Image Analysis and Approximation via Generalized Polyharmonic Local Trigonometric Transforms

Naoki Saito with Jean-François Remy, Noel Smith, Ernest Woei, Katsu Yamatani, & Jim Zhao

saito@math.ucdavis.edu

http://www.math.ucdavis.edu/~saito/

Department of Mathematics

University of California, Davis

Partially supported by ONR N00014-00-1-0469, PECASE, and NSF DMS-0410406

Outline

- Motivations: Issues of Localized Fourier Analysis
- Polyharmonic Local Sine Transform (PHLST) for Rectangular Domains
- Extension to 3D
- Extension to General Shape Domain
- Summary and Future Plan

Motivations

- Want sparse data representation (compression) $\Longrightarrow$ Fast decay of the expansion coefficients
- Want a data representation as less statistically dependent among blocks as possible $\Longrightarrow$ No overlaps among blocks
- Want to develop a local signal analysis tool that can distinguish intrinsic singularities from the artificial discontinuities created by local windowing $\Longrightarrow$ many potential applications
- Want to efficiently represent regions of more general shapes other than rectangular blocks

Life on an Interval (or a Domain)

If data are periodic and smooth, the Fourier basis is very efficient:

- Fast decay of the Fourier coefficients
- Smoothness estimate (e.g., the Lipschitz/Hölder exponents)

However, most signals and images of interest have compact supports, and are neither periodic nor smooth. What to do?

Life on an Interval (or a Domain) . . .

- Design special wavelets (“wavelets on intervals”);
- Use “multiwavelets” of Alpert and Rokhlin (segmented orthogonal polynomials);
- Use “Prolate Spheroidal Wave Functions”;
- Use the “Continuous Boundary Local Fourier Transform” (CBLFT) that periodizes the signal nicely and expand it into a periodic basis; or
- Use the “Polyharmonic Local Trigonometric Transform” (PHLTT) and its generalizations

Basic Ideas of PHLTT

- Split the domain Ω into a set of subregions (often rectangular blocks) $\{\Omega_j\}_j$ and cut the data/function f into pieces using the characteristic functions, $f_j = \chi_{\Omega_j} f$

Basic Ideas of PHLTT

- Split the domain Ω into a set of subregions (often rectangular blocks) $\{\Omega_j\}_j$ and cut the data/function f into pieces using the characteristic functions, $f_j = \chi_{\Omega_j} f$
- f_j is decomposed into two parts $u_j + v_j$

Basic Ideas of PHLTT

- Split the domain Ω into a set of subregions (often rectangular blocks) $\{\Omega_j\}_j$ and cut the data/function f into pieces using the characteristic functions, $f_j = \chi_{\Omega_j} f$
- f_j is decomposed into two parts $u_j + v_j$
- u_j is a solution of the polyharmonic equations $\Delta^m u = 0$ with Dirichlet/Neumann boundary conditions

Basic Ideas of PHLTT

- Split the domain Ω into a set of subregions (often rectangular blocks) $\{\Omega_j\}_j$ and cut the data/function f into pieces using the characteristic functions, $f_j = \chi_{\Omega_j} f$
- f_j is decomposed into two parts $u_j + v_j$
- u_j is a solution of the polyharmonic equations $\Delta^m u = 0$ with Dirichlet/Neumann boundary conditions
- v_j is a residual $f_j - u_j$, which are expanded into the multiple Fourier series (either complex exponentials, sines, or cosines)

Basic Ideas of PHLTT

- Split the domain Ω into a set of subregions (often rectangular blocks) $\{\Omega_j\}_j$ and cut the data/function f into pieces using the characteristic functions, $f_j = \chi_{\Omega_j} f$
- f_j is decomposed into two parts $u_j + v_j$
- u_j is a solution of the polyharmonic equations $\Delta^m u = 0$ with Dirichlet/Neumann boundary conditions
- v_j is a residual $f_j - u_j$, which are expanded into the multiple Fourier series (either complex exponentials, sines, or cosines)
- Sines and cosines are eigenfunctions of the Laplacian on a rectangular box

1D Case

- In 1D, each u_j is a low order algebraic polynomial (e.g., line, cubic poly.)

1D Case

- In 1D, each u_j is a low order algebraic polynomial (e.g., line, cubic poly.)
- v_j is a trigonometric polynomial

1D Case

- In 1D, each u_j is a low order algebraic polynomial (e.g., line, cubic poly.)
- v_j is a trigonometric polynomial
- These two compensate the shortcomings of each other:

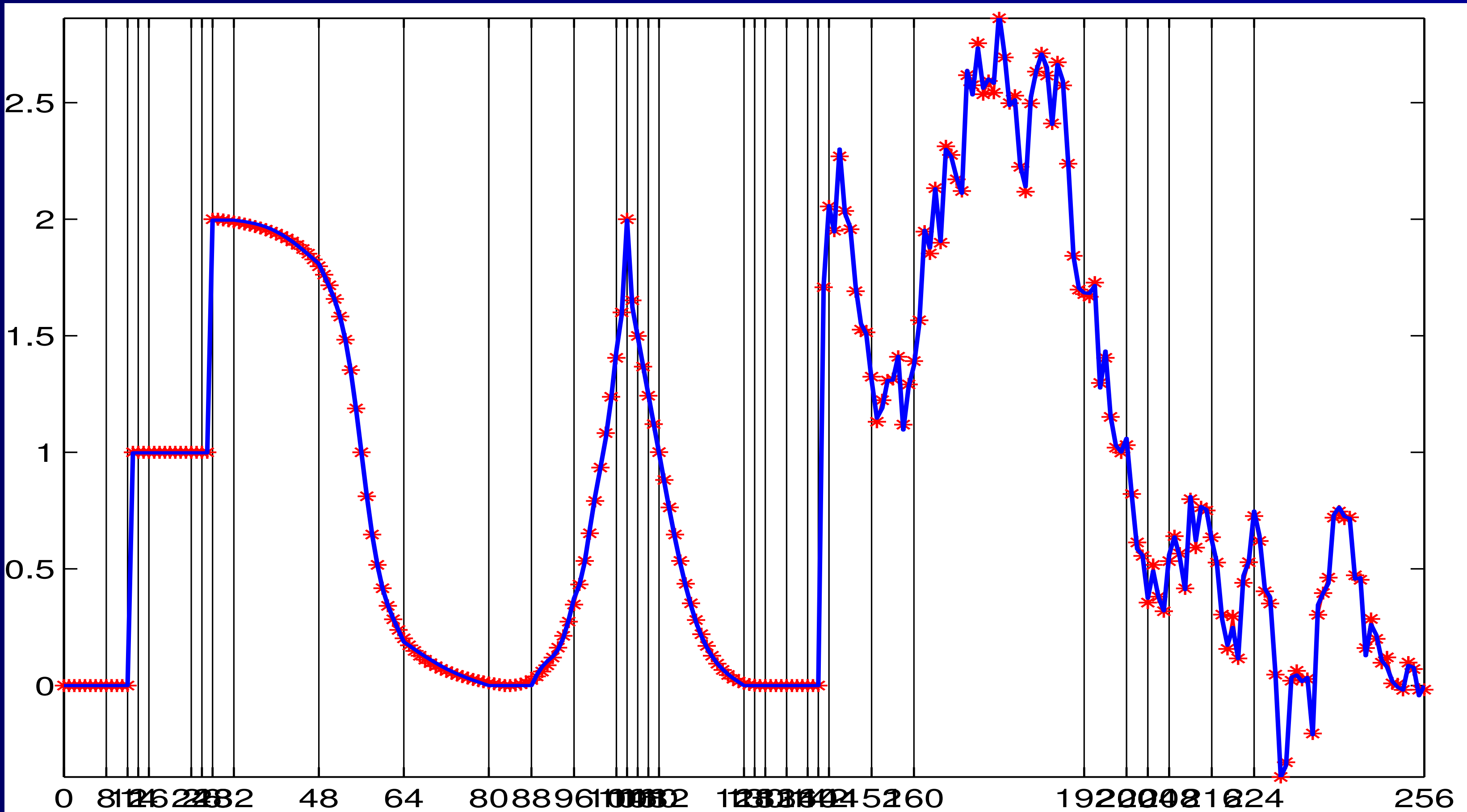
1D Case

- In 1D, each u_j is a low order algebraic polynomial (e.g., line, cubic poly.)
- v_j is a trigonometric polynomial
- These two compensate the shortcomings of each other:
- High order algebraic polynomial $\Longrightarrow$ Runge phenomenon

1D Case

- In 1D, each u_j is a low order algebraic polynomial (e.g., line, cubic poly.)
- v_j is a trigonometric polynomial
- These two compensate the shortcomings of each other:
- High order algebraic polynomial $\Longrightarrow$ Runge phenomenon
- Trigonometric polynomial on an interval $\Longrightarrow$ Gibbs phenomenon

1D Example: Compression Ratio ≈ 6



Polyharmonic Global Sine Transform on a Rectangle

- Consider a function $f \in C^{2m}(\overline{\Omega})$, where $m = 1, 2, \ldots$, and $\Omega \subset \mathbb{R}^n$ (e.g., $\Omega = [0, 1]^n$), but not periodic.

Polyharmonic Global Sine Transform on a Rectangle

- Consider a function $f \in C^{2m}(\overline{\Omega})$, where $m = 1, 2, \ldots$, and $\Omega \subset \mathbb{R}^n$ (e.g., $\Omega = [0,1]^n$), but not periodic.
- Decompose this function into the following two components:

$$f(\boldsymbol{x}) = u(\boldsymbol{x}) + v(\boldsymbol{x}),$$

Polyharmonic Global Sine Transform on a Rectangle

- Consider a function $f \in C^{2m}(\overline{\Omega})$, where $m = 1, 2, \ldots$, and $\Omega \subset \mathbb{R}^n$ (e.g., $\Omega = [0, 1]^n$), but not periodic.
- Decompose this function into the following two components:

$$f(\boldsymbol{x}) = u(\boldsymbol{x}) + v(\boldsymbol{x}),$$

- $u(\boldsymbol{x})$ satisfies the following polyharmonic equation:

$$\Delta^m u = 0 \quad \text{in } \Omega.$$

PHGST on a Rectangle . . .

- The boundary condition for u is:

$$\frac{\partial^{p_\ell} u}{\partial\, \nu^{p_\ell}} = \frac{\partial^{p_\ell} f}{\partial\, \nu^{p_\ell}} \quad \text{on } \Gamma = \partial\Omega, \qquad \ell = 0, \ldots, m-1,$$

where p_ℓ is the order of the normal derivatives to be specified ($p_0 \equiv 0 \Leftrightarrow u = f$ on Γ).

PHGST on a Rectangle . . .

- The boundary condition for u is:

$$\frac{\partial^{p_\ell} u}{\partial\, \nu^{p_\ell}} = \frac{\partial^{p_\ell} f}{\partial\, \nu^{p_\ell}} \quad \text{on } \Gamma = \partial\Omega, \qquad \ell = 0, \ldots, m-1,$$

  where p_ℓ is the order of the normal derivatives to be specified ($p_0 \equiv 0 \Leftrightarrow u = f$ on Γ).

- Now set $v(\boldsymbol{x}) = f(\boldsymbol{x}) - u(\boldsymbol{x})$, which we will call the residual component with

$$\frac{\partial^{p_\ell} v}{\partial\, \nu^{p_\ell}} = 0 \quad \text{on } \Gamma, \qquad \ell = 0, \ldots, m-1.$$

A Specific Example: $m = 1$ (Laplace) Case

$$\begin{cases} \Delta u = 0 & \text{in } \Omega, \\ u = f & \text{on } \Gamma. \end{cases}$$

Variational formulation $\Longrightarrow$ minimum gradient interpolation:

$$\min_{u \in H^1(\Omega)} \int_\Omega |\nabla u|^2 \, \mathrm{d}\boldsymbol{x} \quad \text{subject to the above boundary condition.}$$

Note that in 1D, this is simply a line.

A Specific Example: $m = 2$ (Biharmonic) Case

$$\begin{cases} \Delta^2 u = 0 & \text{in } \Omega, \\ u = f,\ \dfrac{\partial^2 u}{\partial\, \nu^2} = \dfrac{\partial^2 f}{\partial\, \nu^2} & \text{on } \Gamma. \end{cases}$$

Variational formulation $\Longrightarrow$ minimum curvature interpolation:

$$\min_{u \in H^2(\Omega)} \int_\Omega \left(\Delta u + 2 \sum_{j \neq k} \partial_j \partial_k u \right)^2 \mathrm{d}\boldsymbol{x},$$

subject to the above boundary condition.

Note that in 1D, this is simply a cubic polynomial.

PHGST on a Rectangle . . .

- The polyharmonic component u is smooth inside the domain Ω, and its values (and possibly its normal derivatives) matches those of data.

PHGST on a Rectangle . . .

- The polyharmonic component u is smooth inside the domain Ω, and its values (and possibly its normal derivatives) matches those of data.
- The u component can be represented only by the boundary values $f|_\Gamma$. $\Longrightarrow$ No need to store the whole u.

PHGST on a Rectangle . . .

- The polyharmonic component u is smooth inside the domain Ω, and its values (and possibly its normal derivatives) matches those of data.
- The u component can be represented only by the boundary values $f|_\Gamma$. $\Longrightarrow$ No need to store the whole u.
- The residual component v becomes 0 at the boundary Γ.

PHGST on a Rectangle . . .

- Therefore, the v component is suitable for Fourier analysis. In fact, if $\Omega = [0,1]^n$ and $p_\ell = 2\ell$, $\ell = 0, \ldots, m-1$, then the Fourier sine analysis should be used to get the matching normal derivatives up to order $2m-1$ by odd reflection at the boundaries.

PHGST on a Rectangle . . .

- Therefore, the v component is suitable for Fourier analysis. In fact, if $\Omega = [0,1]^n$ and $p_\ell = 2\ell$, $\ell = 0, \ldots, m-1$, then the Fourier sine analysis should be used to get the matching normal derivatives up to order $2m-1$ by odd reflection at the boundaries.
- The frequency content (in particular, mid to high frequency range) of the original is retained in the residual $\Longrightarrow$ textures remain in v; shading is captured by u.

PHGST on a Rectangle . . .

- Therefore, the v component is suitable for Fourier analysis. In fact, if $\Omega = [0,1]^n$ and $p_\ell = 2\ell$, $\ell = 0, \ldots, m-1$, then the Fourier sine analysis should be used to get the matching normal derivatives up to order $2m-1$ by odd reflection at the boundaries.
- The frequency content (in particular, mid to high frequency range) of the original is retained in the residual $\Longrightarrow$ textures remain in v; shading is captured by u.
- We can get the decay rate as follows:

Theorem

Let $\Omega = [0,1]^n$, and $f \in C^{2m}(\overline{\Omega})$, but non-periodic. Assume further that $\partial_i^{2m+1} f$, $i = 1, \ldots, n$, exist and are of bounded variation. Furthermore, let $f = u + v$ be the PHLST representation where the polyharmonic component u is the solution of the polyharmonic equation of order m with the boundary condition

$$\frac{\partial^{2\ell} u}{\partial\, \nu^{2\ell}} = \frac{\partial^{2\ell} f}{\partial\, \nu^{2\ell}} \quad \text{on } \Gamma, \quad \ell = 0, \ldots, m-1.$$

Then, the Fourier sine coefficient $b_{\boldsymbol{k}}$ of the residual v is of $O\left(\|\boldsymbol{k}\|^{-2m-1}\right)$ for all $\boldsymbol{k} \neq \boldsymbol{0}$, where $\boldsymbol{k} = (k_1, \ldots, k_n)$, and $\|\boldsymbol{k}\|$ is the usual Euclidean (i.e., ℓ^2) norm of $\boldsymbol{k}$.

Polyharmonic Local Sine Transform

Now, consider a decomposition of Ω into a disjoint set of subdomains $\{\Omega_j\}$, i.e., $\overline{\Omega} = \cup_{j=1}^{J} \overline{\Omega}_j$. A typical example is $\Omega = (0,1)^n$, and Ω_j is a dyadic subcube. Then, restrict f on Ω_j, i.e., for each j, we decompose f locally as follows:

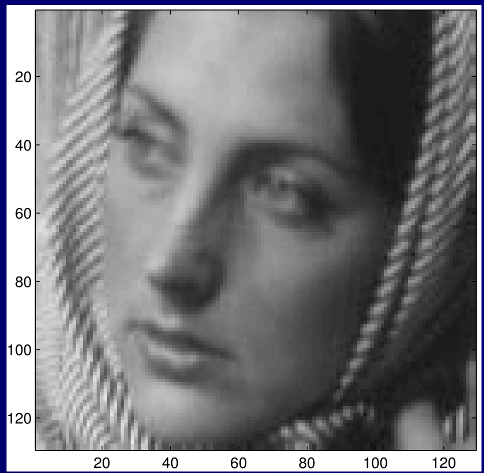
$$f\chi_{\Omega_j} = f_j = u_j + v_j,$$

where we follow the same recipe locally as in the global case. We call this decomposition of f into $\{u_j, v_j\}$ Polyharmonic Local Sine Transform (PHLST).

For $m = 1$: Laplace Local Sine Transform (LLST);

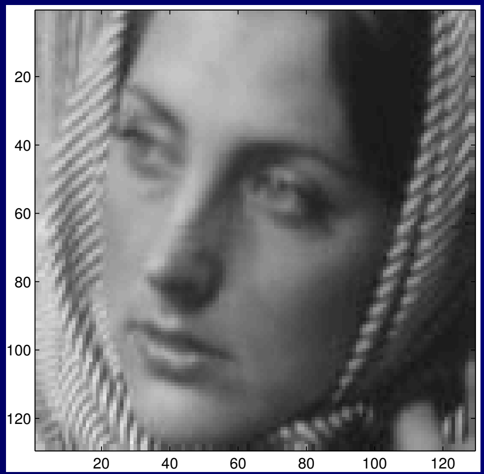
For $m = 2$: Biharmonic Local Sine Transform (BLST).

Polyharmonic Local Sine Transform ...

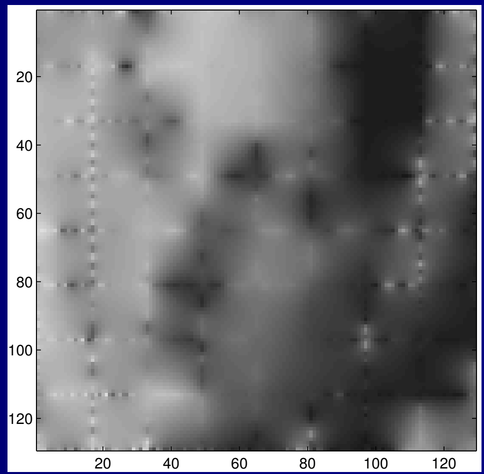


(a) Original

Polyharmonic Local Sine Transform ...

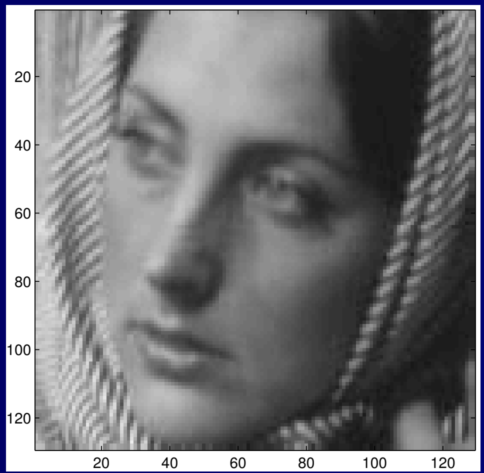


(a) Original

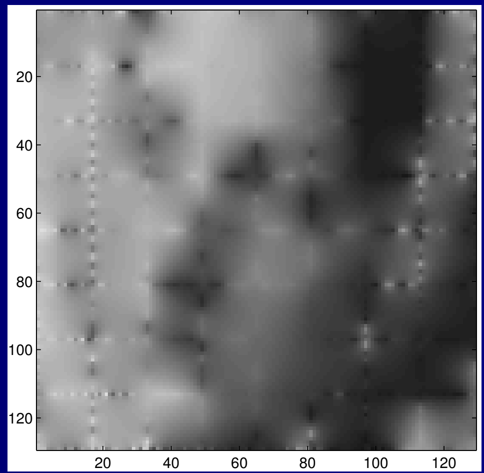


(b) $\cup_j u_j$

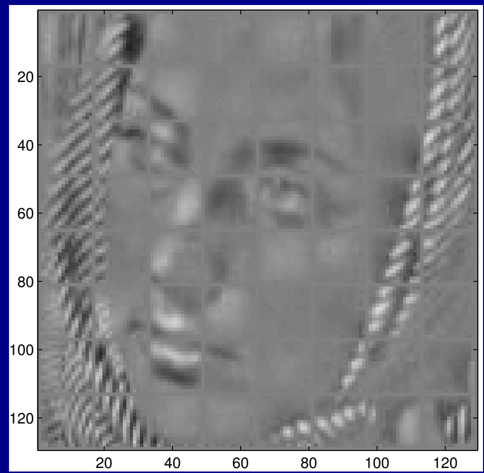
Polyharmonic Local Sine Transform . . .



(a) Original



(b) $\cup_j u_j$



(c) $\cup_j v_j$

Polyharmonic Local Sine Transform . . .

- No spatial overlaps

Polyharmonic Local Sine Transform . . .

- No spatial overlaps
- Decay of the Fourier sine coefficients are fast if Ω_j does not contain any singularity

Polyharmonic Local Sine Transform . . .

- No spatial overlaps
- Decay of the Fourier sine coefficients are fast if Ω_j does not contain any singularity
- Can distinguish intrinsic singularities from the artificial discontinuities at Γ_j imposed by local windowing

Remarks on the Laplace Solver

- A fast and accurate (no finite difference approximation of the Laplace operator) algorithm (for both 2D and 3D) exists via FFT if Ω and Ω_j are dyadic cubes (Averbuch, Braverman, Israeli, and Vozovoi, 1998)

Remarks on the Laplace Solver

- A fast and accurate (no finite difference approximation of the Laplace operator) algorithm (for both 2D and 3D) exists via FFT if Ω and Ω_j are dyadic cubes (Averbuch, Braverman, Israeli, and Vozovoi, 1998)
- The u component can be computed as:

$$u(x,y) = p(x,y) + \sum_{k \geq 1} \{ b_k^{(1)} h_k(x, 1-y) + b_k^{(2)} h_k(y, 1-x) + b_k^{(3)} h_k(x,y) + b_k^{(4)} h_k(y,x) \},$$

where . . .

Remarks on the Laplace Solver

- $p(x, y)$ is a harmonic polynomial that agrees with $f(x, y)$ at the four corner points of the domain, e.g.,

$$p(x, y) = a_3xy + a_2x + a_1y + a_0,$$

Remarks on the Laplace Solver

- $p(x,y)$ is a harmonic polynomial that agrees with $f(x,y)$ at the four corner points of the domain, e.g.,

$$p(x,y) = a_3xy + a_2x + a_1y + a_0,$$

- $h_k(x,y)$ is defined as:

$$h_k(x,y) \triangleq \sin(\pi kx)\frac{\sinh(\pi ky)}{\sinh(\pi k)},$$

where $b_k^{(j)}$ are the kth 1D Fourier sine coefficients of the boundary data of $f-p$.

Remarks on the Laplace Solver

- $p(x, y)$ is a harmonic polynomial that agrees with $f(x, y)$ at the four corner points of the domain, e.g.,

$$p(x, y) = a_3 xy + a_2 x + a_1 y + a_0,$$

- $h_k(x, y)$ is defined as:

$$h_k(x, y) \triangleq \sin(\pi k x) \frac{\sinh(\pi k y)}{\sinh(\pi k)},$$

  where $b_k^{(j)}$ are the kth 1D Fourier sine coefficients of the boundary data of $f - p$.

- The computational cost of u is $O(4N^2 \log N)$.

Remarks on PHLST

- Need to store the boundary values $\Longrightarrow$ can compress them using the lower dimensional version of PHLST

Remarks on PHLST

- Need to store the boundary values $\Longrightarrow$ can compress them using the lower dimensional version of PHLST
- Can use complex exponentials, wavelets, etc., instead of sines with potentially slower decay

Remarks on PHLST

- Need to store the boundary values $\Longrightarrow$ can compress them using the lower dimensional version of PHLST
- Can use complex exponentials, wavelets, etc., instead of sines with potentially slower decay
- Can do in the frequency domain $\Longrightarrow$ better wavelet packets

Remarks on PHLST

- Need to store the boundary values $\Longrightarrow$ can compress them using the lower dimensional version of PHLST
- Can use complex exponentials, wavelets, etc., instead of sines with potentially slower decay
- Can do in the frequency domain $\Longrightarrow$ better wavelet packets
- Can be generalized to other geometries (e.g., discs, spheres, star shapes, general smooth boundaries)

Remarks on PHLST

- Need to store the boundary values $\Longrightarrow$ can compress them using the lower dimensional version of PHLST
- Can use complex exponentials, wavelets, etc., instead of sines with potentially slower decay
- Can do in the frequency domain $\Longrightarrow$ better wavelet packets
- Can be generalized to other geometries (e.g., discs, spheres, star shapes, general smooth boundaries)
- Useful for interpolation and local feature computation (e.g., gradients, directional derivatives, etc.)

$f = u + v$ Models: Interpretation of PHLTT

- $f = u + v$ models of Yves Meyer (2001):

$f = u + v$ Models: Interpretation of PHLTT

- $f = u + v$ models of Yves Meyer (2001):
  - $u \equiv$ objects (e.g., $\in BV(\Omega)$)

$f = u + v$ Models: Interpretation of PHLTT

- $f = u + v$ models of Yves Meyer (2001):
  - $u \equiv$ objects (e.g., $\in BV(\Omega)$)
  - $v \equiv$ textures and noise

$f = u + v$ Models: Interpretation of PHLTT

- $f = u + v$ models of Yves Meyer (2001):
  - $u \equiv$ objects (e.g., $\in BV(\Omega)$)
  - $v \equiv$ textures and noise
- In PHLTT, the polyharmonic component u is what is predictable from only using the boundary behavior of a function.

$f = u + v$ Models: Interpretation of PHLTT

- $f = u + v$ models of Yves Meyer (2001):
  - $u \equiv$ objects (e.g., $\in BV(\Omega)$)
  - $v \equiv$ textures and noise
- In PHLTT, the polyharmonic component u is what is predictable from only using the boundary behavior of a function.
- The residual component v is what is unpredictable using the boundary information.

$f = u + v$ Models: Interpretation of PHLTT

- $f = u + v$ models of Yves Meyer (2001):
  - $u \equiv$ objects (e.g., $\in BV(\Omega)$)
  - $v \equiv$ textures and noise
- In PHLTT, the polyharmonic component u is what is predictable from only using the boundary behavior of a function.
- The residual component v is what is unpredictable using the boundary information.
- Yet another interpretation is: u: trend, v: fluctuation.

$f = u + v$ Models: Interpretation of PHLTT . . .

- Many possible ideas here once we view this as predicition from the available data:

$f = u + v$ **Models: Interpretation of PHLTT . . .**

- Many possible ideas here once we view this as predicition from the available data:
- A. Cohen et al. uses inpainting algorithm for predicting u components

$f = u + v$ Models: Interpretation of PHLTT ...

- Many possible ideas here once we view this as predicition from the available data:
- A. Cohen et al. uses inpainting algorithm for predicting u components
- Relationship with geometric harmonic analysis of Coifman & Lafon

$f = u + v$ Models: Interpretation of PHLTT . . .

- Many possible ideas here once we view this as predicition from the available data:
- A. Cohen et al. uses inpainting algorithm for predicting u components
- Relationship with geometric harmonic analysis of Coifman & Lafon
- Other inverse boundary problems . . .

LLST Approximation Experiment

- Comparison with DCT, the fixed folding local cosine (LCT), and Wavelet (Coiflet 12).

LLST Approximation Experiment

- Comparison with DCT, the fixed folding local cosine (LCT), and Wavelet (Coiflet 12).
- Split into a set of squares of 32×32 pixels or 16×16 pixels

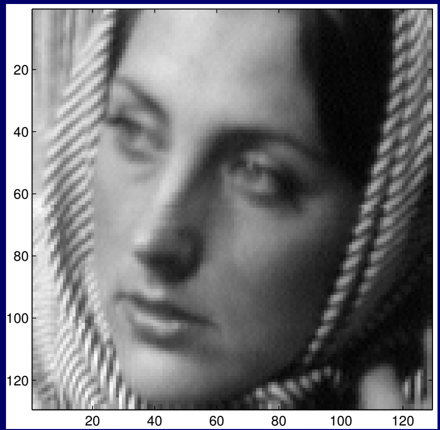
LLST Approximation Experiment

- Comparison with DCT, the fixed folding local cosine (LCT), and Wavelet (Coiflet 12).
- Split into a set of squares of 32×32 pixels or 16×16 pixels
- Select the largest 1% and 5% of coefficients.

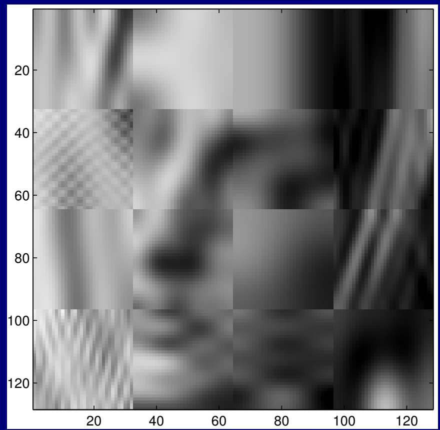
LLST Approximation Experiment

- Comparison with DCT, the fixed folding local cosine (LCT), and Wavelet (Coiflet 12).
- Split into a set of squares of 32×32 pixels or 16×16 pixels
- Select the largest 1% and 5% of coefficients.
- Compression using quantization is in progress.

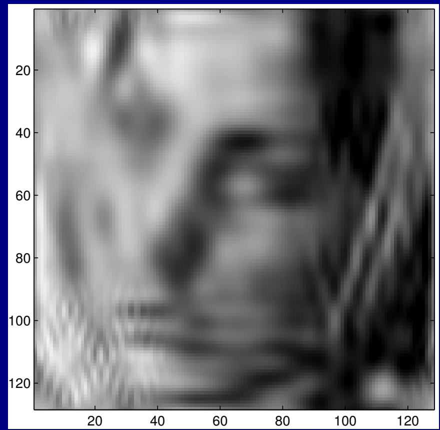
LLST Approximation Experiment: Top 1%



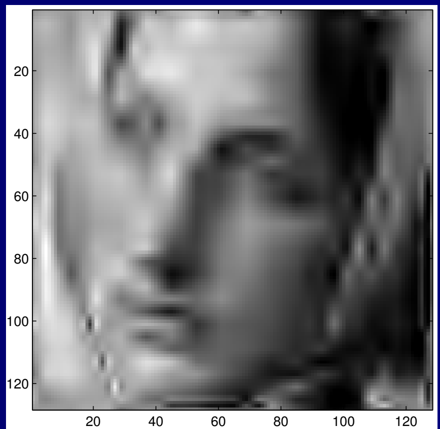
(a) Original



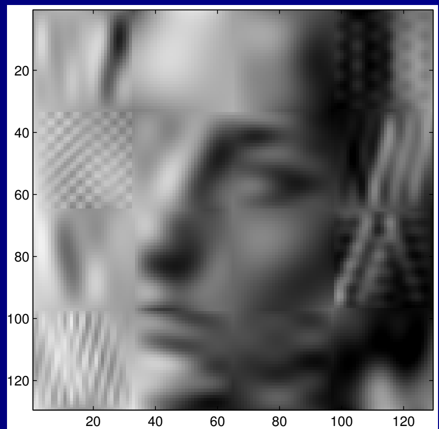
(b) DCT



(c) LCT

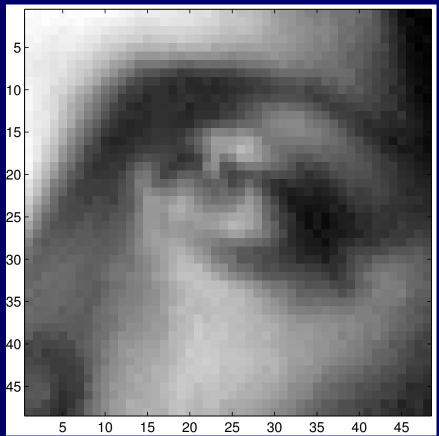


(d) C12

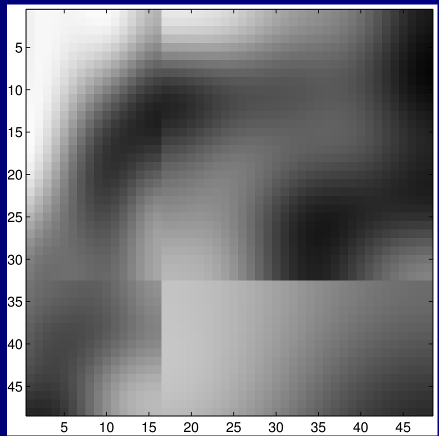


(e) LLST

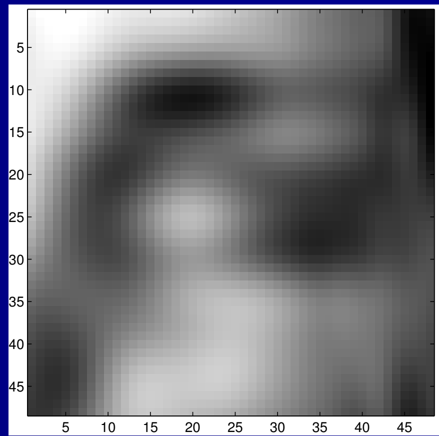
Zoomed up images: Top 1%



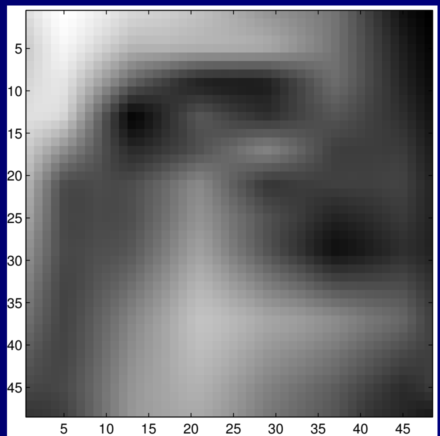
(a) Original



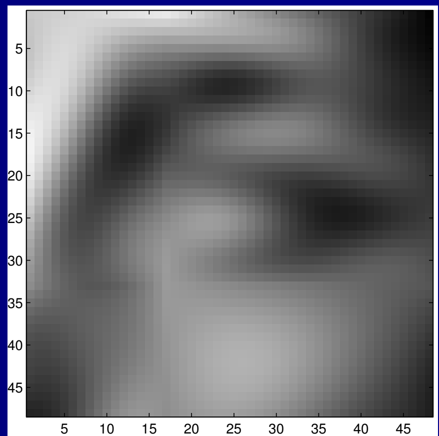
(b) DCT



(c) LCT

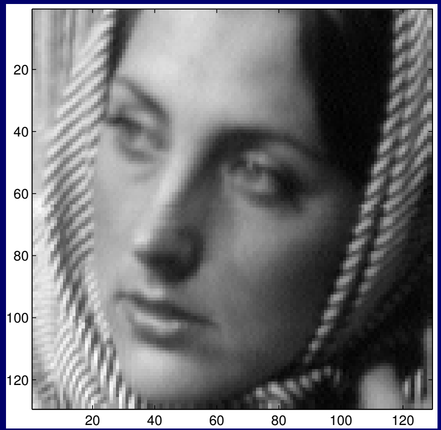


(d) C12

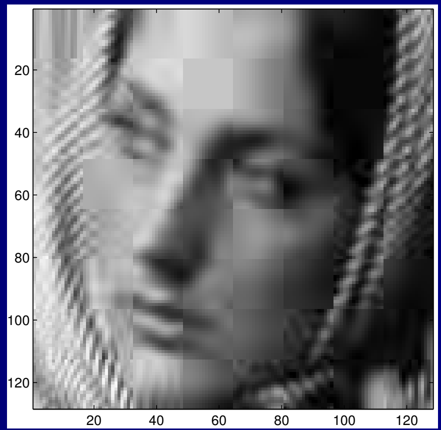


(e) LLST

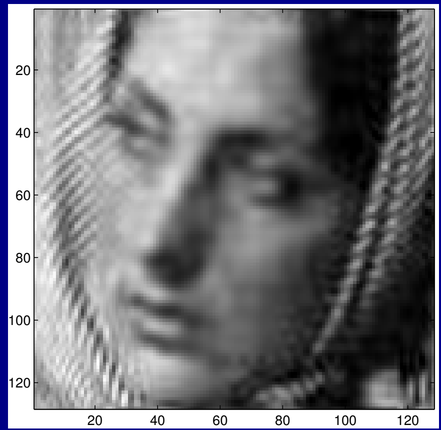
LLST Approximation Experiment: Top 5%



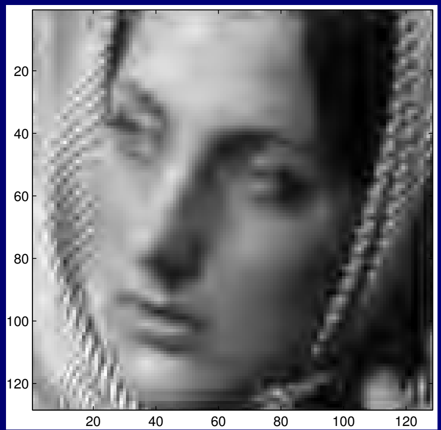
(a) Original



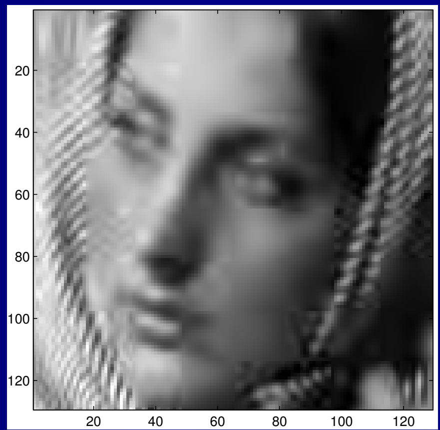
(b) DCT



(c) LCT

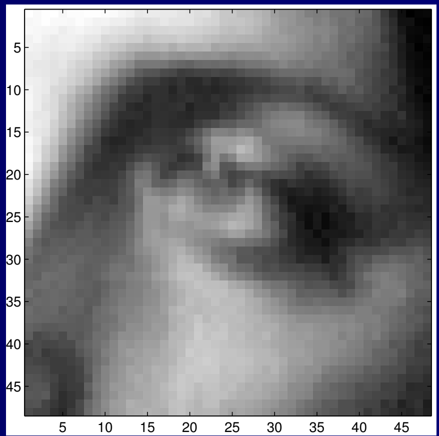


(d) C12

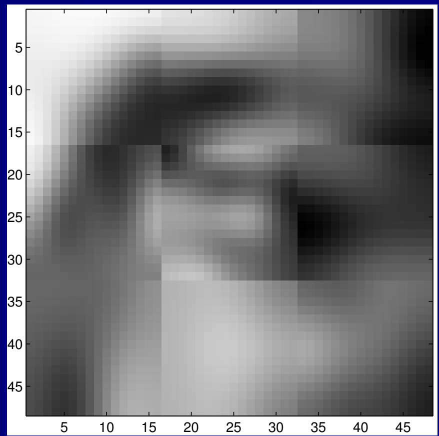


(e) LLST

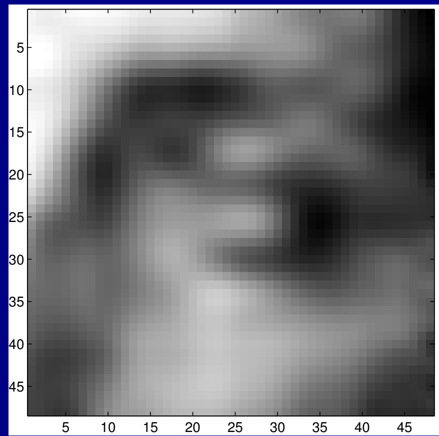
Zoomed up images: Top 5%



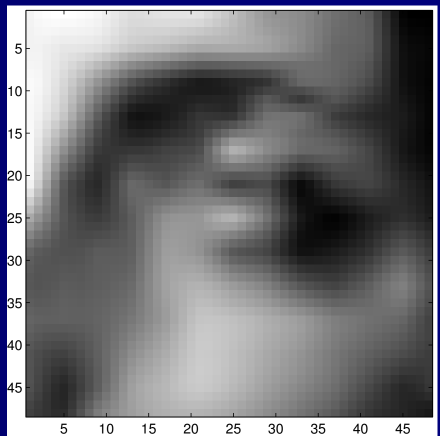
(a) Original



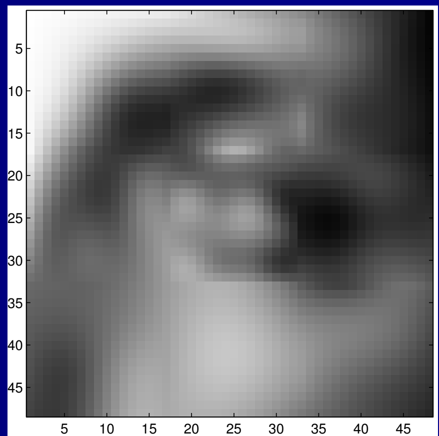
(b) DCT



(c) LCT

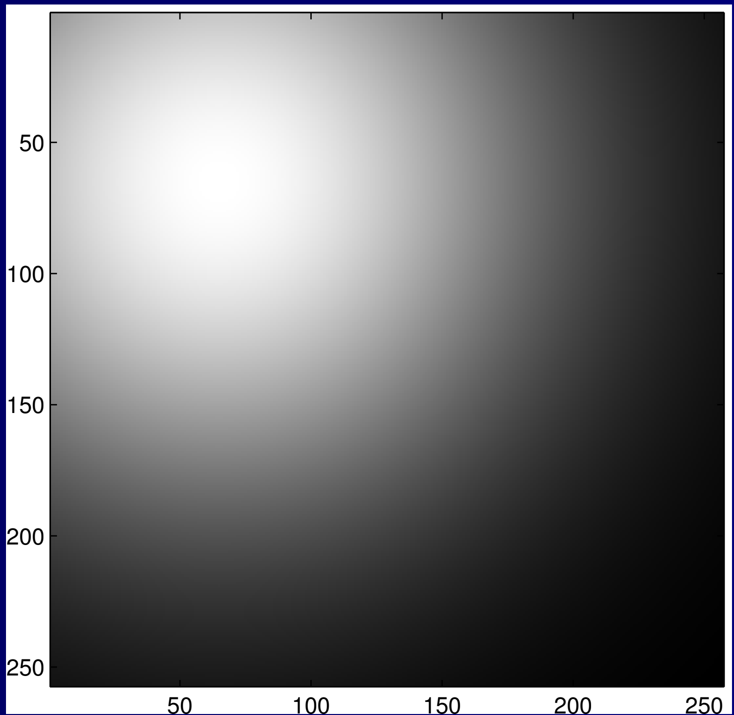


(d) C12

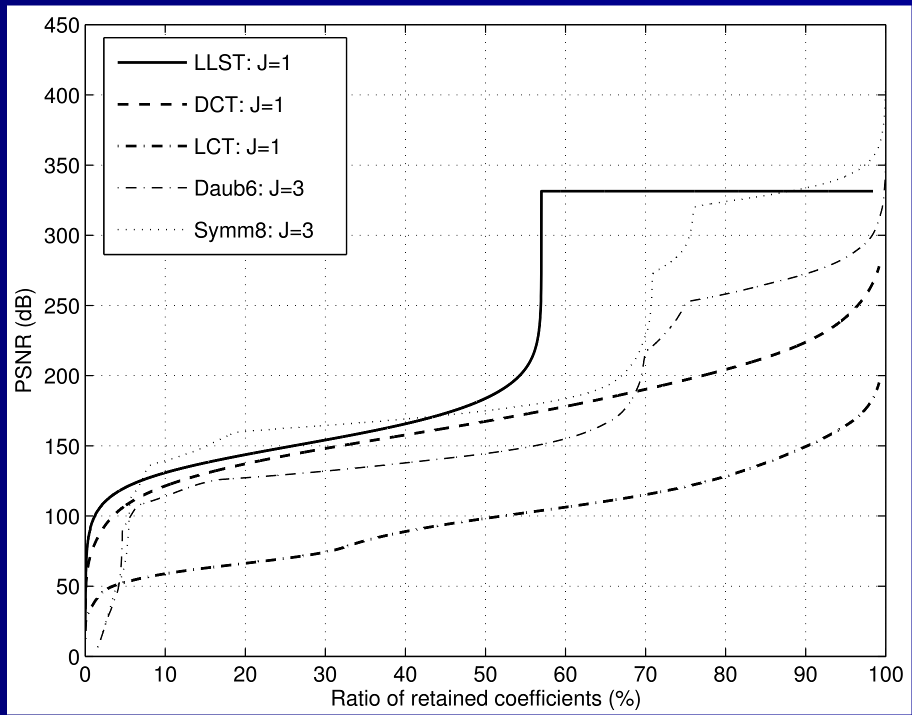


(e) LLST

Approximation Test: Smooth Function

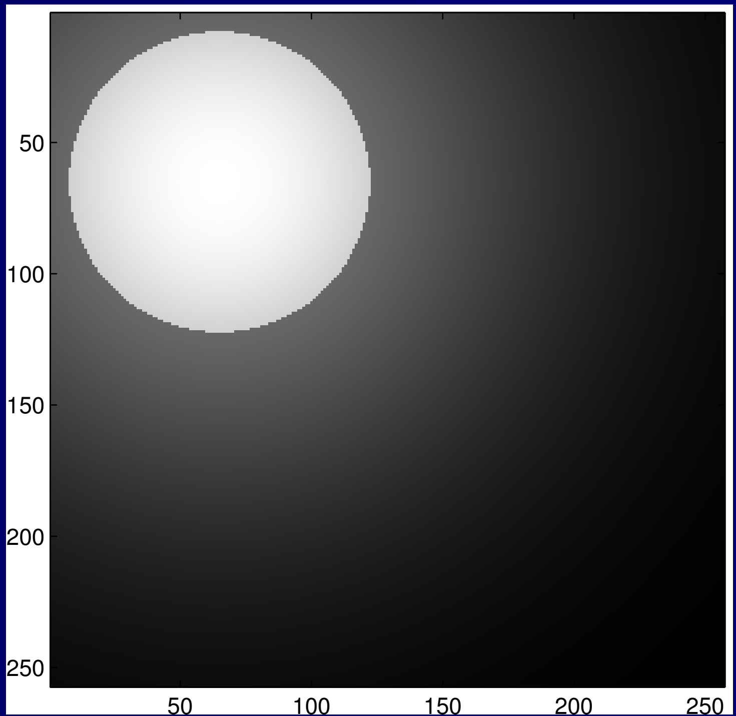


(a) Original

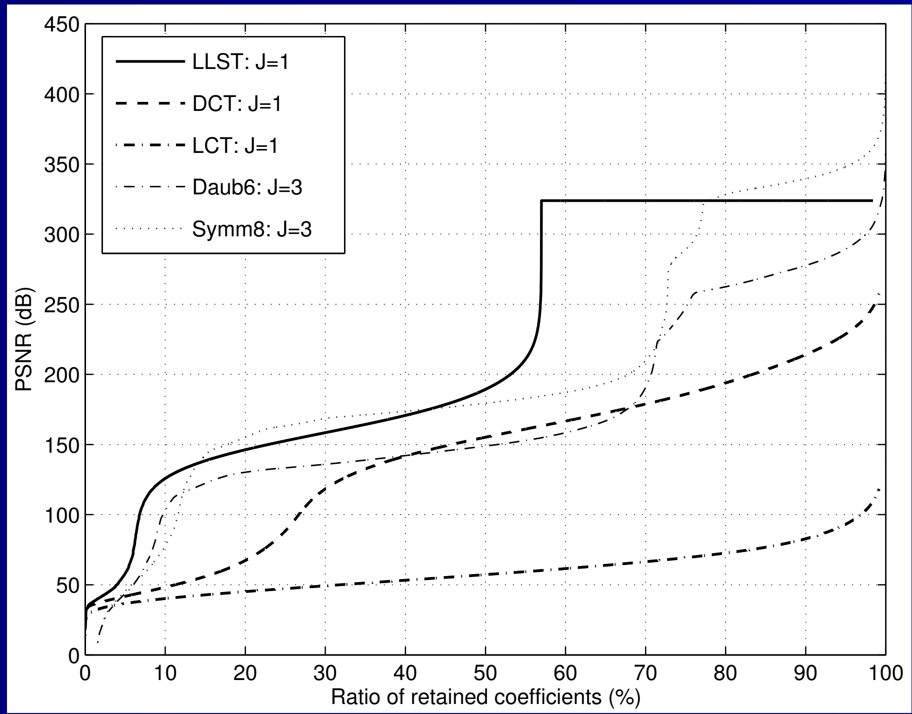


(b) PSNR

Approximation Test: Piecewise Smooth Function

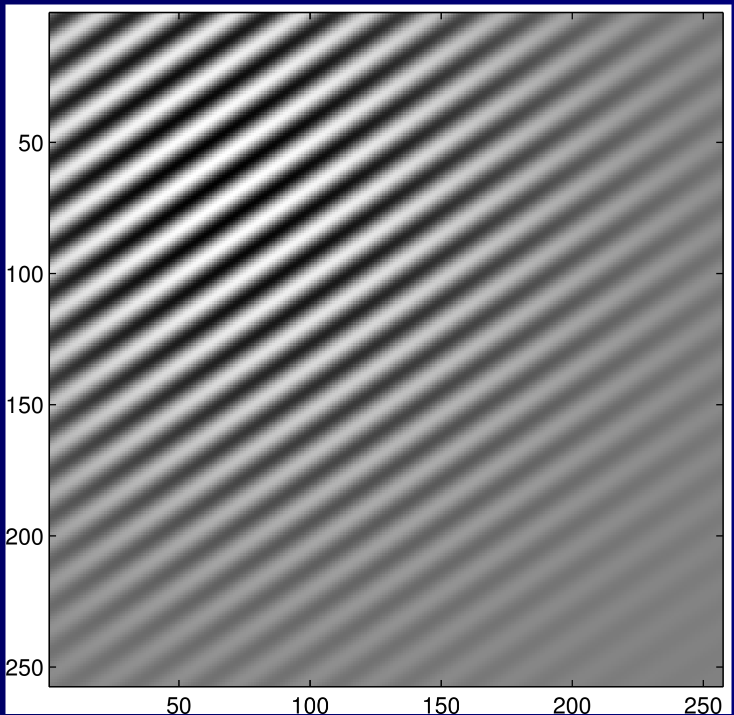


(a) Original

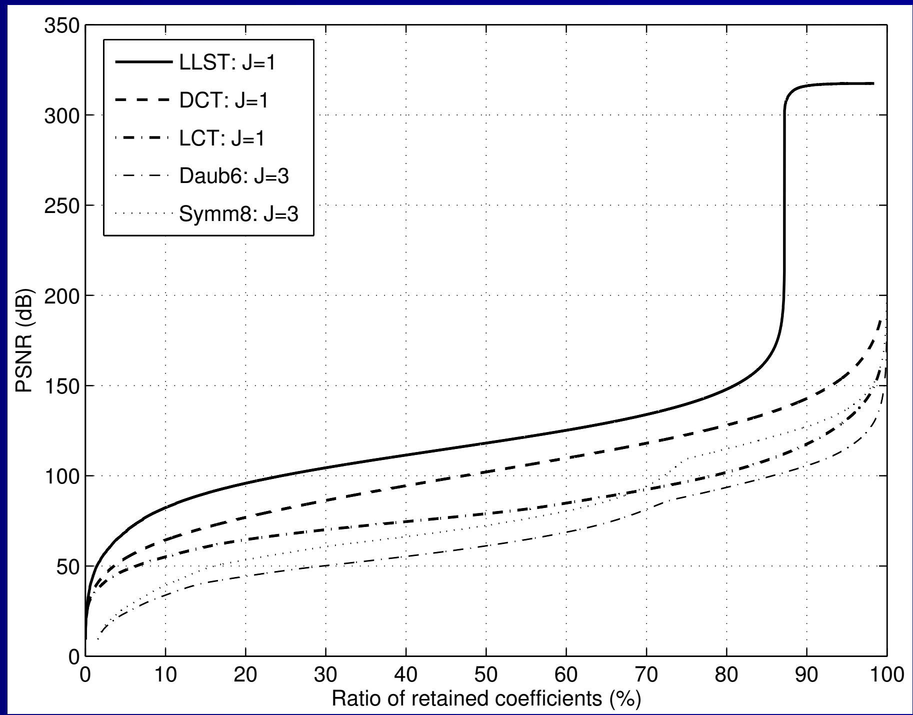


(b) PSNR

Approximation Test: Oscillatory Function

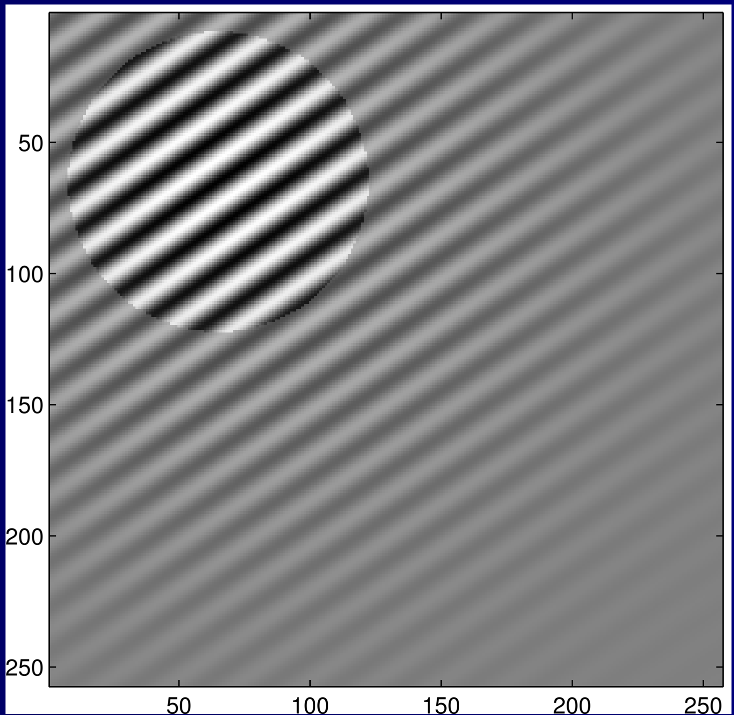


(a) Original

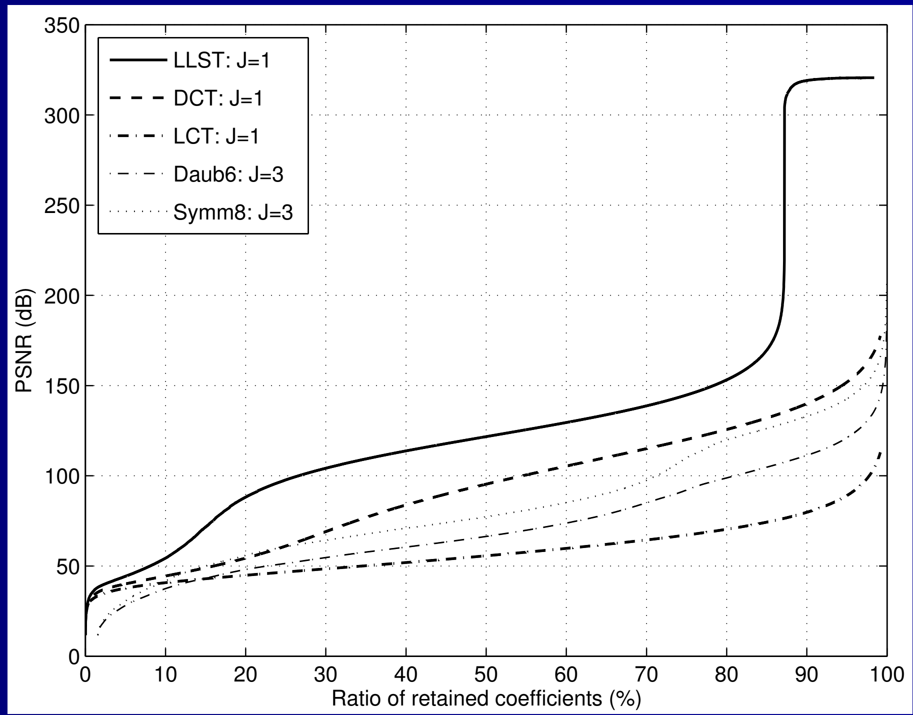


(b) PSNR

Approximation Test: Oscillatory Function with Discontinuity



(a) Original



(b) PSNR

Extension to 3D

- Boundary data contains more information compared to 1D and 2D

Extension to 3D

- Boundary data contains more information compared to 1D and 2D
- Fast 3D Laplace solver exists (the ABIV algorithm with our new organization of data): $O(12N^3 \log N)$

Extension to 3D

- Boundary data contains more information compared to 1D and 2D
- Fast 3D Laplace solver exists (the ABIV algorithm with our new organization of data): $O(12N^3 \log N)$
- Nicely recursive: corners; edges; faces; body contents

Extension to 3D

- Boundary data contains more information compared to 1D and 2D
- Fast 3D Laplace solver exists (the ABIV algorithm with our new organization of data): $O(12N^3 \log N)$
- Nicely recursive: corners; edges; faces; body contents
- Many possibilities: hierarchical decomposition, parallelization, 3D feature extraction, . . .

Extension to 3D

- Boundary data contains more information compared to 1D and 2D
- Fast 3D Laplace solver exists (the ABIV algorithm with our new organization of data): $O(12N^3 \log N)$
- Nicely recursive: corners; edges; faces; body contents
- Many possibilities: hierarchical decomposition, parallelization, 3D feature extraction, . . .
- Promising applications include analysis and compression of 3D seismic, medical, and video data, . . .

Extension to General Shape Domains

- Consider a general-shaped domain $\Omega \subset \mathbb{R}^n$.

Extension to General Shape Domains

- Consider a general-shaped domain $\Omega \subset \mathbb{R}^n$.
- Want to analyze the frequency information inside of the object.

Extension to General Shape Domains

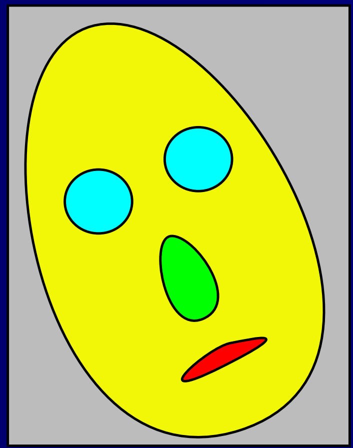
- Consider a general-shaped domain $\Omega \subset \mathbb{R}^n$.
- Want to analyze the frequency information inside of the object.
- Want to avoid the Gibbs phenomenon due to Γ.

Extension to General Shape Domains

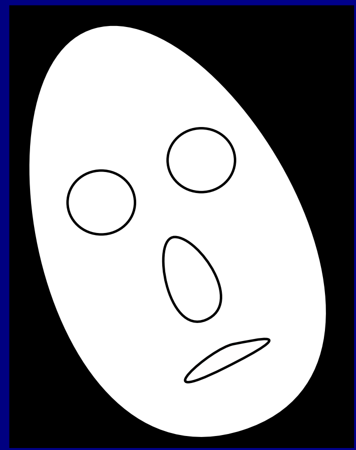
- Consider a general-shaped domain $\Omega \subset \mathbb{R}^n$.
- Want to analyze the frequency information inside of the object.
- Want to avoid the Gibbs phenomenon due to Γ.

$\Longrightarrow$ Smoothly extend the function defined on Ω to the outside!

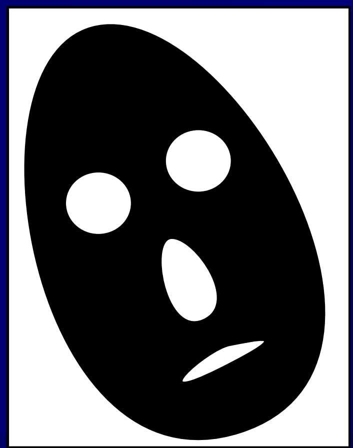
Object-Oriented Image Analysis



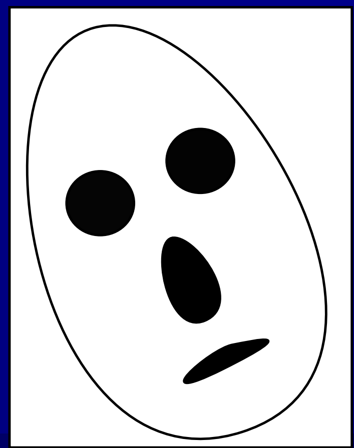
(a) Original



(b) Background



(c) Object



(d) Anomalies

Object-Oriented Image Analysis . . .

- Suppose its boundary Γ is already detected explicitly.

Object-Oriented Image Analysis . . .

- Suppose its boundary Γ is already detected explicitly.
- Let $f \in C^2(\Omega) \cap C(\overline{\Omega})$.

Object-Oriented Image Analysis . . .

- Suppose its boundary Γ is already detected explicitly.
- Let $f \in C^2(\Omega) \cap C(\overline{\Omega})$.
- Let S be a rectangular domain containing $\overline{\Omega}$.

Object-Oriented Image Analysis ...

- Suppose its boundary Γ is already detected explicitly.
- Let $f \in C^2(\Omega) \cap C(\overline{\Omega})$.
- Let S be a rectangular domain containing $\overline{\Omega}$.
- Then, we will find a smooth extension $\tilde{v}$ to the entire S from Ω and the harmonic function u in Ω:

$$\Delta \tilde{v} = \begin{cases} \Delta f & \text{in } \Omega, \\ 0 & \text{in } S \backslash \overline{\Omega}, \end{cases}$$

$$\tilde{v} = 0 \quad \text{on } \partial S.$$

Motivations for $\tilde{v}$

- $\tilde{v}$ is essentially the same as f in Ω modulo some harmonic function (in fact, this harmonic function becomes the u component);

Motivations for $\tilde{v}$

- $\tilde{v}$ is essentially the same as f in Ω modulo some harmonic function (in fact, this harmonic function becomes the u component);
- $\tilde{v}$ is harmonic (i.e., very smooth) in $S\backslash\overline{\Omega}$ (outside of Ω);

Motivations for $\tilde{v}$

- $\tilde{v}$ is essentially the same as f in Ω modulo some harmonic function (in fact, this harmonic function becomes the u component);
- $\tilde{v}$ is harmonic (i.e., very smooth) in $S\backslash\overline{\Omega}$ (outside of Ω);
- can show $\tilde{v} \in C^1(S)$

Motivations for $\tilde{v}$

- $\tilde{v}$ is essentially the same as f in Ω modulo some harmonic function (in fact, this harmonic function becomes the u component);
- $\tilde{v}$ is harmonic (i.e., very smooth) in $S\backslash\overline{\Omega}$ (outside of Ω);
- can show $\tilde{v} \in C^1(S)$
- can also show $\tilde{v}|_{S\backslash\Gamma} \in C^2(S\backslash\Gamma)$

Motivations for $\tilde{v}$

- $\tilde{v}$ is essentially the same as f in Ω modulo some harmonic function (in fact, this harmonic function becomes the u component);
- $\tilde{v}$ is harmonic (i.e., very smooth) in $S\backslash\overline{\Omega}$ (outside of Ω);
- can show $\tilde{v} \in C^1(S)$
- can also show $\tilde{v}|_{S\backslash\Gamma} \in C^2(S\backslash\Gamma)$
- $\tilde{v}|_{\partial S} = 0 \Longrightarrow$ suitable for Fourier sine series expansion, with decay rate $O(\|\boldsymbol{k}\|^{-3})$.

How to compute $\tilde{v}$

- Introduce the potential function:

$$P(\boldsymbol{x}) \triangleq \int_\Omega \Delta f(\boldsymbol{y}) \Phi(\boldsymbol{x}, \boldsymbol{y})\, \mathrm{d}\boldsymbol{y} \quad \text{for } \boldsymbol{x} \in \mathbb{R}^n.$$

where

$$\Phi(\boldsymbol{x}, \boldsymbol{y}) \triangleq \begin{cases} \frac{1}{2\pi} \log \|\boldsymbol{x} - \boldsymbol{y}\| & \text{if } n = 2, \\ \frac{\|\boldsymbol{x}-\boldsymbol{y}\|^{2-n}}{(2-n)\omega_n} & \text{if } n > 2, \end{cases}$$

is the so-called fundamental solution of the Laplace equation.

How to compute $\tilde{v}$ ...

- Can show that P satisfies the above Poisson equation:

$$\Delta P = \begin{cases} \Delta f & \text{in } \Omega, \\ 0 & \text{in } S \backslash \overline{\Omega}, \end{cases}$$

How to compute $\tilde{v}$ ...

- Can show that P satisfies the above Poisson equation:

$$\Delta P = \begin{cases} \Delta f & \text{in } \Omega, \\ 0 & \text{in } S \backslash \overline{\Omega}, \end{cases}$$

- But the boundary condition at the outside box is not satisfied, i.e., $P|_{\partial S} \neq 0$.

How to compute $\tilde{v}$ . . .

- Introduce now a boundary correction function Q:

$$\begin{aligned} \Delta Q &= 0 \quad \text{in } S, \\ Q &= -P \quad \text{on } \partial S; \end{aligned}$$

How to compute $\tilde{v}$ . . .

- Introduce now a boundary correction function Q:

$$\Delta Q = 0 \quad \text{in } S,$$
$$Q = -P \quad \text{on } \partial S;$$

- Finally set $\tilde{v}(\boldsymbol{x}) = P(\boldsymbol{x}) + Q(\boldsymbol{x})$, which satisfies the desired conditions.

How to compute $\tilde{v}$ ...

- Can write the solution via single layer and double layer potentials as:

$$P(\boldsymbol{x}) = \tilde{\chi}_\Omega(\boldsymbol{x}) f(\boldsymbol{x}) + \int_\Gamma \left(\frac{\partial f}{\partial \nu}(\boldsymbol{y}) \Phi(\boldsymbol{x}, \boldsymbol{y}) - f(\boldsymbol{y}) \frac{\partial \Phi(\boldsymbol{x}, \boldsymbol{y})}{\partial \nu_{\boldsymbol{y}}} \right) \mathrm{d}s(\boldsymbol{y}),$$

where

$$\tilde{\chi}_\Omega(\boldsymbol{x}) = \begin{cases} 1 & \text{if } \boldsymbol{x} \in \Omega; \\ 1/2 & \text{if } \boldsymbol{x} \in \Gamma; \\ 0 & \text{if } \boldsymbol{x} \in \mathbb{R}^n \backslash \overline{\Omega}. \end{cases}$$

Object-Oriented Image Analysis . . .

- The u component satisfies:

$$\Delta u = 0 \quad \text{in } \Omega,$$

$$u = f - \tilde{v} \quad \text{on } \Gamma.$$

Object-Oriented Image Analysis ...

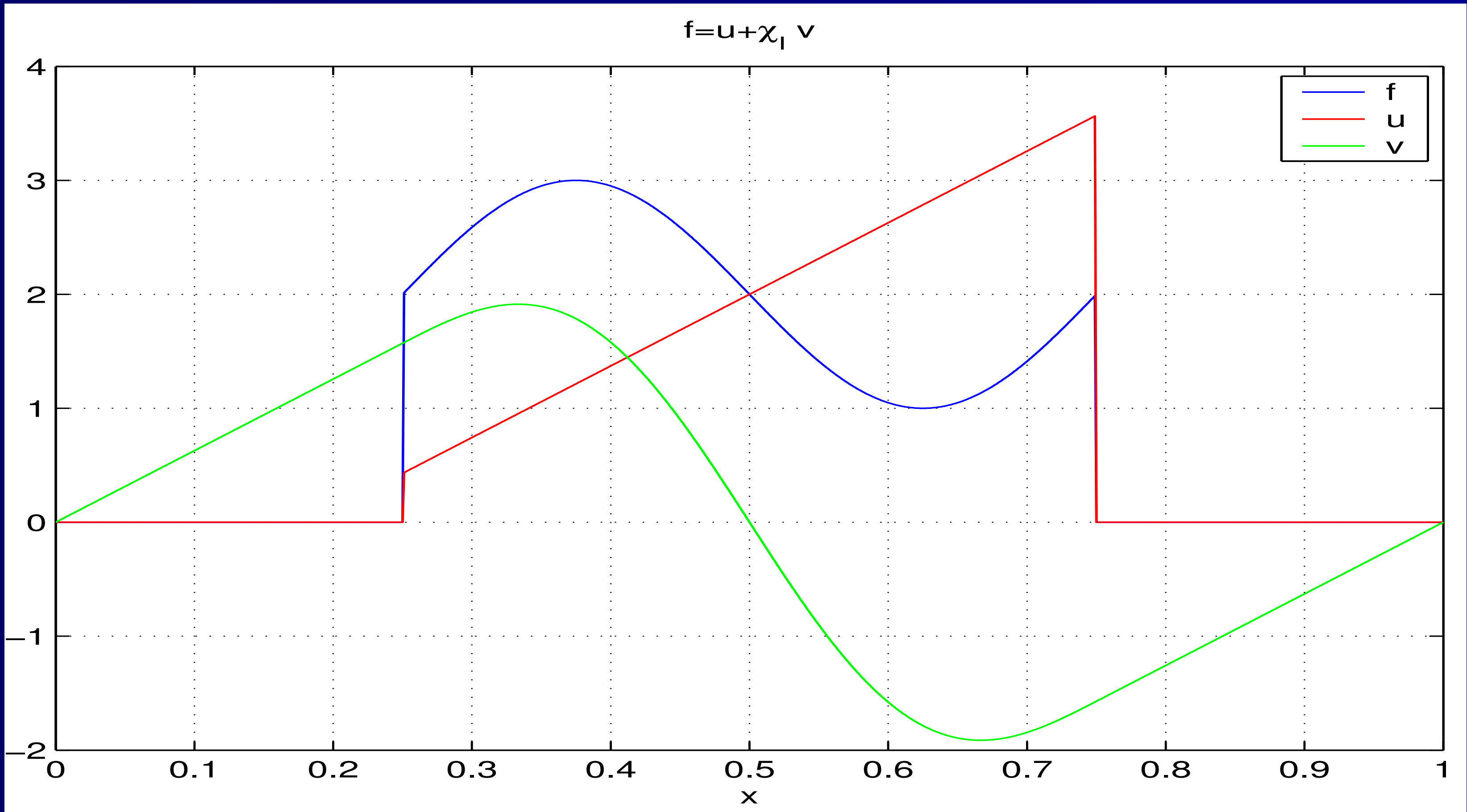
- The u component satisfies:

$$\Delta u = 0 \quad \text{in } \Omega,$$

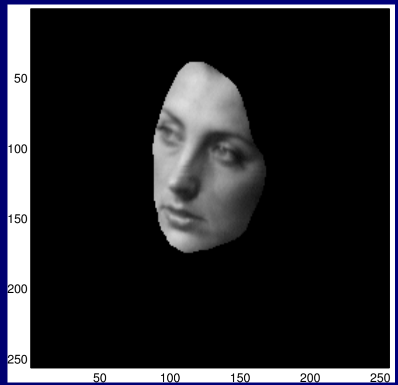
$$u = f - \tilde{v} \quad \text{on } \Gamma.$$

- u can be recovered from $f|_\Gamma$, $\left.\dfrac{\partial f}{\partial \nu}\right|_\Gamma$, and $\tilde{v}$ via Fast Multipole Method (just potential evaluation, no need to solve the Laplace equation).

1D Example

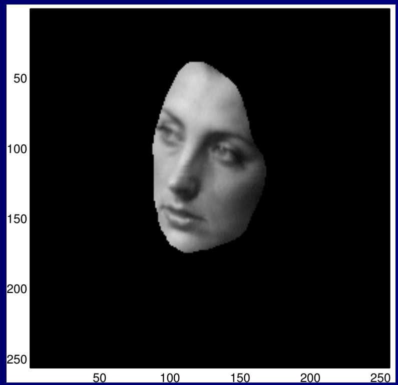


2D Example

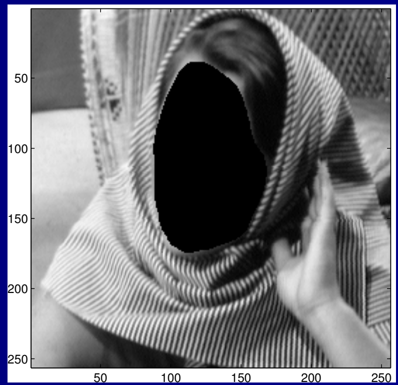


(a) Orig.1

2D Example

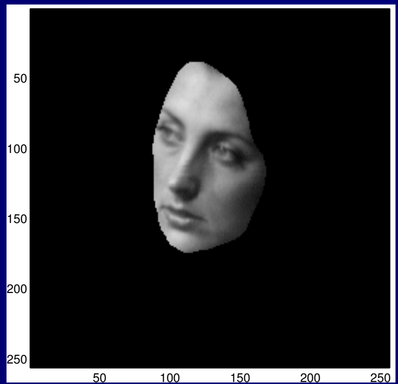


(a) Orig.1

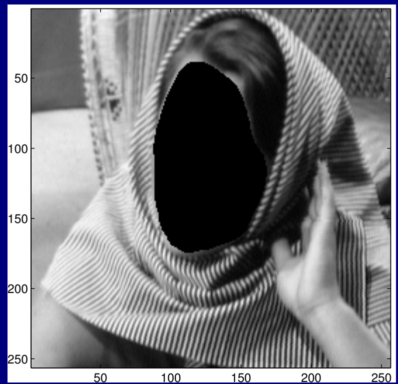


(b) Orig.2

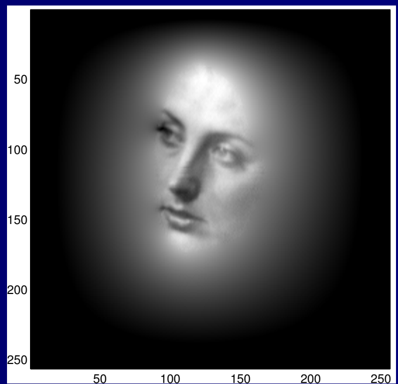
2D Example



(a) Orig.1

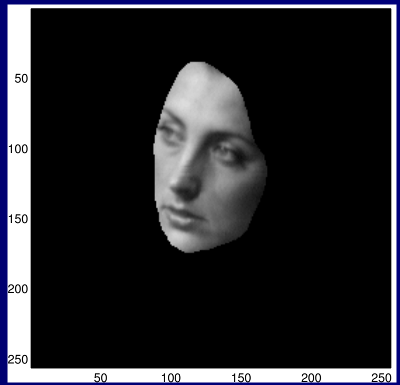


(b) Orig.2

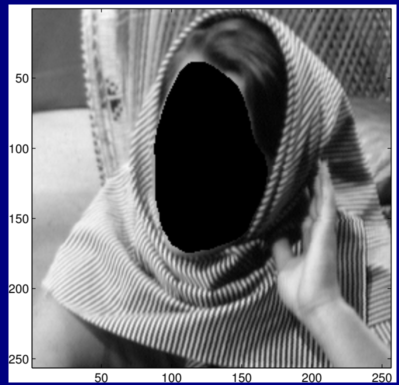


(c) Ext.1

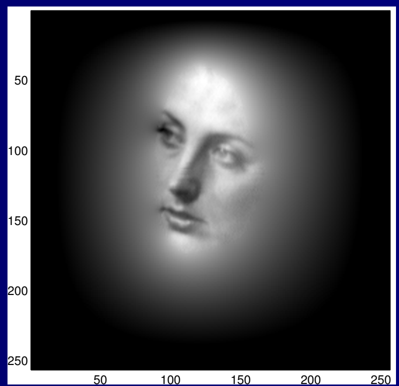
2D Example



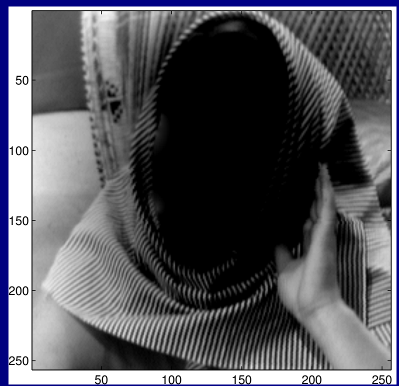
(a) Orig.1



(b) Orig.2

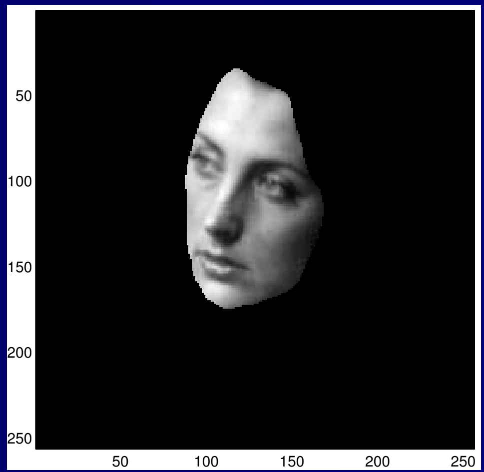


(c) Ext.1



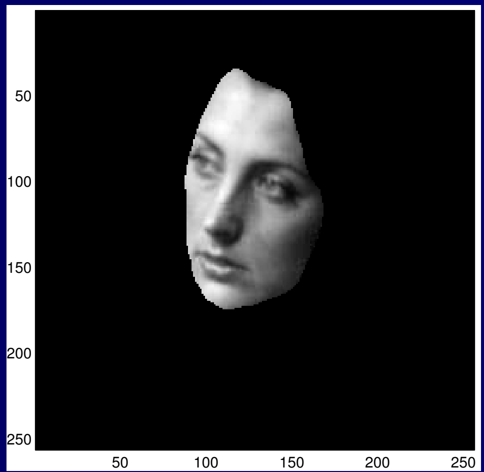
(d) Ext.2

2D Example . . .

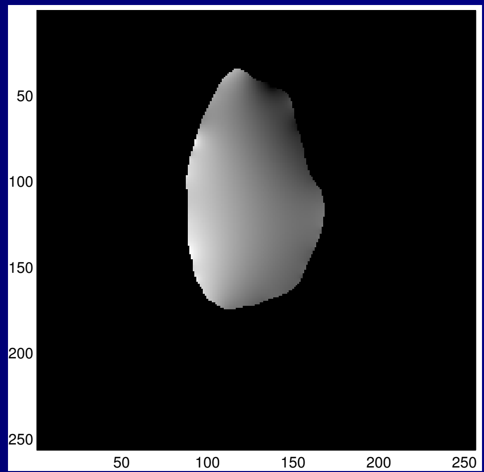


(a) Original

2D Example . . .

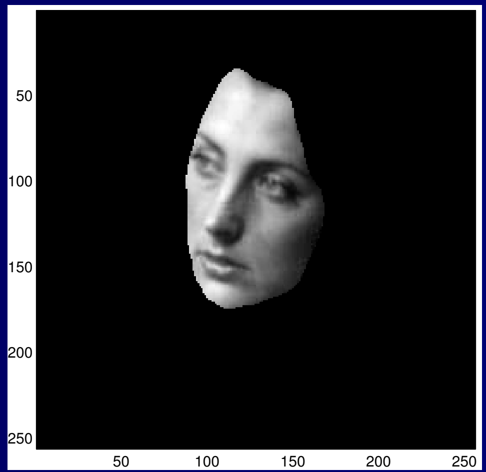


(a) Original

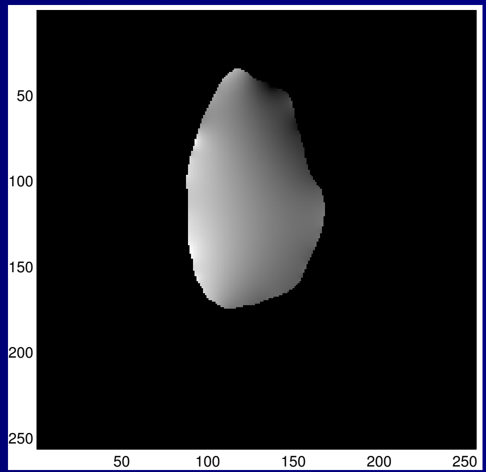


(b) u

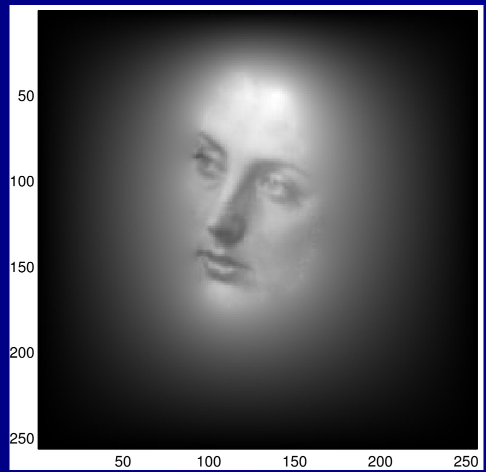
2D Example . . .



(a) Original

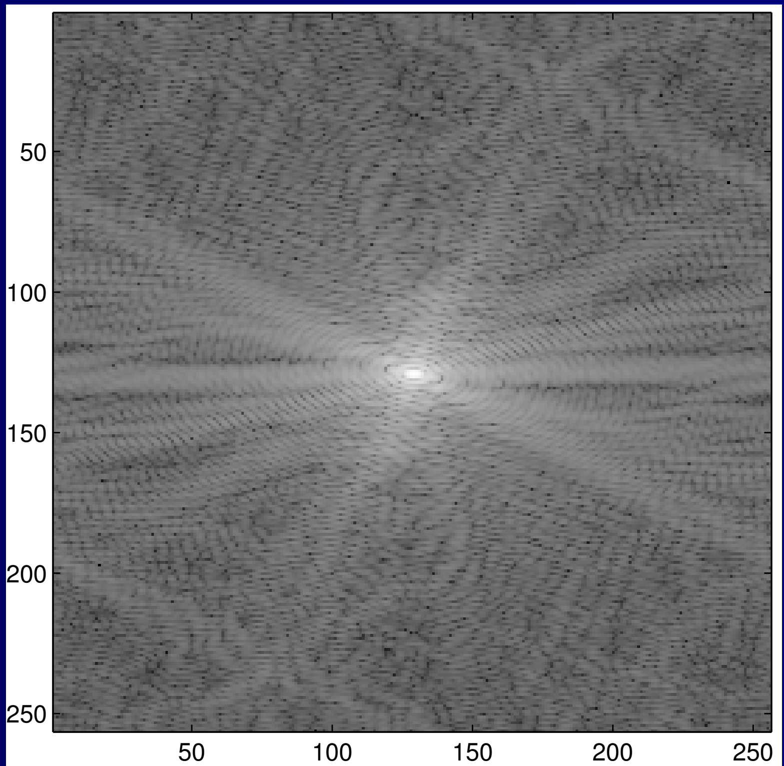


(b) u



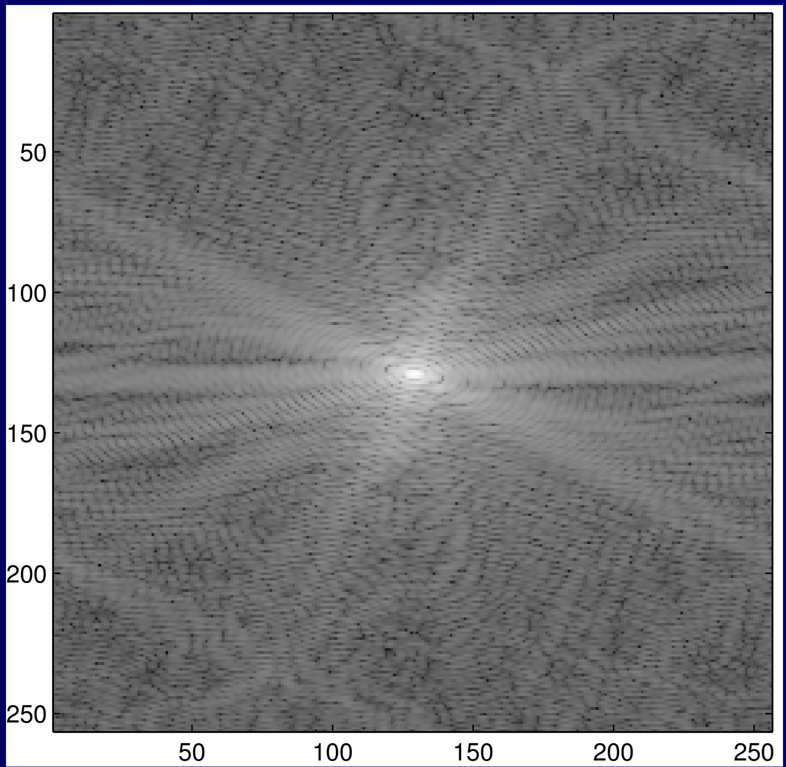
(c) $\tilde{v}$

The Fourier Transform Magnitudes (log-scale)

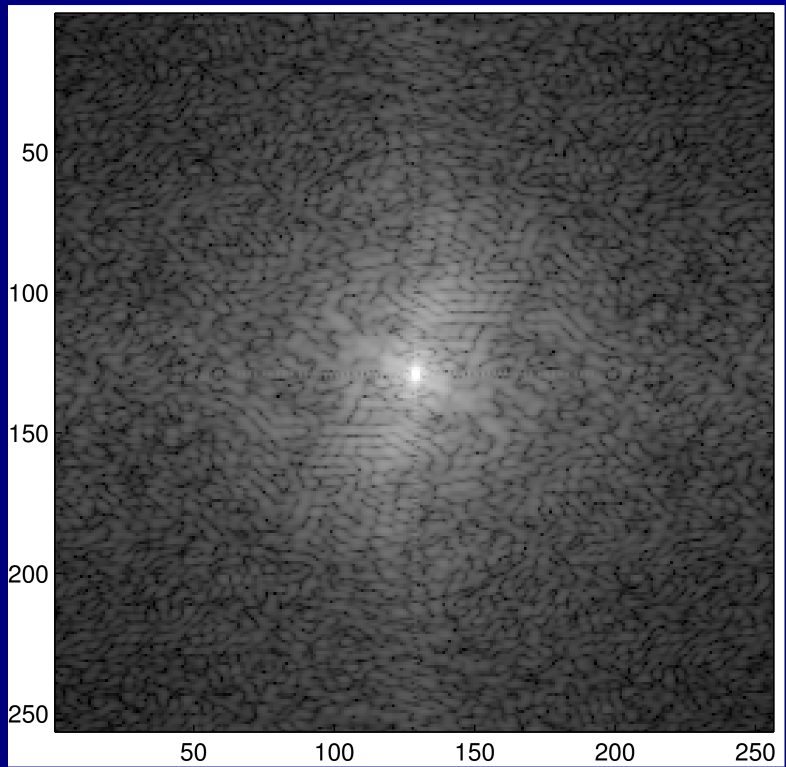


(a) Original

The Fourier Transform Magnitudes (log-scale)

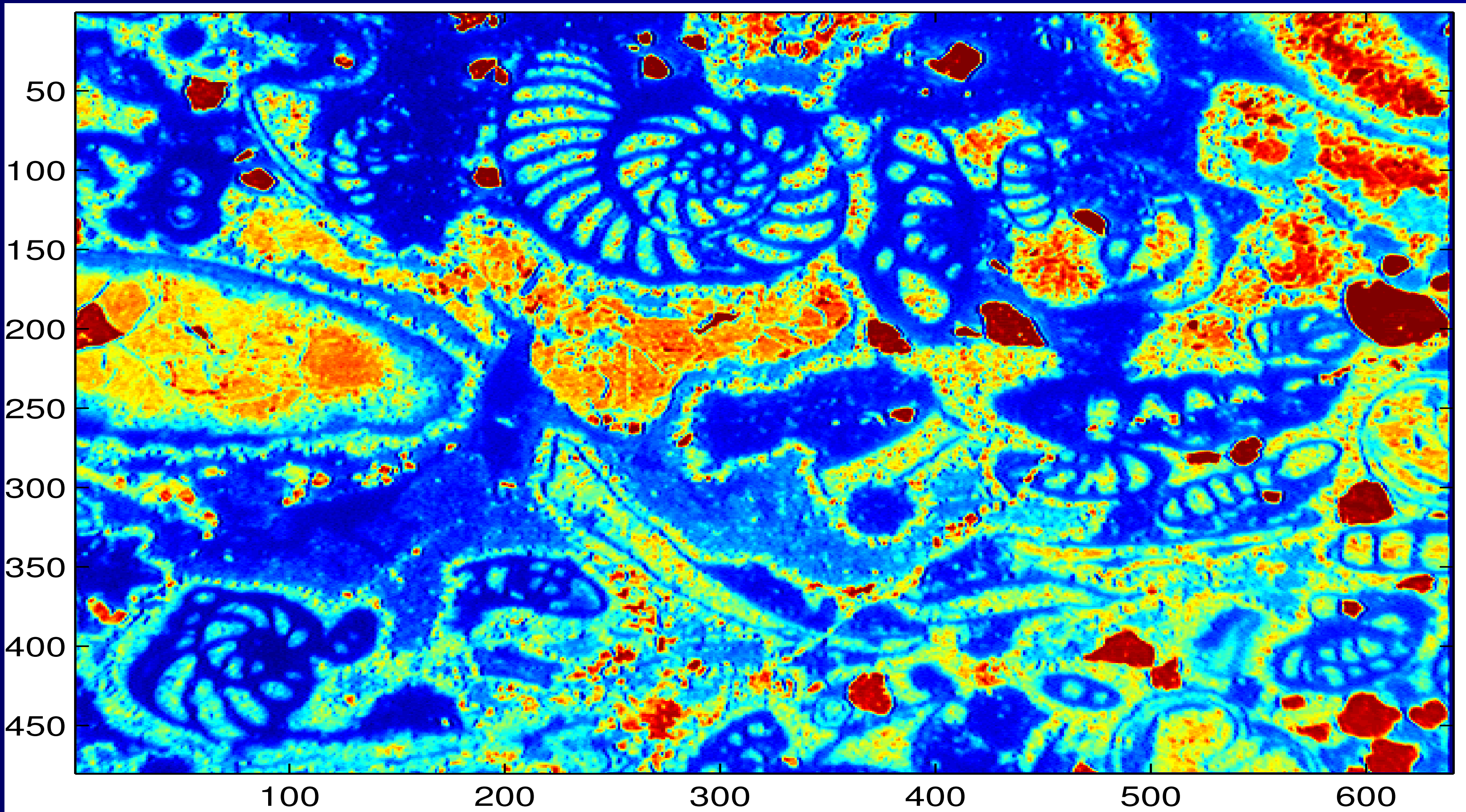


(a) Original

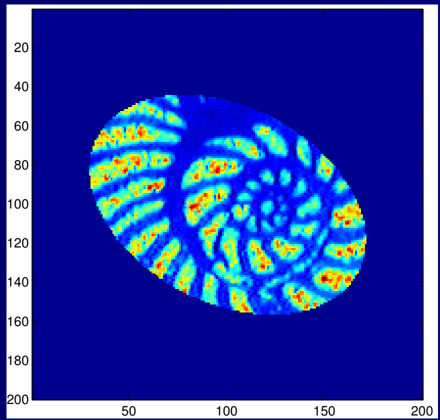


(b) $\tilde{v}$

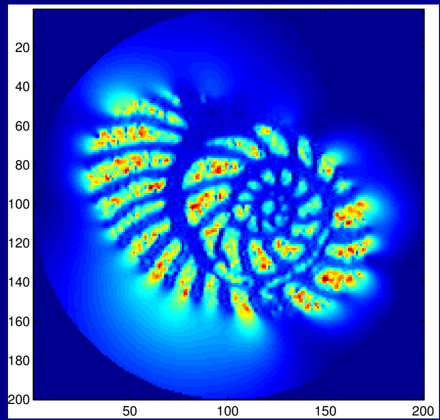
Another example



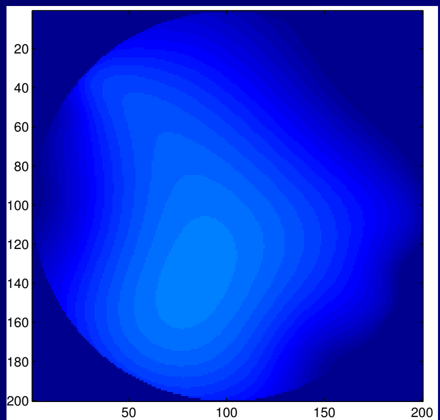
Another example . . .



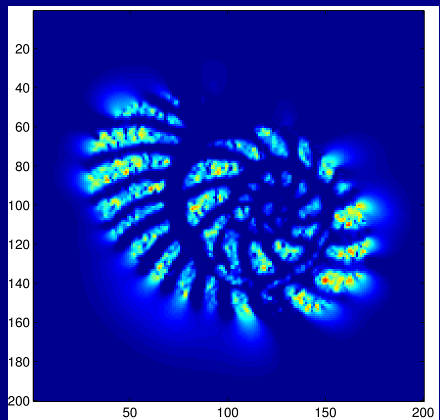
(a) Extracted shell



(b) Extended



(c) u component



(d) $\tilde{v}$ component

Related Previous Work

- Whitney (1934): Smooth extensions of $C^m(\overline{\Omega})$ functions in $\mathbb{R}^n$, and the Whitney decomposition
- Zygmund (1935): Removable discontinuities
- Lanczos (1938,1966): Trigonometric interpolation with linear/polynomial component removal
- Kantrovich & Krylov (1958): Rapidly convergent trigonometric series by polynomial removal
- Briggs (1974): Minimum curvature surface interpolation
- Grimson (1981): Visual surface reconstruction
- Terzopoulos (1983): Multilevel visual surface reconstruction

Related Previous Work . . .

- Leclerc (1989): Image segmentation using MDL and local polynomials
- Wahba (1990): Variational formulation/multivariate splines
- Dimitrov (1996): Polyharmonic functions for quadratures
- Kounchev (1993-2001): Theory of polysplines
- Casas & Torres (1996): Two-stage coding
- Caselles, Morel, & Sbert (1998): Axiomatic interpolation

All of them except Lanczos mainly focus on the u-component. Lanczos explored neither higher dimensions nor multiscale setting.

Conclusions

PHLTT and its generalizations:

- Provide a compact image/data representation scheme

Conclusions

PHLTT and its generalizations:

- Provide a compact image/data representation scheme
- Proactive use of boundary information

Conclusions

PHLTT and its generalizations:

- Provide a compact image/data representation scheme
- Proactive use of boundary information
- Can get faster decay expansion coefficients

Conclusions

PHLTT and its generalizations:

- Provide a compact image/data representation scheme
- Proactive use of boundary information
- Can get faster decay expansion coefficients
- Allow object-oriented image analysis & synthesis

Conclusions

PHLTT and its generalizations:

- Provide a compact image/data representation scheme
- Proactive use of boundary information
- Can get faster decay expansion coefficients
- Allow object-oriented image analysis & synthesis
- Variety of applications: segmentation, compression, interpolation, local feature computation, 3D, . . .

Comments

- This general area is a “meeting ground” of image processing, harmonic analysis, PDEs, and shape optimization

Comments

- This general area is a “meeting ground” of image processing, harmonic analysis, PDEs, and shape optimization
- A large possibility for prediction operators from boundary

Comments

- This general area is a “meeting ground” of image processing, harmonic analysis, PDEs, and shape optimization
- A large possibility for prediction operators from boundary
- Anisotropy and many interesting issues

Comments

- This general area is a “meeting ground” of image processing, harmonic analysis, PDEs, and shape optimization
- A large possibility for prediction operators from boundary
- Anisotropy and many interesting issues
- Reliability of boundary information and noise is an issue

References

- Check http://www.math.ucdavis.edu/˜saito/publications/ periodically for more preprints on this work.
- For preliminary work, see: N. Saito & J.-F. Remy: "A new local sine transform without overlaps: A combination of computational harmonic analysis and PDE," in *Wavelets X* (M. A. Unser, A. Aldroubi, & A. F. Laine, eds.), Proc. SPIE 5207, pp.495–506, 2003.
- See also references therein.

Thank you very much for your attention!