Abstract

- Curvelets is a latest method to represent objects with 1D singularity.
- The development history of curvelets
- Why we need curvelets
- The theory of curvelet transforms (1,2)
- The discrete curvelet transform
- Some recent applications using curvelets.
Development History

- Curvelets -- non-adaptive
- 1D discontinuity (Wavelet fails)
- Time complexity – about 10 times of 2D FFT.
Development History

Google search results for "fourier transform filetype:pdf" showing 3,450,000 results in 0.20 seconds.

A Fast Fourier Transform Compiler
File Format: PDF/Adobe Acrobat - View as HTML

Sponsored Links

Curvelet | David Shen 2006
Development History

Google search: wavelet filetype:pdf

Results 1 - 10 of about 1,080,000 for wavelet filetype:pdf. (0.29 seconds)

[Link: www.acm.org/pubs/articles/proceedings/graph/166117...]

Similar pages - Remove result
Development History

[PDF] Digital Curvelet Transform: Strategy, Implementation and Experiments

Curvelet | David Shen 2006
Development History

- 1998, Ridgelets, Dr. Candès
- 1999, Curvelet 99, Dr. Candès and Dr. Donoho
- 2002-present Curvelet.org, Fast discrete curvelet transform, Curvelab
Development History

- Curvelet Theories by Dr. Candes and Dohono
- Curvelet 99
- Second Generation Curvelet
- Discrete Curvelet Transform
- 3D Discrete Curvelet Transform
Development History

- Applications by Caltech, Stanford, Vanderbilt…
- Image denoising and enhancing
- Pattern recognition
Development History

- Although there are only seven years, curvelet is developing very fast.

Questions:
- Why are people studying curvelets?
- Why do we need curvelets?
Why do we need curvelet?

- Wavelet fails in 1D singularity
- Ideal adaptive methods are hard to implement
- Curvelet is promising
Why do we need curvelet?

- **Approximation Rates of Adaptive & Non-adaptive Schemes**

Suppose we have an object supported in $[0,1]^2$ which has a discontinuity across a nice curve $\Gamma$, and which is otherwise smooth.
Why do we need curvelet?

- The comparing of the approximation using the best $m$ nonzero terms

  Fourier method \[ \| f - \tilde{f}_m \|_2^2 \approx m^{-1/2}, m \to \infty \]

  Wavelet method \[ \| f - \tilde{f}_m \|_2^2 \approx m^{-1}, m \to \infty \]

  Adaptive method \[ \| f - \tilde{f}_m \|_2^2 \approx m^{-2}, m \to \infty \]

  Curvelet method \[ \| f - \tilde{f}_m \|_2^2 \leq C \cdot m^{-2} (\log m)^3, m \to \infty \]
Why do we need curvelet?

- Conclusion:
  - Wavelet failed in the presence of curve discontinuity
  - Adaptive methods are ideal but difficult to implement
  - Non-adaptive methods (curvelet) can represent the ideal behavior of an adaptive representation

So what is curvelet? How does it achieve such good results?
What is curvelet?

--- Basic Idea

- **Basic Idea**
  - We define one curvelet as:
What is curvelet?

--- Basic Idea
What is curvelet?

--- Basic Idea

\[ c_{j,l,k} = \langle f, \varphi_{j,l,k} \rangle = \int_{\mathbb{R}^2} f(x) \varphi_{j,l,k}(x) dx \]

\[ f = \sum_{j,l,k} \langle f, \varphi_{j,l,k} \rangle \varphi_{j,l,k} \]
What is curvelet?
-- Basic Idea
What is curvelet?

-- Basic Idea

The questions here are:

- Is curvelet a **tight frame**?
- Or \( \sum_{\mu} |< f, \gamma_{\mu} >|^2 = \|f\|_{L_2(R^2)}^2 \) ?
- And so shall we have **reconstruction** formula of \( f = \sum_{\mu} < f, \gamma_{\mu} > \gamma_{\mu} \) ?
First generation curvelet

- It is based on:
  - Ridgelets
  - Multiscale Ridgelets
  - Bandpass Filtering
First generation curvelet
---Ridgelets

Ridge functions
\[ \int \psi(t) dt = 0 \]

\[ \Psi_{a,b,\theta}(x_1, x_2) = a^{-1/2} \Psi((x_1 \cos(\theta) + x_2 \sin(\theta) - b) / a) \]
Continuous ridgelet transform

\[ R_f(a, b, \theta) = \left\langle \Psi_{a, b, \theta}(x), f \right\rangle \]

So

\[ f(x) = \int_{0}^{2\pi} \int_{-\infty}^{+\infty} \int_{0}^{\infty} R_f(a, b, \theta) \Psi_{a, b, \theta}(x) \frac{da}{a^3} db \frac{d\theta}{4\pi} \]
First generation curvelet
--Multiscale Ridgelets

Multiscale Ridgelets

Think ortho-ridgelets as objects which have
a “length” of about 1
“width” which can be arbitrarily fine.
First generation curvelet
--Multiscale Ridgelets

- This multiscale ridgelets are not framable

\[ \sum_{\mu \in M} <\Psi_\mu, f>^2 = \infty \]

- Curvelet definition only allows ridgelets of \( width \approx length^2 \)
Subband Filtering

\[ \| f \|_2^2 = \| P_0 f \|_2^2 + \sum_s \| \Delta_s f \|_2^2 \]
First generation curvelet

Curvelet Transform

Curvelet Transform

Curvelet Coefficients
At coarse scale
\[ \alpha_\mu = \langle \varphi_{k_1,k_2}, P_0 f \rangle \]
\[ \mu = (k_1, k_2) \in M \setminus M \]
\( \varphi_{k_1,k_2} \) is the Lemarie scaling function of the Meyer basis

At fine scale
\[ \alpha_\mu = \langle \Delta_s f, \psi_\mu \rangle \]
\[ \mu \in M_s, s = 1, 2, ... \]
First generation curvelet
Curvelet Transform

Curvelet Properties

Tight frame

\[ \|f\|_2^2 = \sum_{\mu \in M'} |\alpha_\mu|^2 \]

Existence of Coefficient Representations (Frame Elements)

\[ \alpha_\mu \equiv \langle f, \gamma_\mu \rangle \]

\( L^2 \) Reconstruction Formula

\[ f = \sum_{\mu \in M'} \langle f, \gamma_\mu \rangle \gamma_\mu \]

Formula for Frame Elements

\[ \gamma_\mu = \Delta_s \psi_\mu \quad \mu \in Q_s \]

Anisotropy Scaling Law

(1) Most curvelet are negligible in norm (most multiscale ridgelets do not survive the Bandpass filtering \( \Delta_s \)).

(2) The non-negligible curvelet obey

\[ \text{length} \approx 2^{-s} \quad \text{width} \approx 2^{-2s} \quad \text{so: width} \approx \text{length}^2 \]
Second Generation Curvelet

Why Do We Need Second Generation?

First generation (Curvelet99)
- Curvelet99 uses 7-index structure $\mu = (s, k_1, k_2, j, k, i, l, e)$
- Curvelet99 uses multiscale window.

Second Generation
- Only use three indices: scale, angle and location
- Doesn’t use the multiscale window
Second Generation Curvelet Definition

Window Definition

\[ \sum_{j=-\infty}^{\infty} W^2(2^j r) = 1 \quad r \in (3/4, 3/2) \]

\[ \sum_{j=-\infty}^{\infty} V^2(t-1) = 1 \quad t \in (-1/2, 1/2) \]

\[ U_j(r,\theta) = 2^{-3j/4} W(2^{-j} r)V\left(\frac{2^{j/2}}{2\pi}\theta\right) \]
Second Generation Curvelet Definition

\[ U_j(r, \theta) = 2^{-3j/4} W(2^{-j} r) V \left( \frac{2^{j/2}}{2\pi} \theta \right) \]

- Curvelet tiling of space and frequency.
Second Generation Curvelet Definition

Define the mother curvelet in frequency space as:

\[ \hat{\phi}_j(w) = U_j(w) \]
Second Generation Curvelet
Definition

\[ \hat{\varphi}_j(w) = U_j(w) \]

At scale \(2^{-j}\), orientation \(\theta_l\), position

\[ x_k^{(j,l)} = R_{\theta_l}^{-1}(k_1 \cdot 2^{-j}, k_2 \cdot 2^{-j/2}) \]

\[ \varphi_{j,l,k}(x) = \varphi_j \left( R_{\theta_l} \left( x - x_k^{(j,l)} \right) \right) \]

\(R_\theta\) is the rotation by \(\theta\) radians.
Curvelet coefficient

\[ c(j, l, k) = < f, \varphi_{j,l,k} > = \int_{\mathbb{R}^2} f(x) \varphi_{j,l,k}(x) dx \]

\[ = \frac{1}{(2\pi)^2} \int \hat{f}(\omega) \hat{\varphi}_{j,l,k}(\omega) d\omega \]

\[ = \frac{1}{(2\pi)^2} \int \hat{f}(\omega) U_{j}(R_{\theta}, \omega) e^{ix_{k}^{(j,l)},\omega} d\omega \]
Second Generation Curvelet Property

(1) **Tight Frame**: we have the reconstruction formula \( f = \sum_{j,l,k} < f, \varphi_{j,l,k} > \varphi_{j,l,k} \) and we have the Parseval relation: \( \sum_{j,l,k} |< f, \varphi_{j,l,k} >|^2 = \| f \|^2_{L^2(R^2)} \quad \forall f \in L^2(R^2) \)

(2) **Anisotropy Scaling Law**: 

\[
\text{length} \approx 2^{-j} \quad \text{width} \approx 2^{-2j} \quad \text{so}: \text{width} \approx \text{length}^2
\]

(3) **Oscillatory Nature**: Curvelet elements display oscillatory components across the ‘ridge’
Why Curvelet generates better result?

There are three types of curvelets.
Why Curvelet generates better result?
Why Curvelet generates better result?

Type C:

\[ |\theta_\mu| = |< f, \gamma_\mu | \leq \| f \|_{L_\infty} \cdot \| \gamma_\mu \|_{L_1} \]

\[ \| \gamma_\mu \|_{L_1} \leq B \cdot 2^{-3j/4} \]

So \[ |\theta_\mu| \leq B \cdot 2^{-3j/4} \cdot \| f \|_{L_\infty} \]
Why Curvelet generates better result?

At scale $2^{-j}$, we have at most $O(2^{j/2})$ coefficients of type C which are bounded by $C \cdot 2^{-3j/4}$.

Assuming other types of coefficients are negligible, the nth largest coefficient

$$|\theta|_{(n)} \leq C \cdot 2^{-3j/4}$$

$$\|f - f_n\|_{L^2}^2 \leq \sum_{m>n} |\theta|_{(m)}^2 \leq C \cdot n^{-2}$$
From 2002 to 2005, Dr. Candès and Dr. Donoho started the research of discrete curvelet transform.

- 2D discrete curvelet transform
- 3D discrete curvelet transform
- Curvelab2.0 [www.curvelet.org](http://www.curvelet.org)
\[ C^D(j,l,k) = \sum_{0 \leq t_1, t_2 < n} f[t_1, t_2] \phi^D_{j,l,k}[t_1, t_2] \]
Digital Curvelet Transform via Unequispaced FFT’s

\[ \tilde{U}_{j,l}(\omega) = W_j(\omega)V_j(S_{\theta_l}(\omega)) \]

\[ V_j(\omega) = V\left(2^{\frac{j}{2}} \frac{\omega_2}{\omega_1}\right) \]

\[ V_j(S_{\theta_l}(\omega)) = V_j(\omega') = V\left(2^{\frac{j}{2}} \frac{\omega_2'}{\omega_1}\right) = V\left(2^{\frac{j}{2}} \frac{\omega_2 - \omega_1 \tan \theta_l}{\omega_1}\right) \]

\[ = V\left(2^{\frac{j}{2}} \left(\frac{\omega_2}{\omega_1} - \tan \theta_l\right)\right) = V\left(2^{\frac{j}{2}} \frac{\omega_2}{\omega_1} - l\right) \]
\[ C^D(j, l, k) = \sum_{0 \leq t_1, t_2 < n} f[t_1, t_2] \overline{\phi^D_{j,l,k}[t_1, t_2]} \]

**Main Steps**

1. 2D FFT \( \rightarrow \) Fourier samples \( \hat{f}[n_1, n_2] \)
2. For any \((j, l)\), resample \( \hat{f}[n_1, n_2] \rightarrow \hat{f}[n_1, n_2 - n_1 \tan \theta_i] \)
3. \( \hat{f}_{j,l}[n_1, n_2] = \hat{f}[n_1, n_2 - n_1 \tan \theta_i] \tilde{U}_j[n_1, n_2] \)
4. Apply the inverse 2D FFT to each \( \tilde{f}_{j,l} \), hence collecting the discrete coefficients \( C^D(j, l, k) \)
Digital Curvelet Transform via Unequispaced FFT’s

Time and Space Consumption

Time: $O(n^2 \log n)$ flops
Space: $O(n^2)$, $n^2$ is number of pixels
\[ C^D(j, l, k) = \sum_{0 \leq t_1, t_2 < n} f[t_1, t_2] \psi_{j,l,k}^D[t_1, t_2] \]

**Main Steps**

1. 2D FFT \( \rightarrow \hat{f}[n_1, n_2] \)
2. For any scale/angle \( \rightarrow \tilde{U}_{j,l}[n_1, n_2] \hat{f}[n_1, n_2] \)
3. Wrap this product around the origin \( \rightarrow \tilde{f}_{j,l}[n_1, n_2] = W(\tilde{U}_{j,l} \hat{f})[n_1, n_2] \)
4. Inverse 2D FFT to each \( \tilde{f}_{j,l} \rightarrow C^D(j, l, k) \)
Digital Curvelet Transform via Wrapping

Time and Space Consumption

Time: $O(n^2 \log n)$ flops
Space: $O(n^2)$, $n^2$ is number of pixels

However, in practice, the time does not exceed 6-10 2D FFT transform. The curvelet inverse transform using Wrapping method is much faster than the USFFT.
3D Discrete Curvelet Transform

(a) 3D frequency tilings. (b)

\[ \omega_1 \omega_2 (1, \alpha, \beta) \]

Curvelet | David Shen 2006
3D Discrete Curvelet Transform

\[
\tilde{V}_{j,l}(w) = \tilde{V}(2^{j/2} \frac{w_2 - \alpha_i w_1}{w_1}) \cdot \tilde{V}(2^{j/2} \frac{w_3 - \beta_i w_1}{w_1})
\]

\[
\tilde{U}_{j,l}(w) = \tilde{W}_{j}(w) \tilde{V}_{j,l}(w)
\]

\[
\hat{\phi}^D_{j,l,k}(w) = \frac{\tilde{U}_{j,l}(\omega)}{\sqrt{L_{1,j,l} \cdot L_{2,j,l} \cdot L_{3,j,l}}} e^{-2\pi i (k_1 w_1 / L_{1,j,l} + k_2 w_2 / L_{2,j,l} + k_3 w_3 / L_{3,j,l})}
\]

\[
C^D(j,l,k) = \sum_{0 \leq t_1, t_2, t_3 < n} f[t_1, t_2, t_3] \hat{\phi}^D_{j,l,k}[t_1, t_2, t_3]
\]
Curvelet applications and demos

- **Denoising**
  - (Top left) Noisy image
  - (top right) filtered images using the decimated wavelet transform,
  - (bottom left) the undecimated wavelet transform
  - (bottom right) curvelet transform.

Reference: Zhibin Lei, Yin Chan, Daniel Lopresti “Image Curvelet Feature Extraction and Matching”
Curvelet applications and demos
-- Multi methods comparison

- Top: Original
- Bottomleft: a trous wavelet
- Bottomright: ridgelet

Reference:
Jean Stark, Michael Elad, David Donoho, “Redundant multiscale transforms and their applications for Morphological component separation”
Curvelet applications and demos --Multi methods comparison

**Reference:**
Jean Stark, Michael Elad, David Donoho, “Redundant multiscale transforms and their applications for Morphological component separation”

Fig. 26. Upper left, galaxy SBS 0335-052 (10 μm), upper middle, upper right, and bottom left, reconstruction respectively from the ridgelet, the curvelet and wavelet coefficients. Bottom middle, residual image. Bottom right, artifact free image.
Curvelet applications and demos
--Line Feature Extraction

Used features of Curvelab 2.0
Today, we talked about:
- The history of curvelets,
- How curvelet looks like and why it generates good results,
- Second generation of curvelet transform,
- Discrete curvelet transform,
- Applications.
Summarize

- We see clearly curvelet is developing very fast in the recent years.
- Many researchers and institute are interested in this theory.
- We believe curvelets is a very promising theory!
Resources

- www.curvelet.org
- Dr. Candes homepage: http://www.acm.caltech.edu/~emmanuel
- Dr. Dohono homepage: http://www-stat.stanford.edu/~donoho/
The End

- Thanks for your time!