Surface Reconstruction Based on Modified Gauss Formula

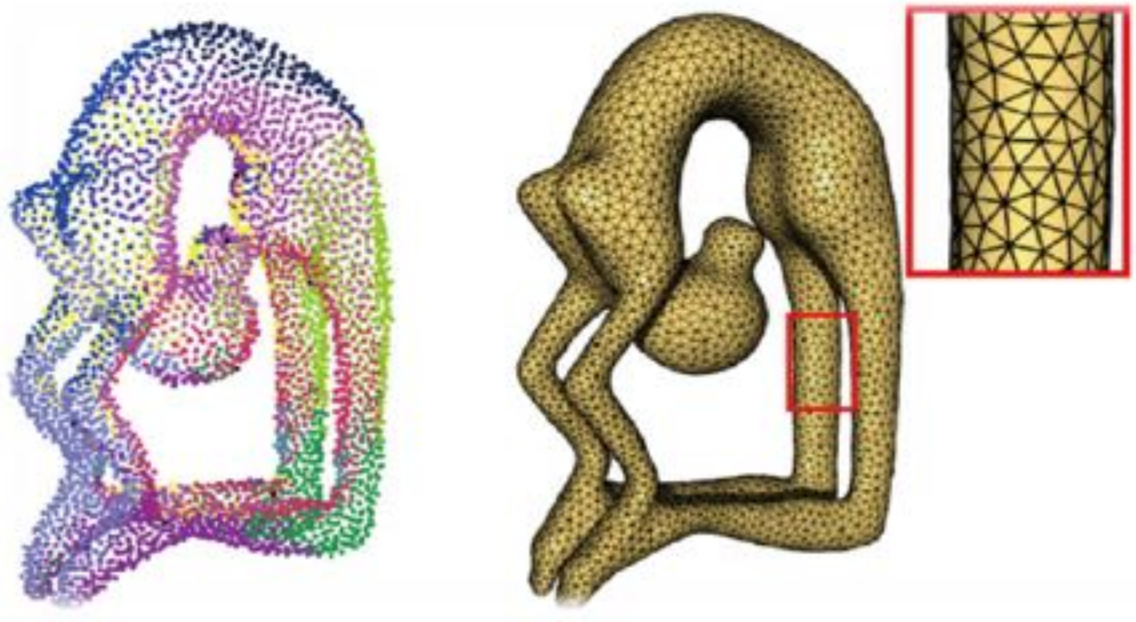
Wenjia Lu, Zuoqiang Shi, Jian Sun, Bin Wang

Tsinghua University

Outline

1 Related Works

- Surface reconstruction from point clouds
  - Well studied
- Applications
  - Manufacture
  - Animation
  - Visualization
  - Etc.



1 Related Works

■Combinatorial methods

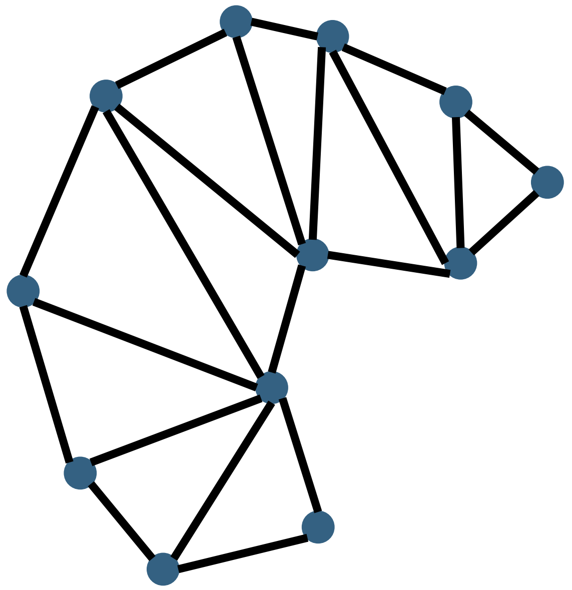
- Utilize (part of) the input sample points as vertices.
- Usually sensitive to noise and may produce jagged surfaces.

[Amenta et al. 2002]

[Kolluri et al. 2004]

[Xiong et al. 2014]

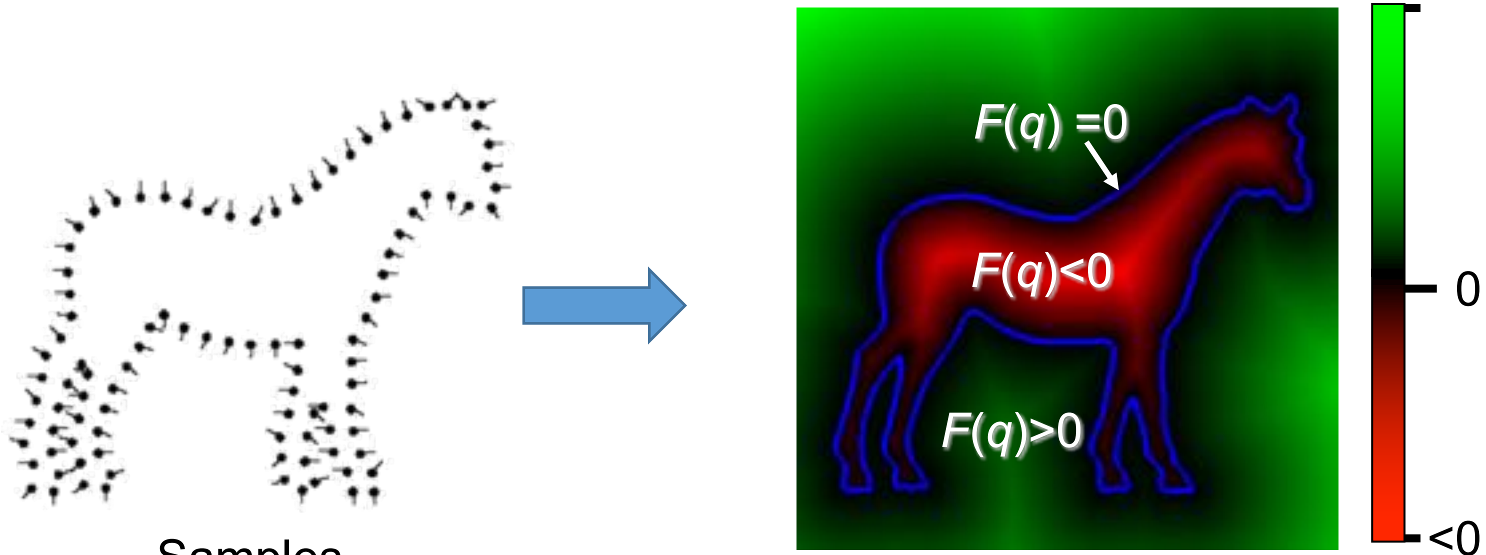
etc…



1 Related Works

■Implicit methods

- Estimate implicit functions from the input samples and extract iso-surfaces to generate triangle meshes.

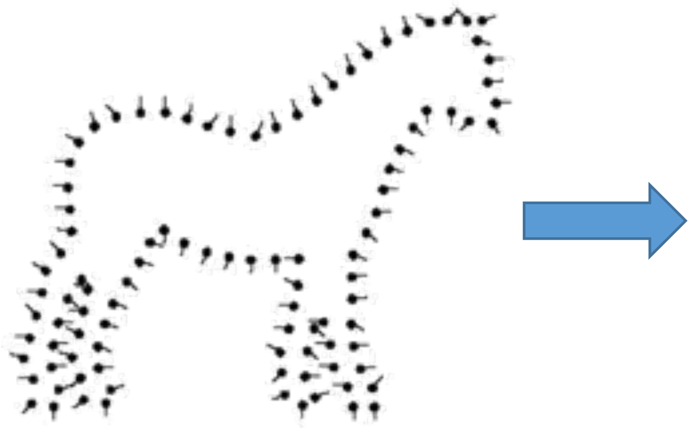


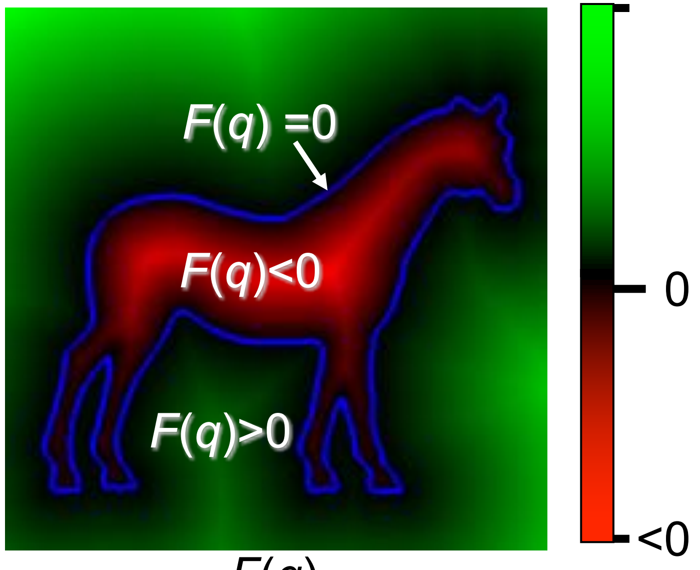
Samples   F(q)

1 Related Works

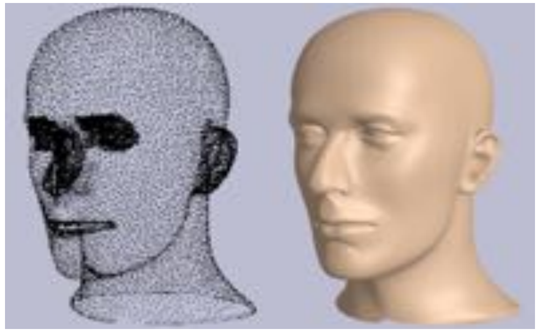
■Implicit methods

- How to choose implicit function and how to solve it?





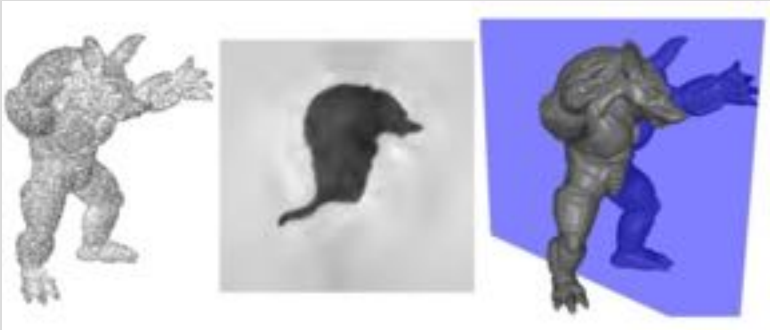
1 Related Works



[Hoppe *et al*. 1992]

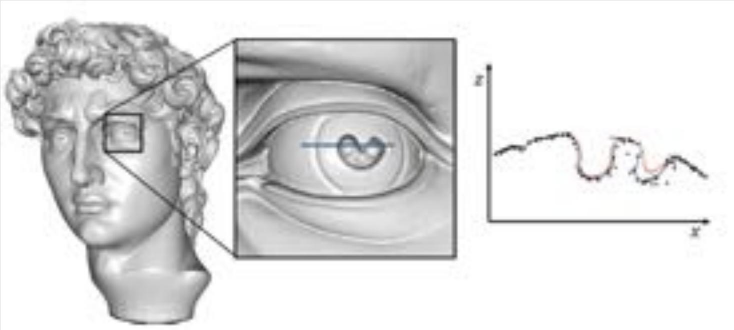
[Curless and Levoy 1996]

[Carr *et al*. 2001]



[Kazhdan *et al*. 2006]

[Calakli and Taubin 2011]

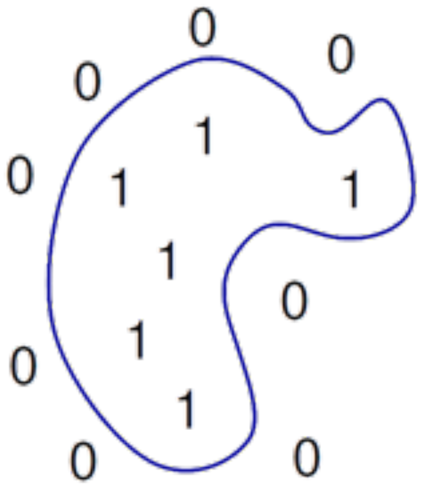


[Kazhdan *et al*. 2013]

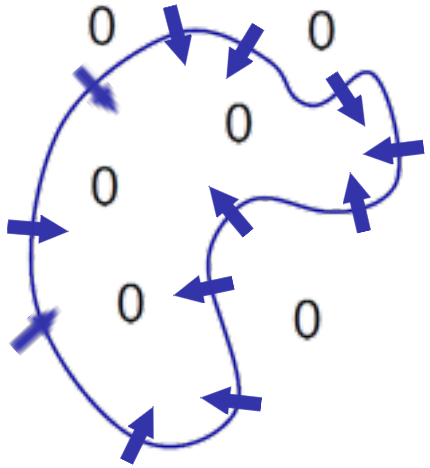
… …

1 Indicator function

- Indicator function : $\chi = \begin{cases} 0 & x \in R^3 \setminus \bar{\Sigma} \\ 1/2 & x \in \partial\bar{\Sigma} \\ 1 & x \in \Sigma \end{cases}$

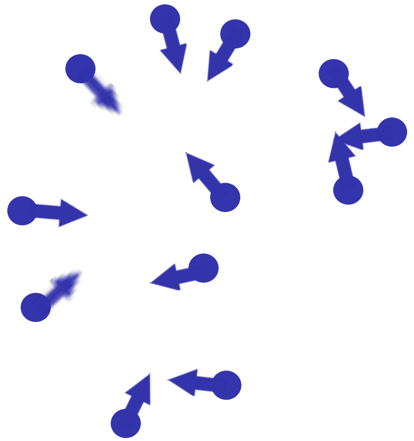


Indicator function



Indicator function gradients

$$\nabla\chi = \vec{N}$$



Samples

1 Indicator function algorithms

- Poisson surface reconstruction [Kazhdan *et al.* 2006]
  - Convolution of the indicator function
  - Over smoothed
- Screened Poisson surface reconstruction[Kazhdan *et al.* 2013]
  - A scalar function fitting term
  - Over fitting

1.1 Signed distance function

■Signed distance function methods

- A signed distance function is smooth near the surface
- Easy to interpolate
- Sensitive to noise

1 A hybrid solution

■Observation

- Indicator function : robust to noise
- Signed distance function : easy to interpolate near the surface

■Objective

- Away from the surface : indicator function
- Near the surface : signed distance function
- Balance between data fidelity and resiliency against noise

1 A hybrid solution

Our method

- Based on Gauss Lemma in the potential theory.
- Estimated directly from an explicit integral formula without solving any linear system.
- Easy to parallelize with small overhead.

Outline

2 Gauss Lemma

- Let Σ be an open region in $\mathbb{R}^3$, $\bar{\Sigma}$ denotes the closure of Σ. Consider the following double layer potential: For any $x \in \mathbb{R}^3, y \in \partial\Sigma$,

$$\chi(x) = \int_{\partial\Sigma} \frac{\partial G}{\partial \vec{N}(y)}(x, y) d\tau(y).$$

- G is the fundamental solution of the Laplace equation, which can be stated explicitly as

$$G(x, y) = -\frac{1}{4\pi||x-y||}.$$

2 Gauss Lemma

- Let Σ be an open region in $\mathbb{R}^3$, $\bar{\Sigma}$ denotes the closure of Σ. Consider the following double layer potential: For any $x \in \mathbb{R}^3, y \in \partial\Sigma$,

$$\chi(x) = \int_{\partial\Sigma} \frac{\partial G}{\partial \vec{N}(y)}(x, y) d\tau(y).$$

- Note that

$$\frac{\partial G}{\partial \vec{N}(y)}(x, y) = -\frac{1}{4\pi} \frac{(x-y)\cdot\vec{N}(y)}{\left||x-y|\right|^3}.$$

which we call the kernel function, and denote by $K(x, y)$.

2 Gauss Lemma

- Then , the indicator function can be formalized as :

$$\chi(x) \approx -\frac{1}{4\pi}\sum_{y\in P}\frac{(x-y)\cdot\vec{N}(y)}{||x-y||^3}y.A.$$

- Where $y.A$ is a small region near the sample y , the set $\{y.A\}_{y\in P}$ cover the surface $\partial\Sigma$.

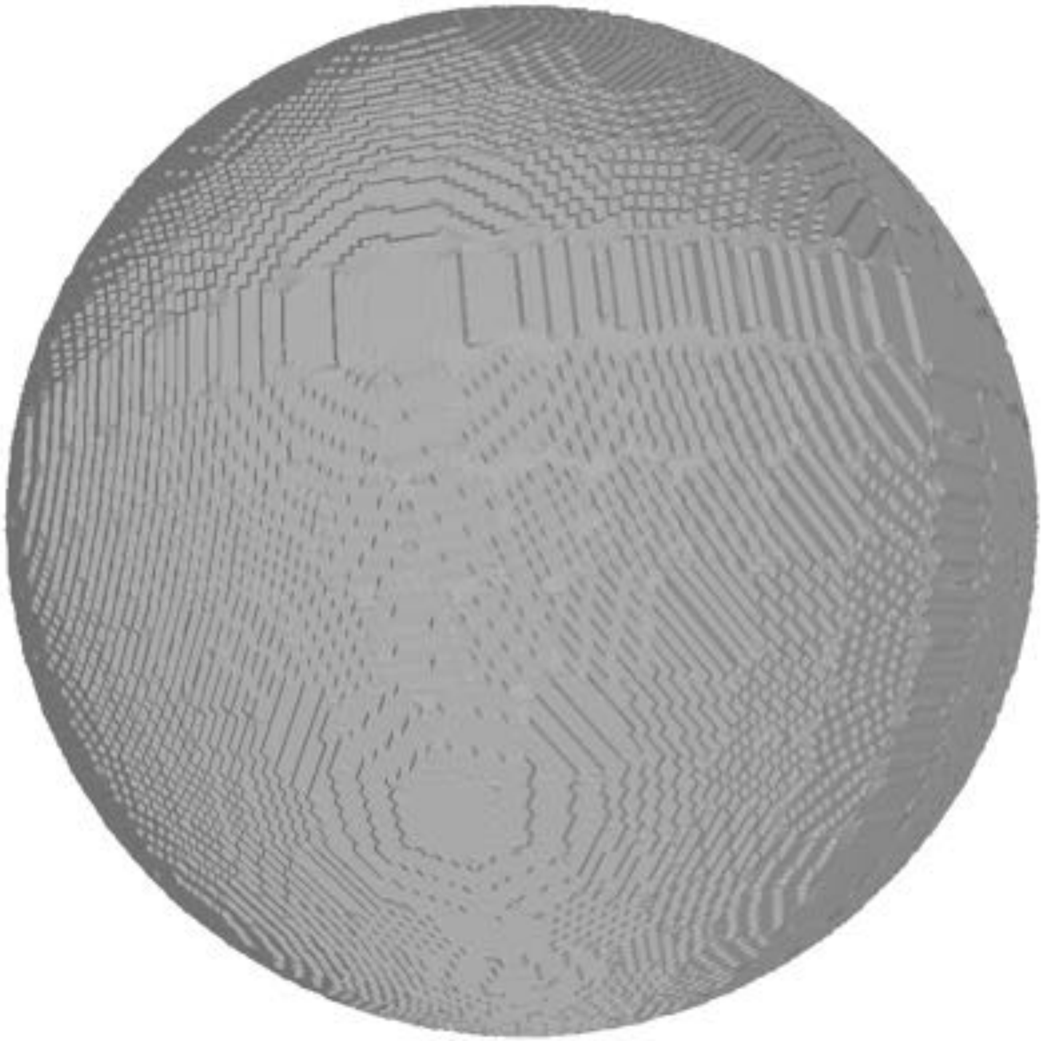
$$\text{Indicator function}: \chi(x) = \begin{cases} 0 & x \in R^3 \setminus \bar{\Sigma} \\ 1/2 & x \in \partial\bar{\Sigma} \\ 1 & x \in \Sigma \end{cases}$$

DIFFICULTIES

2.1 **Discontinuity**

2.2 Singularity

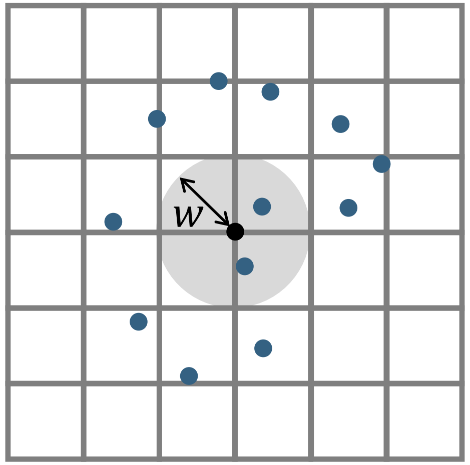
2.3 Globalness



2.1 Discontinuity

- Original kernel: $K(x,y) = \frac{(x-y)\cdot\vec{N}(y)}{4\pi\cdot||x-y||^3}$

- $\widetilde{K}(x,y) = \begin{cases} K(x,y)\,, & ||x-y|| \geq w(x) \\ -\alpha\frac{(x-y)\cdot\vec{N}(y)}{4\pi\cdot w^3(x)} & ||x-y|| < w(x), \alpha \in \{0,1\} \end{cases}$

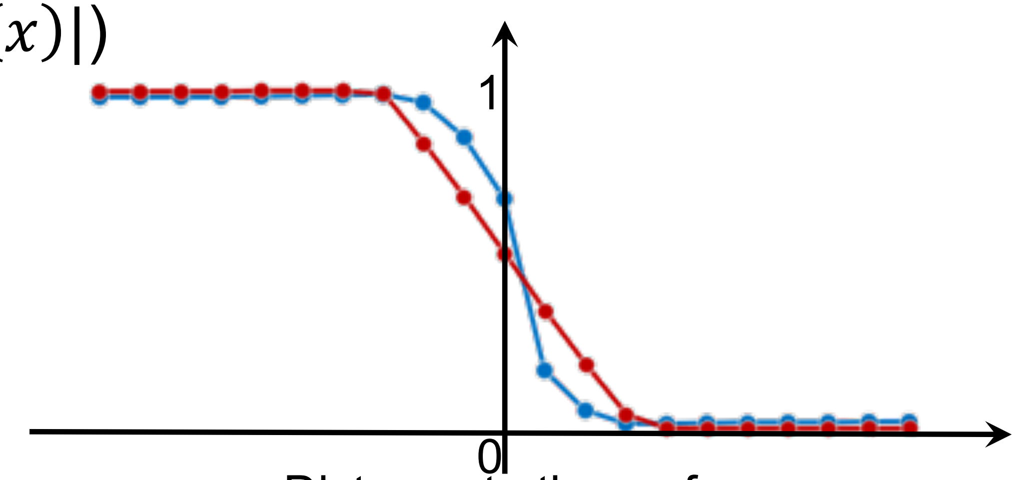


2.1 Discontinuity

$$\tilde{\chi}(x) = \int_{B(x,w(x))\cap\partial\Sigma} \widetilde{K}(x,y)\mathrm{d}\tau(y) + \int_{\partial\Sigma\setminus B(x,w(x))} K(x,y)\mathrm{d}\tau(y)$$

$$= \frac{1}{2} + \left(\frac{1}{2} - \frac{\alpha}{4}\right)\frac{d(x)}{w(x)} + \frac{\alpha d^3(x)}{4w^3(x)} + O(|w(x)|)$$

$$\tilde{\chi}(x) \approx \begin{cases} \frac{1}{2} + \frac{d(x)}{4w(x)} + \frac{d^3(x)}{4w^3(x)} & \alpha = 1 \\ \frac{1}{2} + \frac{d(x)}{2w(x)} & \underline{\underline{\alpha = 0}} \end{cases}$$

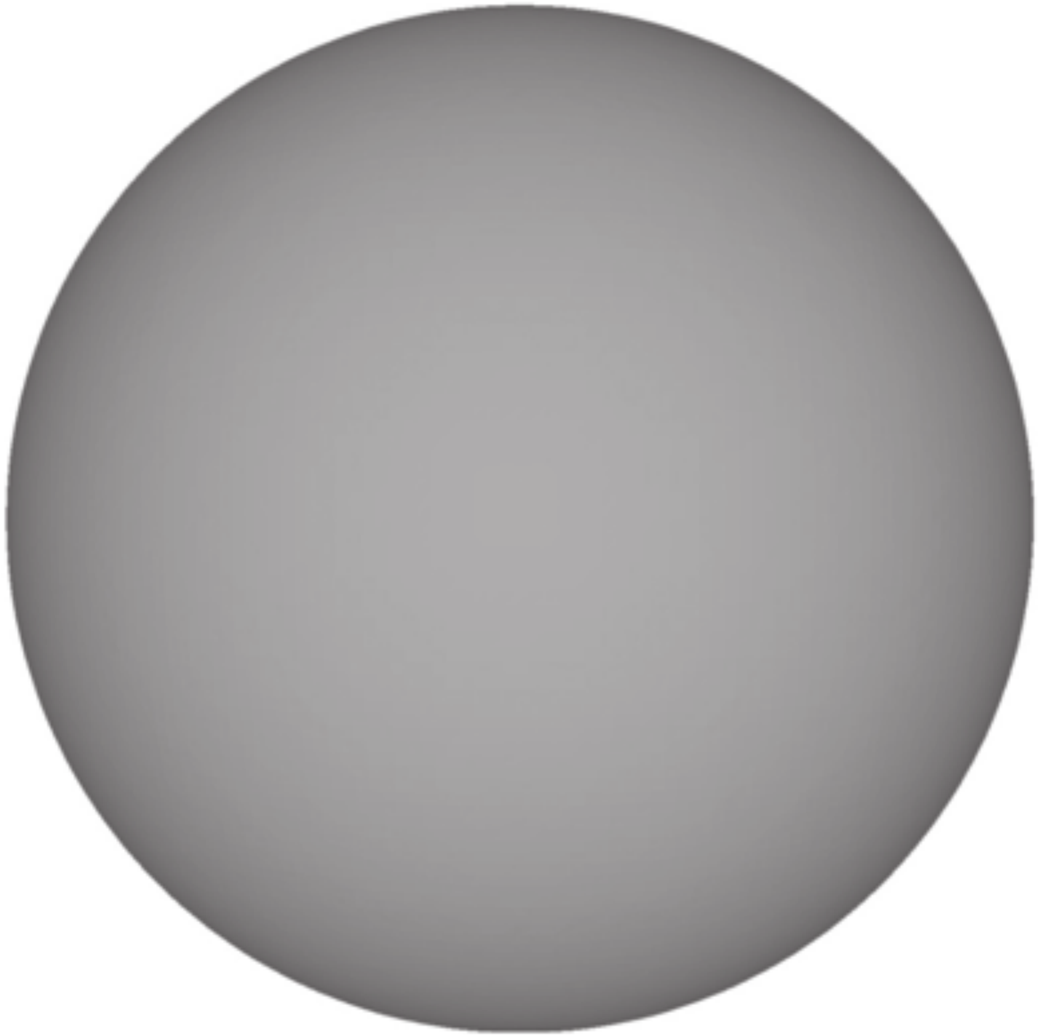


DIFFICULTIES

2.1 **Discontinuity**

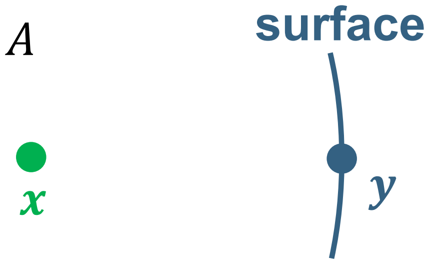
2.2 Singularity

2.3 Globalness



DIFFICULTIES

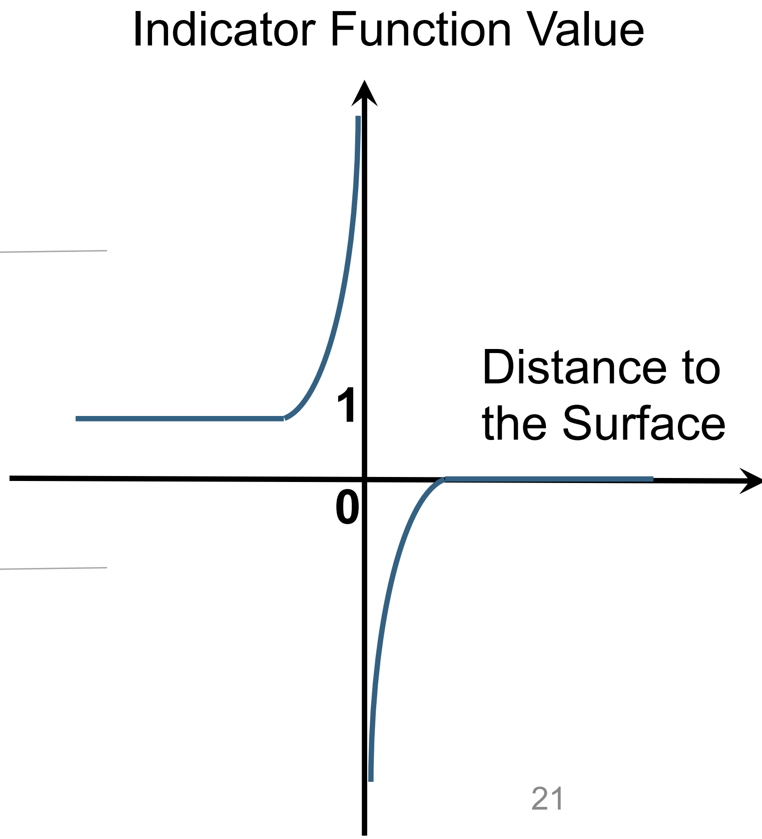
Indicator function： $C(x,y) = -\frac{1}{4\pi}\frac{(x-y)\cdot\vec{N}(y)}{||x-y||^{3}}y.A$



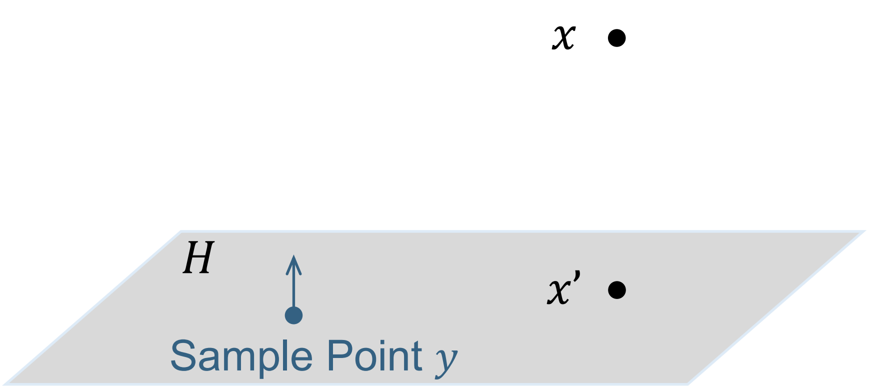
2.1 Discontinuity

2.2 Singularity

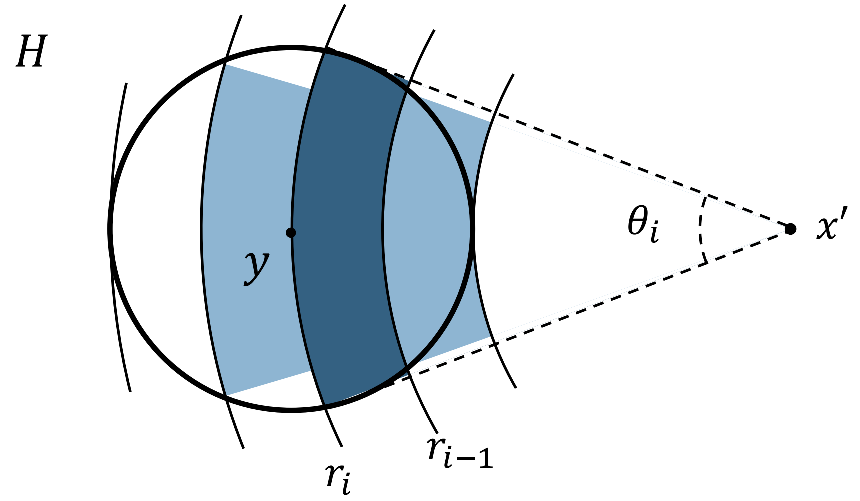
2.3 Globalness

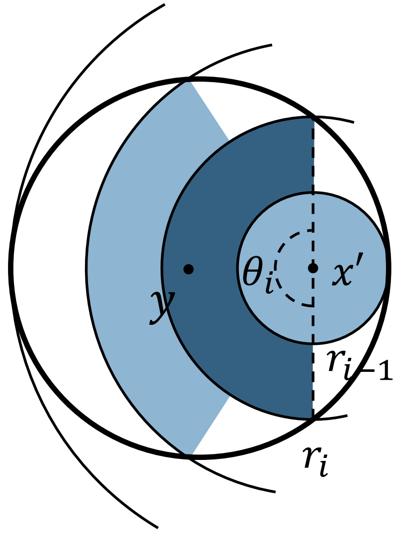


2.2 Disk integration



$$
\begin{aligned}
c_i &= \int_{F_i \cap A(r_{i-1}, r_i)} \tilde{K}(x, y)\mathrm{d}y \\
&= -\frac{1}{4\pi} \int_0^{\theta_i} \int_{r_{i-1}}^{r_i} \frac{d}{(d^2 + r^2)^{3/2}} r\mathrm{d}r\mathrm{d}\theta \\
&= \frac{\theta_i d}{4\pi} \left(\frac{1}{\sqrt{d^2 + r_{i-1}^2}} - \frac{1}{\sqrt{d^2 + r_i^2}} \right)
\end{aligned}
$$





DIFFICULTIES

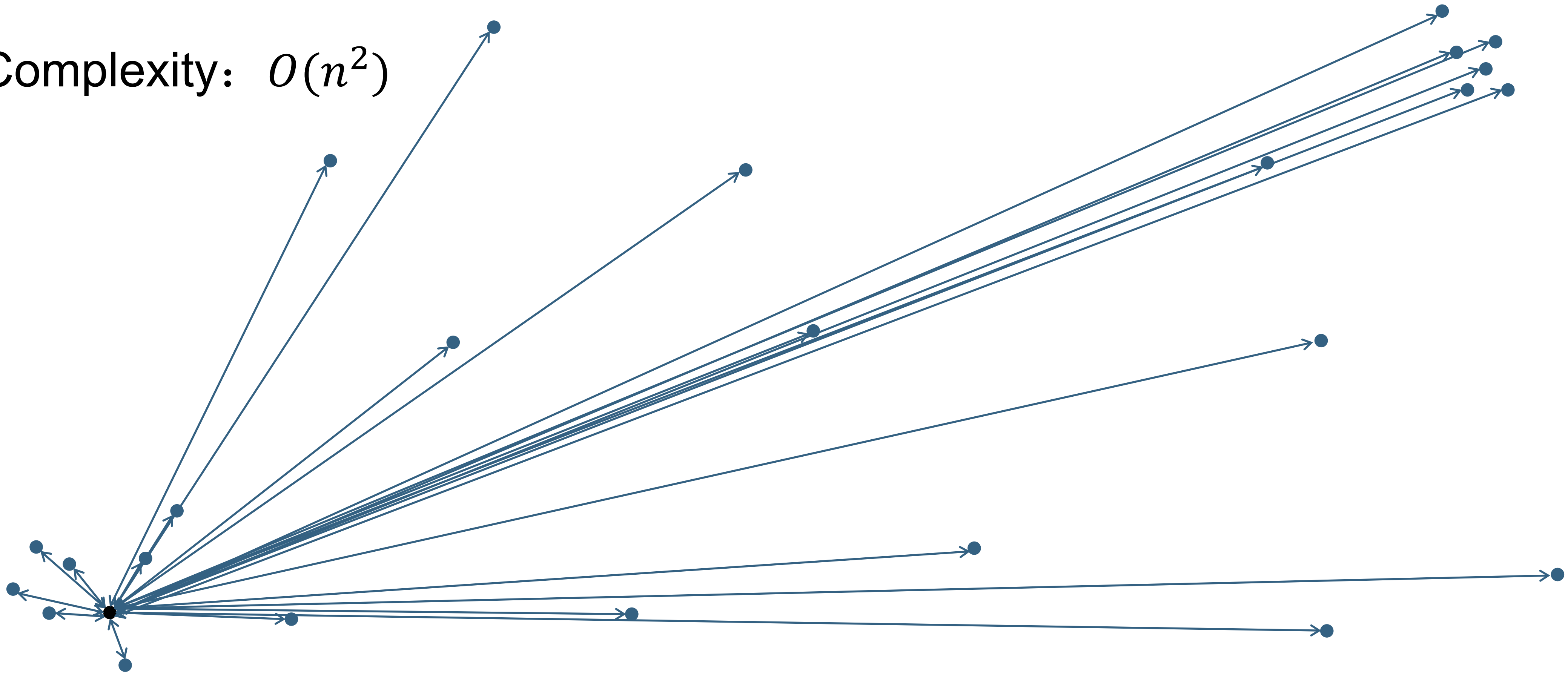
2.1 Discontinuity

2.2 Singularity

2.3 Globalness

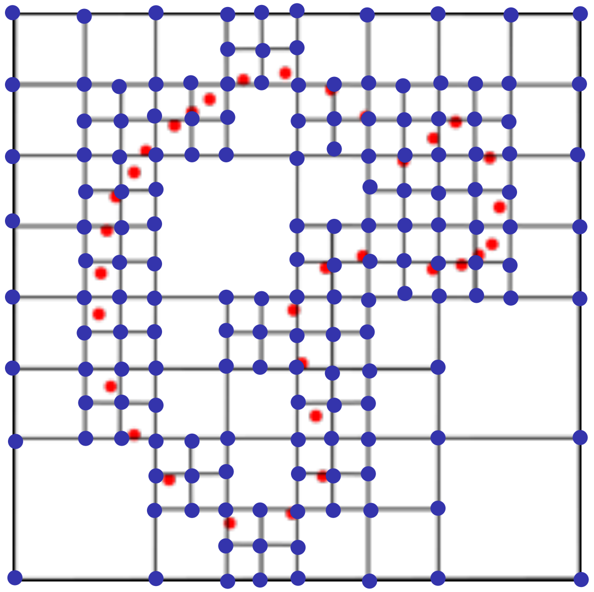
2.3 Globalness

Complexity： $O(n^2)$



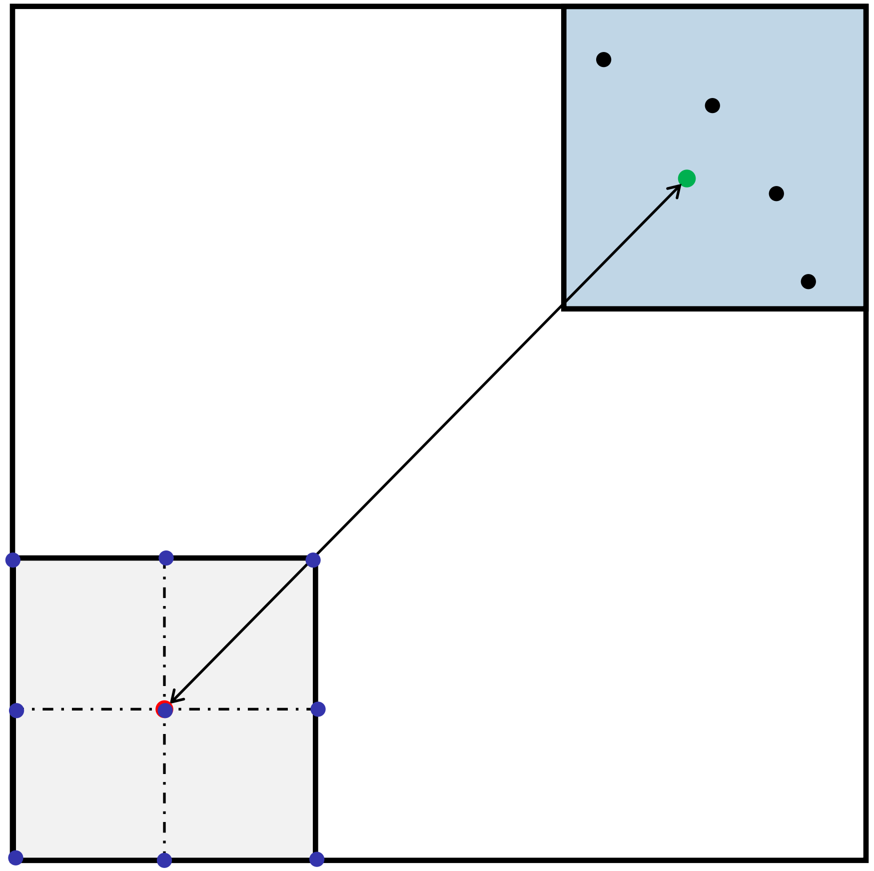
2.3 Fast Multipole Method

- Data Structure：Adaptive Octree
- Objective：Estimate function value for each cell



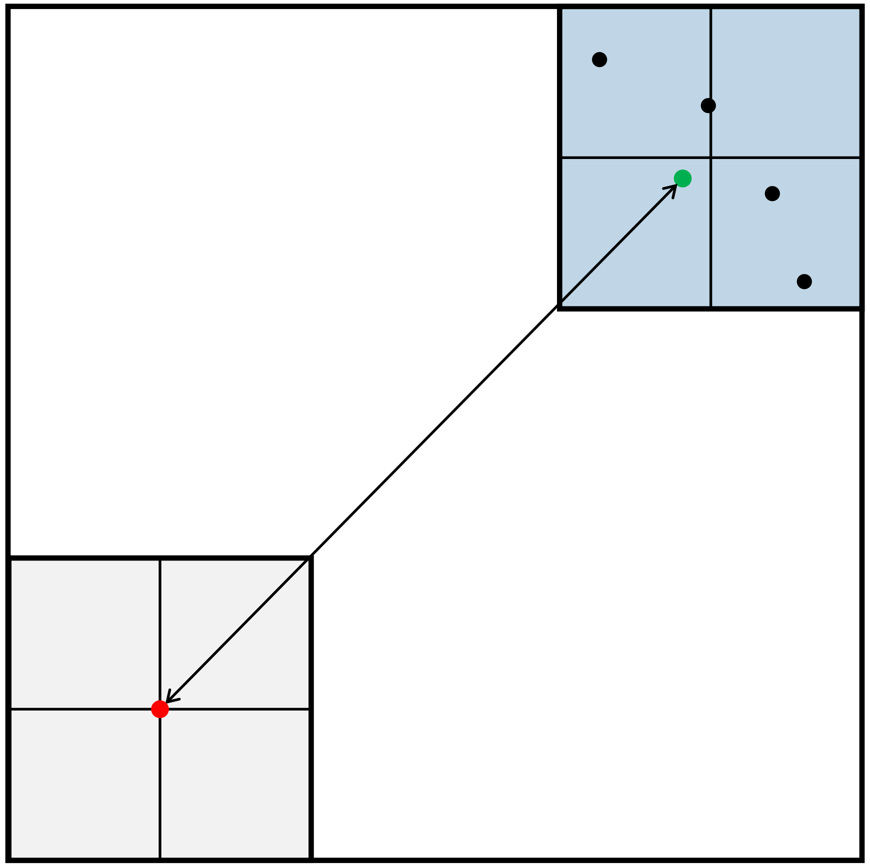
2.3 Fast Multipole Method

- $Dist > Threshold$



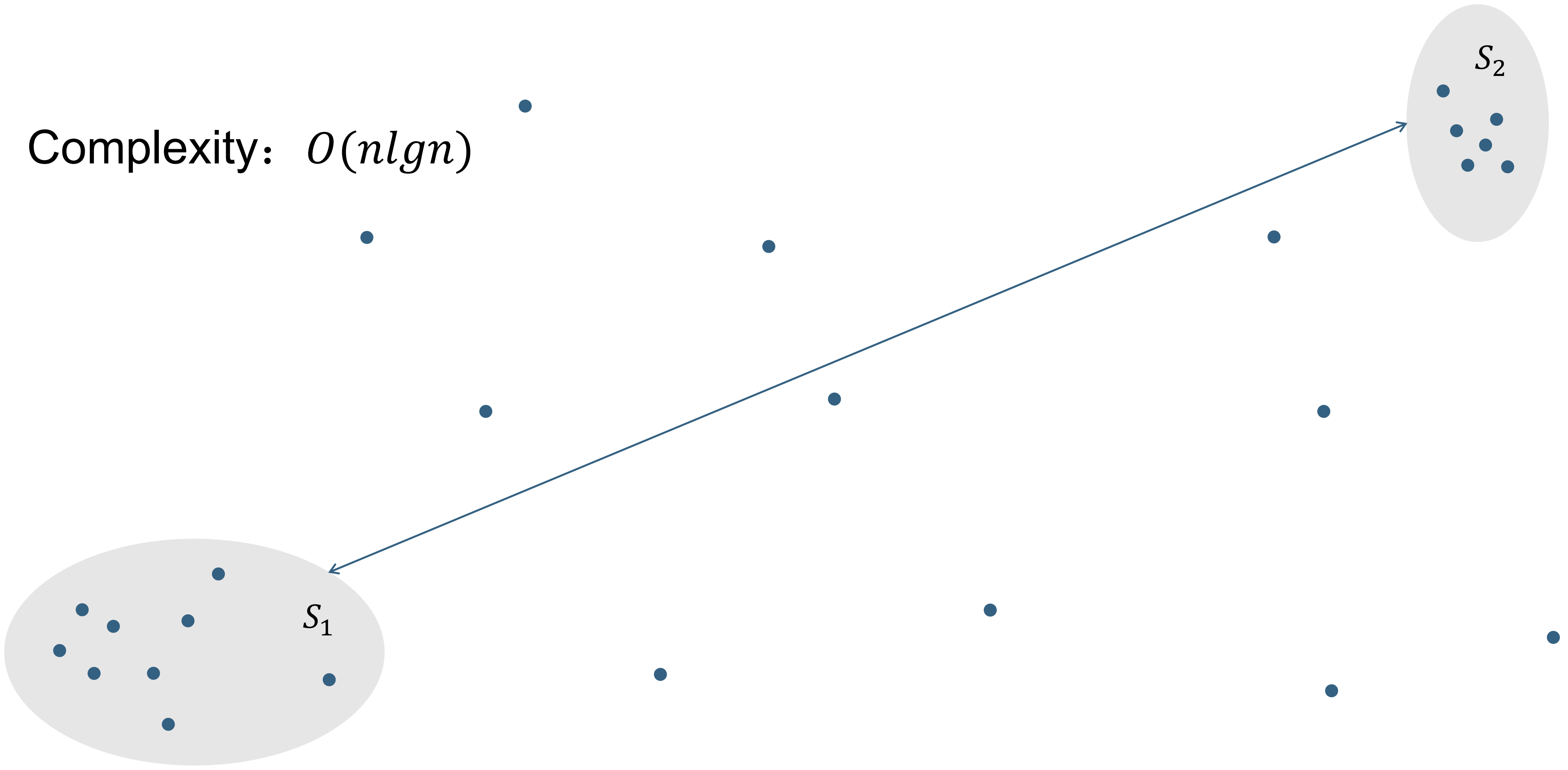
2.3 Fast Multipole Method

- $Dist < Threshold$



2.3 Fast Multipole Method

Complexity： $O(nlgn)$



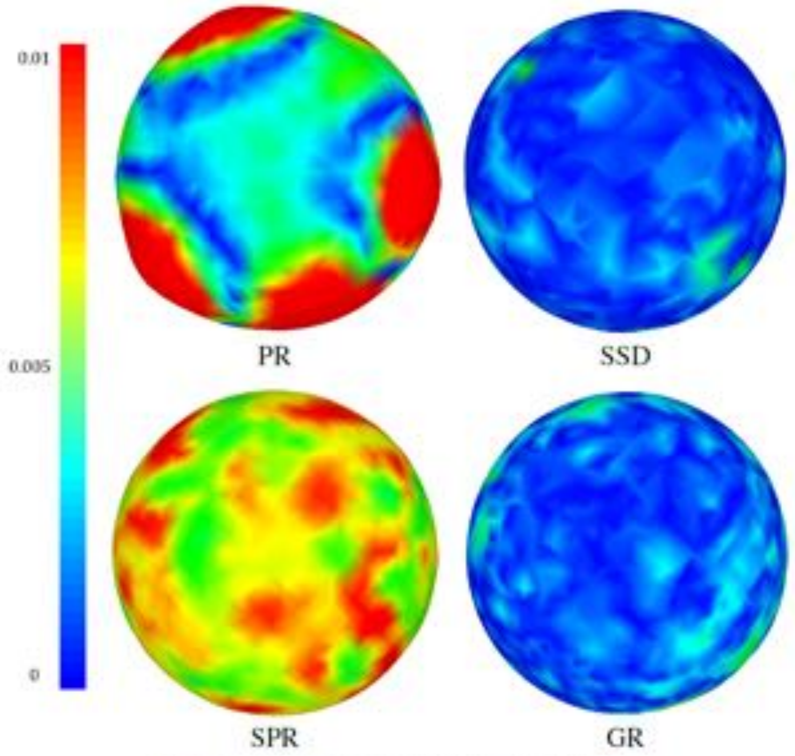
Outline

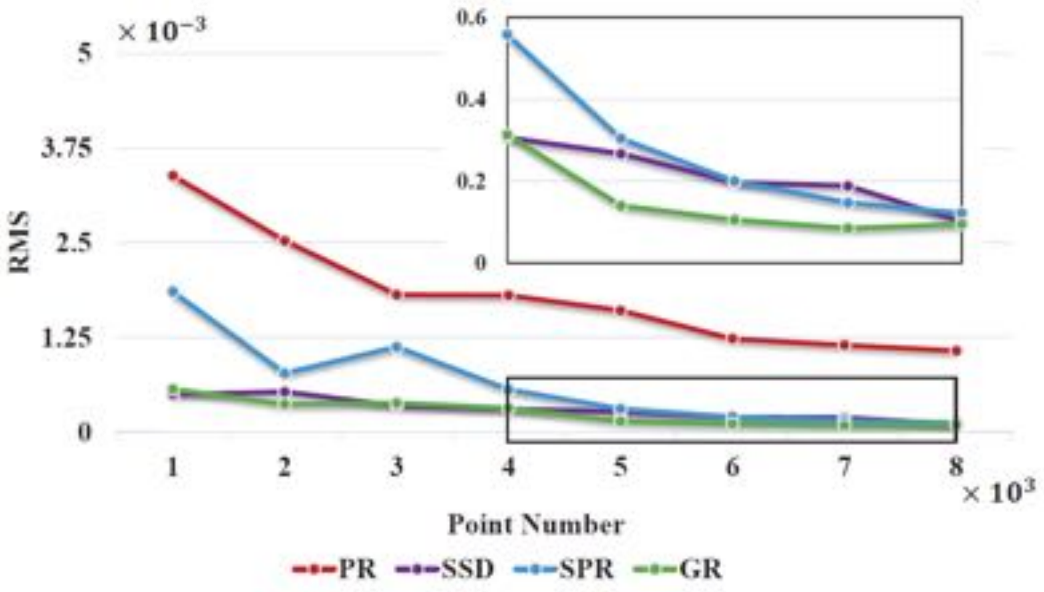
3 Algorithms for comparisons

- Poisson SR (Kazhdan *et al.* 2006, SGP)
- Screened Poisson SR (Kazhdan *et al.* 2013, TOG)
- SSD (Calakli *et al.* 2011, CGF)
- Dictionary learning (Xiong *et al.* 2014, TOG)

3 Accuracy

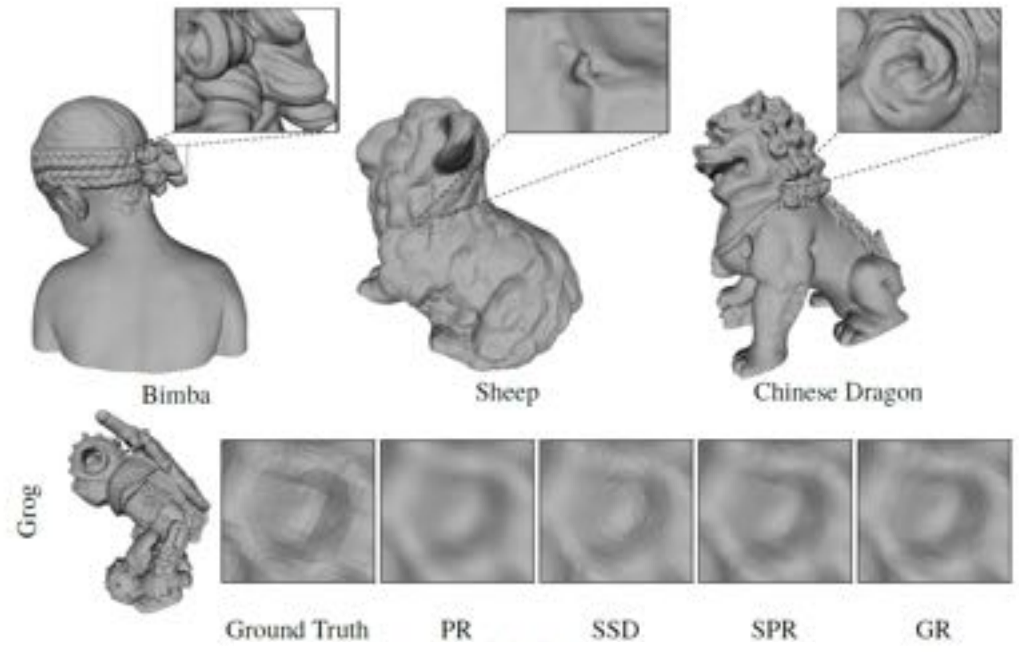
■Data：Samples on unit sphere (1000 samples)

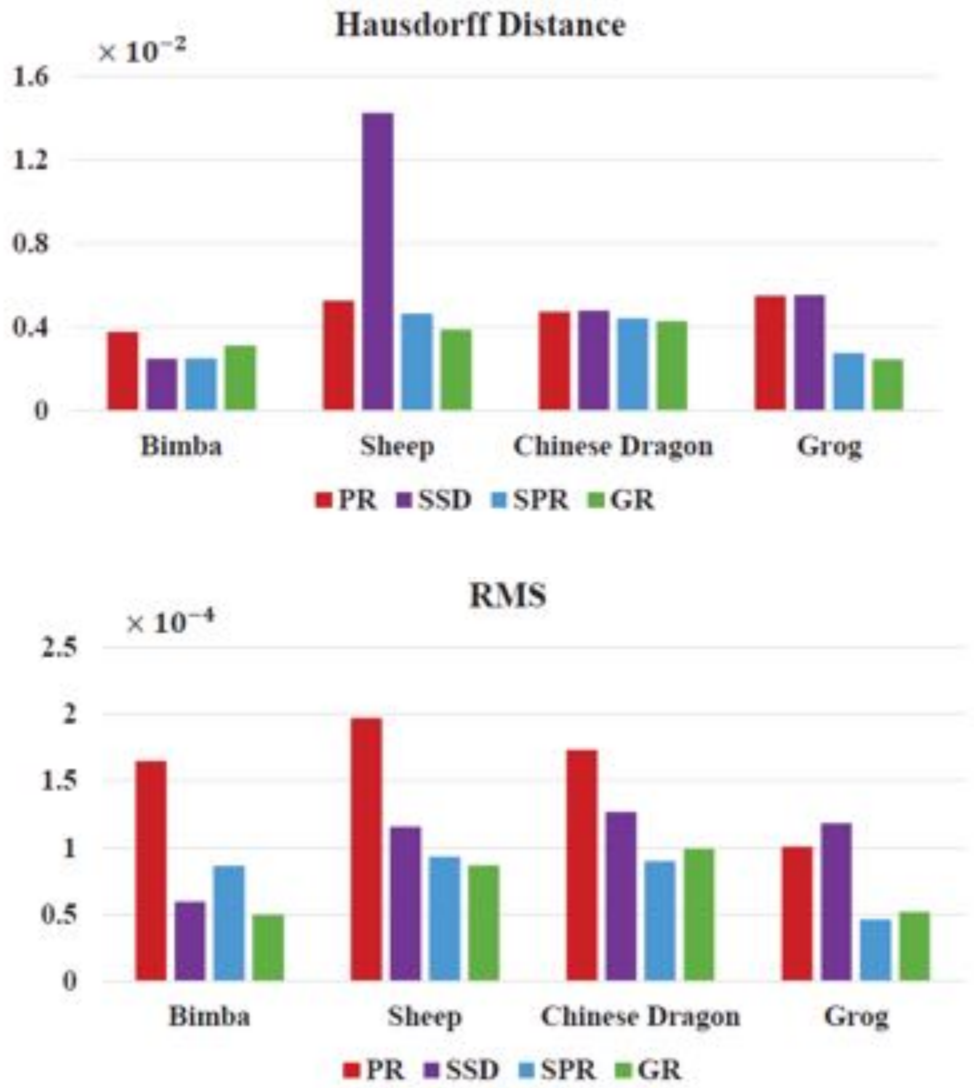




3 Accuracy

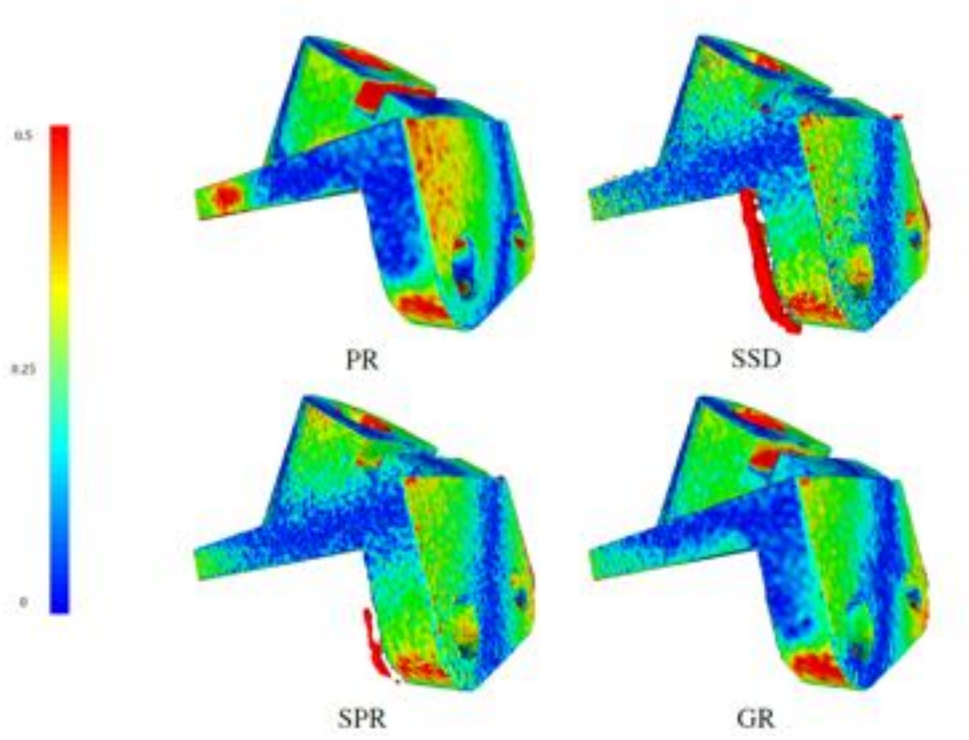
■Data：AimShape

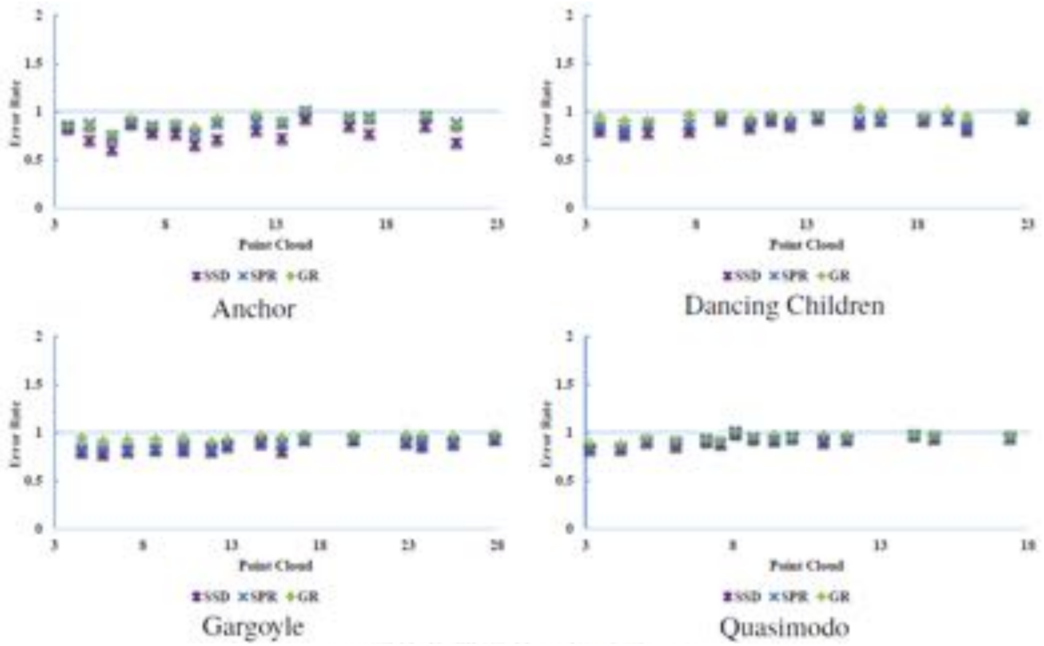




3 Accuracy

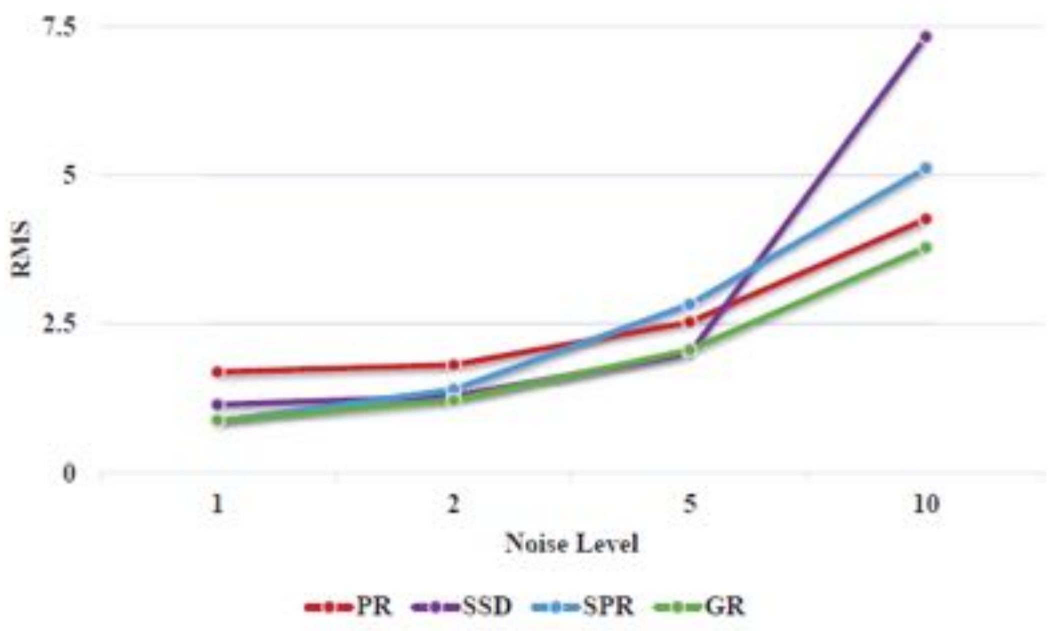
- **Data**： Berger et al. A benchmark for surface reconstruction [TOG 2013]

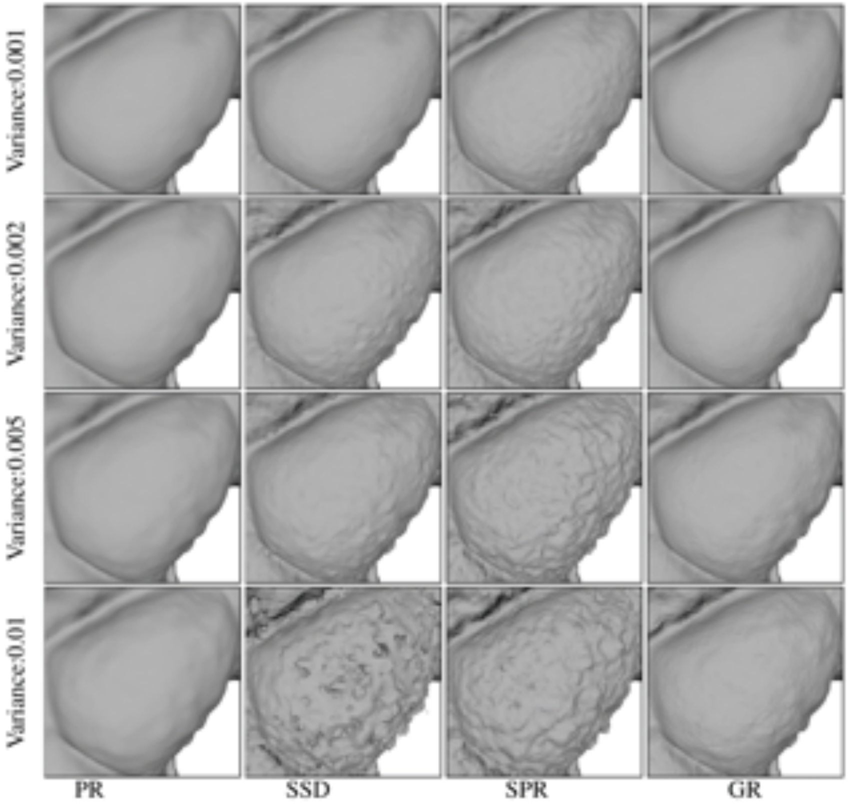




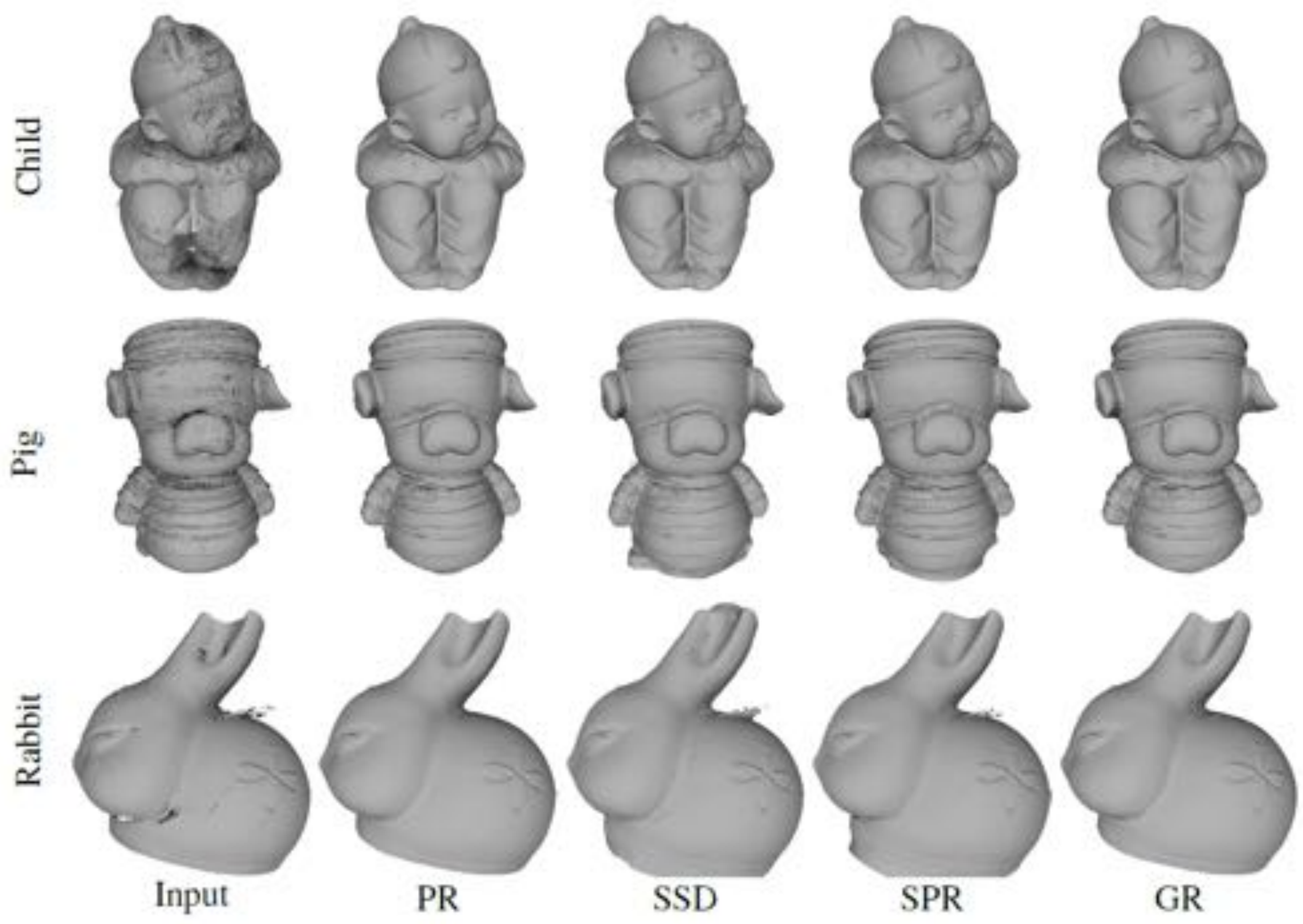
3 Synthetic noise

■Data：Gaussian distribution noises





3 Real-world Scanned Data



3 Real-world Scanned Data

3 Merlion model

■Data：Xiong et al. Robust surface reconstruction via dictionary [TOG 2014]

GR Result(octree depth = 10)

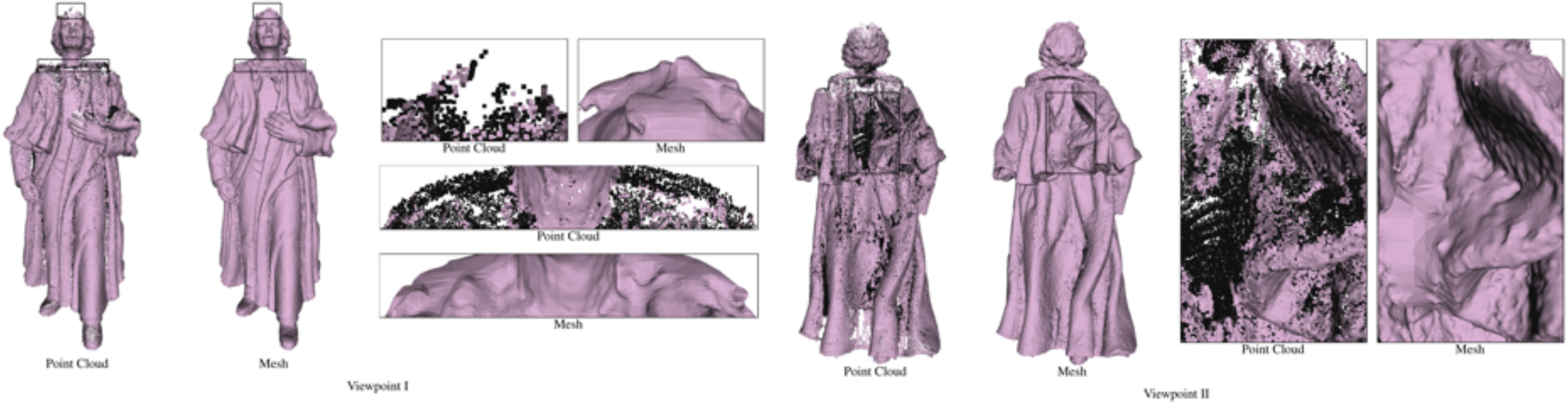
[Xiong et al. 2014], $|v| = 0.28$

GR (octree depth = 9), $|v| = 0.32$

GR (octree depth = 10), $|v| = 1.32$

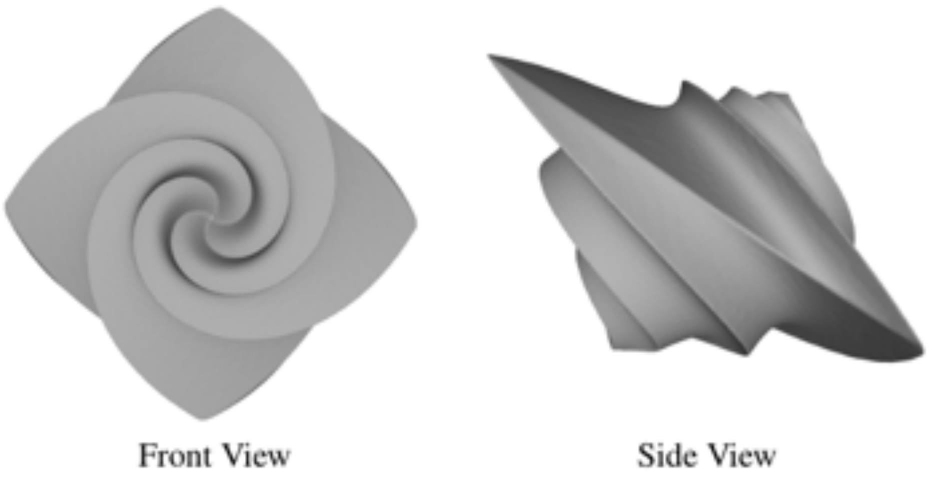
3 Missing Data

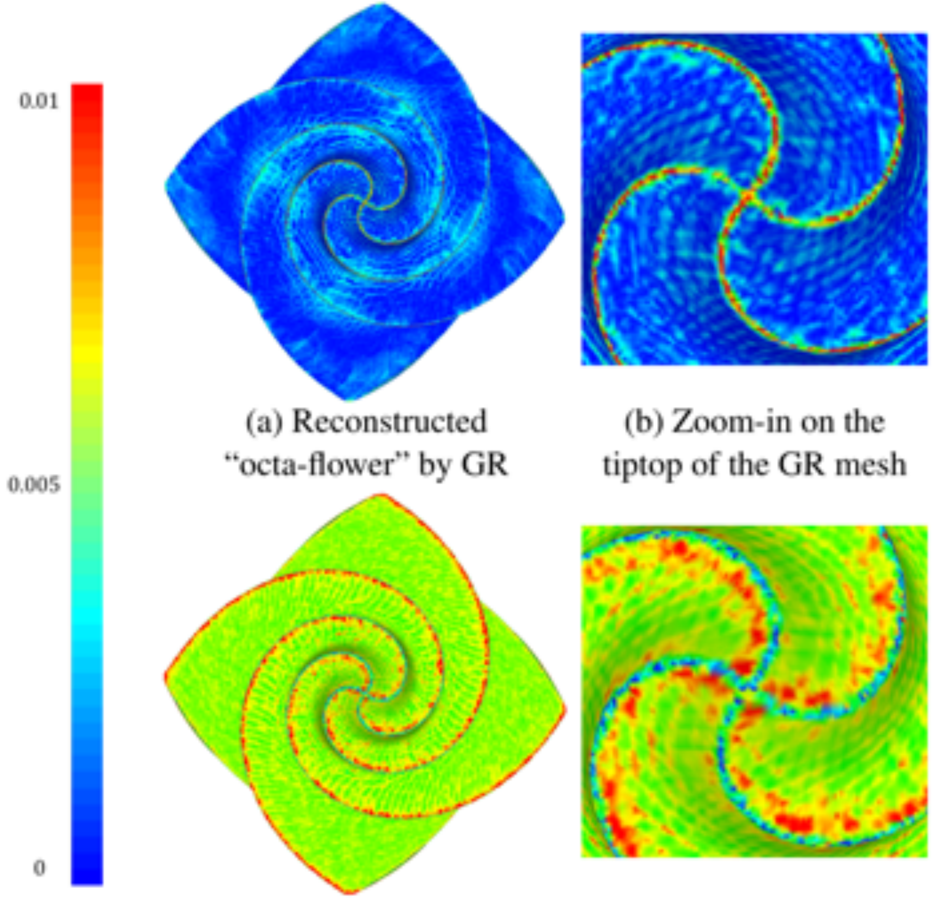
- Data：EPFL(http://lgg.epfl.ch/code.php)



3 Sharp features

- Data：Oct-Flower (Aimshape)





(a) Reconstructed "octa-flower" by GR

(b) Zoom-in on the tiptop of the GR mesh

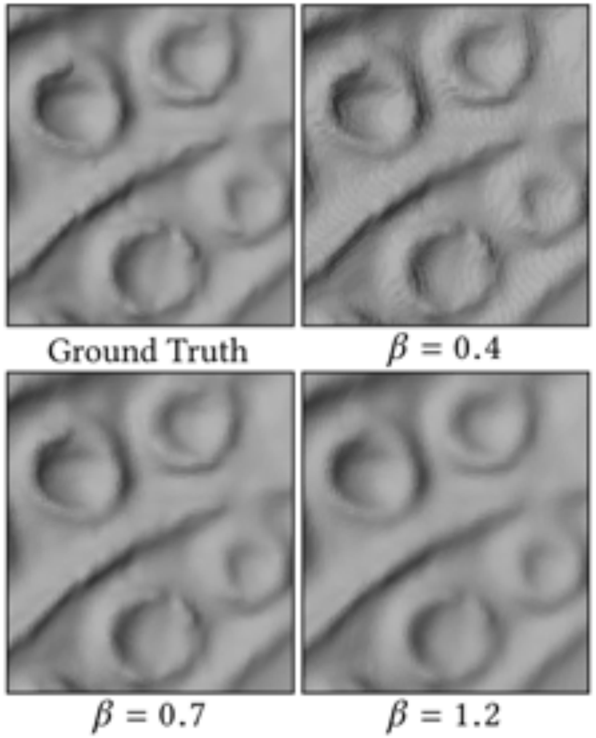
(c) Reconstructed "octa-flower" by SPR

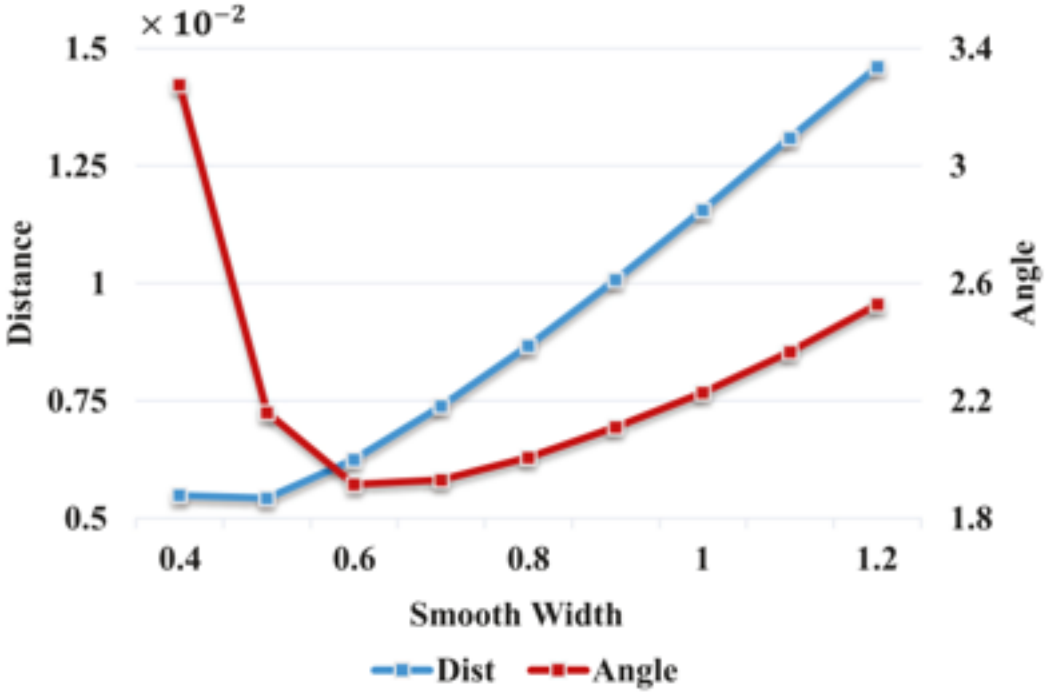
(d) Zoom-in on the tiptop of the SPR mesh

3 Efficiency

Method	Cores	Time in Seconds							
		Bimba				Sheep			
		Tree Cons	Func-Eval	Iso-Surf	Total	Tree Cons	Func-Eval	Iso-Surf	Total
PR	CPU 1 core	6.00	63.79	30.94	100.72	2.08	30.63	13.20	45.92
SSD	CPU 1 core	2.90	27.83	9.59	40.33	0.96	14.38	4.51	19.85
SPR	CPU 1 core	8.13	95.85	29.93	133.92	2.78	31.41	9.58	43.77
	CPU 10 cores	8.36	14.86	10.55	33.77	2.80	5.60	3.30	11.70
GR	CPU 1 core	13.90	36.05	17.12	67.07	6.03	14.26	8.39	28.69
	CPU 10 cores	10.37	4.21	8.28	22.86	4.39	1.71	4.18	10.28
	GPU	-	1.72	-	-	-	0.77	-	-
Method	Cores	Time in Seconds							
		Chinese Dragon				Grog			
		Tree Cons	Func-Eval	Iso-Surf	Total	Tree Cons	Func-Eval	Iso-Surf	Total
PR	CPU 1 core	8.55	110.69	54.56	173.79	10.96	182.34	93.20	286.50
SSD	CPU 1 core	3.78	34.77	12.21	50.76	5.13	45.57	16.96	67.66
SPR	CPU 1 core	10.75	124.65	38.32	173.72	13.03	176.05	54.22	243.30
	CPU 10 cores	10.65	19.34	12.88	42.86	13.04	26.21	18.55	57.80
GR	CPU 1 core	22.10	64.88	27.87	114.85	36.00	99.28	46.50	181.78
	CPU 10 cores	16.42	7.32	13.68	37.42	24.72	11.14	23.18	59.04
	GPU	-	3.03	-	-	-	5.30	-	-
Method	Cores	Time in Seconds							
		Child				Pig			
		Tree Cons	Func-Eval	Iso-Surf	Total	Tree Cons	Func-Eval	Iso-Surf	Total
PR	CPU 1 core	9.48	128.60	58.15	196.22	12.29	168.62	71.00	251.91
SSD	CPU 1 core	4.27	42.00	12.01	58.29	5.27	45.37	12.17	62.80
SPR	CPU 1 core	11.85	127.83	37.96	177.64	13.27	143.48	42.00	198.74
	CPU 10 cores	11.74	21.91	12.59	46.24	13.18	22.90	13.84	49.92
GR	CPU 1 core	25.40	70.19	29.39	124.98	29.42	91.05	33.11	153.58
	CPU 10 cores	18.67	8.15	15.15	41.97	21.64	10.50	17.19	49.33
	GPU	-	3.44	-	-	-	4.37	-	-

3 Choice of width coefficient





Outline

4 Conclusion

- A hybrid implicit function ;
- An explicit integral formula based on Gauss Lemma ;
  - without solving any linear system ;
  - has a natural parallel implementation ;
- Disk integration ;
- A concise and efficient version of FMM.

4 Future work

- Hash Octree
- Enhanced implementation of the FMM
- A closed form formula for disk integration
- Completely GPU Version

- Thanks!