JBEAM: Multi-scale Curve Coding via Beamlets

Xiaoming Huo

School of Industrial and Systems Engineering
Georgia Institute of Technology
9 / 23 / 2004

w. Jihong Chen
Beamlets & Beamlet Dictionary

Donoho, Ann. of Stat. 1999

Donoho & Huo, 2001, 2002, etc.
Objective

- Coding linear and curvilinear features
- Combine Beamlet and Zero-tree Coding
- Outperform existing industrial standards?
- Variations/Improvements/Software

Wavelet ↔ Image Coding
Beamlet ↔ Curve Coding
Outline for JBEAM

- I. Motivations
- II. Preparation:
  - Coding Single Beamlet
  - Beamlet representation
- III. Beamlet coder (JBEAM)
  - Rate-Distortion Optimized Representation
  - Zero-tree Coding
- IV. Simulation / Discussion
- V. Conclusion
I. Motivation

(1) Shape Coding

- Video object-based coding

Recent publications in IEEE Trans. I.P.
Motivation (2) Line Images

Line drawings
## II. Prep.-1: Coding a Single Beamlet

<table>
<thead>
<tr>
<th>Scale, $s$</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Size, $2^s$</td>
<td>1</td>
<td>2</td>
<td>4</td>
<td>8</td>
<td>16</td>
</tr>
<tr>
<td># beamlets</td>
<td>1</td>
<td>(4)</td>
<td>(12)</td>
<td>(28)</td>
<td>(60)</td>
</tr>
<tr>
<td># bits</td>
<td>(1)</td>
<td>(3)</td>
<td>7</td>
<td>9</td>
<td>11</td>
</tr>
</tbody>
</table>

Number of bits that are required

Coded beamlets with partial bit stream
Coding Scheme for a beamlet

- Beamlets to intervals.
- Is X covered? 0/1.
- Coding two ends 0/1, 0/1.
- Recursion
A Progressive Scheme to Code Single Beamlets

1) For a $2^s$ by $2^s$ image, it takes $2s + 3$ bits to code EXACTLY.

2) While $2k(k \leq s + 1)$ bits are available, the MAX distortion is $2^{s+1-k}$, which is lower bound in IT.

3) Partial bits, partial reconstruction, as in illustration.
Illustrations

4 bits

8 bits
II. Prep.-2: Beamlet Representation

<table>
<thead>
<tr>
<th>scale 0</th>
<th>scale 1</th>
<th>scale 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Q</td>
<td>Q</td>
<td>N</td>
</tr>
<tr>
<td>N</td>
<td>N</td>
<td>–</td>
</tr>
<tr>
<td>B₂</td>
<td>B₃</td>
<td>–</td>
</tr>
<tr>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>1st vertex</td>
<td>2nd vertex</td>
<td>binary represent</td>
</tr>
<tr>
<td>B₁ 16</td>
<td>45</td>
<td>10011111011</td>
</tr>
<tr>
<td>B₂ 5</td>
<td>16</td>
<td>100100000</td>
</tr>
<tr>
<td>B₃ 8</td>
<td>24</td>
<td>101101010</td>
</tr>
</tbody>
</table>
A Re-list of Beamlet Representation

\[ l_0 \rightarrow Q \]
\[ l_1 \rightarrow Q \]
\[ l_2 \rightarrow \]

\[ \begin{align*}
B_1 & : 10011111011 \\
B_2 & : 1001000000 \\
B_3 & : 1011010100
\end{align*} \]

Bits, locations of beamlets

Geometry
III. JBEAM
Overview of Beamlet Coder

- Binary shape image
- Beamlet transform
- Beamlet representation
- Zerotree coding
- Bit stream
- Entropy Coding

Reconstruction
Decoding

Zerotree algorithm (Shapiro, 93; Said & Pearlman)
- The bit-stream includes information in order of importance: Msb first and Lsb last.
Rate-Distortion-Optimized Representation

\[
\min_{\mathcal{P}} R(\mathcal{P}), \quad \text{subject to } D(y, \mathcal{P}) \leq \overline{D},
\]

- Use Lagrange multiplier to solve

\[
\min_{\mathcal{P}} R(\mathcal{P}) + \tau D(y, \mathcal{P}).
\]

- Bottom-up Tree pruning algorithm, fast
Pruning a Quad-tree

1. A complete quad-tree

2. An admissible sub-tree

3. An admissible sub-tree with beamlet decoration
Distortion

- Two components in distortion:
  - Sum of square of Euclidean distance from image to beamlet representation (~ Hausdorff distance)
  - Degree of overlapping

\[
d_S(y, b) = L_1(y \cap S, b) + \lambda \cdot L_2(y \cap S, b)
\]
Zero-Tree Coding

- Representation
- Symbol Stream
- Rule that generates the symbol stream

\[
\begin{align*}
l_0 & \rightarrow Q \\
l_1 & \rightarrow Q \quad Q \
& \quad N \quad N \
& \quad B_1 \quad 1 \quad 0 \quad 0 \quad 1 \quad 1 \quad 1 \quad 1 \quad 0 \quad 1 \quad 1 \\
l_2 & \rightarrow N \
& \quad N \
& \quad B_2 \quad 1 \quad 0 \quad 0 \quad 1 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \\
& \quad B_3 \quad 1 \quad 0 \quad 1 \quad 1 \quad 0 \quad 1 \quad 0 \quad 1 \quad 0
\end{align*}
\]

Q, Q, N, N, B, 1, 0, 0, N, N, B, B, 1, 1, 1, 0, 0, 1, 0, 1, 1, 1, 0, 1, 0, 1, 0, 0, 1, 0, 1, 1, 0, 0, 1, 0.

Q, Q, N, N, B, B_1(1), B_1(2), B_1(3), N, N, B, B, B_1(4), B_1(5), B_2(1), B_2(2), B_2(3), B_3(1), B_3(2), B_3(3), ... B_1(2J), B_1(2J + 1), B_2(2J − 2), B_2(2J − 1), B_3(2J − 2), B_3(2J − 1).
IV. Simulation Results: Comparison with JBIG

<table>
<thead>
<tr>
<th>Country</th>
<th>Pixels</th>
<th>JBIG 2 (lossless)</th>
<th>JBEAM (lossless)</th>
<th>JBEAM (lossy)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Belgium</td>
<td>1233</td>
<td>7888</td>
<td>6114</td>
<td>3098</td>
</tr>
<tr>
<td>Belize</td>
<td>822</td>
<td>5768</td>
<td>4391</td>
<td>2021</td>
</tr>
<tr>
<td>Canada</td>
<td>2911</td>
<td>15224</td>
<td>14555</td>
<td>7366</td>
</tr>
<tr>
<td>Switzerland</td>
<td>845</td>
<td>6088</td>
<td>4171</td>
<td>2135</td>
</tr>
<tr>
<td>China</td>
<td>813</td>
<td>5792</td>
<td>4609</td>
<td>2041</td>
</tr>
<tr>
<td>Cuba</td>
<td>658</td>
<td>4360</td>
<td>3303</td>
<td>1542</td>
</tr>
<tr>
<td>Germany</td>
<td>871</td>
<td>5792</td>
<td>4916</td>
<td>2138</td>
</tr>
<tr>
<td>Denmark</td>
<td>1350</td>
<td>7992</td>
<td>6780</td>
<td>3735</td>
</tr>
<tr>
<td>France</td>
<td>920</td>
<td>6448</td>
<td>5300</td>
<td>2339</td>
</tr>
</tbody>
</table>

No CAE and PWC...
Effects of Distortion
Simulation: Rate-distortion tradeoff

\[ \min D + \tau R \]

- Distortion decreases.
- Rate increases.
Simulation: Progressive Reconstruction

# of bits: 437
# of bits: 608
# of bits: 658
# of bits: 858
Rate & Distortion Analysis

- Polygonal, or model with finite number of parameters, an oracle can do:

\[ D(R) \sim C_1 2^{-C_2 R} \]

- JBEAM:

\[ D(R) \sim C_3 2^{-C_4 \sqrt{R}} \]

- Horizon model ($C^2$)

\[ D(R) \sim \log_2^2(R) \cdot R^{-2} \quad \text{versus} \quad D(R) \sim R^{-2} \quad \text{Best} \]
Discussion

- Multiscale curve representation: Strip Trees, 70’s; and many more; no beamlets, no zero-tree, no R-D optimization
- Recent works in image compression, e.g. wedgeprint, bandelets, curvelets, ridgelets, contourlets, etc
- Partial progressivity versus full progressivity?
Discussion (2)

- Shape coding versus curve coding

- Hidden Markov Model
Discussion (3)

- More innovative way to take advantage of SPIHT.

- Target
- Coarse scale beamlet
- Fine scale beamlets

Many many more…
V. Conclusion

- Combining zero-tree algorithm and beamlet.
- Numerical experiments show promises.
- First multiscale curve coder.
- Software, more research.

Beamlets: Multiscale, Curves: Geometry.
“Connect-the-dots” Problems and Dynamic Programming Solutions

Xiaoming Huo
School of Industrial and Systems Engineering
Georgia Institute of Technology

Joint work with David L. Donoho, Craig Tovey, and Ery Arias-Castro
9 / 23 / 2004
What is a Connect-The-Dots (CTD) Problem?

- Size of maximum subset of dots on a function from a given functional class.

Maximum number of dots on a Lipschitz graph
Why CTD Problem

- Test the Uniformality

(a) 22 points on a Lip. curve
(b) point set
(c) 31 points on a Lip. curve

\[ N = 256, \ C = 1 \]
Other Classes

- Convex functions
- Vectors; Connect-the-darts
Applications (1)

- Illusory contour (Psychophysics)
This talk...

- Different formulation, due to different motivations
  - CTD problems
- Efficient *dynamic programming* algorithms to solve these problems.
- Simulations / Insights / Software.
- Main objective: Fundamental understanding (asymptotic distributions).
Outline

- I. Classes of problems and *dynamic programming* approaches
- II. Complexity
- III. Theoretical Insights
- IV. Software
- V. Conclusion
I. Classes of Problems & DP Approaches

- Increasing, Unimodal, Lipschitz
- Extra variable: convex, Holder-2
- High-dimensional cases
  - Increasing
  - Lipschitz
  - Holder-2
- Connect-the-Darts
I. Increasing Functions

- **Incr. Function**
  \[ x < x' \implies f(x) \leq f(x') \]

- **M(i) =** maximum number of point on an incr. func. up to point \( i \).
  \[
  M(i) = 1 + \max_{(j < i, y_j \leq y_i)} M(j)
  \]

- **DP approach**
Fast Algorithm for Incr. Functions

- There is an $O(n \log(n))$ time and $O(n)$ space algorithm to solve CTD problem with increasing functions, Fredman 1975.
- While the data are ordinal, there is an $O(n \log \log(n))$ time and $O(n)$ space algorithm, Bespamyatnikh and Segal (2000).
- We will extend it to high dimensions.
Unimodal

- Non-decreasing up to a point, then non-increasing
- Solve it by running the algorithm for Incr. twice.
- Fast algorithms have the same complexity
Lipschitz

- Lipschitz function
  \[ |f(x) - f(x')| \leq C |x - x'| \]

- Convert it into Incr. function problem.

\[
(x'_i, y'_i) = (x_i, y_i) \begin{pmatrix}
\frac{1}{2} \sqrt{1 + C^2} & \frac{1}{2C} \sqrt{1 + C^2} \\
-\frac{1}{2C} \sqrt{1 + C^2} & \frac{1}{2} \sqrt{1 + C^2}
\end{pmatrix}
\]

\[|y_1 - y_2| \leq C |x_1 - x_2| \quad \text{and} \quad x'_1 \leq x'_2 \text{ and } y'_1 \leq y'_2.\]
Lipschitz (2)

- Illustration of the previous transform

- Solving CTD for Lipschitz functions is as hard as for increasing functions.
Convex

- Convex function

\[ f(\lambda x + (1 - \lambda)x') \leq \lambda f(x) + (1 - \lambda)f(x'). \]

- DP with additional variable
  - \( \text{Mi}(s) = \text{max. number of points up to i-th point with left slope s. (s is an additional variable)} \)
Convex (2)

- **DP algorithm:**
  \[ \forall x_k < x_i : M_i(s_{ki}) = 1 + \max_{x_j < x_i} M_j(s_{ji}) 1(s_{ji} \leq s_{ki}) , \]

  Where
  \[ s_{ji} \equiv \frac{y_i - y_j}{x_i - x_j} : x_j < x_i \]

- **Hence**
  \[ L_N(\text{Conv.}) = \max_i \max_s M_i(s) \]
Convex (3)

- There is an $O(n \log(n))$ algorithm to solve the CTD for convex functions.
Holder-2

- **Holder-2 function**

\[ G_2 = \{ f : [0, 1] \rightarrow [0, 1] : |f'(t) - f'(s)| \leq C \cdot |t - s| \} \]

- **DP approach**

\[ V_{i+1}(z, t) = 1_{z=y_{i+1}} + \max_{(x_j, y, s) \in R_{i+1}(z, t)} V_j(y, s) \]
Holder-2 (2)

- Essence of the derivation
  - Solving a variational problem
  - Computing influential intervals
- Complexity: unclear, but empirically, it seems polynomial.

<table>
<thead>
<tr>
<th>Sample size (N)</th>
<th>Run time (sec.)</th>
<th>Ave. no. of breakpoints</th>
<th>Max. no. of breakpoints</th>
</tr>
</thead>
<tbody>
<tr>
<td>16</td>
<td>0.125</td>
<td>16.2</td>
<td>31</td>
</tr>
<tr>
<td>32</td>
<td>0.734</td>
<td>20.8</td>
<td>36</td>
</tr>
<tr>
<td>64</td>
<td>4.437</td>
<td>32.2</td>
<td>53</td>
</tr>
<tr>
<td>128</td>
<td>28.28</td>
<td>53.6</td>
<td>99</td>
</tr>
<tr>
<td>256</td>
<td>175.2</td>
<td>80.3</td>
<td>168</td>
</tr>
</tbody>
</table>
High-dimensional

- Longest increasing subsequence in k-dimension.
- We designed an $O(N \log^K N)$ algorithm.
- Example when $k=3$, $N=8$. 
Illustrate k-D increasing subsequences
High-Dimensional Lipschitz and Holder

- Modified DP approach works for high dimensional Lipschitz functions
- Modified DP approach works for high dimensional Holder-2 functions
Connect-the-Darts

- Test of Randomness

- Similar approach as in Holder-2
- Knowing the angle `saves’ many computing, connection in psychophysics.
## II. Complexity

<table>
<thead>
<tr>
<th>Case Description</th>
<th>Complexity</th>
</tr>
</thead>
<tbody>
<tr>
<td>Increasing</td>
<td>$O(N \log(N))$</td>
</tr>
<tr>
<td>Unimodal</td>
<td>$O(N \log(N))$</td>
</tr>
<tr>
<td>Lipschitz 1-graph in $\mathcal{R}^2$</td>
<td>$O(N \log(N))$</td>
</tr>
<tr>
<td>Convex</td>
<td>$O(N^2 \log(N))$</td>
</tr>
<tr>
<td>Increasing in $\mathcal{R}^d$</td>
<td>$O(N(\log(N))^{d-1})$</td>
</tr>
<tr>
<td>Lipschitz 1-graph in $\mathcal{R}^d$, $d &gt; 2$</td>
<td>$O(N^2d)$</td>
</tr>
<tr>
<td>Vector Field under (C1)</td>
<td>$O(N^2)$</td>
</tr>
<tr>
<td>Finite Length Curve</td>
<td>NP-hard</td>
</tr>
</tbody>
</table>
III. Insights

- Limit distributions
- Asymptotic rates
- Concentration of measures
- Typical cases
III-1. Limit Distributions

- Consistent with known theory in geometric probability.
Limit Distributions (2)

- New cases
  - Lipschitz
  - Holder-2
  - High dimension
  - Connect the darts

(b) $L_N(\text{Lip}_C)$

(c) $L_N(\mathcal{G}_2)$
III-2. Asymp. Rates

- For increasing functions, we have

\[
\lim_{N \to \infty} \frac{L_N(\mathcal{F})}{2\sqrt{N}} = 1, \quad \text{w.p. 1}
\]

- In general, we have

\[
L_N(\mathcal{F}) \asymp \text{Constant} \cdot N^\rho
\]

<table>
<thead>
<tr>
<th>Case Description</th>
<th>Slope</th>
<th>Predicted Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>increasing</td>
<td>0.5182</td>
<td>1/2</td>
</tr>
<tr>
<td>unimodal</td>
<td>0.5282</td>
<td>1/2</td>
</tr>
<tr>
<td>Lipschitz in $\mathcal{R}^2$</td>
<td>0.5065</td>
<td>1/2</td>
</tr>
<tr>
<td>convex</td>
<td>0.4387</td>
<td>1/3</td>
</tr>
<tr>
<td>Hölder-2</td>
<td>0.3087</td>
<td>1/3</td>
</tr>
</tbody>
</table>
III-3. Concentration of Measure

- What is concentration of measure?
  \[ \log(\text{IQR}) < 0.5 \log(\text{median}). \]
- The square root seems to be true in all cases.

<table>
<thead>
<tr>
<th>Case Description</th>
<th>increasing</th>
<th>unimodal</th>
<th>Lipschitz in $\mathbb{R}^2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Slope</td>
<td>0.6378</td>
<td>0.4733</td>
<td>0.6582</td>
</tr>
</tbody>
</table>
III-4. Typical Cases

- Increasing, consistent with current literature.
Typical Cases (2)

- Unimodal
- Lipschitz
- Convex
IV. Software

- CTDLab.
- Implementation of all algorithms
- Demos for the figures
- Documentation
V. Conclusion

- Introducing CTD problems.
- Dynamic programming approach for a set of CTD problems - New algorithms, with low computational complexity.
- Theoretical insights.
- Applications: hypothesis testing, data analysis, fundamental detection theorems, …
Multi-scale Significance Run Algorithm

Xiaoming Huo
School of Industrial and Systems Engineering
Georgia Institute of Technology

Joint work with David L. Donoho, Ery Arias-Castro
9 / 23 / 2004
I. Introduction — The Problem

- What's the problem?
  ---- Detecting the presence of a filament (or a filament-like feature) in a noisy picture.
II. Multiscale Analysis and Significance Run

- Multiscale components:
  - Axoids
  - Beamlets
- Significance run and detection
- Fundamental theory in detecting filamentary structures
- Most powerful test!
- Scan statistics
Multiscale Analysis

- Axoids
- Beamlets

Figure 1: axoids.
Significance Run

- Significance run and Bernoulli Table (significance graph)
Significance Run

Longest significance run

(a) White Noises

(b) Longest Run in White Noises

(c) with an Underlying Feature

(d) Longest Run for (c)
Statistical Theorems

- **Fundamental theorem in detecting filamentary structures in point clouds**

  **Theorem 3.1** There is a single choice of thresholds $N^*$ and $(L_n^*)_n$ so that for every $\alpha \in (1, 2]$ and $\beta > 0$, there is $T_*(\alpha, \beta, \rho, S)$ so that, for each $\tau > T_*$

  $\Pr\{\text{Test rejects } H_0 \mid H_1(\alpha, \beta, S, \tau) \} \to 1$ as $n \to \infty$;

  at the same time

  $\Pr\{\text{Test rejects } H_0 \mid H_0 \} \to 0$ as $n \to \infty$.

  In words, under $H_0$ the longest significant path is overwhelmingly unlikely to be substantially longer than $L_n^*$ for large $n$, while under each indicated $H_1$, the longest significant path is overwhelmingly likely to be substantially longer than $L_n^*$. The threshold $T_*$ is within a constant factor $T_*/T_\infty$ of the optimal detection threshold; this shows that, up to constants, we can adaptively test for existence of fragments of $C^\alpha$ graphs.

- **Most powerful test in significance runs**

  If there is a test with power $\sim 1$, then MSRA will have that power.
III. Simulations — Some Figures

Noisy images with an underlying trigonometric curve having a range of amplitudes (provided in the titles). It shows that the proposed method can detect an underlying feature which is not obviously visible.

Table 4: Detections out of 10 simulations for a trigonometric curve while $n = 64$ and $t = 1$.

<table>
<thead>
<tr>
<th>$A_n$</th>
<th>0</th>
<th>2.000</th>
<th>2.333</th>
<th>2.666</th>
<th>3.000</th>
<th>4.000</th>
</tr>
</thead>
<tbody>
<tr>
<td>Detected Cases</td>
<td>0</td>
<td>0</td>
<td>4</td>
<td>7</td>
<td>9</td>
<td>10</td>
</tr>
</tbody>
</table>

9/23/04 IPAM Sep. 04
Simulations

- Increased sample size

The random images that are used in the experiments that are reported in Table 5. The thickness of the underlying feature is \( t = 8 \).

Table 5: Detections out of 10 simulations for a trigonometric curve while \( n = 128 \) and \( t = 8 \).

<table>
<thead>
<tr>
<th>( A_n )</th>
<th>1/4</th>
<th>1/2</th>
<th>3/4</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Detected Cases</td>
<td>0</td>
<td>10</td>
<td>10</td>
<td>10</td>
</tr>
</tbody>
</table>
Simulations — more

- Thinner objects

The random images that are used in the experiments that are reported in Table 6. The thickness of the underlying feature is $t = 1$.

Table 6: Detections out of 10 simulations for a trigonometric curve while $n = 128$ and $t = 1$.

<table>
<thead>
<tr>
<th>$A_n$</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Detected Cases</td>
<td>0</td>
<td>3</td>
<td>10</td>
<td>10</td>
</tr>
</tbody>
</table>

9/23/04

IPAM Sep. 04
Simulations

Illustration of an increment of length of the longest run when an underlying feature (a trigonometric function for (a) and (b), and a beam for (c) and (d)) is present. When there is an underlying feature, the length of the longest run is significantly larger.
IV. Conclusion

- Multiscale detection
- Automatic adaptation
- Fast algorithms
- Length of the longest significance run; limit distribution

**Multiscale + Geometry**