Inpainting-Based Image Compression

Joachim Weickert
Mathematical Image Analysis Group
Saarland University
Saarbrücken, Germany

long term research funded by

Deutsche Forschungsgemeinschaft DFG

Gottfried Wilhelm Leibniz Prize

erc

ERC Advanced Grant INCOVID

Collaborators

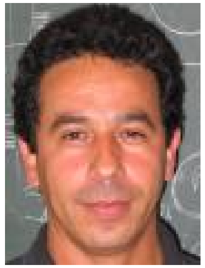

Tobias Alt


Naoufal Amrani

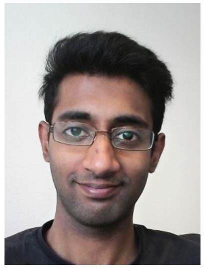

Sarah Andris


Matthias Augustin


Egil Bae


Zakaria Belhachmi


Matthias Berg


Pinak Bheed


Andrés Bruhn


Dorin Bucur


Marcelo Cárdenas


Vassillen Chizhov


Viktor Daropoulos


Martin Fuchs


Irena Galić


Laurent Hoeltgen


Sebastian Hoffmann


Ferdinand Jost


Niklas Kämper


Lena Karos


Markus Mainberger

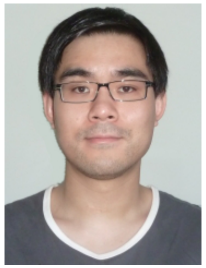

Rahul Mohideen


Pascal Peter


Gerlind Plonka


Christian Schmaltz


Joan Serra-Sagrista


Ching Hoo Tang


Martin Welk

Introduction

Inpainting with Diffusion

original

inpainted

Left: Colour image (438×297 pixels) with severe degradations by a text. From Bertalmio et al. (2000). **Right:** Result of inpainting with an anisotropic diffusion process.

How Far Can This be Pushed ?

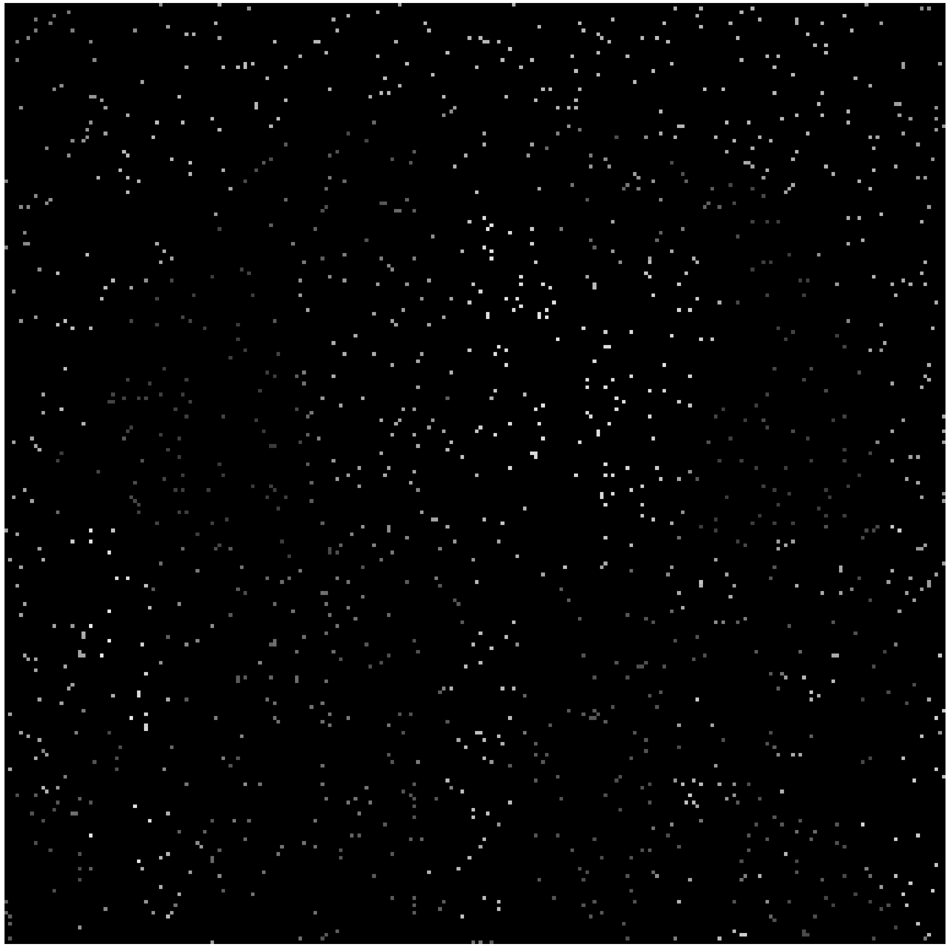
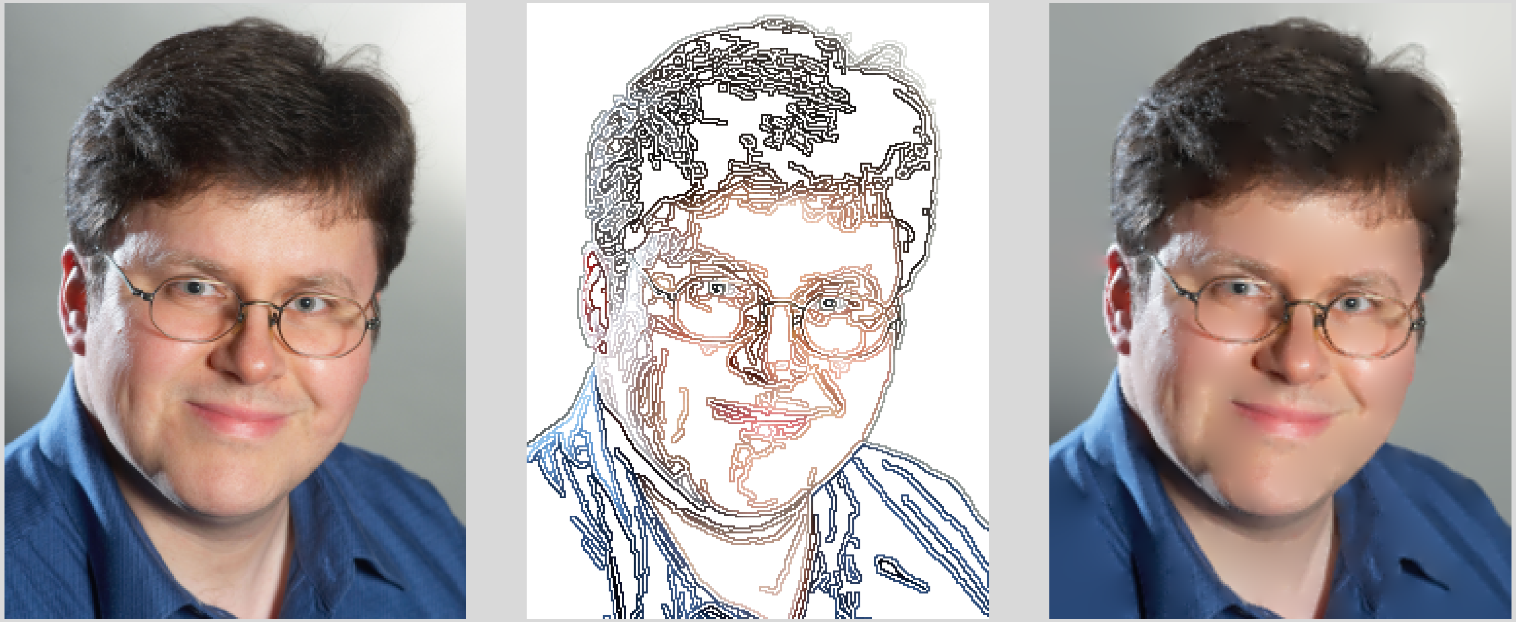


Image where only 2 percent of all pixels are known. Can you recognise what is depicted ?

What Happens if We Select Semantically More Important Data ?



original, 237×316 | data near edges kept | inpainted

Inpainting from contour data by solving the homogeneous diffusion equation in each RGB channel.

Do We Still Need Data Compression Methods ?

- Creating huge amounts of digital images and videos has never been easier. However, also memory has become much cheaper.
- How many hours of 4K video with 60 fps can we store with 128 GB in *raw* format, i.e. with 3 bytes per RGB pixel?

Do We Still Need Data Compression Methods ?

- Creating huge amounts of digital images and videos has never been easier. However, also memory has become much cheaper.
- How many hours of 4K video with 60 fps can we store with 128 GB in *raw* format, i.e. with 3 bytes per RGB pixel?

Only 86 seconds !

- Data compression is necessary to reduce the file size for storing and transmission.
- For high compression rates, we need inexact methods *(lossy codecs).*
- Common lossy codecs such as JPEG, JPEG 2000, HEVC, BPG:
  aim at simplified representations in a transform domain (DCT, wavelets)

A Popular Method for Lossy Image Compression: JPEG

- decomposes image into 8×8 pixel blocks
- frequency representation within each block (DCT)
- higher frequencies are coded with coarser quantisation, requiring less bits
- typical compression rate: 10 : 1
- quality deteriorates significantly for high compression rates

original (256 $\times$ 256)

compression rate 10 : 1

compression rate 40 : 1

Introduction (6)

We need methods that perform better for high compression rates.

Our Goals

- Explore the potential of inpainting for image compression:
  Keep some selected pixels and reconstruct the missing ones by inpainting.
- looks simpler and more intuitive than transform-based methods
- justified from biology: brain uses inpainting-like filling-in (Werner 1935)
- generic framework for various data types:
  - 1-D, 2-D, 3-D data
  - scalar-, vector-, matrix-valued
  - still images and videos
  - surface data, graphs
  - dedicated codecs for specific applications
- challenging demonstrator for inpainting quality and speed

What are the Challenges ?

- We have to solve several difficult problems:
  1. Which pixels should be kept ?
  2. Which inpainting process gives a good restoration ?
  3. How can the selected pixels be encoded in an efficient way ?
  4. Can one prove theoretical results ?
  5. Are the sufficiently fast algorithms for encoding and decoding ?
- These problems are interrelated:
  - Optimal pixels also depend on the inpainting process.
  - Suboptimal pixels can pay off if they are cheaper to encode.
  - There is a natural tradeoff between quality and speed.

How Does this Differ from Some Related Approaches?

- **Classical Inpainting Methods:**

  e.g. Masnou/Morel 1998, Bertalmio et al. 2000, Efros/Leung 1999

  rarely used for very sparse data; do not optimise inpainting data

- **Scattered Data Interpolation Methods, Radial Basis Functions:**

  e.g. Shepard 1968, Duchon 1976, DiBlasi et al. 2010, Achanta et al. 2017

  no data optimisation; usually not applied to compression

- **Image Compression with Subdivision:**

  e.g. Dyn et al. 1990, Sullivan et al. 1994, Distasi et al. 1997, Demaret et al. 2006

  often very simple inpainting (linear splines); more localised

- **Reconstructions from Image Features:**

  e.g. Johansen et al. 1986, Carlsson 1988, Weinzaepfel et al. 2011

  usually not for compression purposes; suboptimal for general imagery

My Strongest Motivation ...

"Forget about these ideas. You will never beat JPEG and JPEG 2000."

Stanley Osher, 2005.

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52

Outline

- Optimising the Data
- Optimising the Inpainting Operator
- Optimising the Encoding
- Theoretical Achievements
- Fast Algorithms and Real-time Aspects
- Extensions and Applications
- Summary and Outlook

Outline

- Optimising the Data
- Optimising the Inpainting Operator
- Optimising the Encoding
- Theoretical Achievements
- Fast Algorithms and Real-time Aspects
- Extensions and Applications
- Summary and Outlook

Optimising the Data

Our Baseline Codec

- **Encoding**:
  - store (greyscale) image $f\colon \Omega \to \mathbb{R}$ only in some small subset $K \subset \Omega$.
- **Decoding**:
  - In this subset K *(inpainting mask)*, the reconstruction u is known:

$$u(\boldsymbol{x}) \;=\; f(\boldsymbol{x}).$$

  - In the domain $\Omega \setminus K$ where the data are unknown:
    Compute the steady state $(t \to \infty)$ of the homogeneous diffusion equation

$$\partial_t u \;=\; \Delta u$$

    with the known data as fixed boundary conditions
  - In short: Inpaint with the Laplace equation $\Delta u \;=\; 0.$

Continous Optimisation of the Data Set K

(Belhachmi et al. 2009)

- established analytic theory based on the mathematics of shape optimisation
- result: choose density of the data points as an increasing function of $|\Delta f|$
- real-time encoding: just compute Laplacian and apply dithering

Felix Klein

Laplacian magnitude

10% mask by dithering

inpainting

A Better Performing Alternative: Discrete Data Optimisation

(Mainberger et al. 2012)

- Solving the discrete data selection problem exactly is combinatorically hopeless: Selecting the best $5\ \%$ pixels of a 256×256 image offers
  $$\binom{65536}{3277} \approx 1.72 \cdot 10^{\,5648}$$ possibilities.

- realistic: optimisation heuristics for finding a good suboptimal solution

A Better Performing Alternative: Discrete Data Optimisation

(Mainberger et al. 2012)

- Solving the discrete data selection problem exactly is combinatorically hopeless: Selecting the best $5\ \%$ pixels of a 256×256 image offers
  $\binom{65536}{3277} \approx 1.72 \cdot 10^{\,5648}$ possibilities.

- realistic: optimisation heuristics for finding a good suboptimal solution

- probabilistic sparsification:
  - start with full mask
  - gradually discard pixels that do not give large errors when they are removed

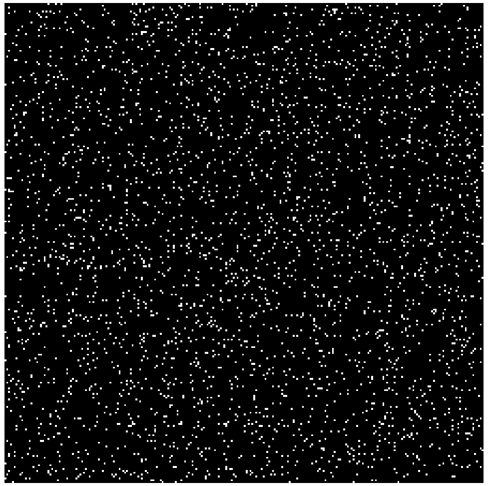
- further improvements by
  - nonlocal pixel exchange to escape from bad local minima
  - tonal optimisation: optimise also the grey / colour values

- MSE reduction by factor 10 compared to unoptimised random mask

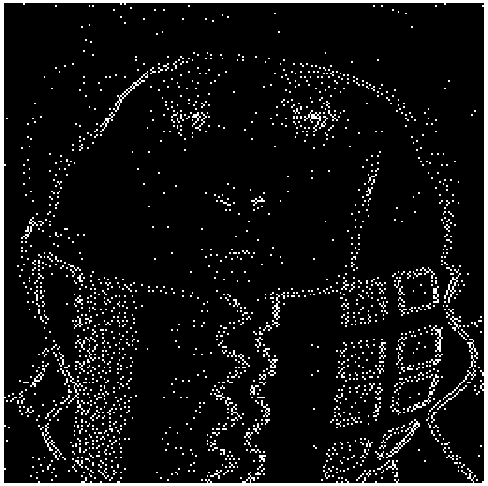
Example: Quality Improvements through Discrete Data Optimisation

original

random 5 % mask



inpainting, MSE=189.90

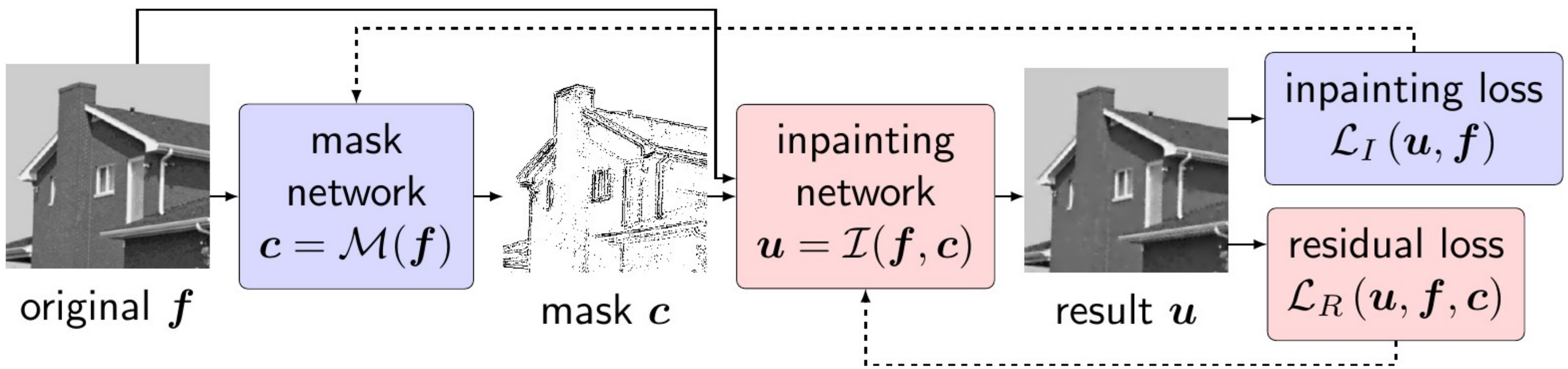


optimised 5 % mask

inpainting, MSE=17.17

Alternative Data Optimisation Approaches

- **Densification Methods**
  (Karos et al. 2018, Daropoulos et al. 2021, Chizhov/W. 2021)
- **Bilevel Optimisation Models**
  (Hoeltgen et al. 2013, Ochs et al. 2014, Bonettini et al. 2017)
- **Deep Learning Approaches**
  (Alt et al. 2022, Peter et al. 2023, Schrader et al. 2023)



Sparsification, densification, bilevel optimisation, and deep learning approaches are qualitatively similar. Deep learning, however, avoids inpainting at inference time, and is thus several order of magnitude faster. It takes 85 milliseconds on an *Intel Core i7-7700K CPU @ 4.20 GHz* for a 256×256 image.

Outline

Optimising the Inpainting Operator

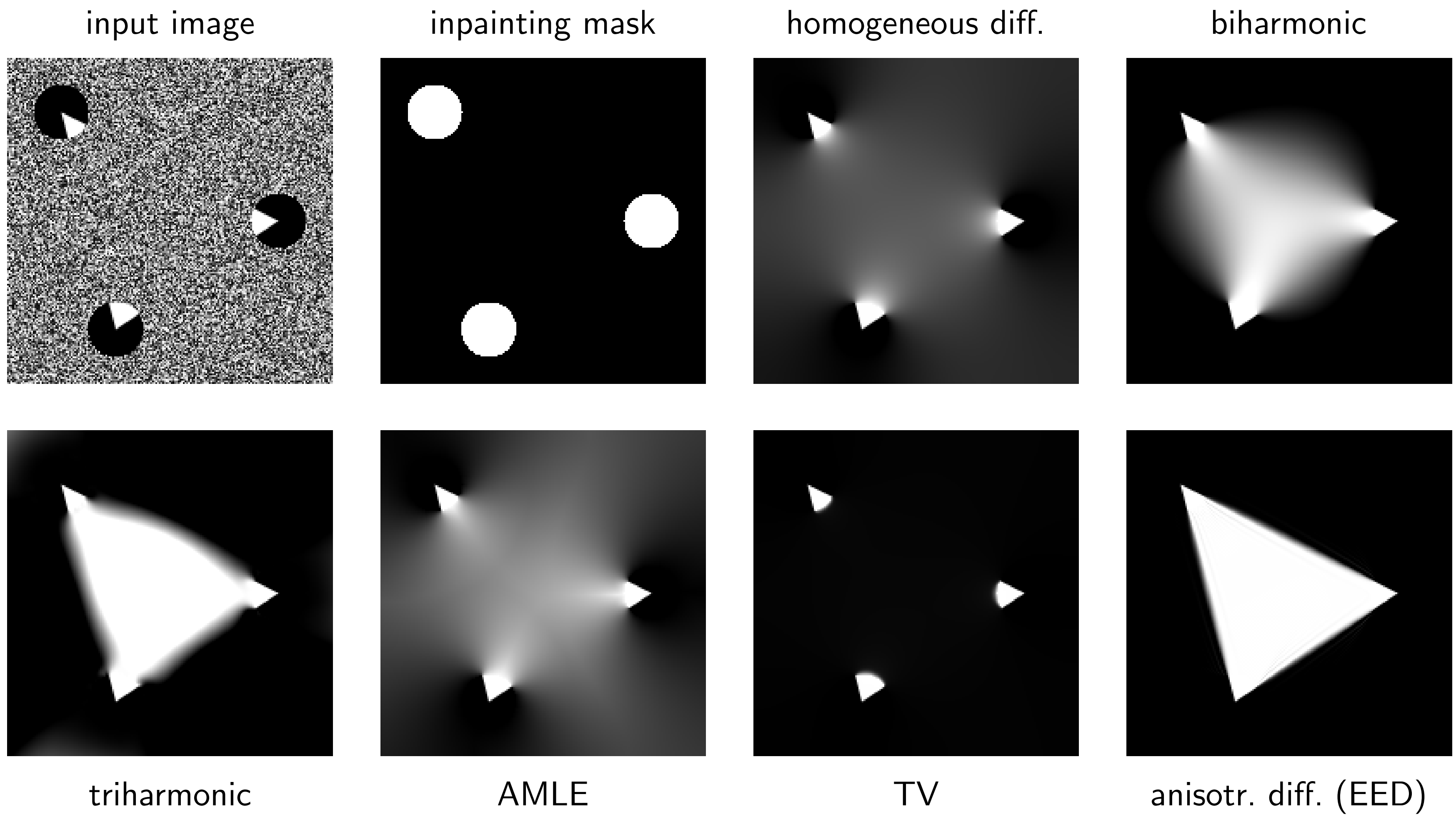
- so far: inpainting with the homogeneous diffusion operator Δu;
  requires many data to represent edges

- Galić et al. 2005, Schmaltz et al. 2009, Chen et al. 2014:
  - experimental evaluations of various differential operators
  - motivated from spline theory or inpainting literature

interpolation strategy	authors	differential oper.	max–min
homogeneous diffusion	Iijima 1959	Δu	yes
biharmonic interpol.	Duchon 1976	$-\Delta^2 u$	no
triharmonic interpol.		$\Delta^3 u$	no
AMLE	Caselles et al. 1998	$u_{\eta\eta} \quad (\eta \parallel \boldsymbol{\nabla} u)$	yes
TV inpainting	Rudin et al. 1992	$\operatorname{div}\left(\frac{\boldsymbol{\nabla} u}{\lvert\boldsymbol{\nabla} u\rvert}\right)$	yes
anisotropic diff. (EED)	W. 1996	$\operatorname{div}\left(\boldsymbol{D}(\boldsymbol{\nabla} u_\sigma)\,\boldsymbol{\nabla} u\right)$	yes

Optimising the Inpainting Operator (2)

Comparison of Different Inpainting Operators



Best performing inpainting operator in this and other experiments:

Edge-Enhancing Anisotropic Diffusion (EED)

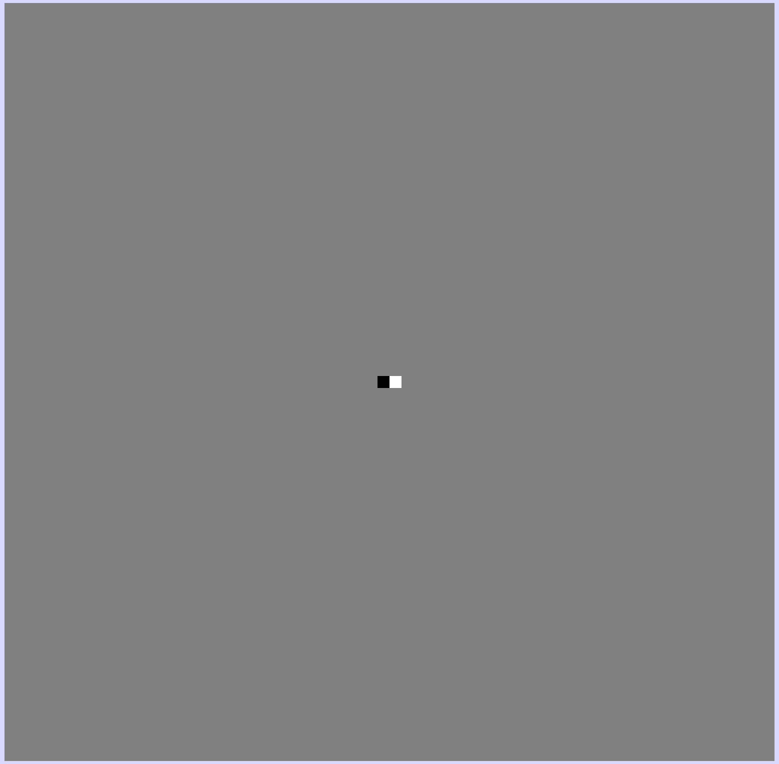
- originally for denoising (W. 1996), later for inpainting (W./Welk 2006)
- uses nonlinear inpainting operator

$$\operatorname{div}\left(\boldsymbol{D}(\boldsymbol{\nabla} u_\sigma)\, \boldsymbol{\nabla} u\right)$$

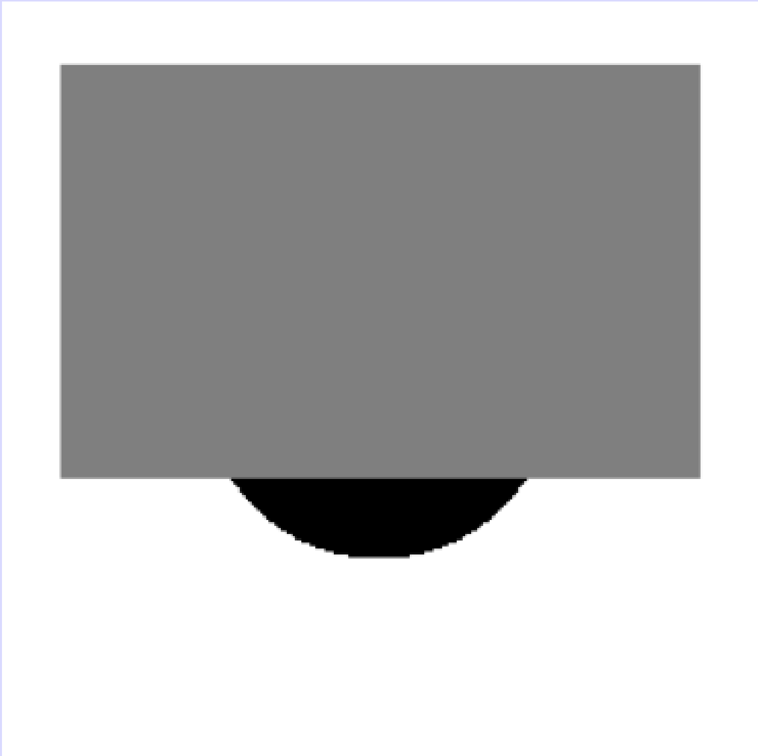
- u_σ is a Gaussian-smoothed version of u
- diffusion tensor $\boldsymbol{D}(\boldsymbol{\nabla} u_\sigma)$:
  - symmetric 2×2 matrix
  - adapts itself to (semi-) local image structure
  - prefers inpainting along edges over inpainting across them

Why Does Anisotropic Diffusion (EED) Work so Well ?

Two Simple but Instructive Experiments



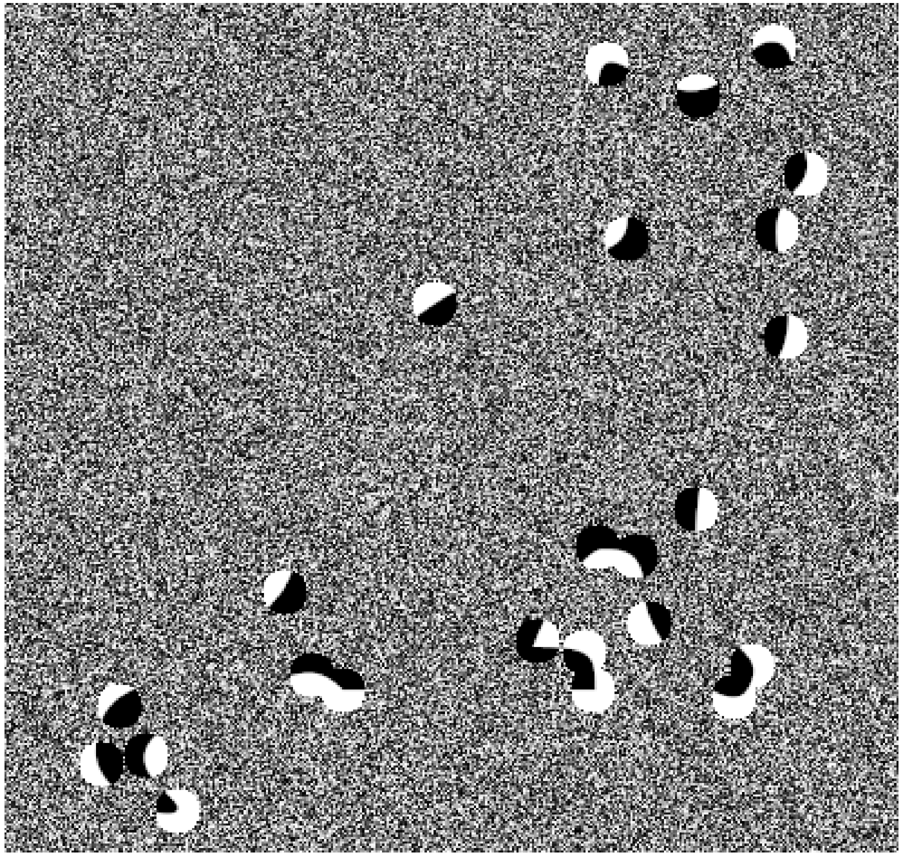
dipole



occluded disk

inpainting domain initialised in grey

EED has Interesting Shape Reconstruction Qualities



Binary image, where data are specified within 24 small disks. The unknown pixels are initialised with uniform noise. Reconstruction is performed with anisotropic diffusion (EED). Author: W. (2012).

What is so Special about EED ?

- **anisotropy due to the diffusion tensor:**
  - makes it well-suited for reconstructing edges
  - no need for high mask pixel density at edges

- **nonlocality due to Gaussian convolution:**
  - allows to propagate edge structures
  - creates a curvature-reducing effect (cf. Merriman et al. 1994)

- **very natural:**
  - EED-like operators reflect statistics of natural images (Peter et al. 2015)

Optimising the Inpainting Operator (7)

Other Classes of Inpainting Operators Researched

- **Euler's Elastica**
  almost competitive, but much slower (Schrader et al. 2022)
- **Radial Basis Functions / Pseudodifferential Inpainting**
  interesting theoretical insights (Augustin et al. 2019)
- **Smoothed Particle Hydrodynamics**
  almost competitive (Daropoulos et al. 2021)
- **Shepard Interpolation**
  simple and very fast (Peter 2019)
- **Inpainting with Deep Learning**
  almost competitive, promising (Peter 2022)
- **Exemplar-based Inpainting**
  excellent for highly textured images (Karos et al. 2018)

Outline

- Optimising the Data
- Optimising the Inpainting Operator
- Optimising the Encoding
- Theoretical Achievements
- Fast Algorithms and Real-time Aspects
- Extensions and Applications
- Summary and Outlook

Optimising the Encoding

Lessons Learnt

- Homogeneous diffusion requires careful data optimisation.
- Anisotropic diffusion (EED) can be a better alternative:
  - It needs less data to reconstruct edges (and texture).
  - Thus, the data distribution can be more evenly.
  - This means that the data location is a bit less critical.

Key Aspect Ignored so Far

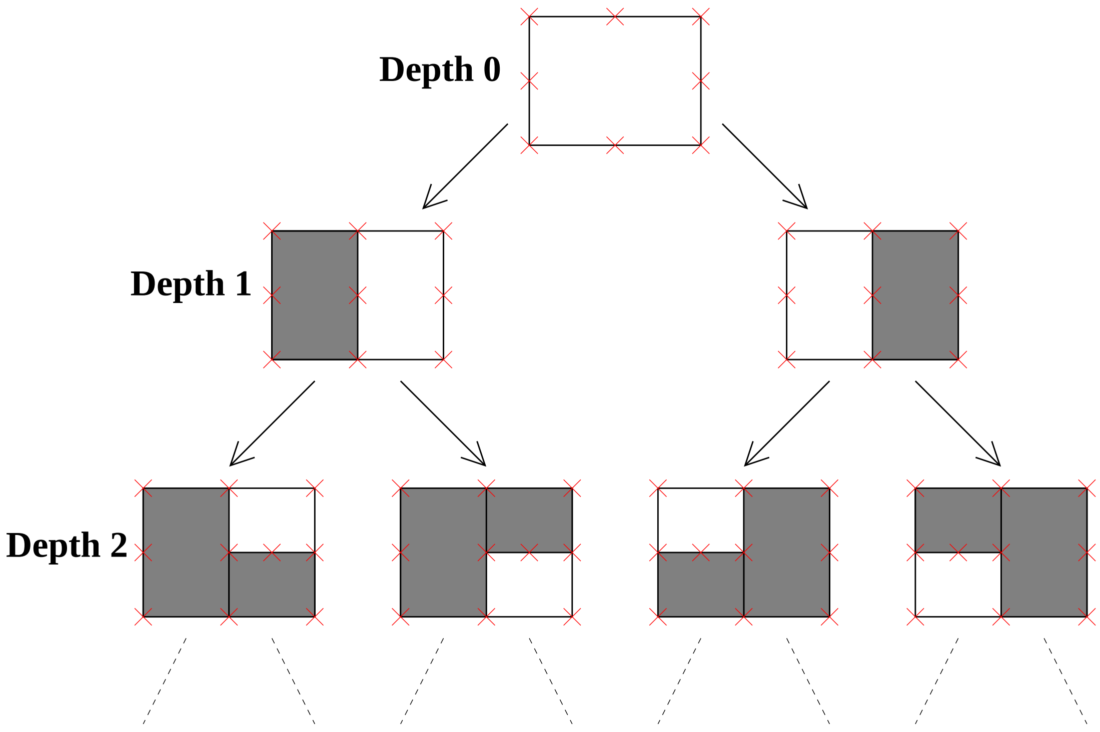
- encoding costs:
  storing a freely optimised mask is expensive (Mohideen et al. 2021)

Balancing all Requirements

- use EED and find a slightly suboptimal mask that is much cheaper to encode

Finding a Useful Mask that is Inexpensive to Encode

- rectangular subdivision where approximation quality of EED is insufficient (cf. Dyn et al. 1990, Sullivan et al. 1994, Distasi et al. 1997)
- encoding in a binary tree costs about 1 bit per pixel location



Rectangular subdivision. Each subdivision step splits the white area and inserts one point.

Optimising the Encoding (3)

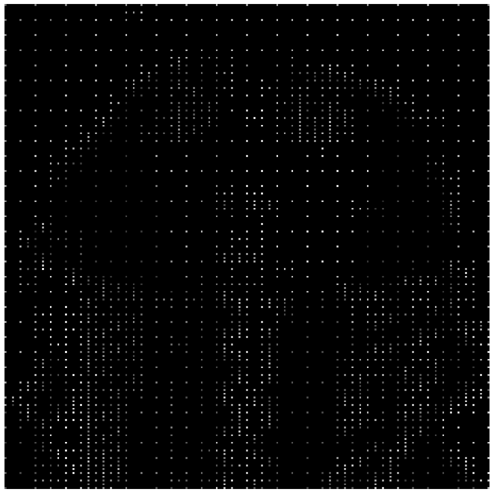
Designing a Full Codec

(Schmaltz et al. 2014)

- benefit from established concepts on the encoding side:
  - – coarser quantisation of grey values
  - – remove redundancy from entire bitstring by entropy coding (e.g. PAQ)
- decoding: inpaint with EED

Example at a Compression Rate of 57 : 1

original image



data from subdivision

EED-based decoding

Optimising the Encoding (4)

Comparison to JPEG and JPEG 2000 at 57 : 1

original

JPEG, MSE $= 111.01$

JPEG 2000, MSE $= 70.79$

EED, MSE $= 42.77$

What Happens for Extremely High Compression Rates ?

JPEG 2000, 162 : 1

JPEG 2000, 202 : 1

JPEG 2000, 261 : 1

EED, 163 : 1

EED, 211 : 1

EED, 264 : 1

Outline

- Optimising the Data
- Optimising the Inpainting Operator
- Optimising the Encoding
- Theoretical Achievements
- Fast Algorithms and Real-time Aspects
- Extensions and Applications
- Summary and Outlook

Theoretical Achievements

Continuous Theory

- rigorous theory for continuous data selection (Belhachmi et al. 2009)
- existence results for continuous EED inpainting (Bildhauer et al. 2021)

Discrete Theory

- well-posedness of the discrete inpainting problem (Mainberger et al. 2011)
- uniqueness result for tonal optimisation (Mainberger et al. 2012)

Connections and Equivalences

- sparsity interpretation of linear approaches (Hoffmann et al. 2015)
- equivalence of RBF and pseudodifferential inpainting (Augustin et al. 2019)

Scale-space Theories

- sparsification scale-spaces (Cárdenas et al. 2019)
- quantisation scale-spaces (Peter 2021)

Outline

- Optimising the Data
- Optimising the Inpainting Operator
- Optimising the Encoding
- Theoretical Achievements
- Fast Algorithms and Real-time Aspects
- Extensions and Applications
- Summary and Outlook

Fast Algorithms and Real-time Aspects

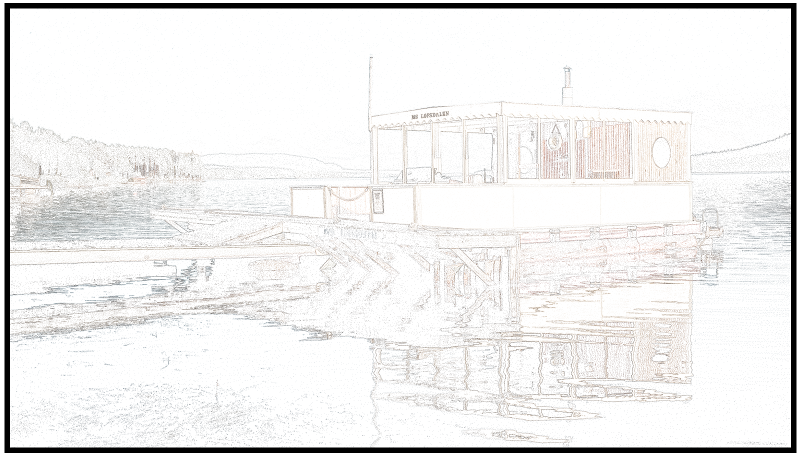
- **Encoding: Use data optimisation without inpainting.**
  - analytic approach with Floyd-Steinberg dithering (Belhachmi et al. 2009)
  - CNN-based mask optimisation (Alt et al. 2022, Schrader et al. 2023)

- **Decoding: Adapt state-of-the-art numerics for inpainting.**
  - adaptive finite elements to reduce number of unknowns (Chizhov/W. 2021)
  - CNN-based surrogate models for Euler's elastica (Schrader et al. 2022)
  - fast explicit diffusion schemes with super-time stepping (Peter et al. 2015)
  - multigrid on CPUs (Mainberger et al. 2011)
  - discrete Green's functions for very sparse data (Hoffmann et al. 2015)
  - domain decomposition on GPUs (Kämper/W. 2022)

Domain Decomposition for Real-time Inpainting in 4K (Kämper/W. 2022)

4K image, 3840×2160 pixels

optimised data, 5 % density



homogeneouos diffusion inpainting, solves 3 systems with 8 million unknowns in 25.2 ms

Outline

- Optimising the Data
- Optimising the Inpainting Operator
- Optimising the Encoding
- Theoretical Achievements
- Fast Algorithms and Real-time Aspects
- Extensions and Applications
- Summary and Outlook

Extensions and Applications

Edge-based Codecs for Piecewise Almost Constant Images

(Mainberger et al. 2011)

original

data kept

reconstruction

Left: Original image. **Middle:** Stored data with tonal optimisation. The data are subsampled along the contour, requantised and entropy coded. **Right:** Reconstruction with homogeneous diffusion in each colour channel. Compression rate: 200:1.

Comparison to JPEG and JPEG 2000 at 200 : 1

original image

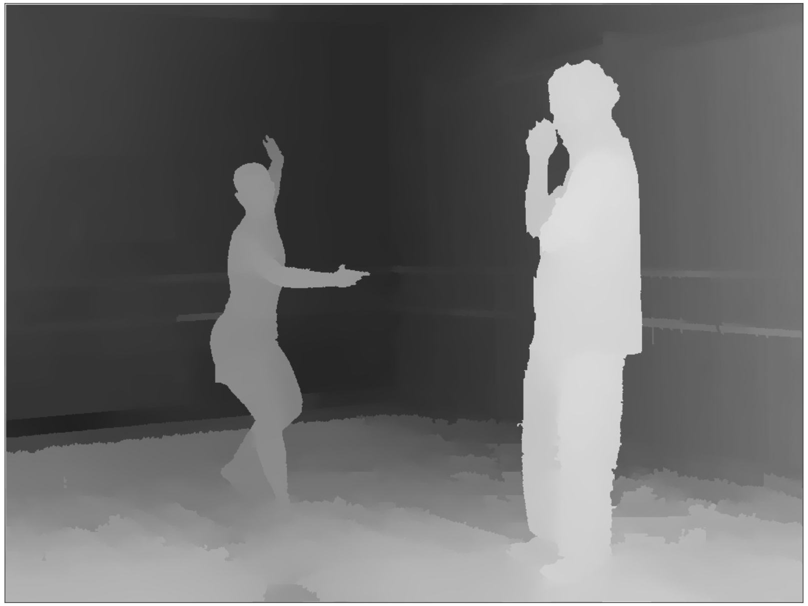
JPEG, MSE=441.20

JPEG 2000, MSE=164.53

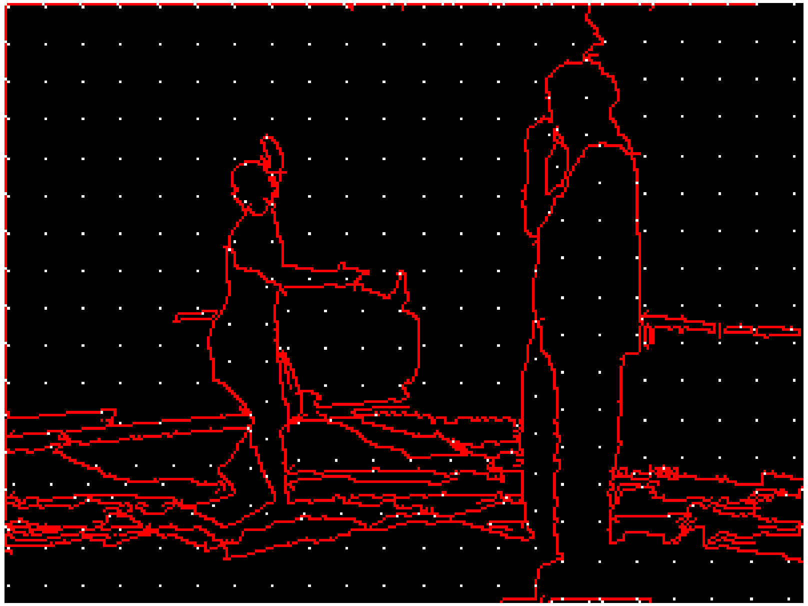
diffusion, MSE=25.97

Segmentation-based Codecs for Piecewise Smooth Images

(Jost et al. 2020)



depth map

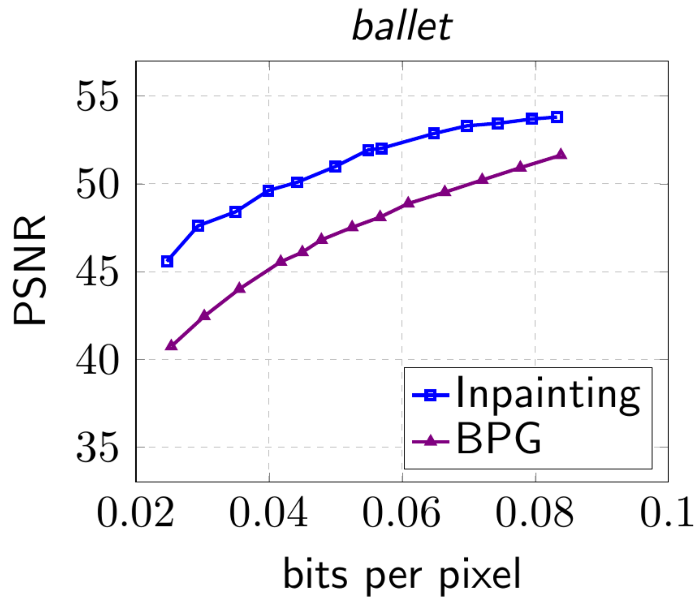


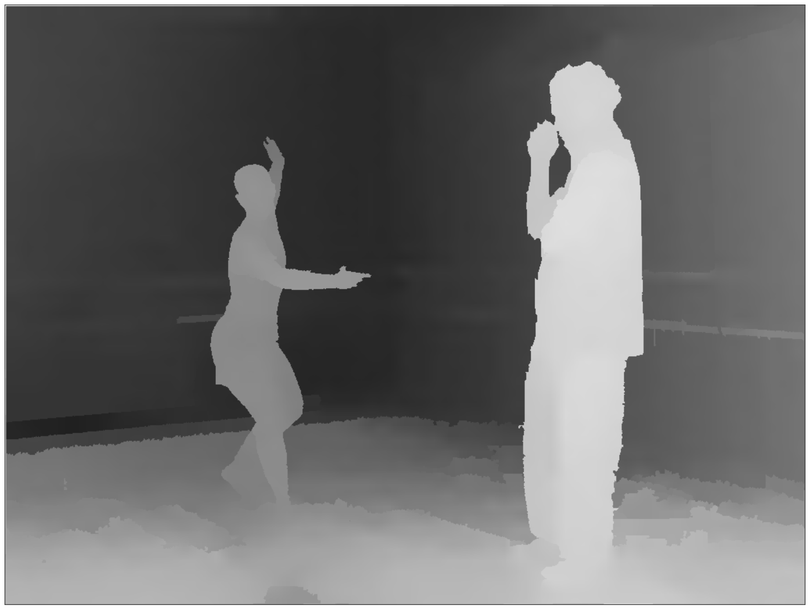
stored data

Left: A depth map as an example of a piecewise smooth image. **Right:** Data used for homogeneous diffusion inpainting. We store the segmentation boundaries and the grey values at a coarse regular grid. The segmentation uses a Mumford–Shah cartoon model with region merging.

Outperforming State-of-the-Art Transform-Based Compression

- BPG is based on HEVC and designed as successor of JPEG and JPEG 2000.
- Inpainting-based compression outperforms it by up to 5 dB in PSNR.





inpainted at compression ratio 200 : 1

M I A
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52

Exemplar-based Inpainting for Highly Textured Images

(Karos et al. 2018)

- based on sparse exemplar-based inpainting of Facciolo et al. (2009)
- data optimisation via gradual densification in 10 steps:
  – in each step, insert 10 % of desired pixels by dithering the error map
  – computationally feasible: only 10 exemplar-based inpaintings
- postprocessing with nonlocal pixel exchange

original image

optimised data, 5 %

inpainted

Extension to Colour Images

(Peter et al. 2014)

original

JPEG 2000, MSE=151.31

EED, MSE=77.53

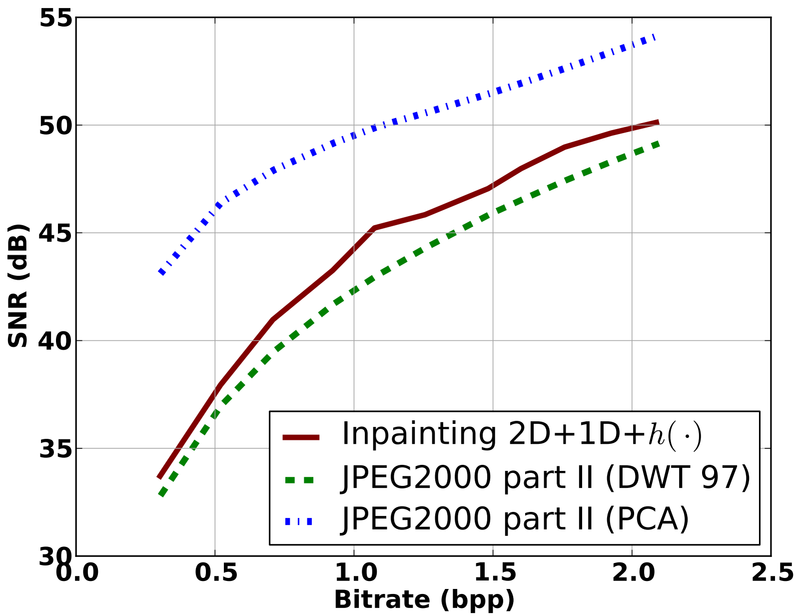
Compression of a 256×256 subimage of *peppers* with compression rate 110:1. **Left:** Original image. **Middle:** JPEG 2000. **Right:** EED in Luma Preference Mode. This mode works in the YCbCr space. It compresses the luma channel Y with higher quality. The resulting diffusion tensor steers the inpainting in the chroma channels Cb and Cr.

M I A
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52

Extension to Hyperspectral Data

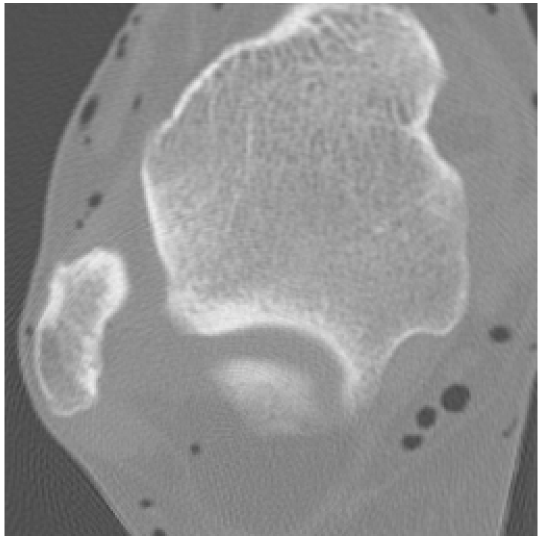
(Amrani et al. 2017)

- consist of hundreds of channels (frequency bands)
- homogeneous diffusion inpainting in space
- biharmonic inpainting in spectral dimension
- qualitatively between JPEG 2000 Part II with DWT 9/7 and JPEG 2000 with PCA

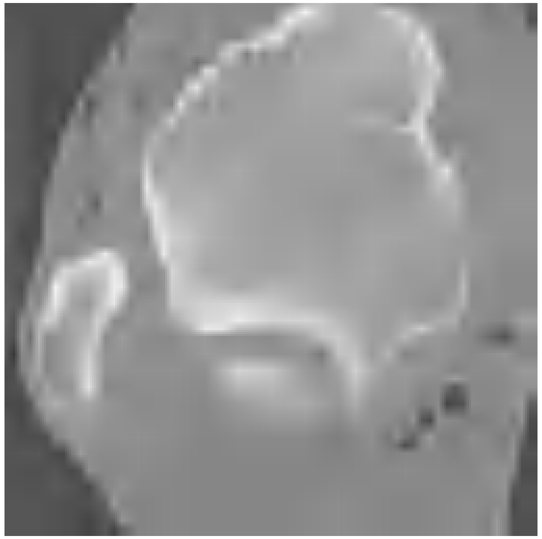


Extension to Three-Dimensional Data Sets

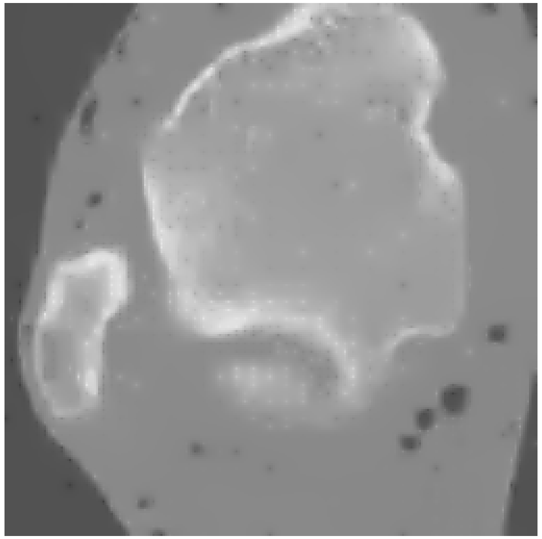
(Peter 2013, Schmaltz et al. 2014)



original



DICOM, MSE=65.02



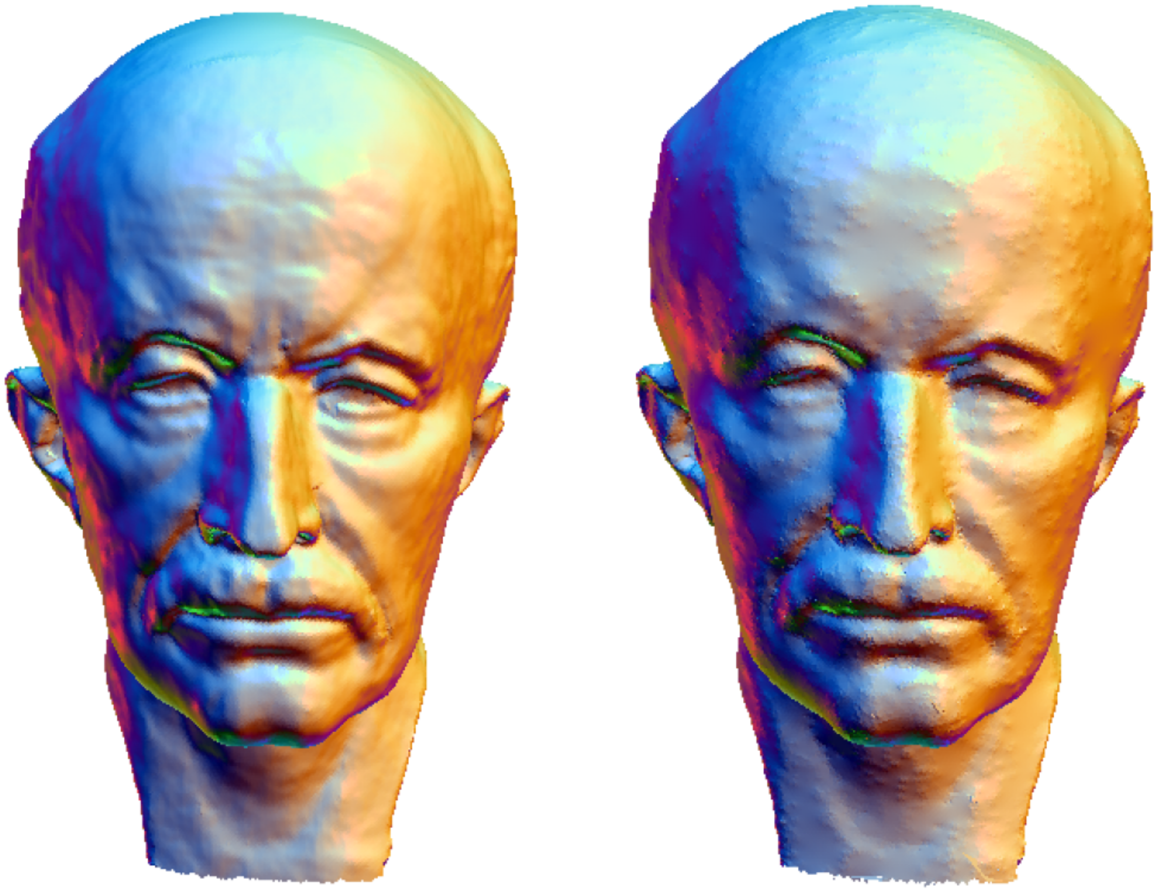
EED, MSE=38.66

Compression of a 3-D data set with a compression rate of 207:1. **Left:** Original slice from a 3-D CT data set of a trabecular bone. **Middle:** Result after compression in the DICOM standard, based on JPEG 2000. **Right:** Compression with 3-D EED with cuboidal subdivision.

M I A
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52

Extension to Surface Data

(Bae/W. 2010)



Left: Original data set. **Right:** Reconstruction with optimised 10 % of the points and geometric linear diffusion.

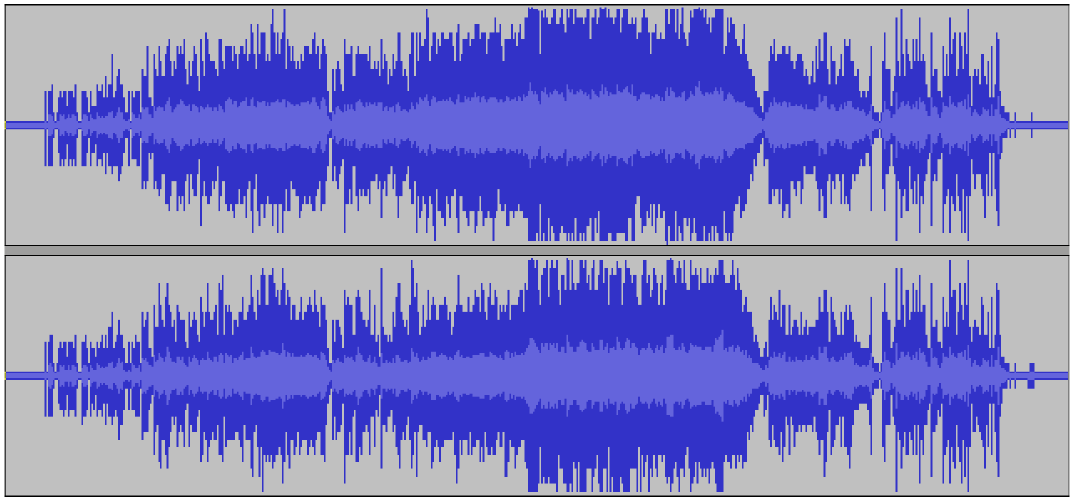
M I A
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52

Adaptation to Audio Compression

(Peter et al. 2018)

- sparse representation of audio signal in sample domain
- homogeneous diffusion inpainting becomes linear interpolation
- data sparsification approach, entropy coding with LPAQ
- competitive to mp3, AAC, and Vorbis in terms of SNR

original audio signal



inpainting-based compression

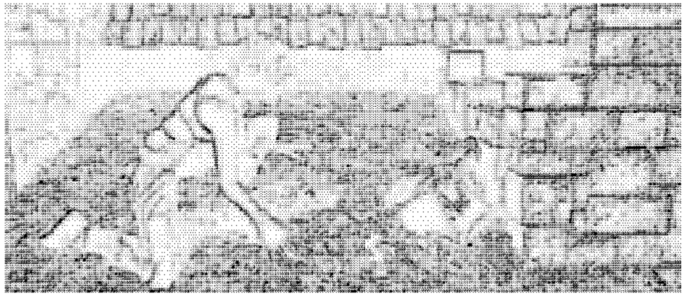
Extension to Video Compression

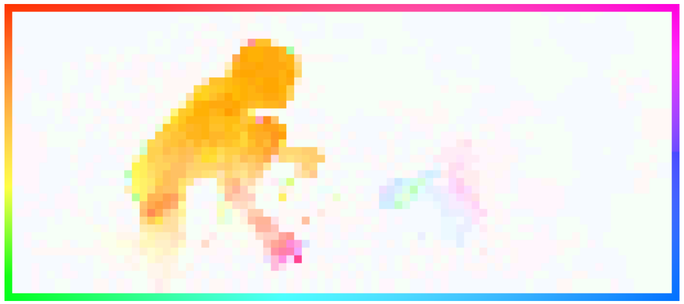
(Andris et al., PCS 2016 Best Poster Award)

- inpainting-based coding of selected images (intraframes)
- prediction of images inbetween (interframes) with variational optic flow
- additional coding of the residual errors

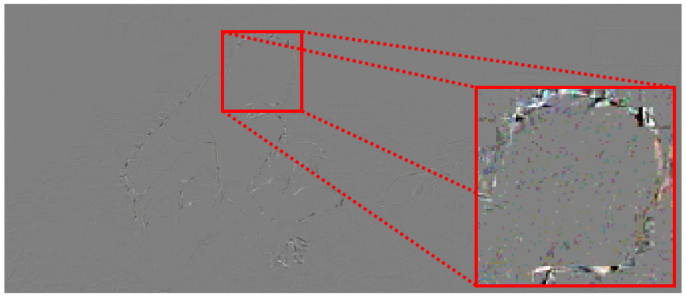
original frame

inpainting mask (intraframe)





subsampled motion field (interframe)



interframe residual

Steganographic Application: Censoring

(Mainberger et al. 2012)

original

censored

restored

Left: Original image. **Middle:** Censored version. The censored part is EED-encoded and hidden in the least significant bits of the non-censored part. **Right:** Reconstruction.

- Try it with your own image: `stego.mia.uni-saarland.de`

Outline

- Optimising the Data
- Optimising the Inpainting Operator
- Optimising the Encoding
- Theoretical Achievements
- Fast Algorithms and Real-time Aspects
- Extensions and Applications
- Summary and Outlook

Summary

- inpainting offers novel and promising ways for compressing visual data
- store subset of pixels and inpaint inbetween
- can achieve results of competitive or superior quality, if data, inpainting process, and coding are optimised
- real-time performance possible
- flexible and widely applicable

Outlook

- exploring the potential of deep learning further
- treating all optimisation problems simultaneously

Thank you!

https://www.mia.uni-saarland.de

M I A
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52