Wavelets for Surface Reconstruction

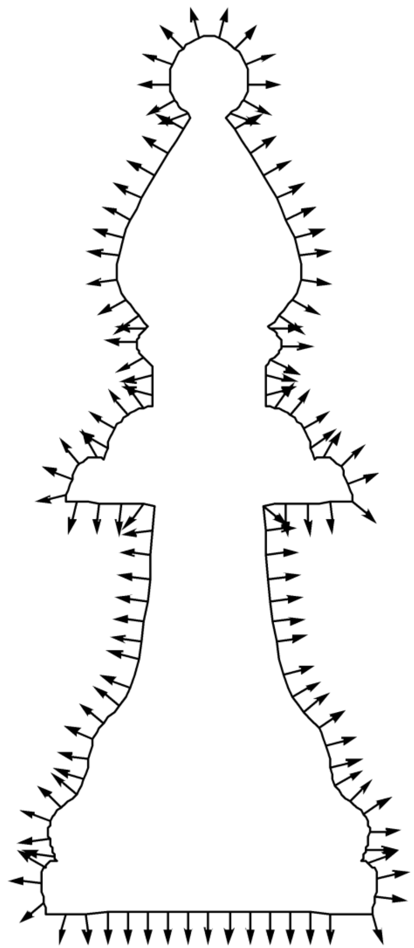
Josiah Manson

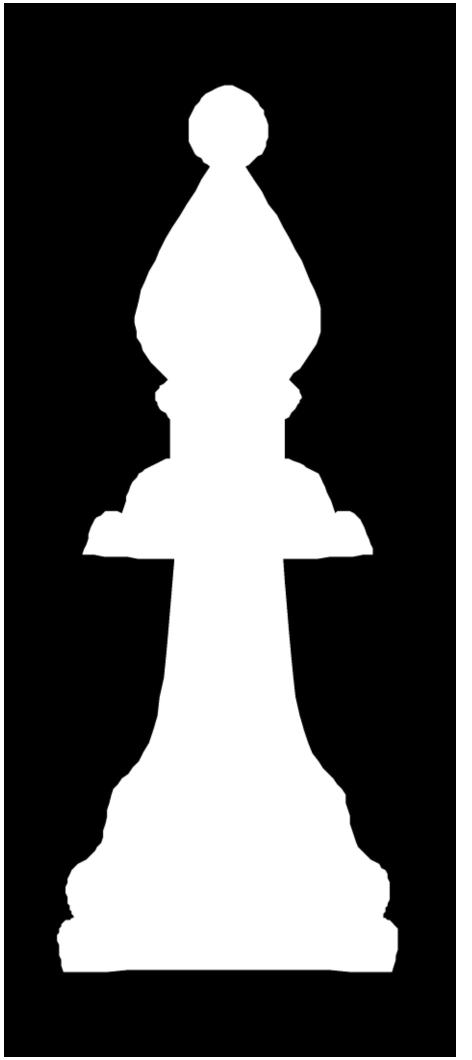
Guergana Petrova

Scott Schaefer

TAMU Computer Science

Convert Points to an Indicator Function





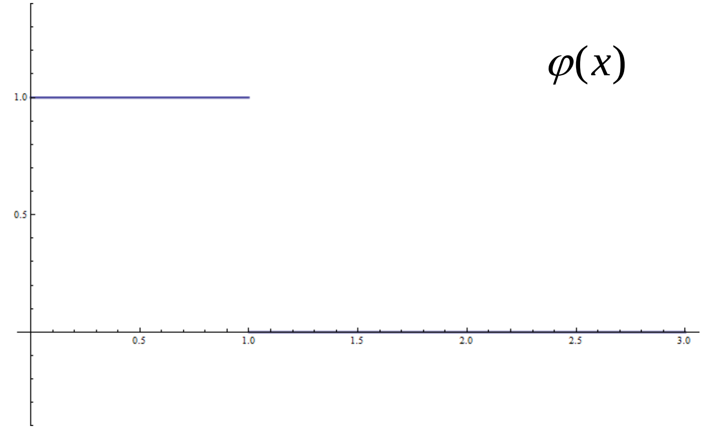
Data Acquisition

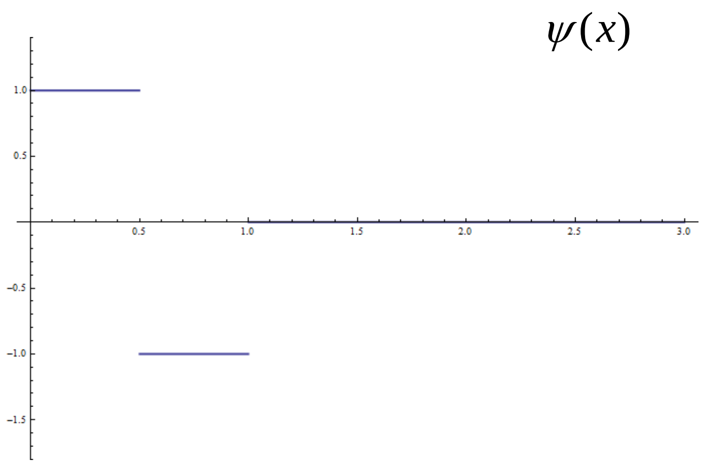
Properties of Wavelets

	Fourier Series	Wavelets
Represents all functions	✓	✓
Locality	✗	✓
Smoothness	✓	Depends on wavelet

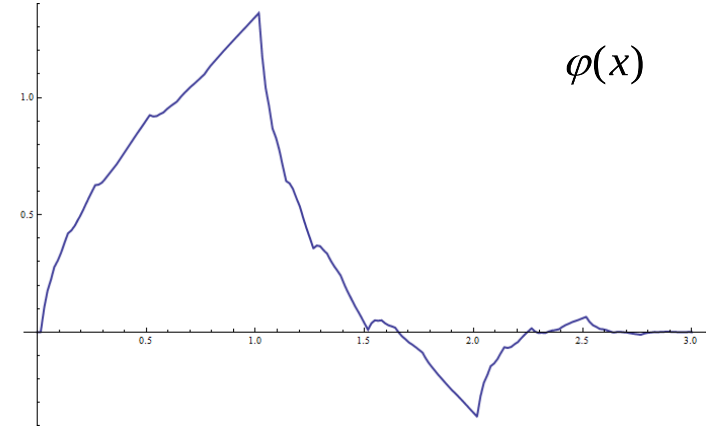
Wavelet Bases

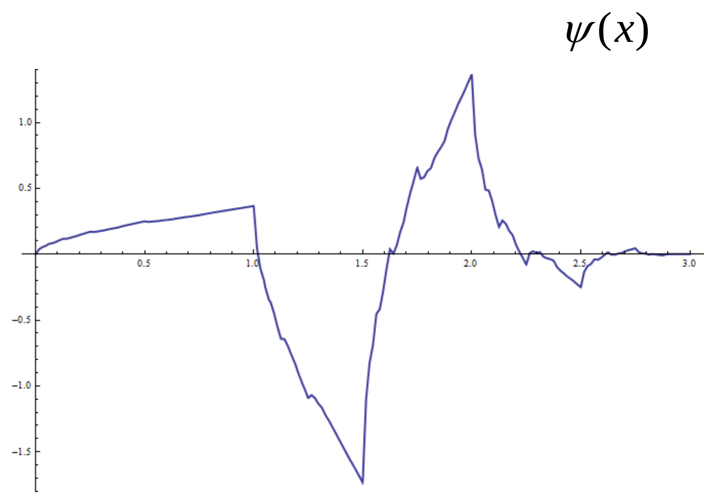
Haar



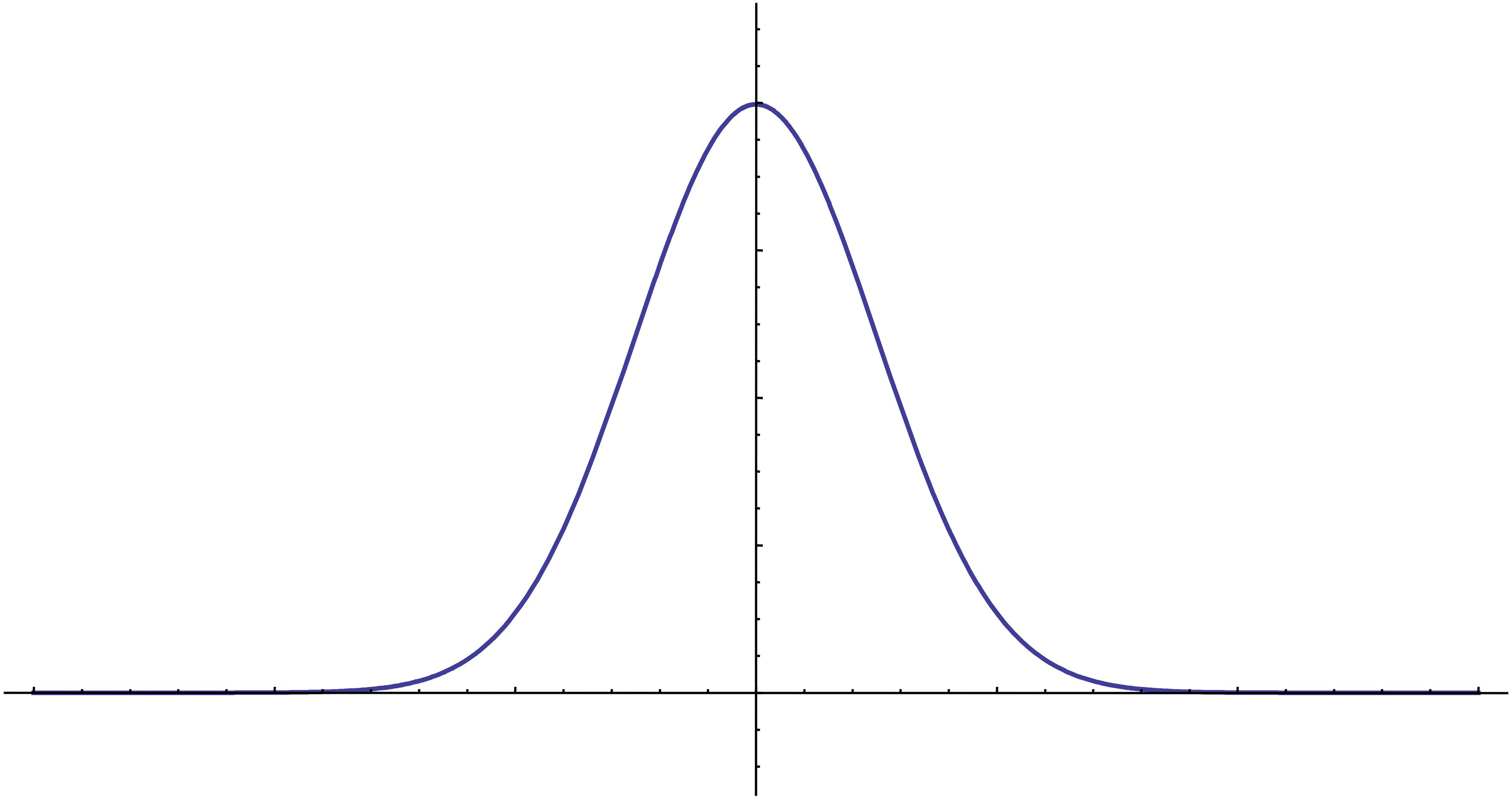


D4



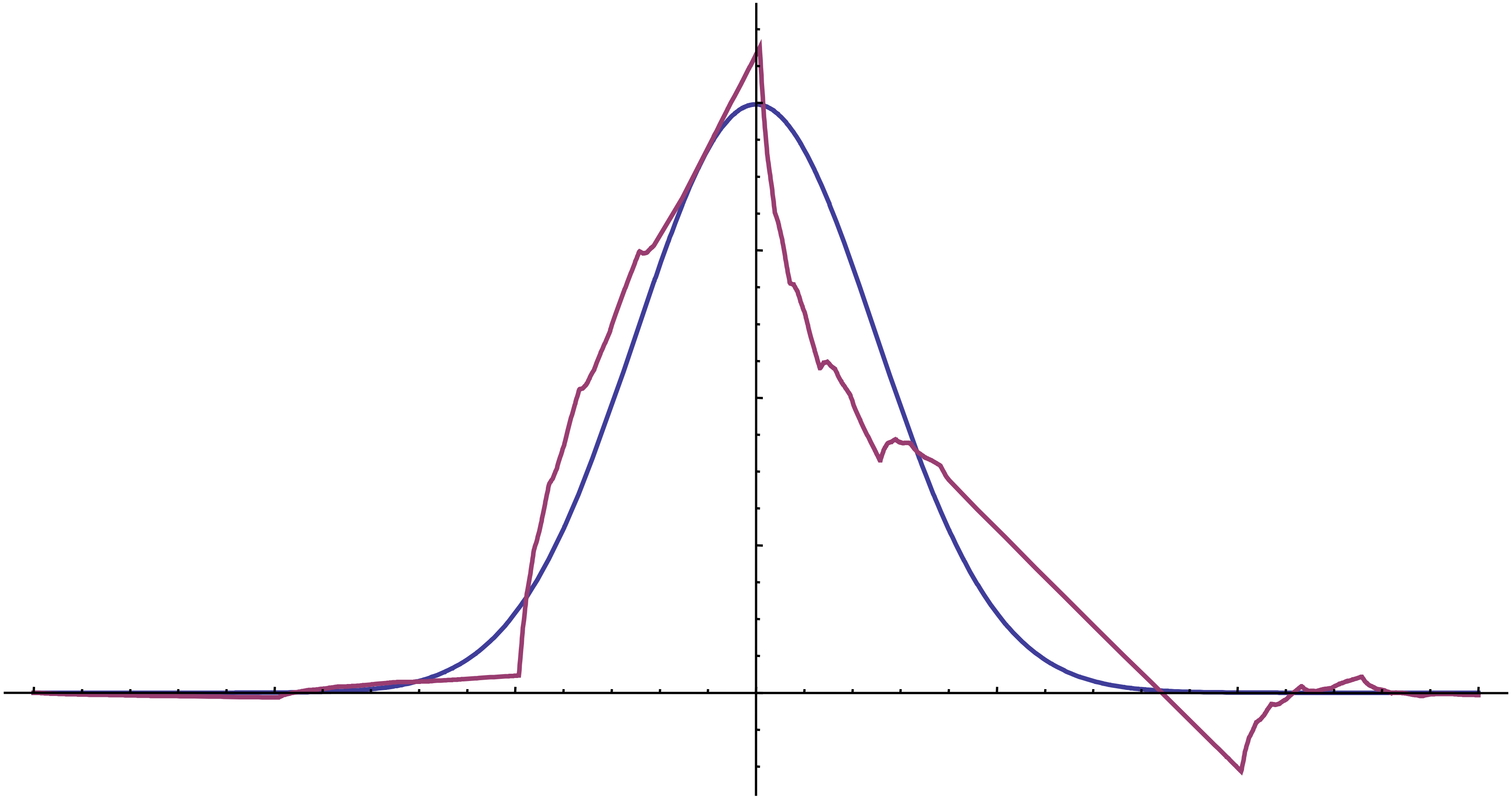


Example of Function using Wavelets



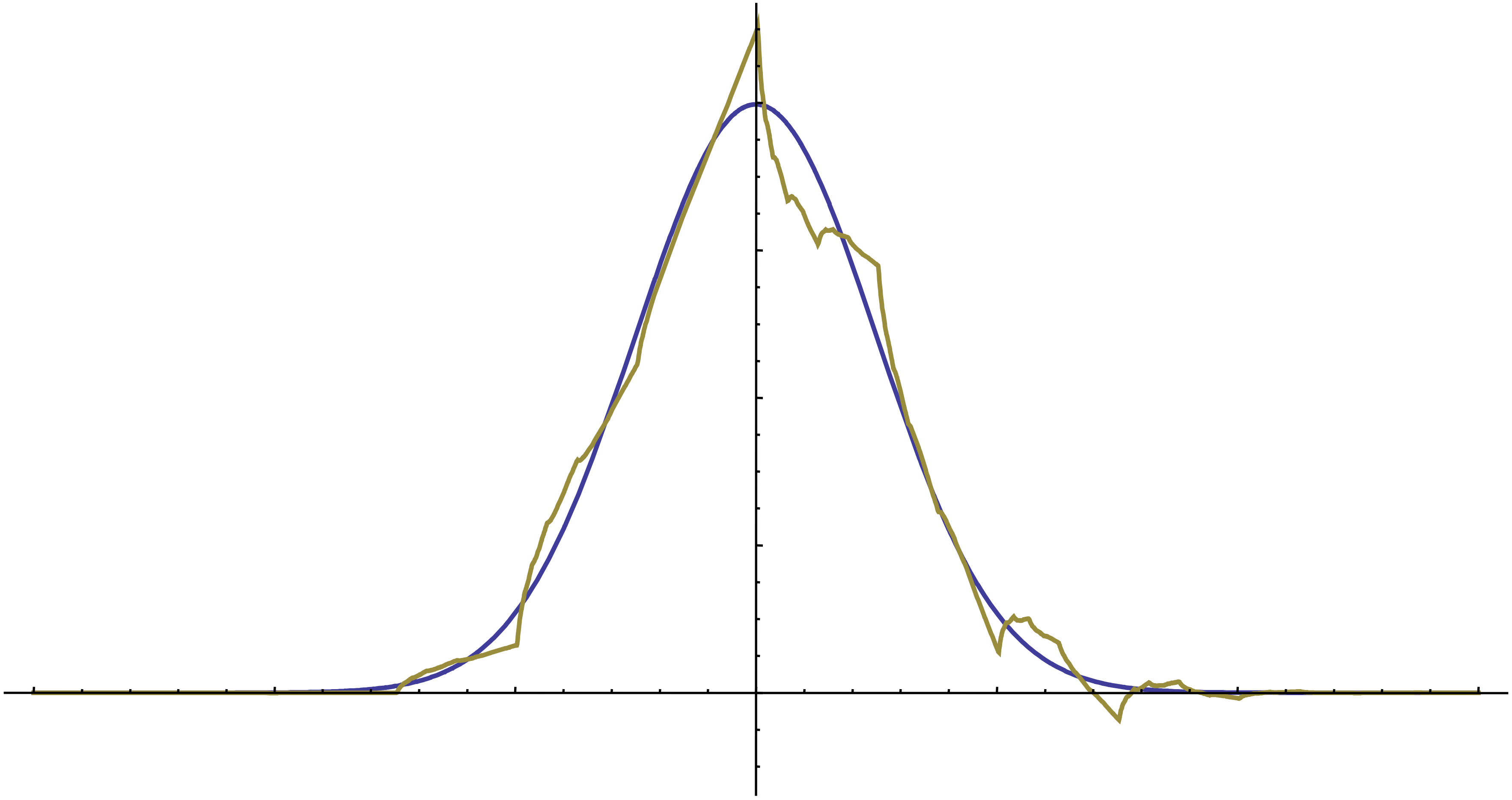
$$f(x) = \sum_k c_k \varphi(x-k) + \sum_{j,k} c_{j,k} \psi(2^j x - k)$$

Example of Function using Wavelets



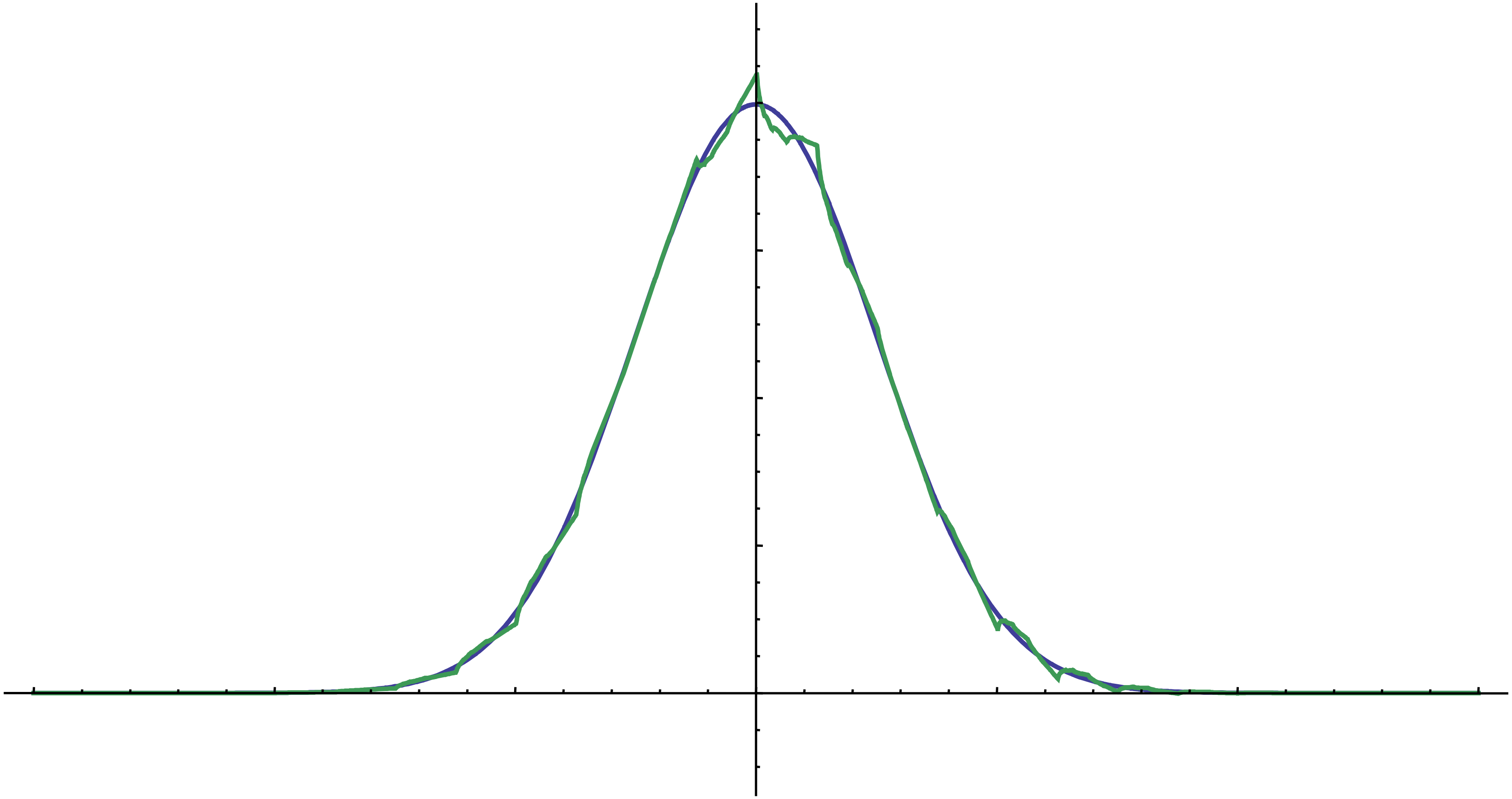
$$f(x) = \sum_{k} c_k \varphi(x-k)$$

Example of Function using Wavelets



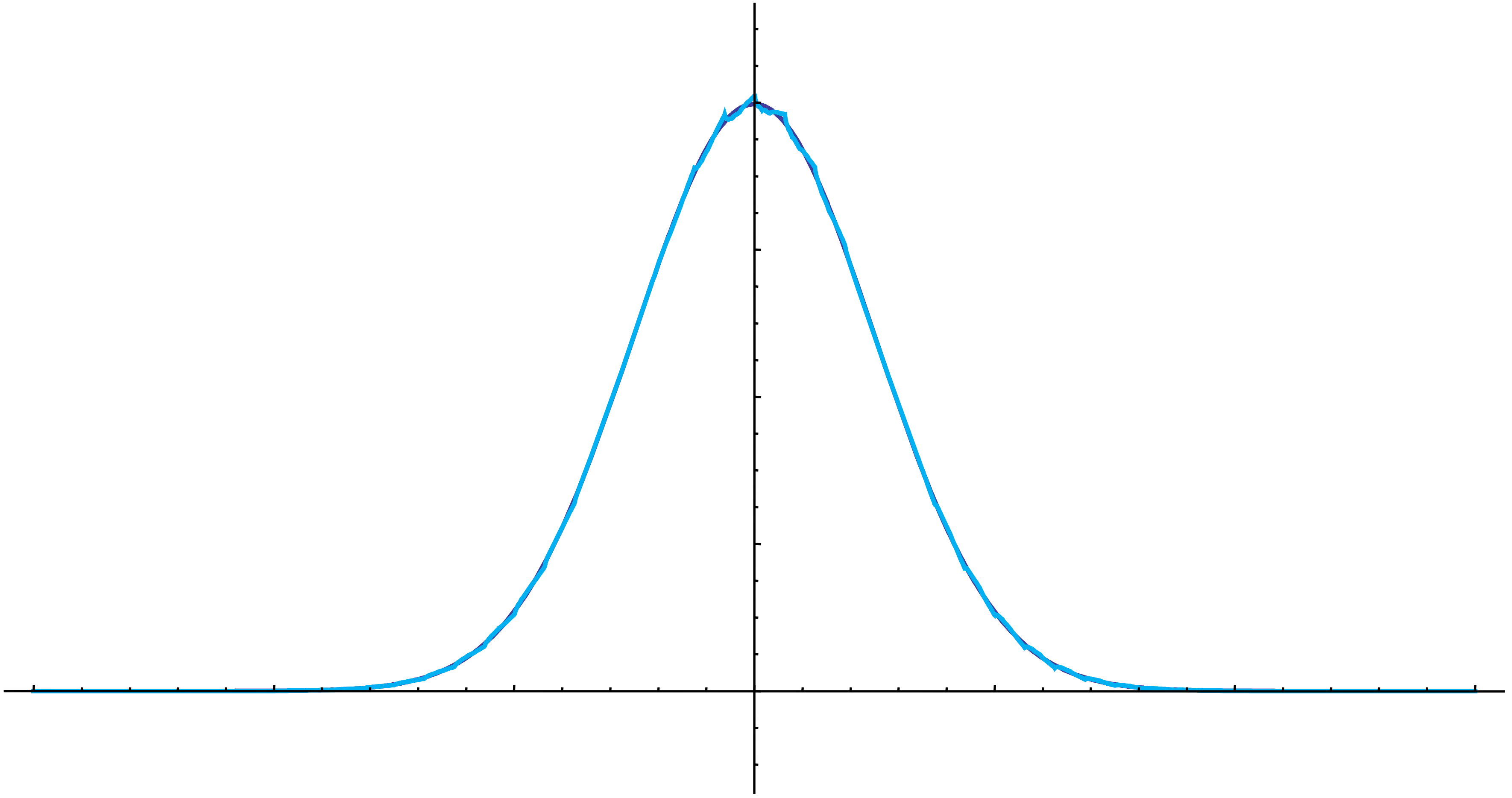
$$f(x) = \sum_k c_k \varphi(x-k) + \sum_{j=0,k} c_{j,k} \psi(2^j x - k)$$

Example of Function using Wavelets



$$f(x) = \sum_{k} c_k \varphi(x-k) + \sum_{j\in\{0,1\},k} c_{j,k} \psi(2^j x - k)$$

Example of Function using Wavelets



$$f(x) = \sum_k c_k \varphi(x-k) + \sum_{j\in\{0,1,2\},k} c_{j,k}\psi(2^j x - k)$$

Strategy

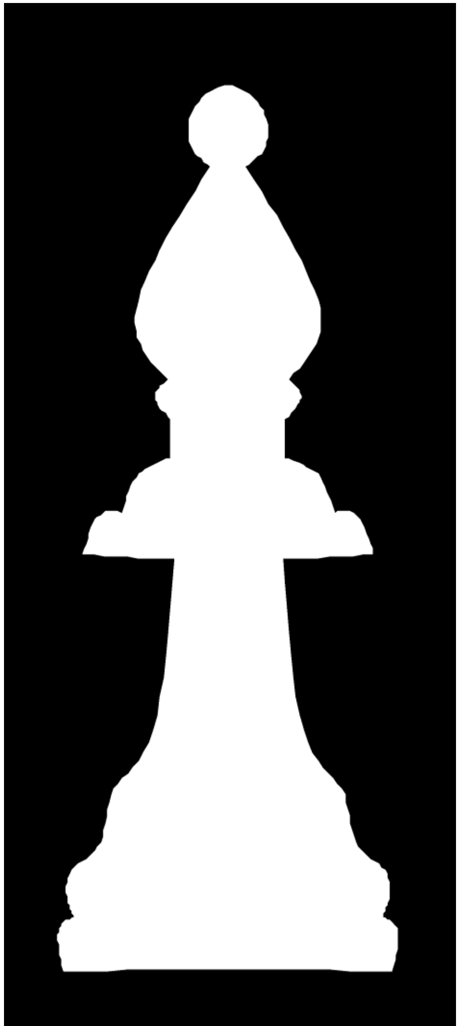
- Estimate wavelet coefficients of indicator function
- Use only local combination of samples to find coefficients

Computing the Indicator Function

[Kazhdan 2005]

$$\chi(x) = \sum_k c_k \varphi(x-k) + \sum_{j,k} c_{j,k} \psi(2^j x - k)$$

$\chi(x)$



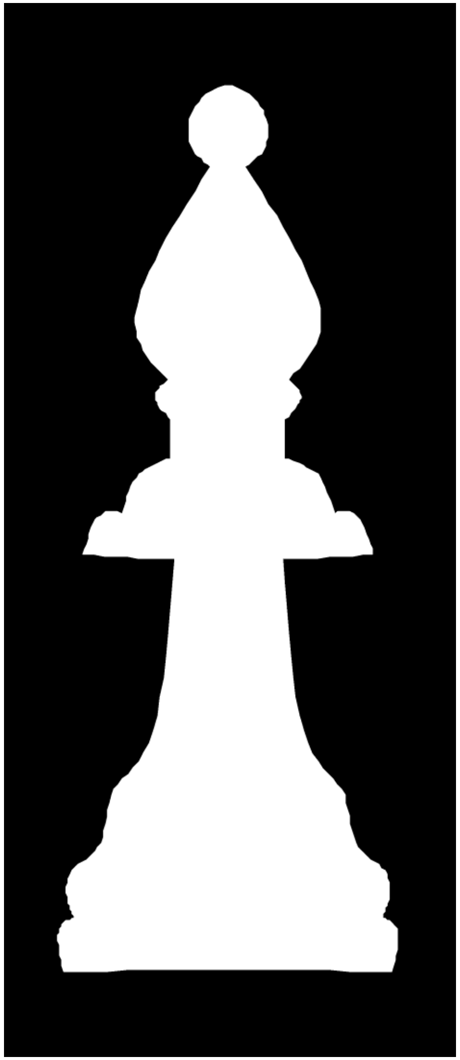
Computing the Indicator Function

[Kazhdan 2005]

$$\chi(x) = \sum_k c_k \varphi(x-k) + \sum_{j,k} c_{j,k} \psi(2^j x - k)$$

$$c_{j,k} = \int_R \chi(x) \psi(2^j x - k) dx$$

$\chi(x)$



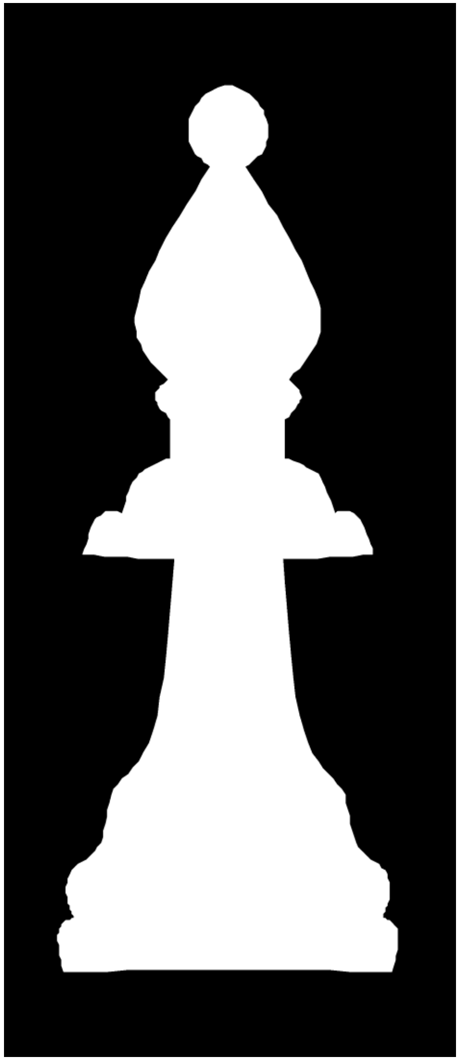
Computing the Indicator Function

[Kazhdan 2005]

$$\chi(x) = \sum_k c_k \varphi(x-k) + \sum_{j,k} c_{j,k} \psi(2^j x - k)$$

$$c_{j,k} = \int_R \chi(x) \psi(2^j x - k) dx = \int_M \psi(2^j x - k) dx$$

$\chi(x)$



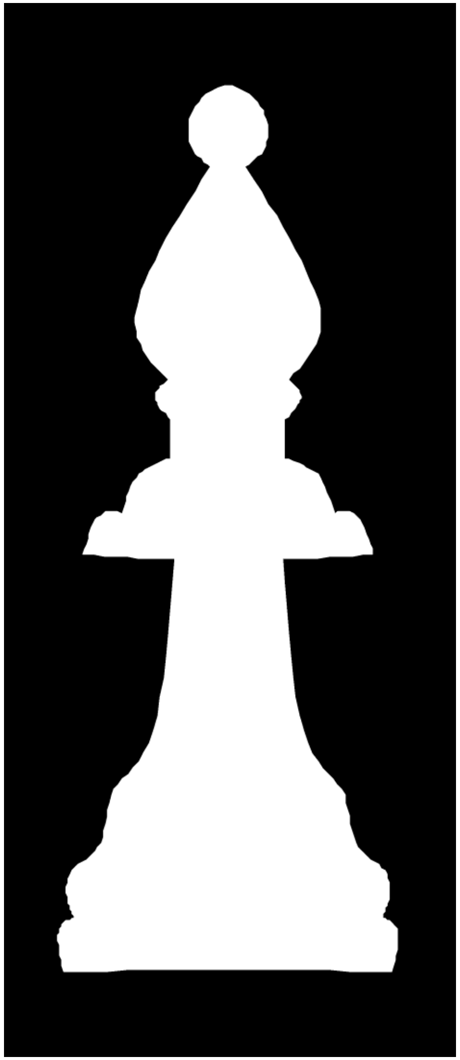
Computing the Indicator Function

[Kazhdan 2005]

$$\chi(x) = \sum_k c_k \varphi(x-k) + \sum_{j,k} c_{j,k} \psi(2^j x - k)$$

$$c_{j,k} = \int_R \chi(x)\psi(2^j x - k)dx = \int_M \psi(2^j x - k)dx$$

$\chi(x)$



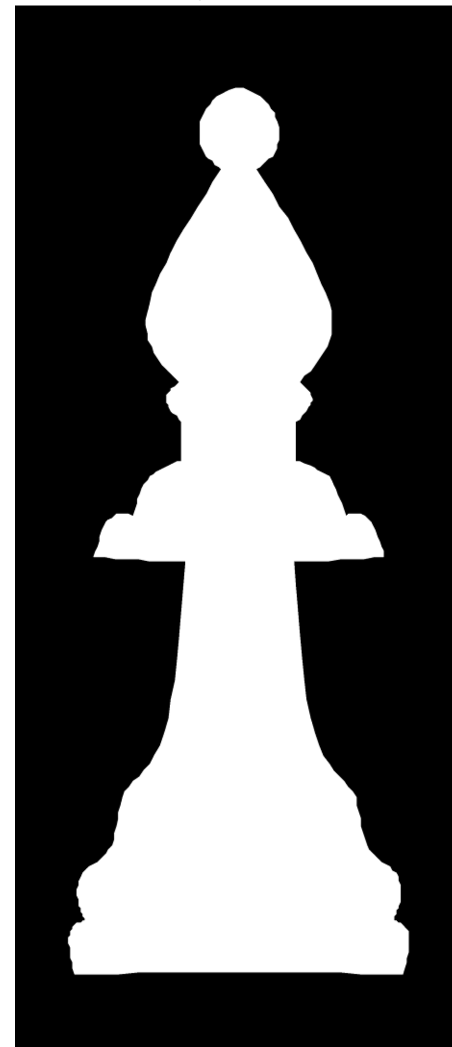
Divergence Theorem

$$\int_M \nabla \cdot \vec{F}(x)dx = \int_{p \in \partial M} \vec{F}(p) \cdot \vec{n}(p) d\sigma$$

Computing the Indicator Function

[Kazhdan 2005]

$\chi(x)$



$$\chi(x) = \sum_k c_k \varphi(x-k) + \sum_{j,k} c_{j,k} \psi(2^j x - k)$$

$$c_{j,k} = \int_R \chi(x)\psi(2^j x - k)dx = \int_M \boxed{\psi(2^j x - k)} dx$$

Divergence Theorem

$$\int_M \boxed{\nabla \cdot \vec{F}(x)} dx = \int_{p\in\partial M} \vec{F}(p)\cdot\vec{n}(p)d\sigma$$

Computing the Indicator Function

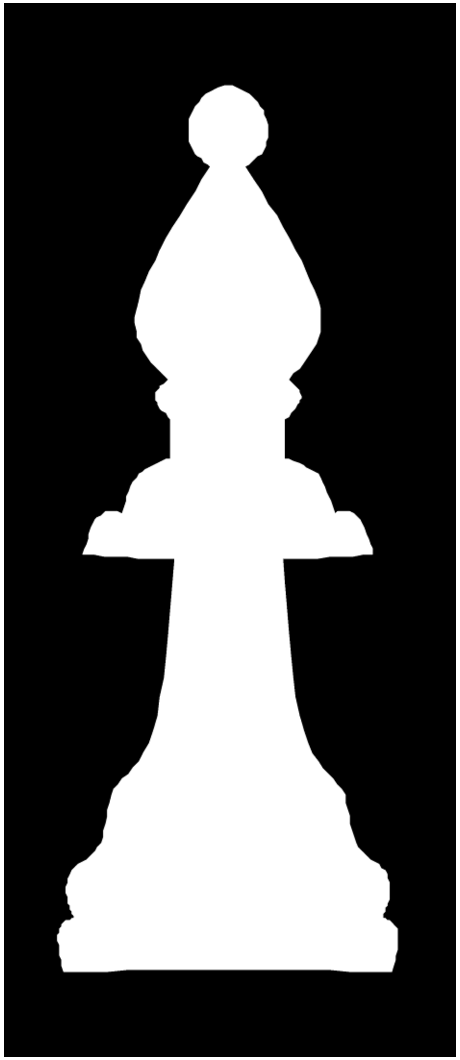
[Kazhdan 2005]

$$\chi(x)=\sum_{k} c_k \varphi(x-k)+\sum_{j,k} c_{j,k}\psi(2^j x-k)$$

$$c_{j,k}=\int_R \chi(x)\psi(2^j x-k)dx=\int_M \psi(2^j x-k)dx$$

$$=\int_{p\in\partial M} \vec{F}_{j,k}(p)\cdot\vec{n}(p)d\sigma$$

$\chi(x)$



Computing the Indicator Function

[Kazhdan 2005]

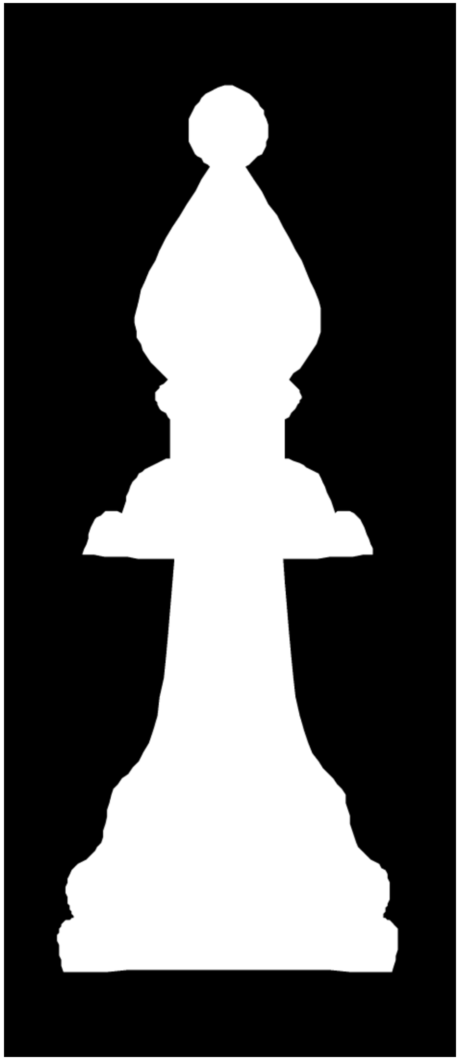
$$\chi(x) = \sum_k c_k \varphi(x-k) + \sum_{j,k} c_{j,k} \psi(2^j x - k)$$

$$c_{j,k} = \int_R \chi(x)\psi(2^j x - k)dx = \int_M \psi(2^j x - k)dx$$

$$= \int_{p \in \partial M} \vec{F}_{j,k}(p) \cdot \vec{n}(p) d\sigma$$

$$\approx \sum_i \vec{F}_{j,k}(p_i) \cdot \vec{n}_i d\sigma_i$$

$\chi(x)$



Finding $\vec{F}(x)$

$$\nabla \cdot \vec{F}(x) = \psi(2^j x - k)$$

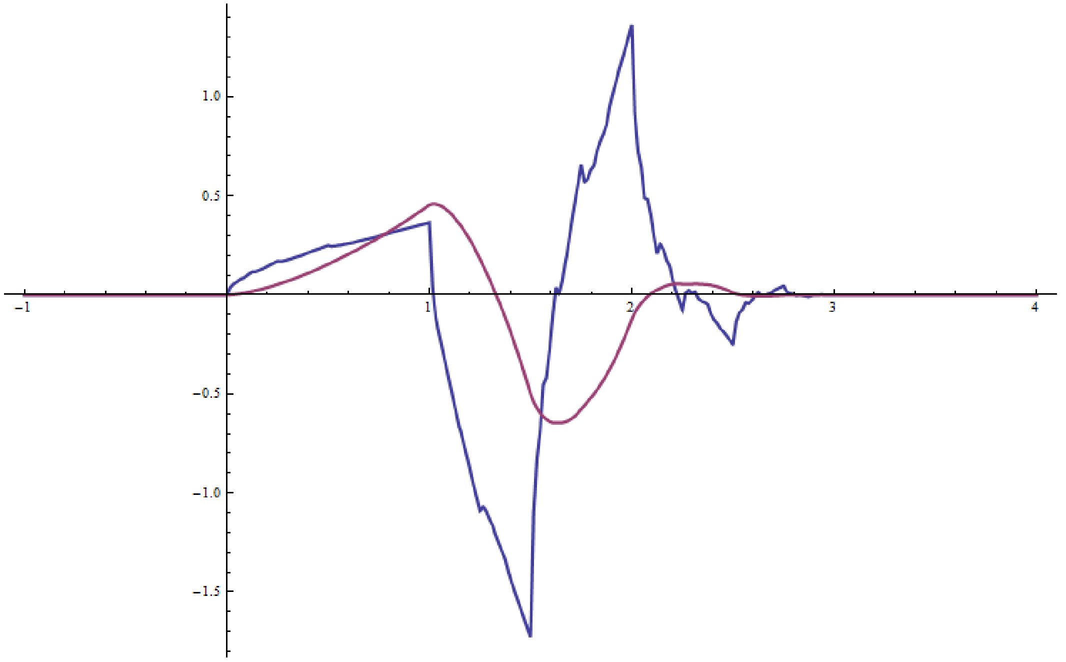
Finding $\vec{F}(x)$

$$\nabla \cdot F(x) = \frac{d}{dx} F(x) = \psi(2^j x - k)$$

Finding $\vec{F}(x)$

$$\nabla \cdot F(x) = \frac{d}{dx} F(x) = \psi(2^j x - k)$$

$$F(x) = \int_{-\infty}^{x} \psi(2^j s - k)\, ds$$



Extracting the surface

Coefficients

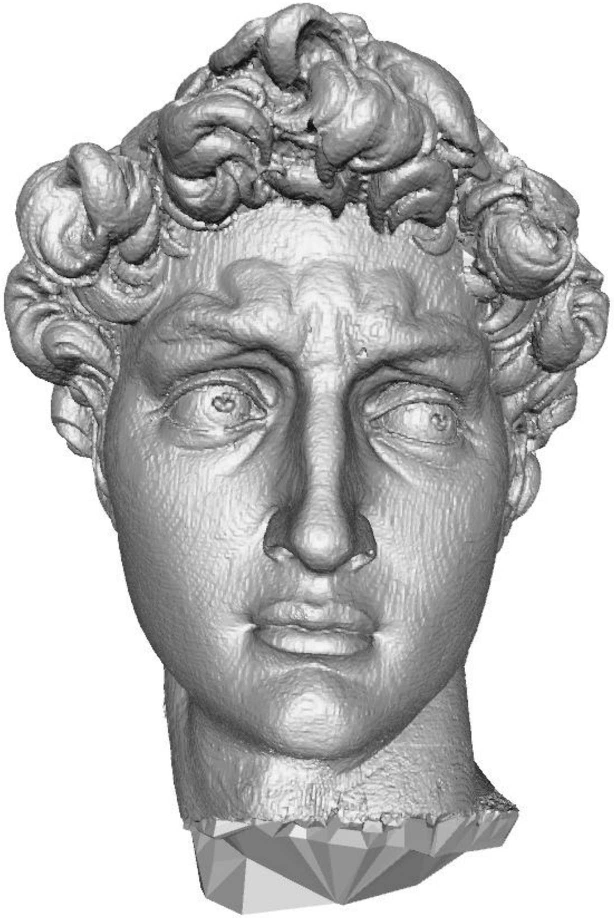
$$\chi(x) = \sum_k c_k \varphi(x-k) + \sum_{j,k} c_{j,k} \psi(2^j x - k)$$

Indicator function

Dual marching cubes

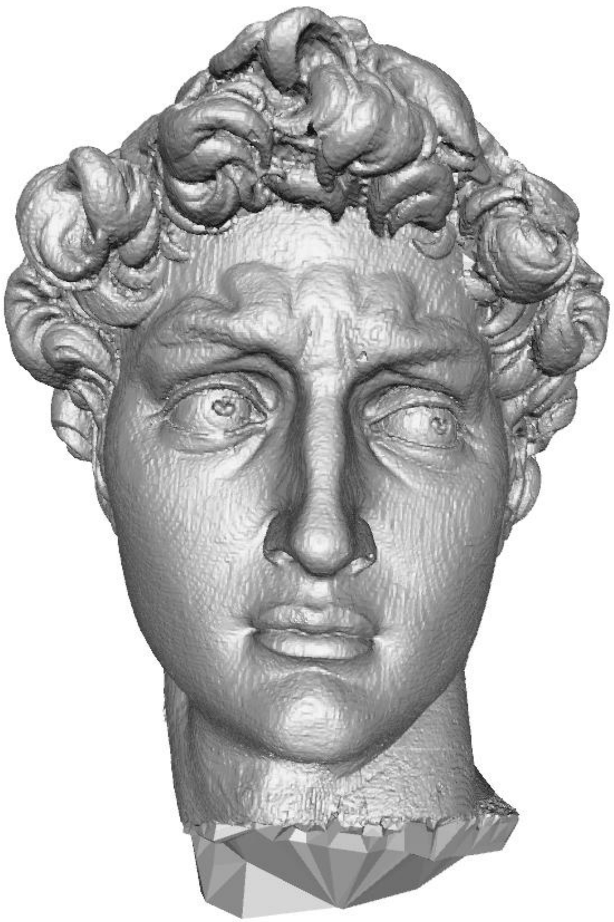
Surface

Smoothing the Indicator Function



Haar unsmoothed

Smoothing the Indicator Function



Haar unsmoothed

Haar smoothed

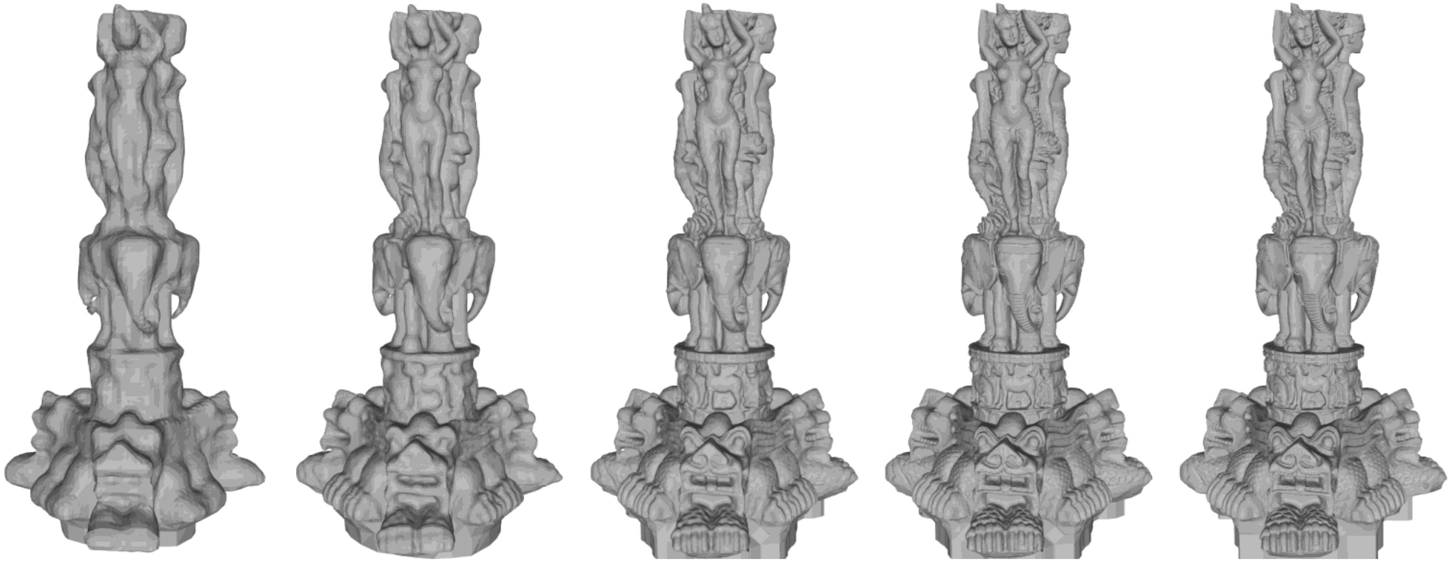
Comparison of Wavelet Bases

Haar

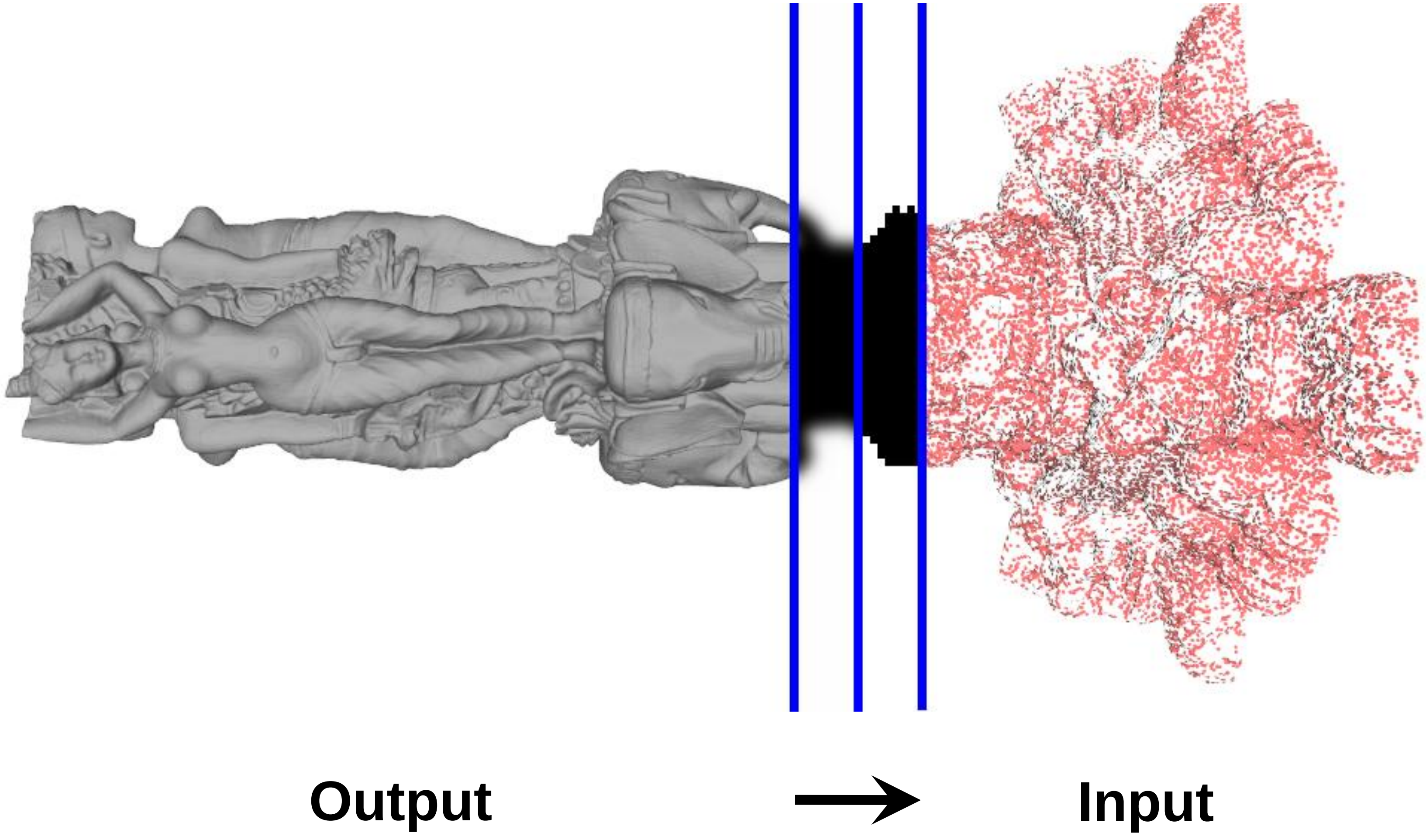
D4

Advantages of Wavelets

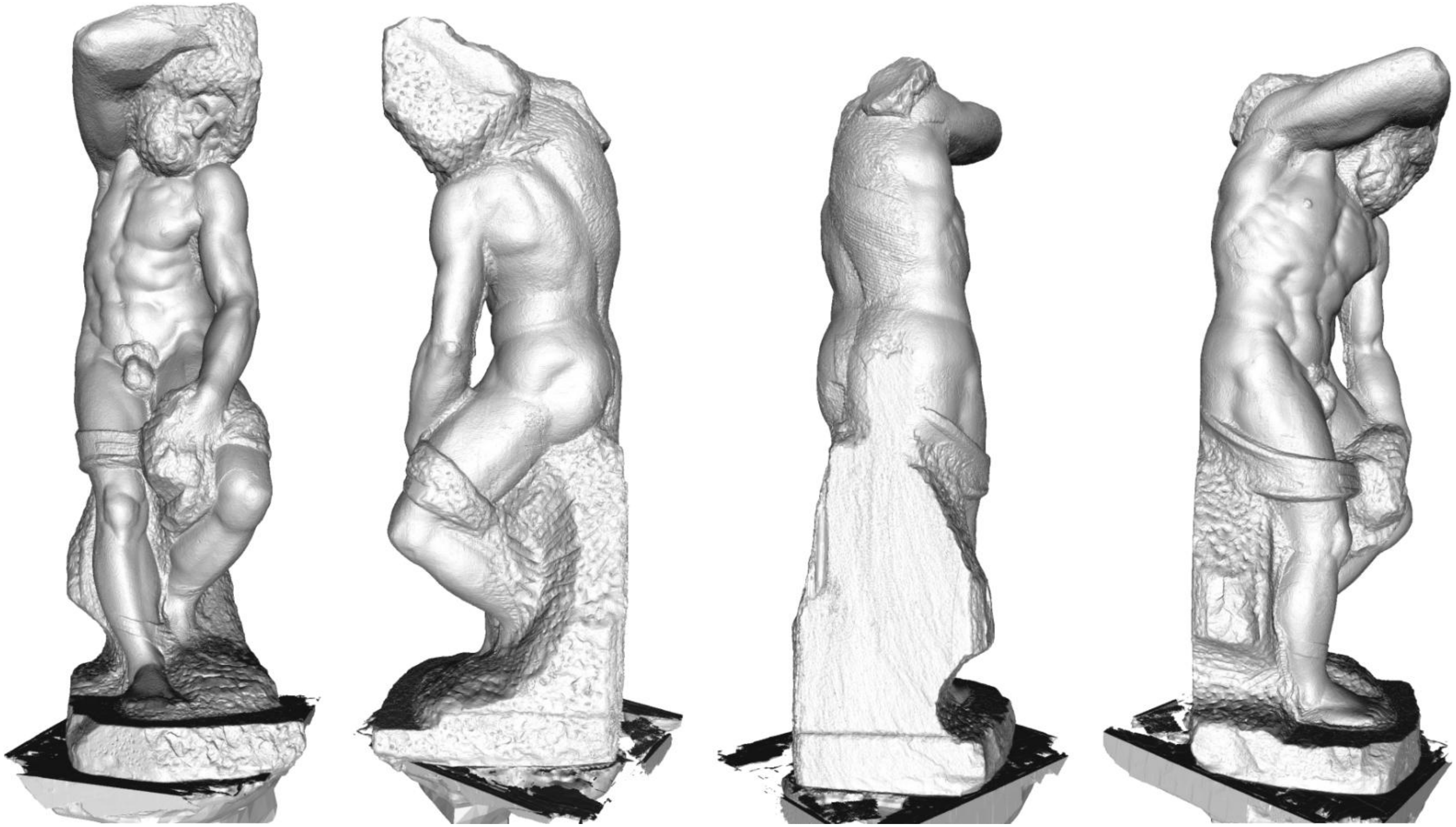
- Coefficients calculated only near surface
  - Fast
  - Low memory
- Multi-resolution representation
- Out of core calculation is possible



Streaming Pipeline



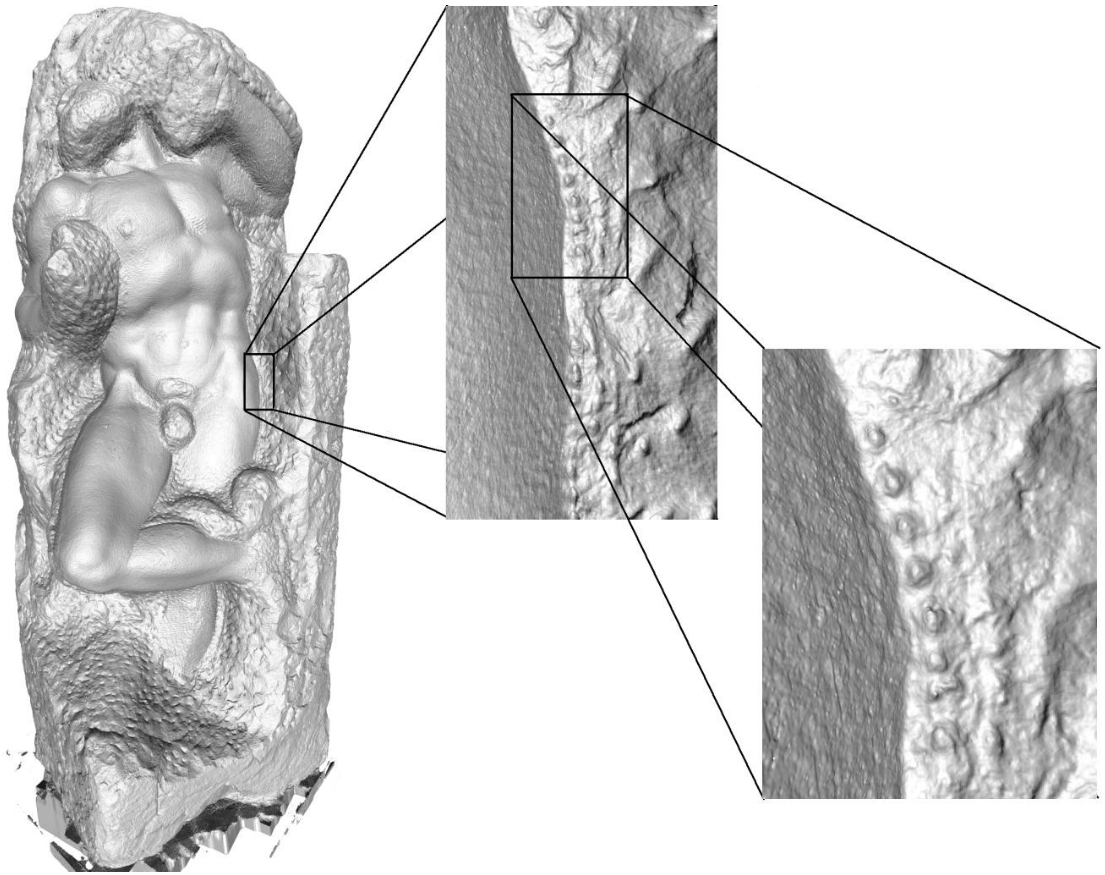
Results



Michelangelo's Barbuto

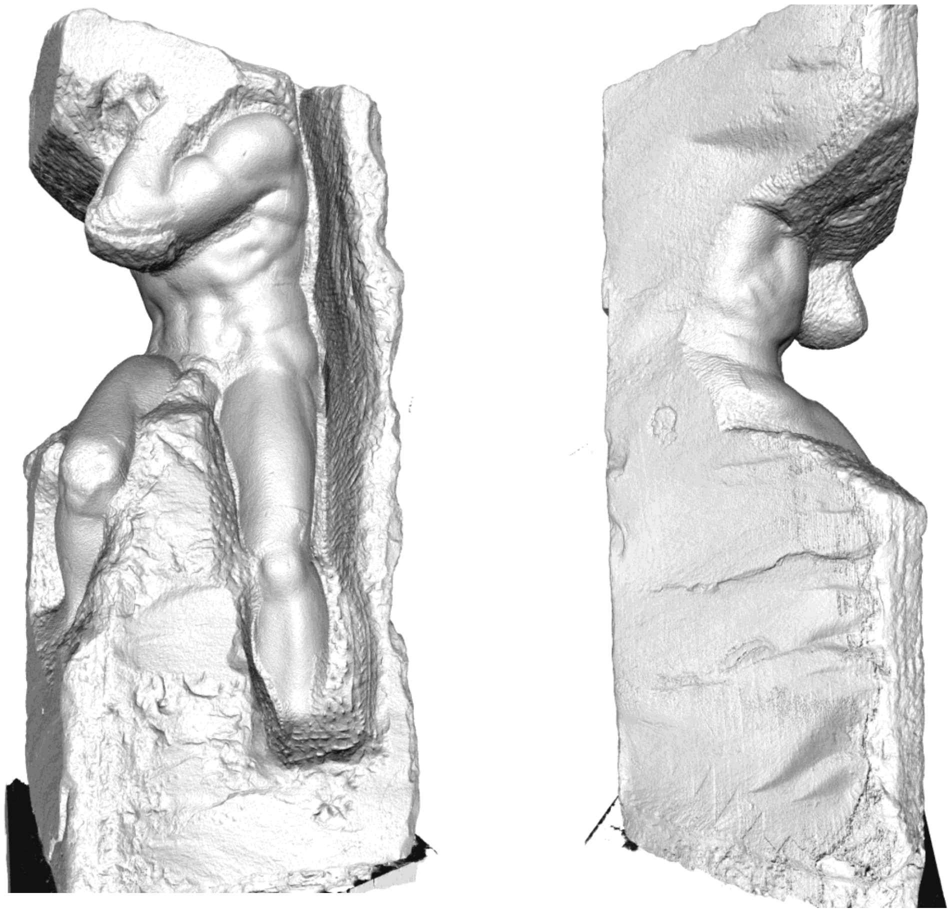
329 million points (7.4 GB of data), 329MB memory, 112 minutes

Results



Michelangelo's Awakening
381 million points (8.5 GB), 573MB memory, 81 minutes
Produced 590 million polygons

Results

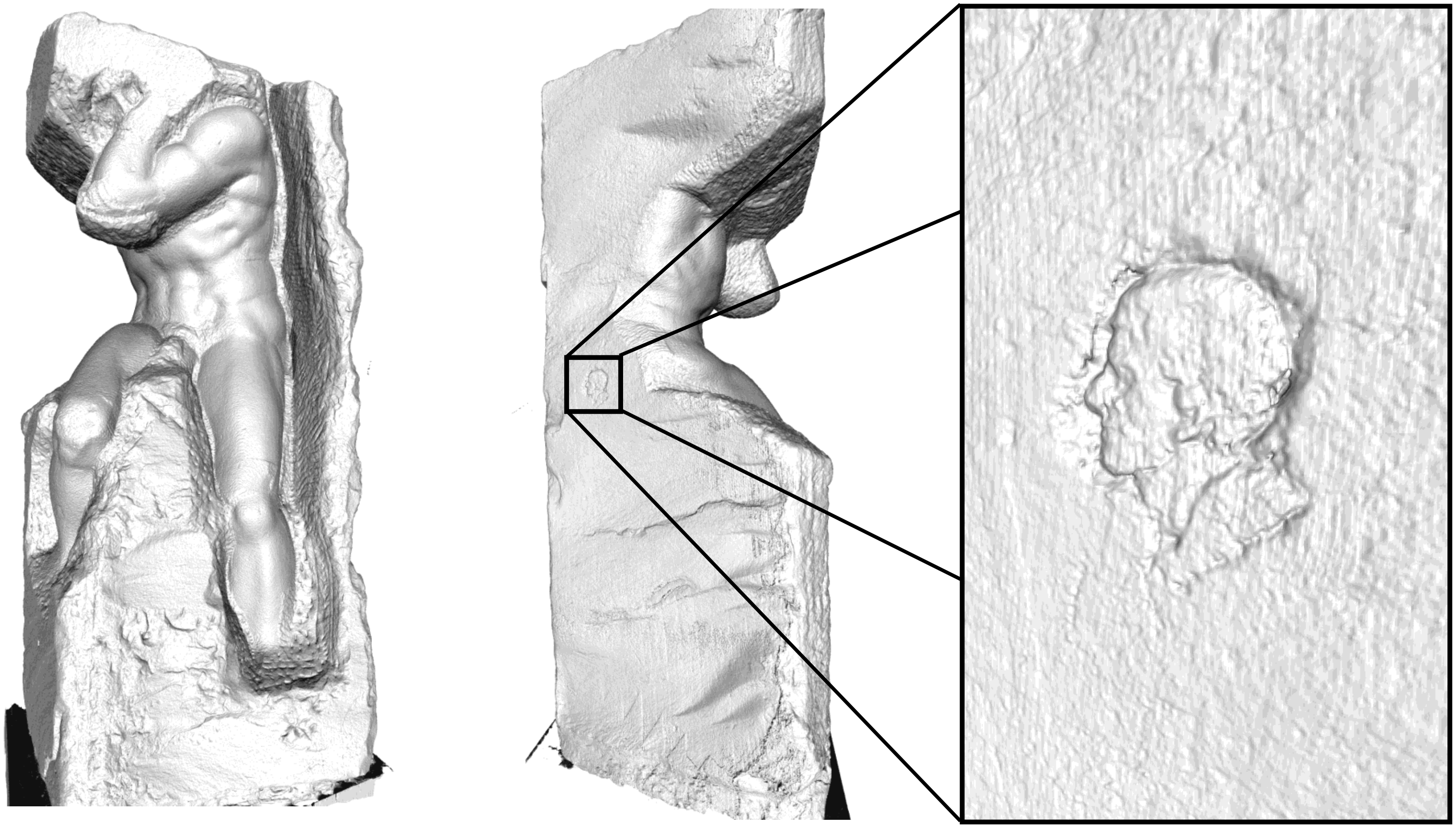


Michelangelo's Atlas

410 million points (9.15 GB), 1188MB memory, 98 minutes

Produced 642 million polygons

Results

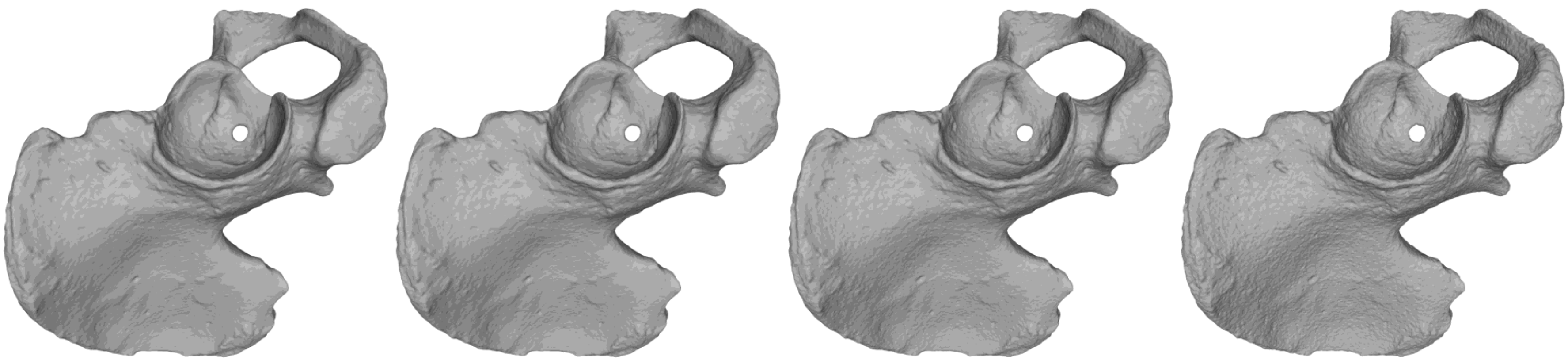


Michelangelo's Atlas

410 million points (9.15 GB), 1188MB memory, 98 minutes

Produced 642 million polygons

Robustness to Noise in Normals

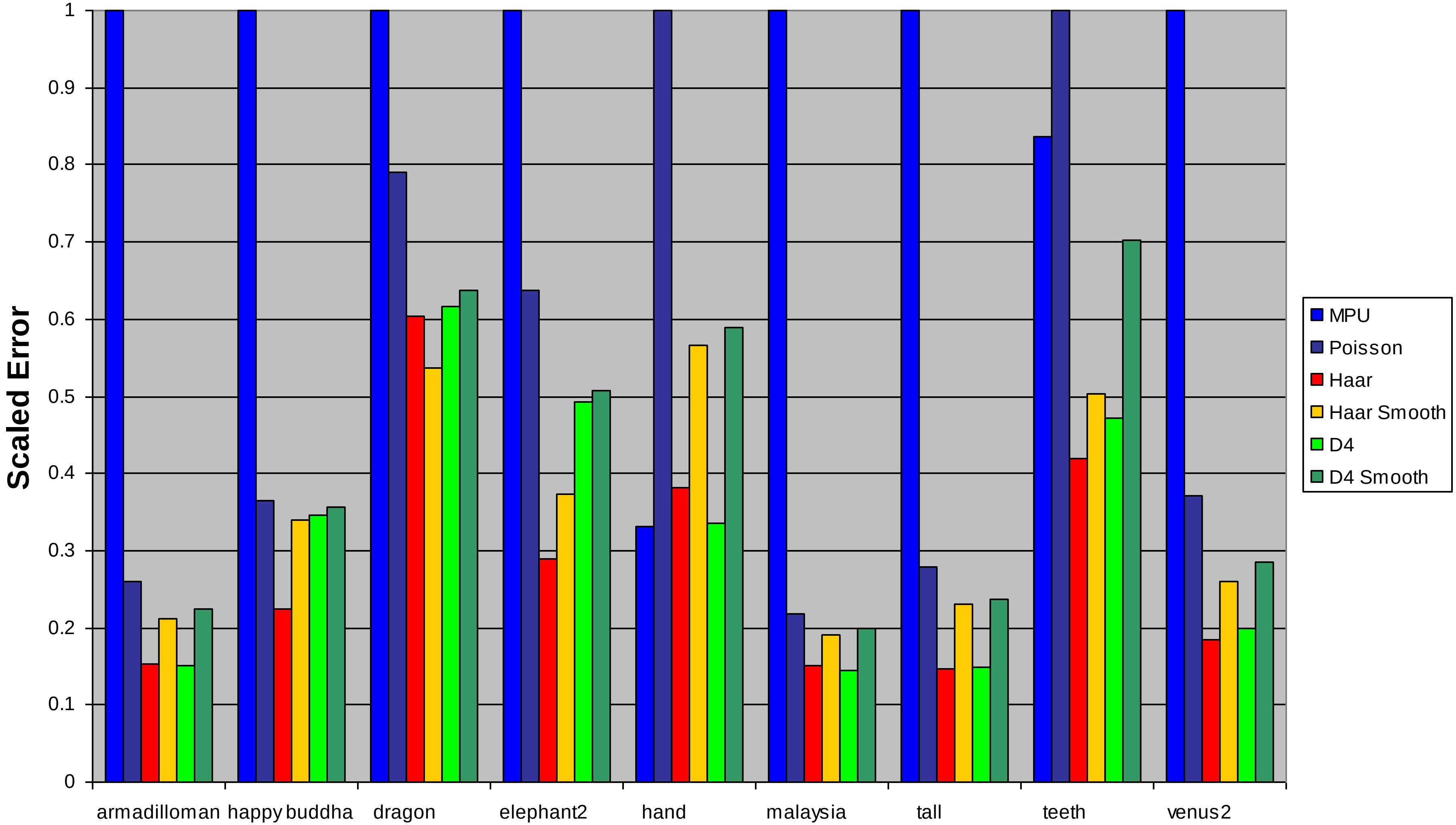


0° 30° 60° 90°

Comparison of Methods

MPU	Poisson	Haar	D4
551 sec	289 sec	17 sec	82 sec
750 MB	57 MB	13 MB	43 MB

Relative Hausdorf Errors



Conclusions

- Wavelets provide trade-off between speed/quality
- Works with all orthogonal wavelets
- Guarantees closed, manifold surface
- Out of core