

# Объединение карт глубины в облако точек. Poisson реконструкция

Фотограмметрия. Лекция 8



- Depth Maps filtering
- Dense Cloud filtering
- Poisson Reconstruction

Полярный Николай  
[polarnick239@gmail.com](mailto:polarnick239@gmail.com)

## Где мы сейчас?

- Есть разреженное облако 3D положений ключевых точек
- Есть хорошо оптимизированные внутренние калибровки камер (intrinsics)
- Есть хорошо оптимизированные положения и ракурсы камер (extrinsics)
- Есть (шумные) карты глубины описывающие подробно и точно геометрию

## Что хотим дальше?

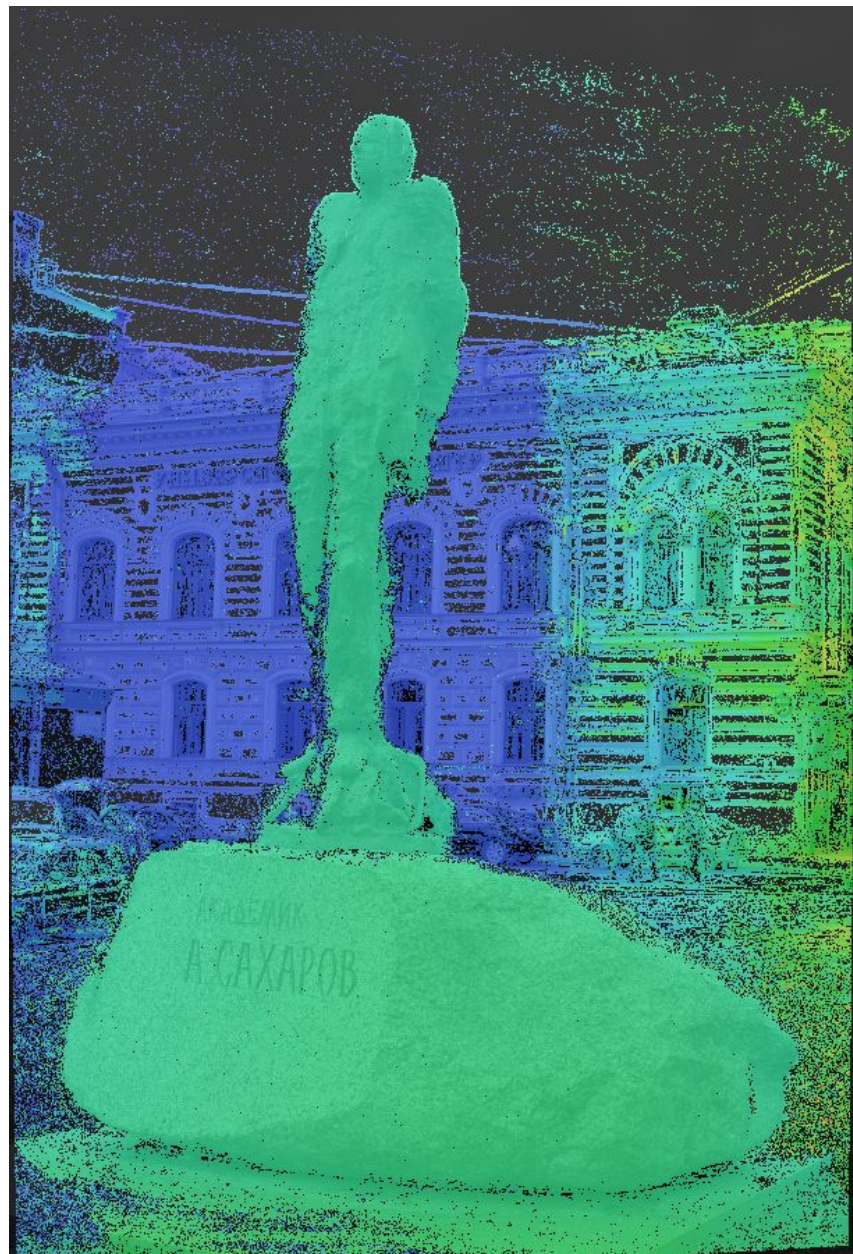
- **Чистить карты глубины от шума на уровне каждой карты глубины (локально)**
- Чистить карты глубины на базе их противоречий (глобально)
- Объединять эти карты глубины в одно общее плотное облако точек
- Строить полигональную 3D модель методом Пуассона

# Фильтрация шума в построенной по стереопаре карте глубины



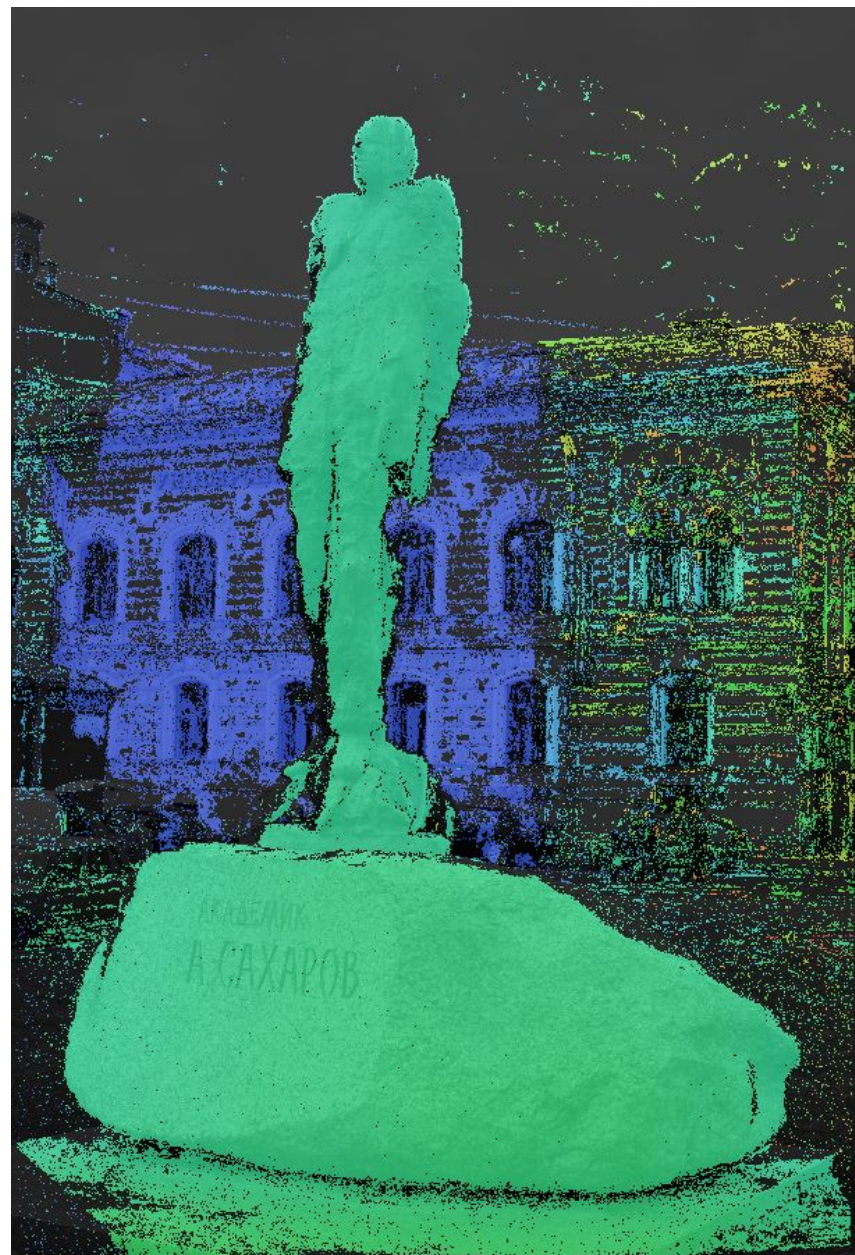


1. Удалим всех с  $\text{cost} > \text{threshold}$ :



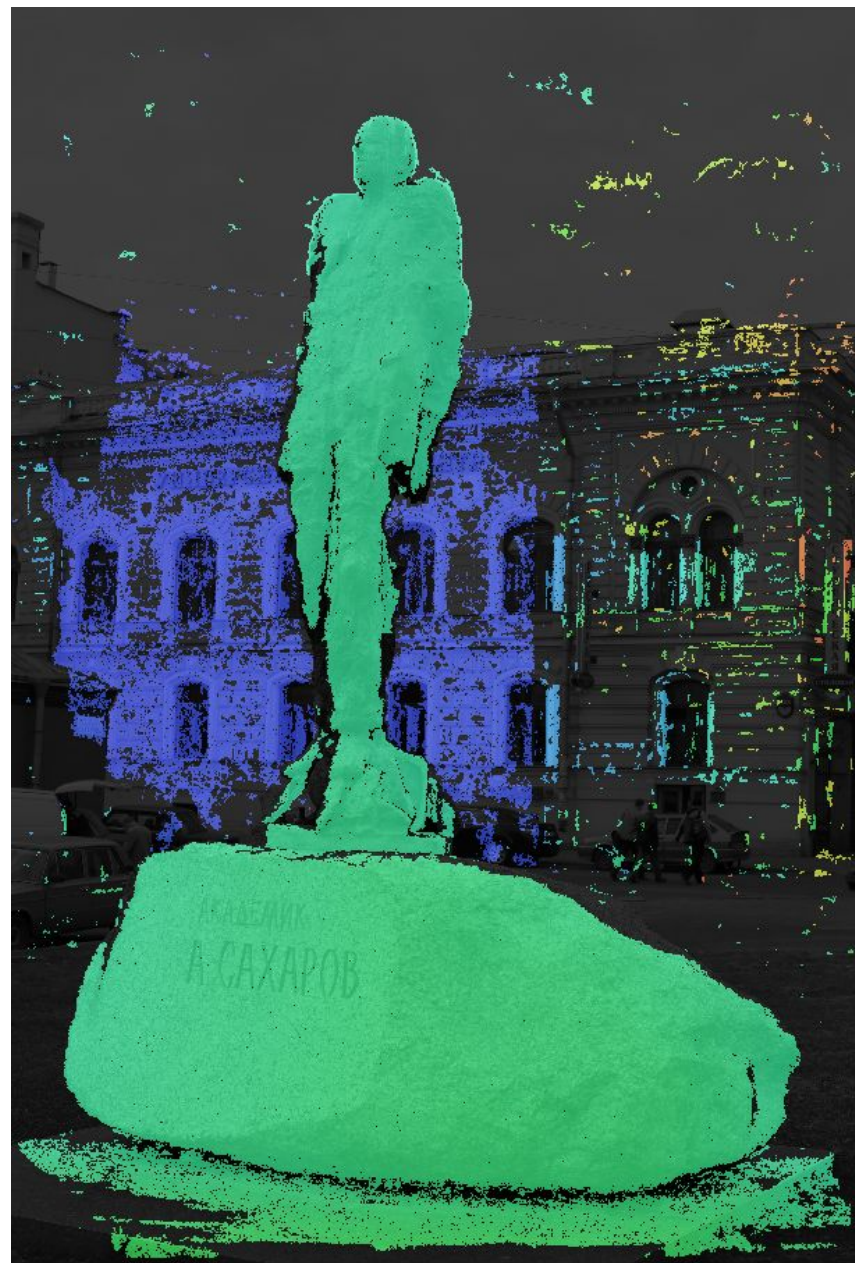


2. Удалим всех чья нормаль не согласована с глубинами вокруг:





### 3. Удалим всех чья компонента связности **маленького** размера:



Ссылки:

[Про CHM1](#)  
[Про CHM2](#)



### 3. Удалим всех чья компонента связности **среднего** размера:



Ссылки:

[Про CHM1](#)  
[Про CHM2](#)

## Фильтрация шума в построенной по стереопаре карте глубины

- Как можно проверить что нормаль не согласована? (два способа)



## Фильтрация шума в построенной по стереопаре карте глубины

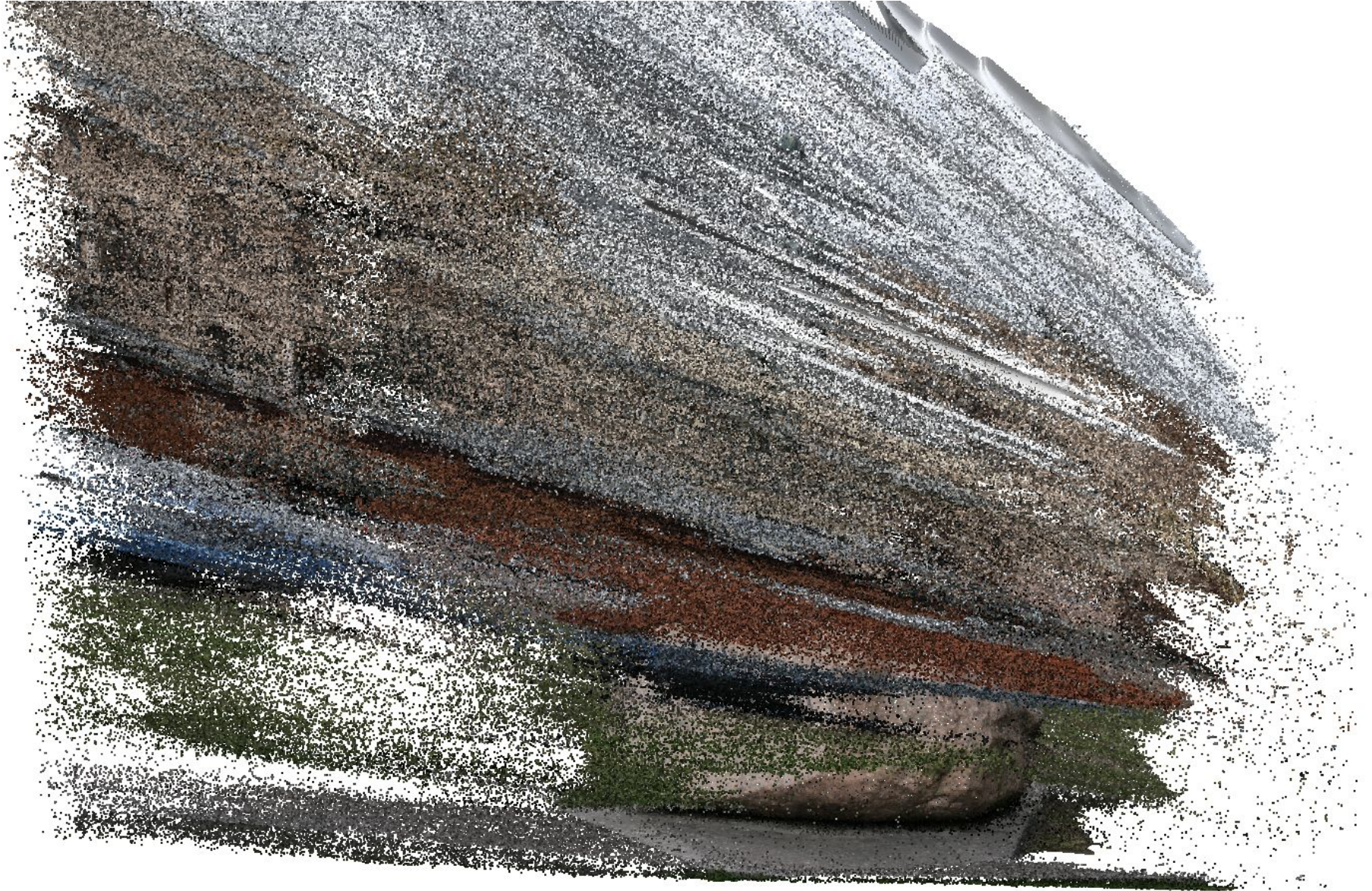
- Как можно проверить что нормаль не согласована? (два способа)
- Всегда ли если две компоненты связности смежны по пикселям - их надо объединять?

## Фильтрация шума в построенной по стереопаре карте глубины

- Как можно проверить что нормаль не согласована? (два способа)
- Всегда ли если две компоненты связности смежны по пикселям - их надо объединять?
- Как ускорить фильтрацию компонент связности на GPU?



0. Везде что-то да нашли:





1. Удалим всех с **cost** > **threshold**:





2. Удалим всех чья нормаль не согласована с глубинами вокруг:



3. Удалим всех чья компонента связности **маленького** размера:





3. Удалим всех чья компонента связности **среднего** размера:



## Где мы сейчас?

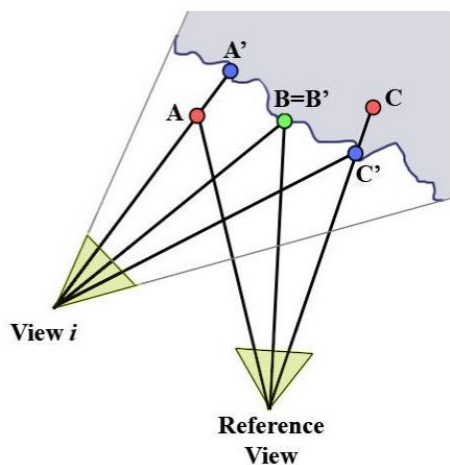
- Есть разреженное облако 3D положений ключевых точек
- Есть хорошо оптимизированные внутренние калибровки камер (intrinsics)
- Есть хорошо оптимизированные положения и ракурсы камер (extrinsics)
- Есть (шумные) карты глубины описывающие подробно и точно геометрию

## Что хотим дальше?

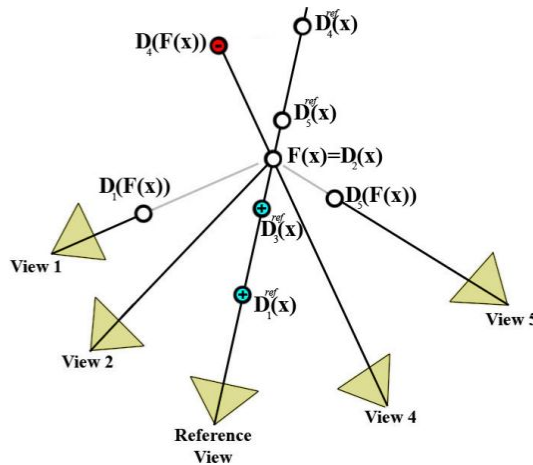
- Чистить карты глубины от шума на уровне каждой карты глубины (локально)
- **Чистить карты глубины на базе их противоречий (глобально)**
- Объединять эти карты глубины в одно общее плотное облако точек
- Строить полигональную 3D модель методом Пуассона



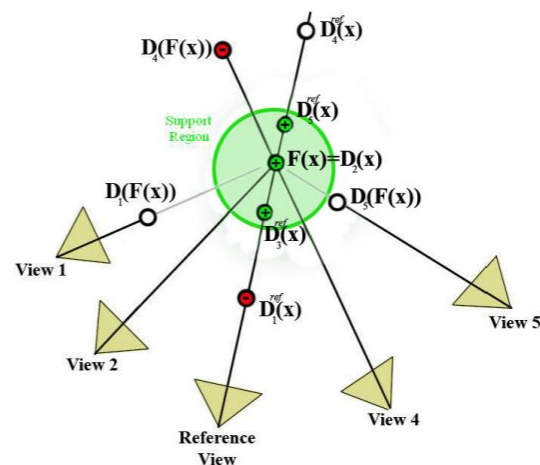
# Геометрическая согласованность карт глубины



(a) Visibility relations between points



(b) Stability calculation



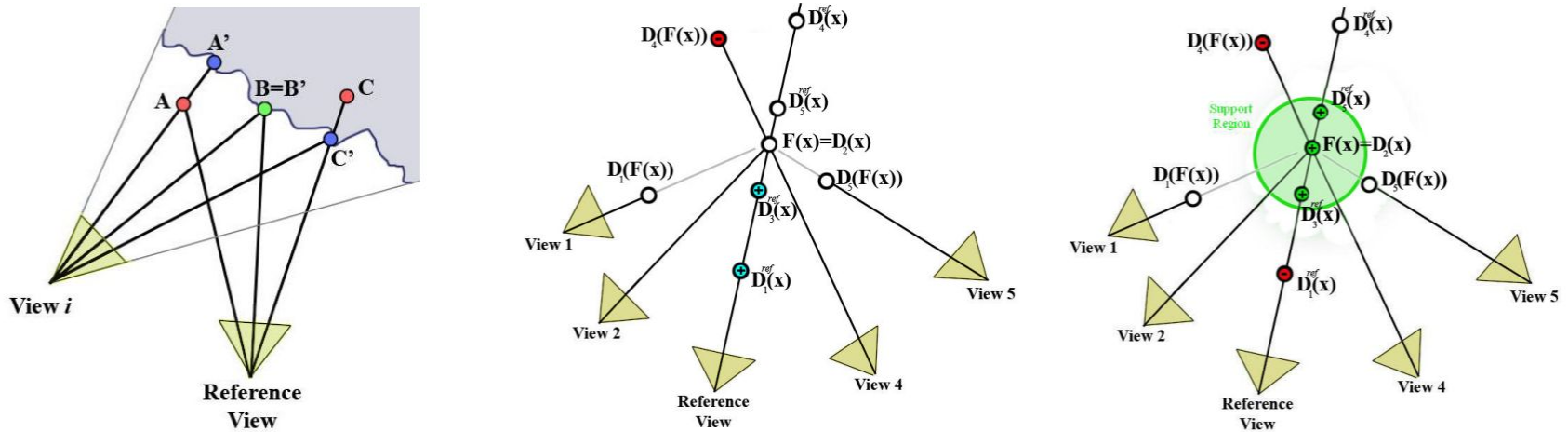
(c) Support estimation

Figure 2. (a) Visibility relations between points. The point  $A'$  seen in view  $i$  has its free space violated by  $A$  seen in the reference view.  $B'$  supports  $B$ .  $C$  seen in the reference view is occluded by  $C'$ . (b) Stability Calculation. In this example, there are two occlusions which raise stability and one free-space violations which lowers it. The stability is +1. (c) Support calculation. Three measurements are close to the current estimate and add support to it. Outside the support region, there is one occlusion and one free-space violation which lower the support.

The stability of a point  $S(x)$  is defined as the number of depth maps that occlude  $F(x)$  minus the number of free-space violations.

При каком значении стабильности точка хорошая и может считаться истинной глубиной на этом луче?

# Геометрическая согласованность карт глубины



(a) Visibility relations between points

(b) Stability calculation

(c) Support estimation

Figure 2. (a) Visibility relations between points. The point  $A'$  seen in view  $i$  has its free space violated by  $A$  seen in the reference view.  $B'$  supports  $B$ .  $C$  seen in the reference view is occluded by  $C'$ . (b) Stability Calculation. In this example, there are two occlusions which raise stability and one free-space violations which lowers it. The stability is +1. (c) Support calculation. Three measurements are close to the current estimate and add support to it. Outside the support region, there is one occlusion and one free-space violation which lower the support.

The stability of a point  $S(x)$  is defined as the number of depth maps that occlude  $F(x)$  minus the number of free-space violations.

The final fused depth is selected to be the closest depth to the camera for which stability is non-negative. This depth is balanced in the sense that the amount of evidence that indicates it is too close is equal to the amount of evidence that indicates it is too far away.

[Real-Time Visibility-Based Fusion of Depth Maps, Merrell et. al., 2007](#)



# Проблемный пример



# Проблемный пример





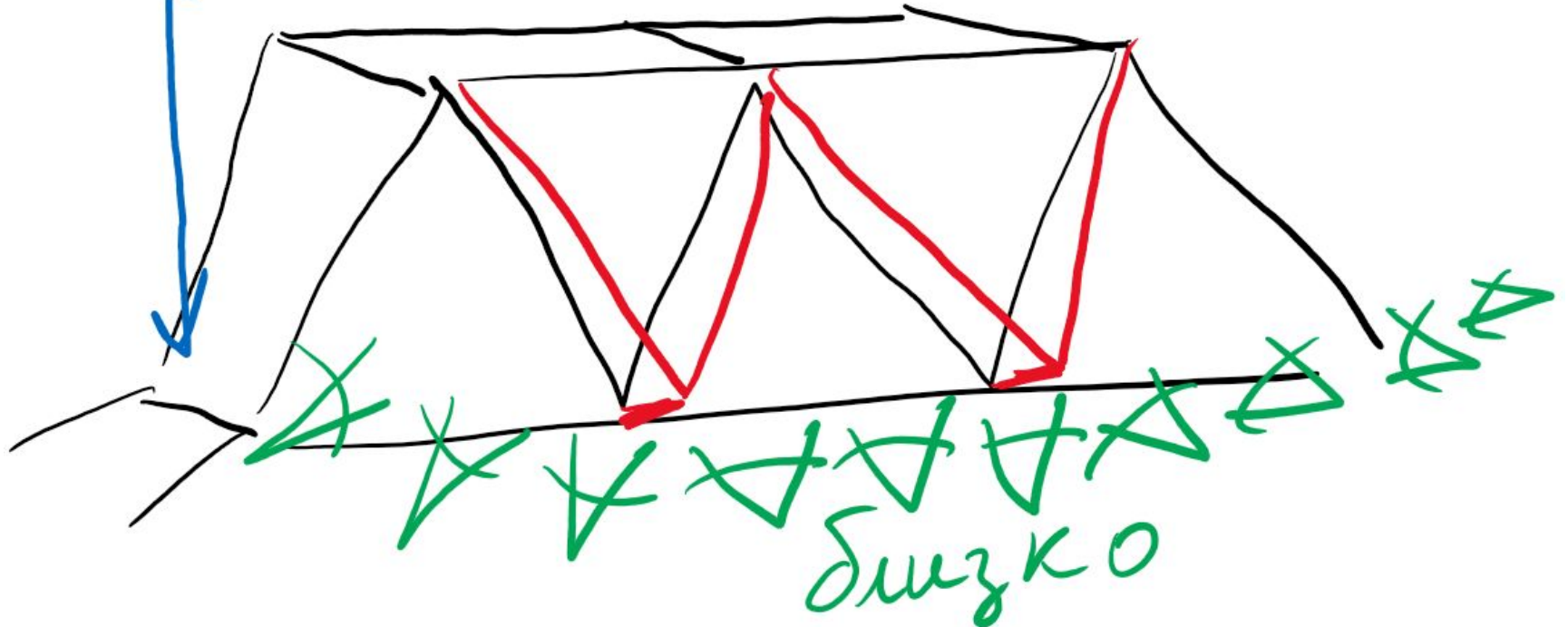
# Проблемный пример



Как исследовать проблему? Куда смотреть?

А А А А А А

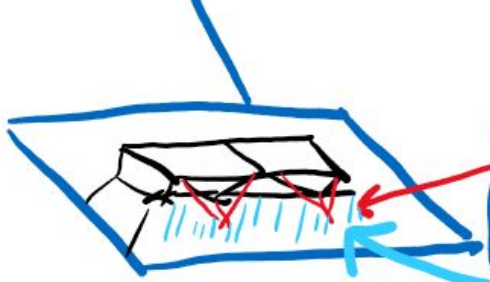
галеко





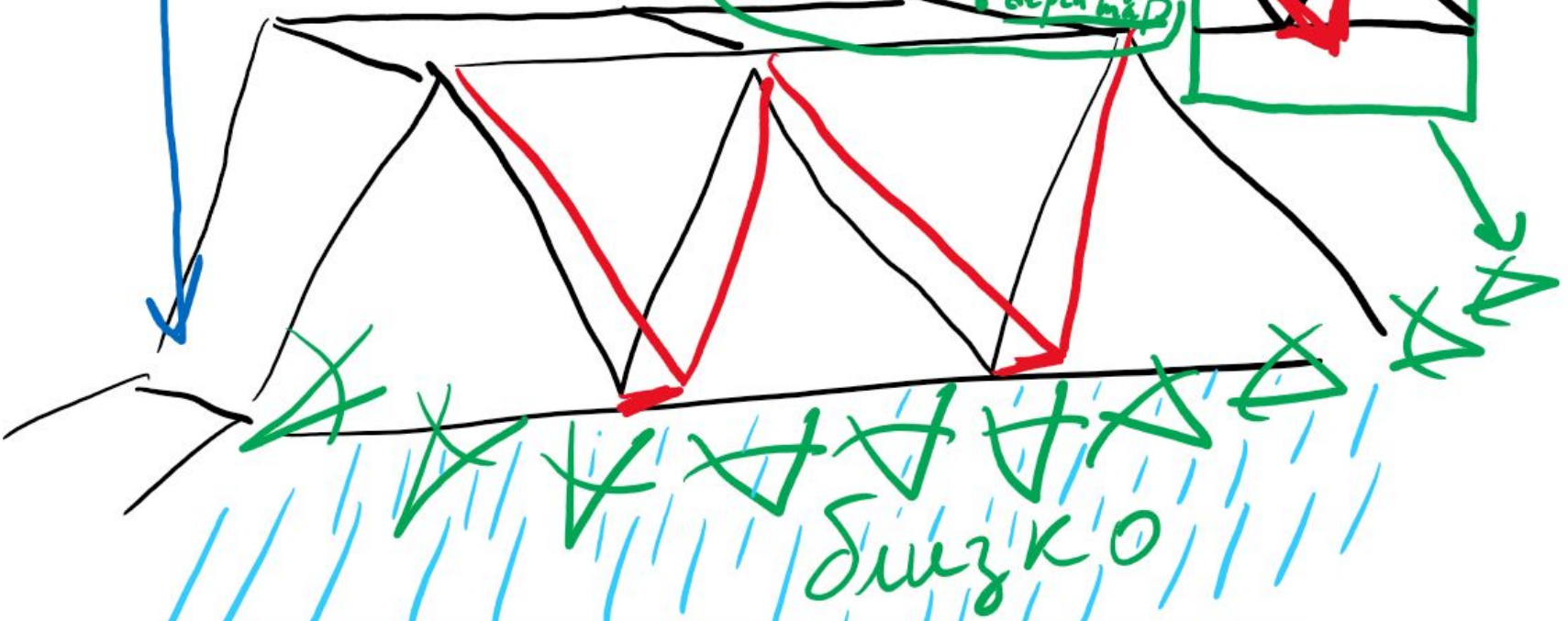
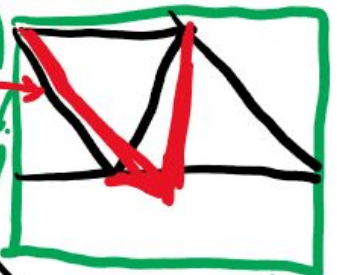
далеко

А А А А А



орень молчане  
ремья реконструкция  
уполнать & depth map

орень молчане  
учебно реконструкция  
& depth map



близко

## Где мы сейчас?

- Есть разреженное облако 3D положений ключевых точек
- Есть хорошо оптимизированные внутренние калибровки камер (intrinsics)
- Есть хорошо оптимизированные положения и ракурсы камер (extrinsics)
- Есть (шумные) карты глубины описывающие подробно и точно геометрию

## Что хотим дальше?

- Чистить карты глубины от шума на уровне каждой карты глубины (локально)
- Чистить карты глубины на базе их противоречий (глобально)
- **Объединять эти карты глубины в одно общее плотное облако точек**
- Строить полигональную 3D модель методом Пуассона



## Где мы сейчас?

- Есть разреженное облако 3D положений ключевых точек
- Есть хорошо оптимизированные внутренние калибровки камер (intrinsics)
- Есть хорошо оптимизированные положения и ракурсы камер (extrinsics)
- Есть (шумные) карты глубины описывающие подробно и точно геометрию

## Что хотим дальше?

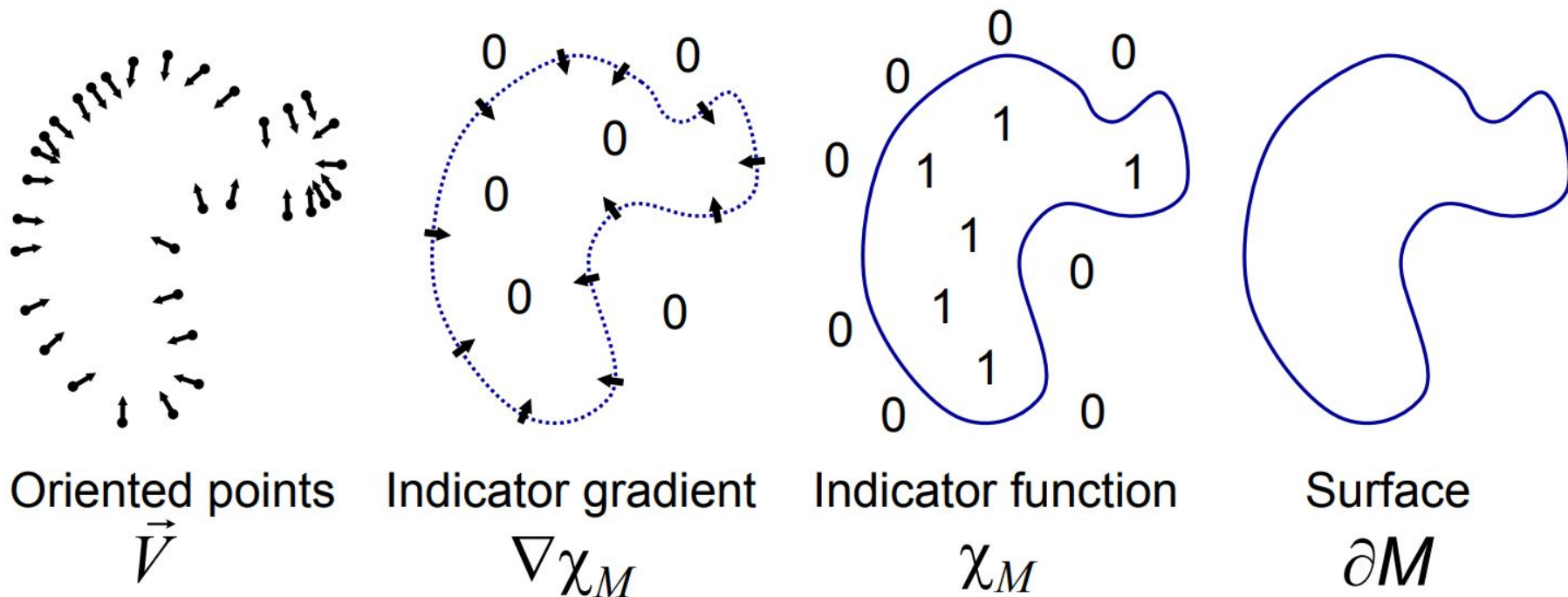
- Чистить карты глубины от шума на уровне каждой карты глубины (локально)
- Чистить карты глубины на базе их противоречий (глобально)
- Объединять эти карты глубины в одно общее плотное облако точек
- **Строить полигональную 3D модель методом Пуассона**

# Poisson surface reconstruction

На входе  $\vec{V}$  - точки с инвертированными нормальными векторами.

Находим индикатор  $\chi$  - скалярное поле, чей градиент приближает векторное поле норма  $\min_{\chi} \|\nabla\chi - \vec{V}\|$

Результирующая поверхность  $\partial M$  - изоповерхность в поле индикатора.



**Figure 1:** *Intuitive illustration of Poisson reconstruction in 2D.*



# Сведение к уравнению Пуассона

Итого хочется найти такой индикатор  $\chi$  чтобы:

$$\nabla \tilde{\chi} = \vec{V}$$

Явного решения нет, но можно приближенно решить минимизируя квадратичную ошибку оператором дивергенции сведя к уравнению Пуассона:

$$\Delta \chi \equiv \nabla \cdot \nabla \chi = \nabla \cdot \vec{V}$$

$$\Delta \tilde{\chi} = \nabla \cdot \vec{V}$$

# Дискретизация

Регулярная решетка - кубическая память.

На самом деле интересно лишь пространство около поверхности. Поэтому можно использовать адаптивное октодерево.

В каждом узле октодерева с центром  $o.c$  и шириной  $o.w$  находится базовая функция  $F : \mathbb{R}^3 \rightarrow \mathbb{R}$

$$F_o(q) \equiv F \left( \frac{q - o.c}{o.w} \right) \frac{1}{o.w^3}$$

Т.е. каждая точка делает вклад в базовую функцию своего узла.



# Решение

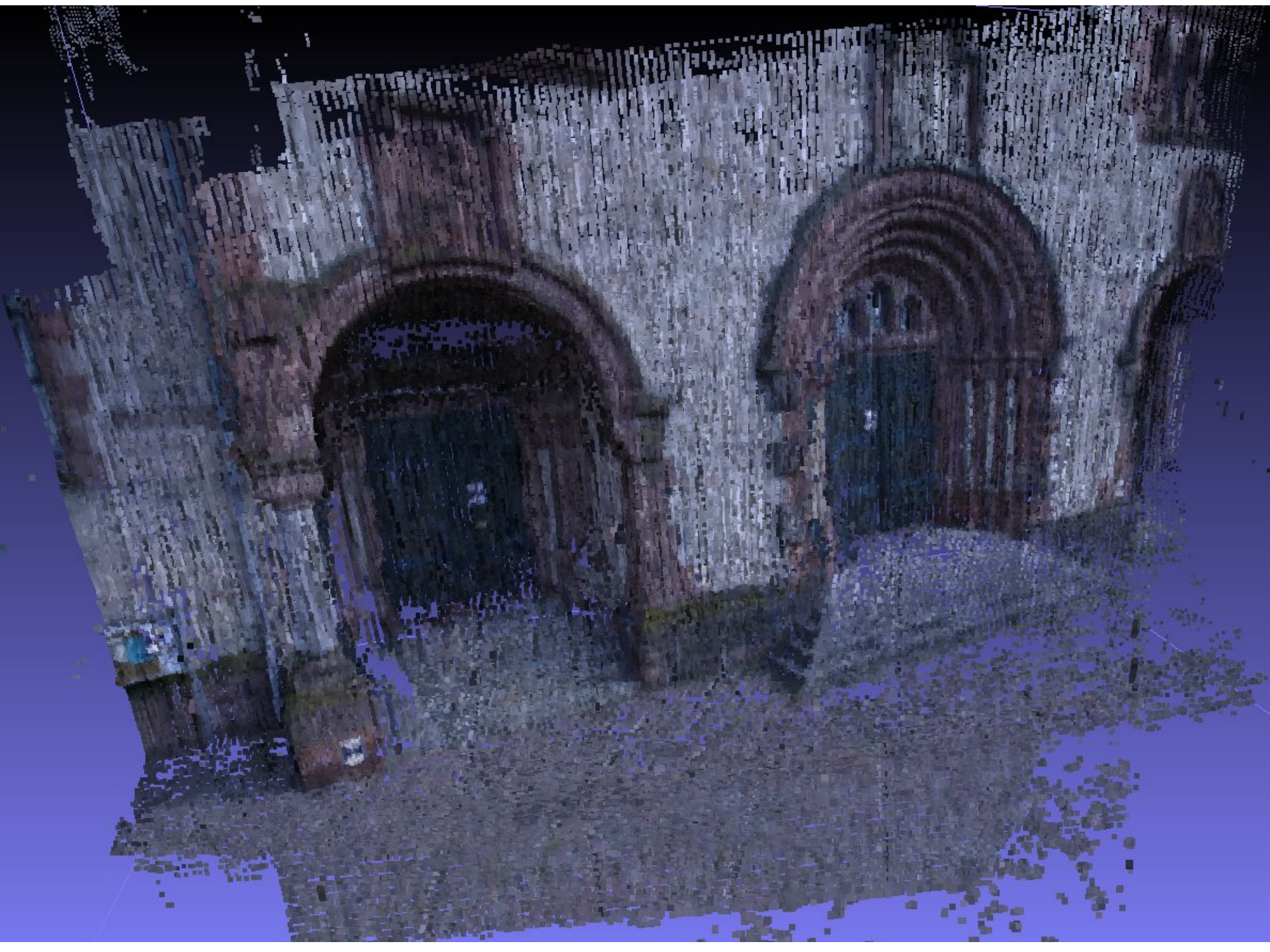
Задача сводится к:

$$\min_{x \in \mathbb{R}^{|\mathcal{O}|}} \|Lx - v\|^2$$

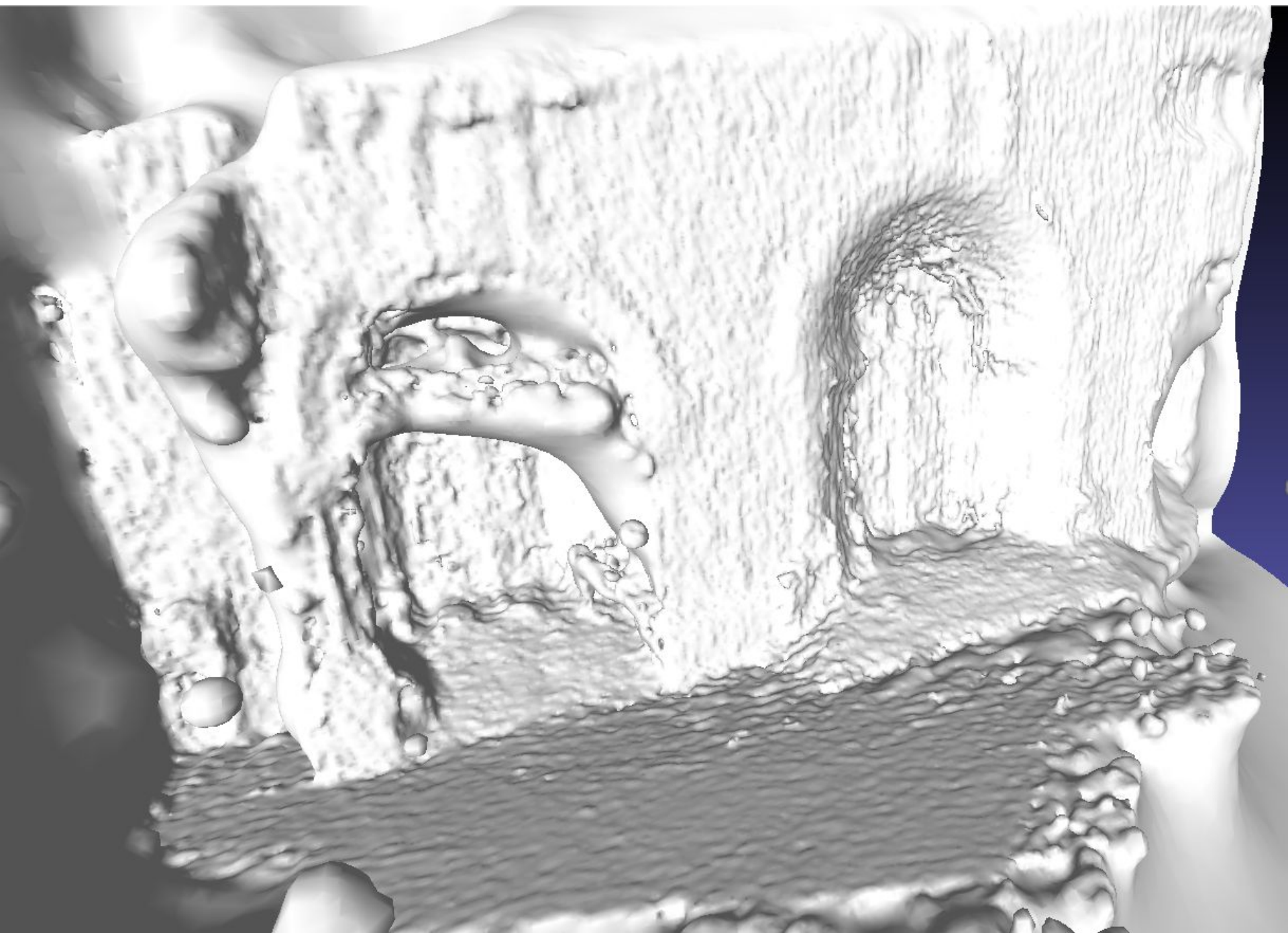
Где  $L$  - разреженная, симметричная матрица размера  $|\mathcal{O}| \times |\mathcal{O}|$  (квадрат размера октодеревя).

Решается методом сопряженных градиентов.

Число столбцов - число соседних узлов чья базисная функция пересекается с узлом текущего ряда. Поэтому  $|\mathcal{O}| \times 125$





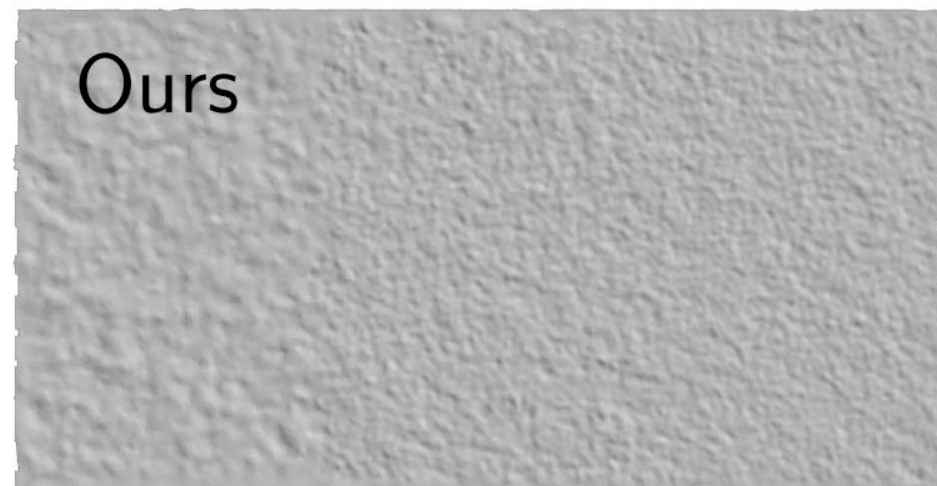
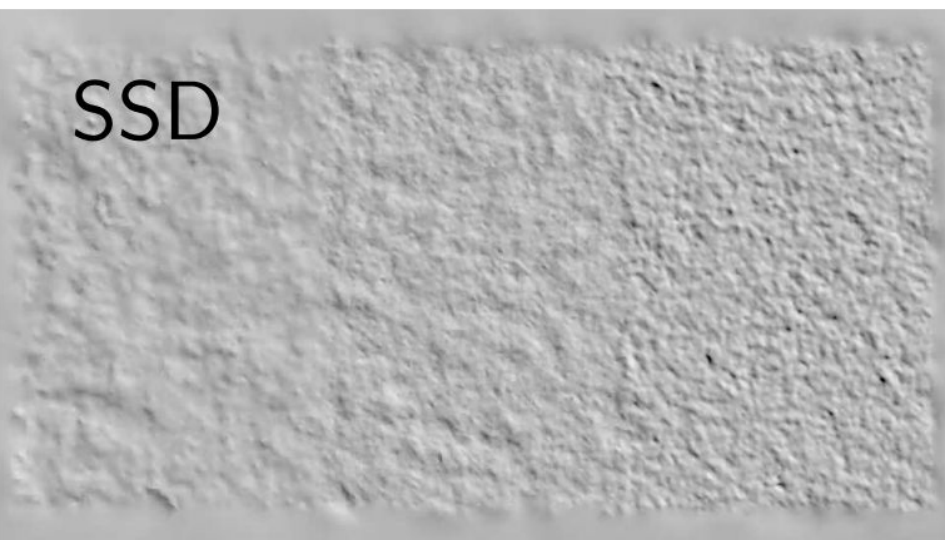
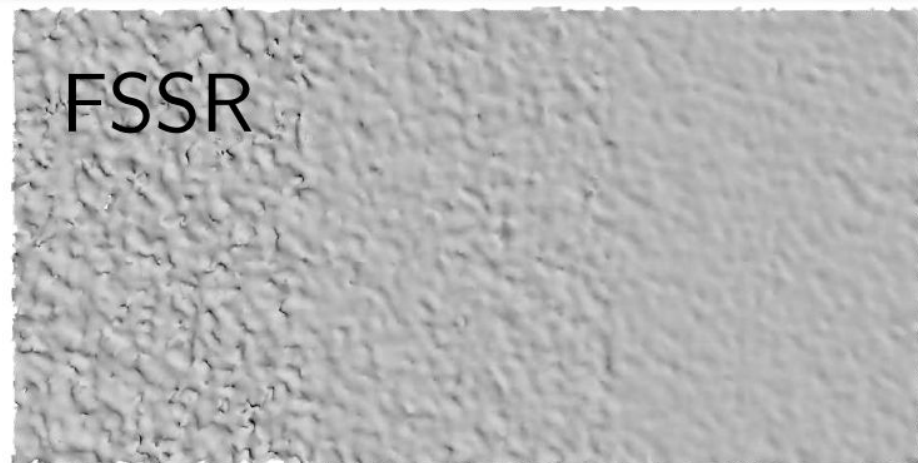
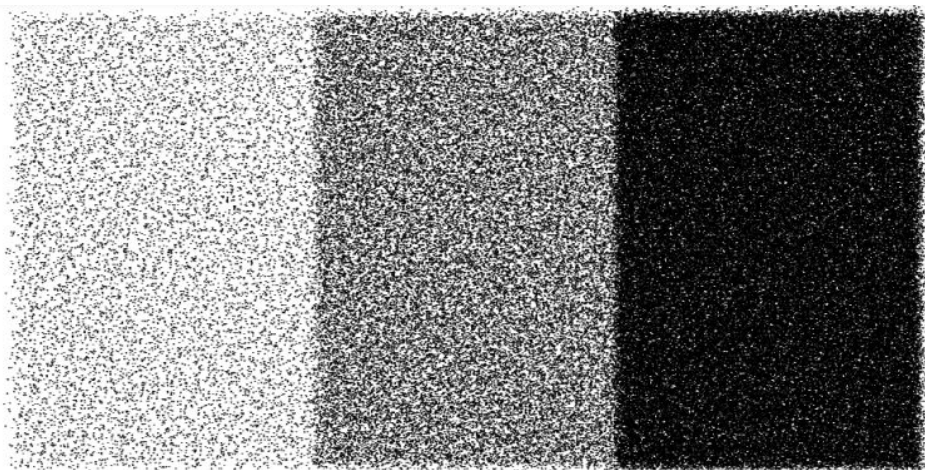


# Адаптивность к детализации

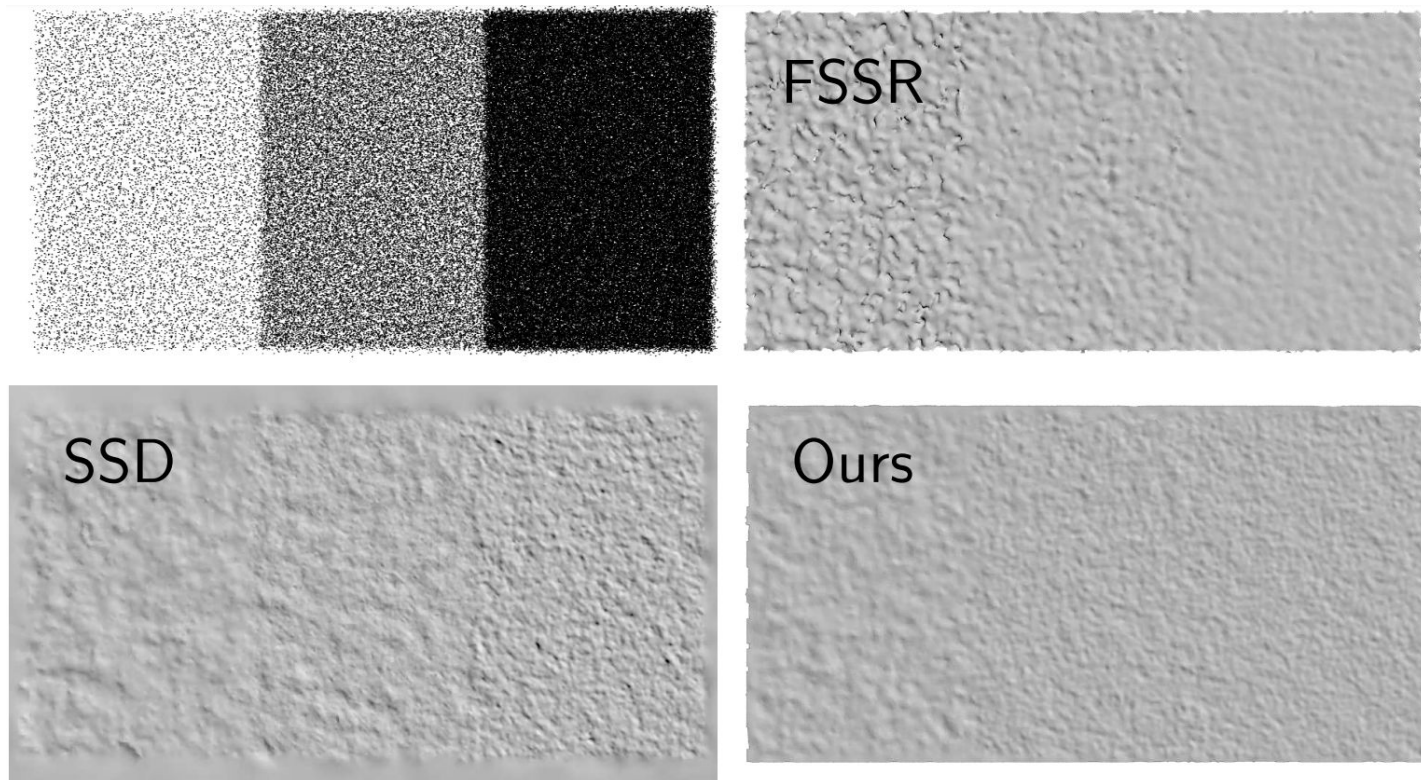
Как решить насколько маленький кубик нужен для дискретизации каждой точки в адаптивном октодереве?



# Адаптивность к детализации



# Адаптивность к детализации



**Figure 11.** Reconstruction of a plane with three regions, each with a different uniform point density. The scale value assigned to the points corresponds to the sampling density of the central region. Gaussian white noise was added to the points' position and normal. **Top left:** Input point cloud. **FSSR:** With increasing density, FSSR effectively cancels out noise. In the low density region it suffers from a too sparse sampling. **SSD:** SSD adapts the scale to the point density and therefore models the noise in the high density region. In the low density region noise is suppressed by using a coarser scale for reconstruction. **Ours:** Our reconstruction looks more even in the high density region than that of the other methods. The high density leads to a strong data term but is also effective to cancel out noise. In the low density region the smoothness term dominates and the reconstruction looks smoother than with SSD.



# ССЫЛКИ

## Depth Maps filtering:

- [Real-Time Visibility-Based Fusion of Depth Maps, Merrell et. al., 2007](#)

## Poisson:

- [Poisson Surface Reconstruction, Kazhdan et. al., 2006](#)  
[+ github.com/PoissonRecon](#)
- [Multilevel Streaming for Out-of-Core Surface Reconstruction, Bolitho et. al., 2007](#)  
[+ Code/StreamingRecon\\_918.zip](#)

## Poisson + GPGPU:

- [Data-Parallel Octrees for Surface Reconstruction, Zhou et al., 2011](#)
- [NV Forum: YouTube: Introduction to CUDA & GPU Poisson Surface Reconstruction](#)
- [YouTube: Introduction to CUDA & GPU Poisson Surface Reconstruction](#)

# Вопросы?



Полярный Николай  
[polarnick239@gmail.com](mailto:polarnick239@gmail.com)