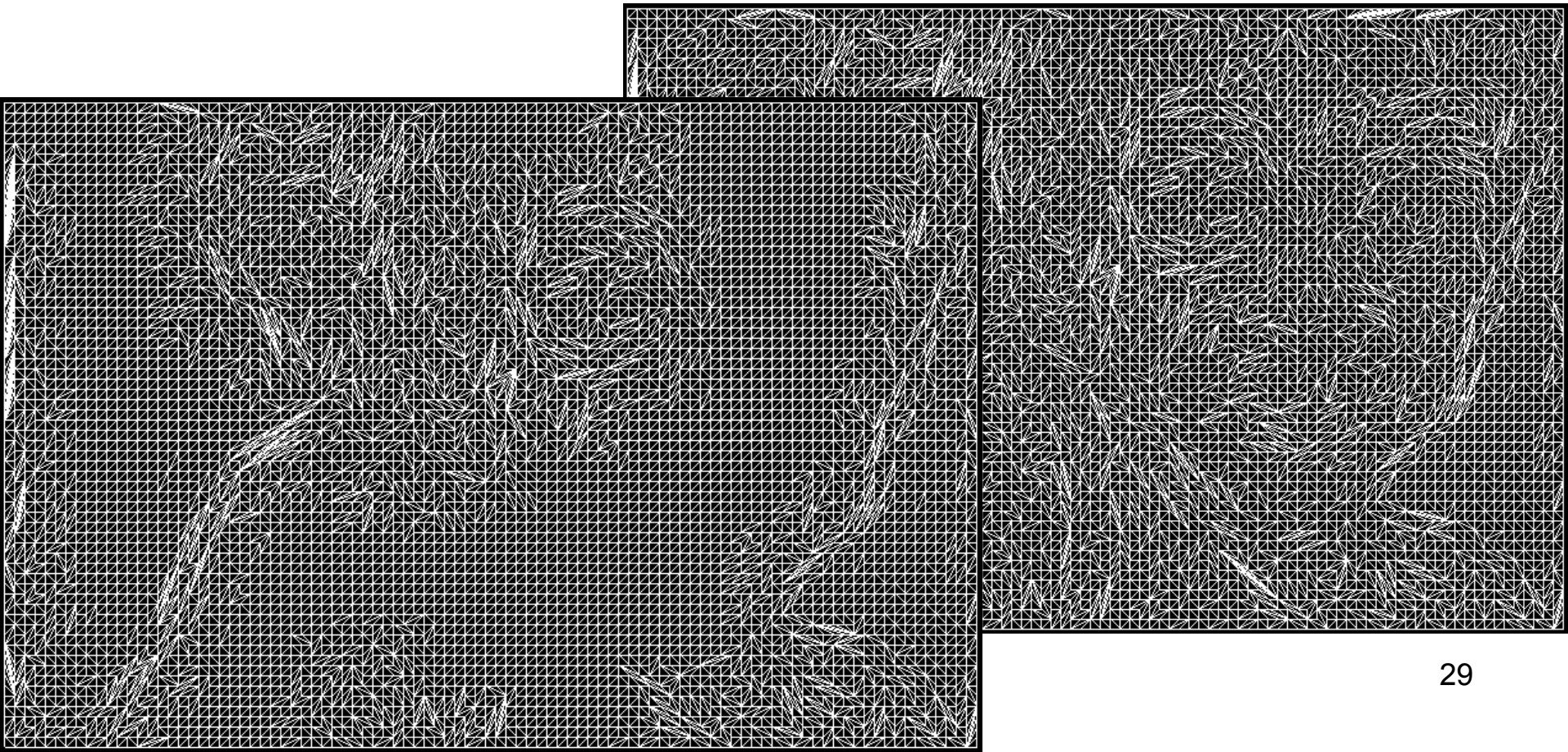
- Masks – DDT (wide), Combination (narrow)
- 4 steps – x, y, x-slope, y-slope
- Size based on zoom factor



Implementation
DDT

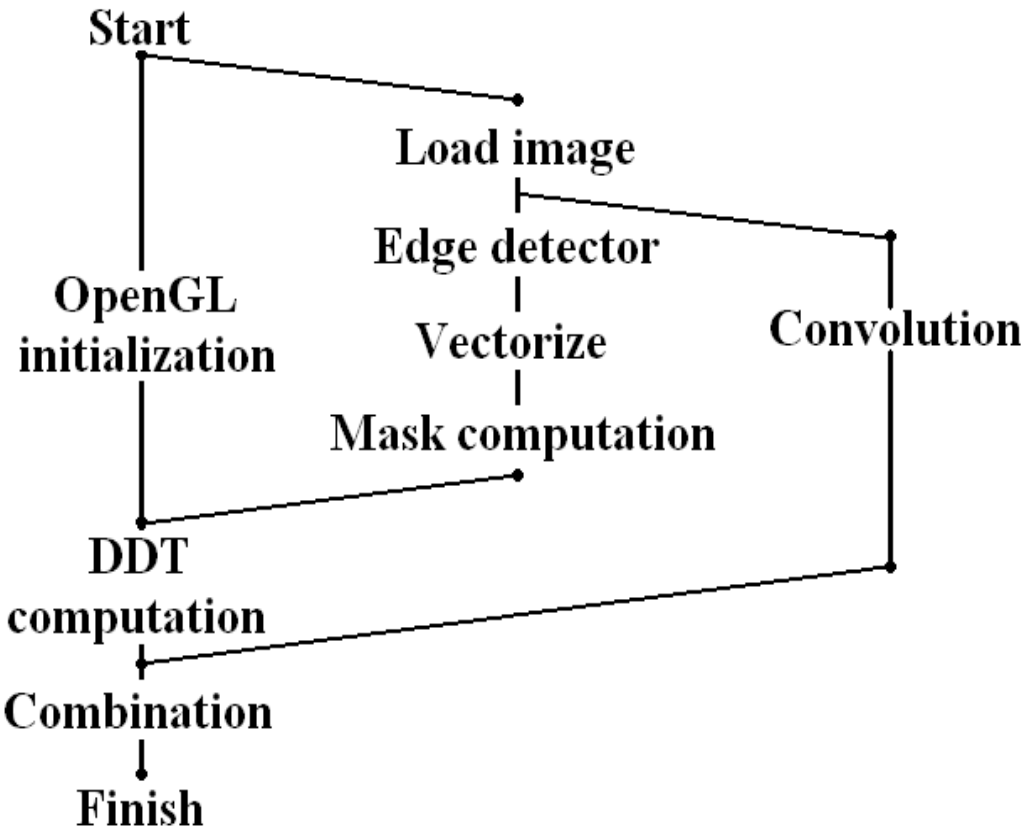
- DDT computation
  - Non-homogeneous area (DDT mask)





Implementation

- All computations on GPU
  - Cg, glsl implementation
- Threads (pthreads)
  - CPU+GPU

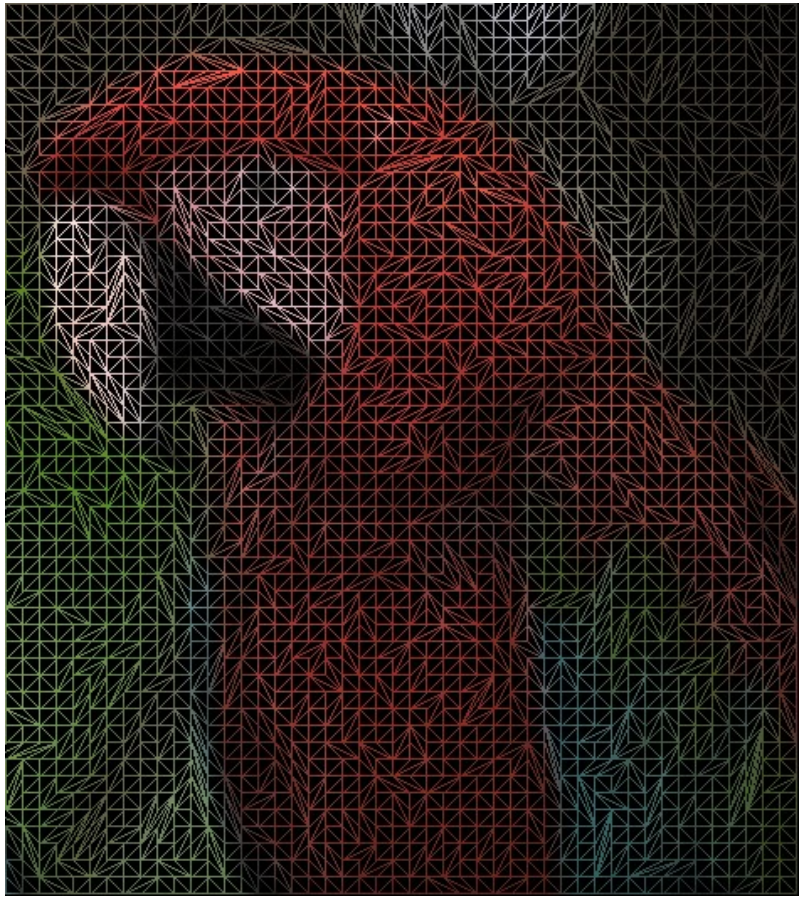


Results

- Many parameters
  - Suitable setting
  - Not yet tested

Summary

- Various methods
  - Image based methods
    - Small contribution
  - Computation based
    - Satisfactory



Future Work

- Algorithm optimization
  - Computations on meaningful fragments
  - Faster computation
  - Reordering of candidates
    - slow step
- OpenCL
  - More possibilities (memory management)
- New Algorithm
  - Not universal → Image magnification
  - Based on edge colouring

Thank you