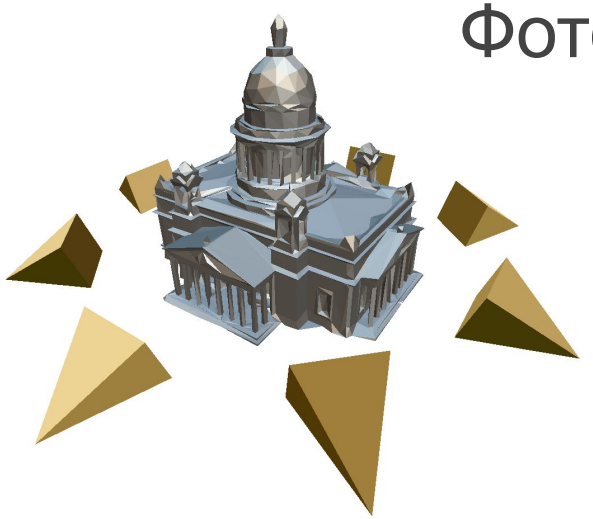
Реконструкция 3D модели из карт глубины. (Delaunay + Min-cut)

Фотограмметрия. Лекция 9



- **Delaunay Triangulation**
- **Graph Max-Flow = Min-Cut**
- **3D model reconstruction**

Полярный Николай
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

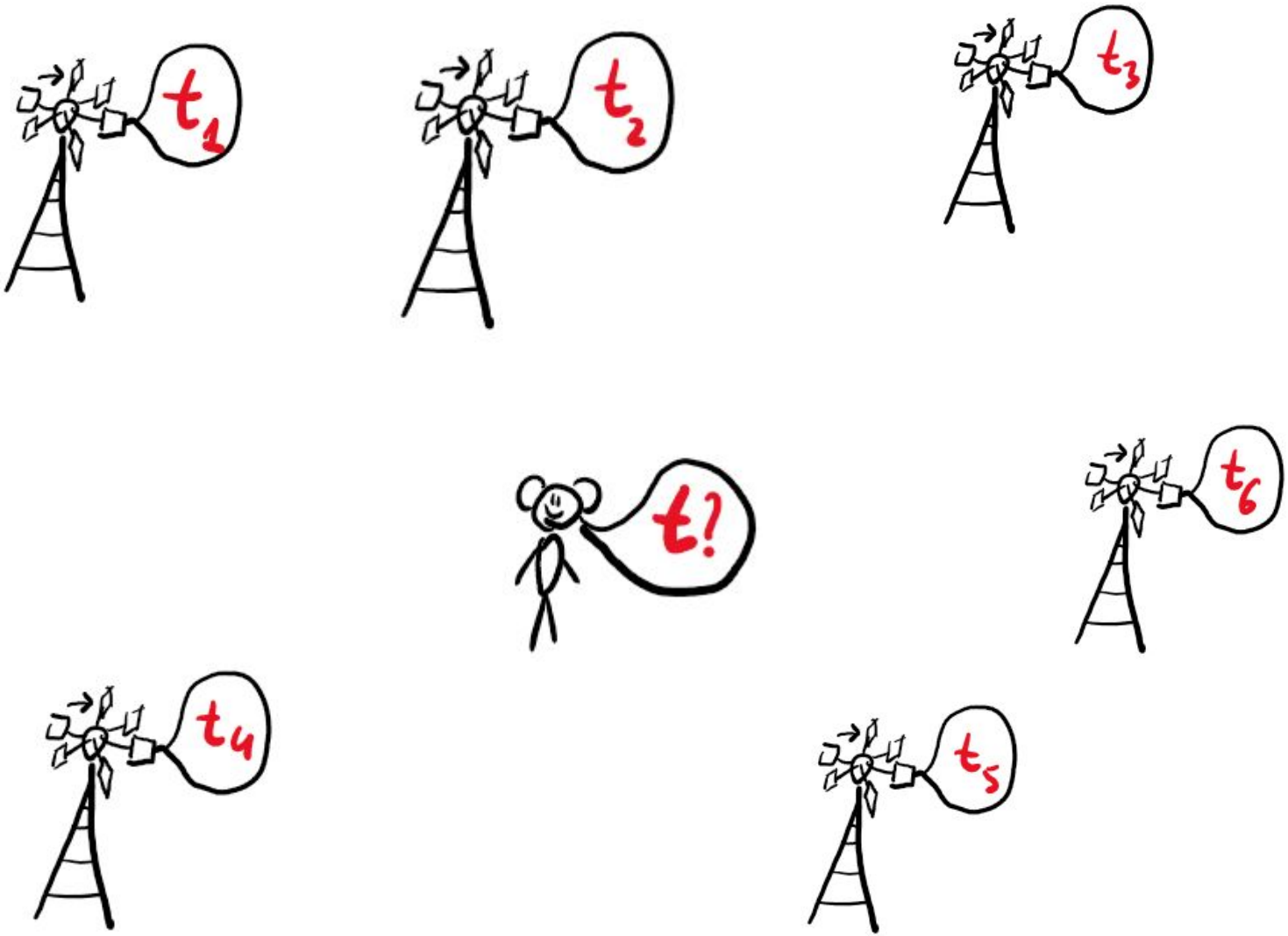
Где мы сейчас?

- 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

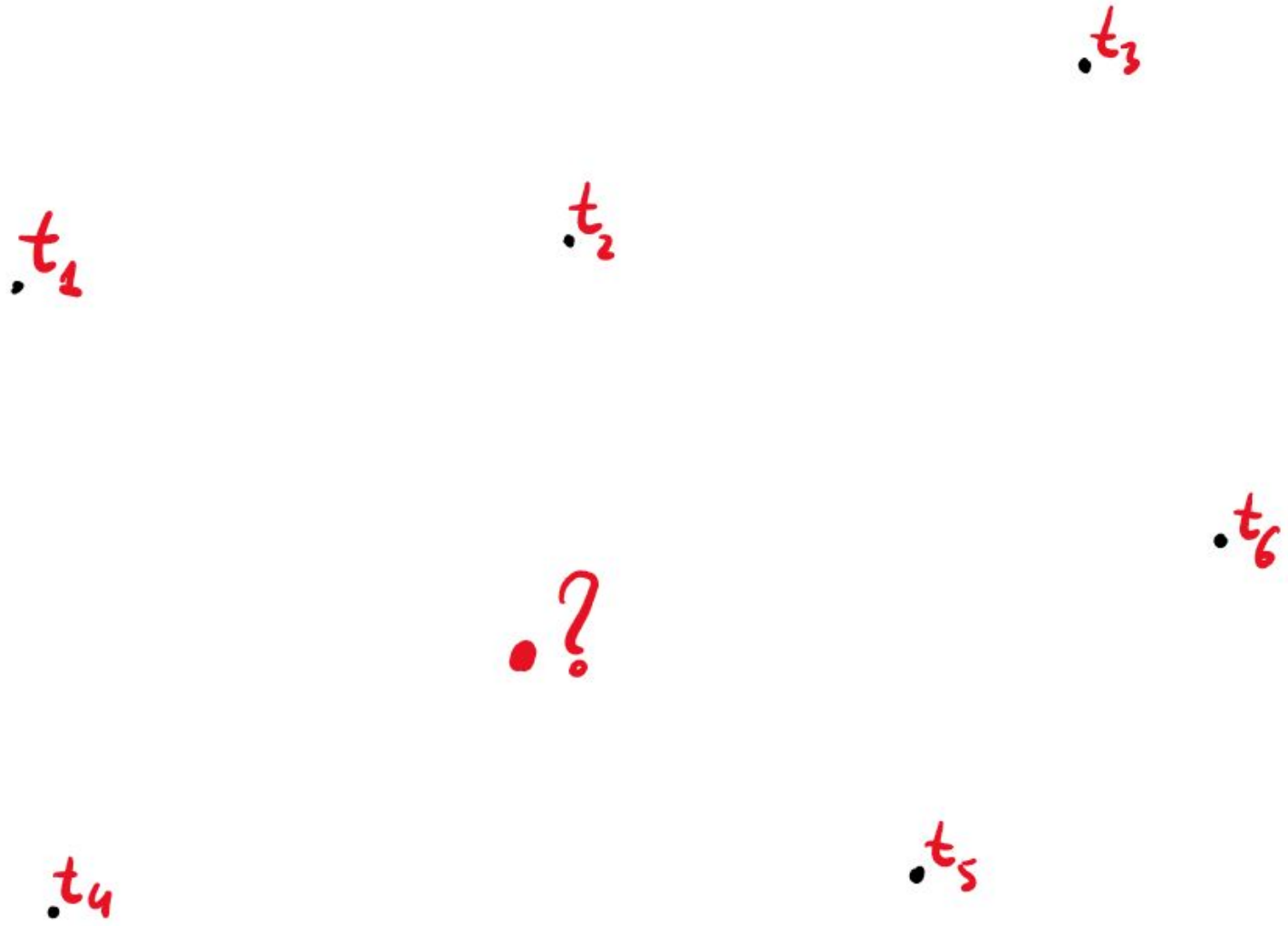
Что хотим дальше?

- 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 **напрямую из карт глубины**

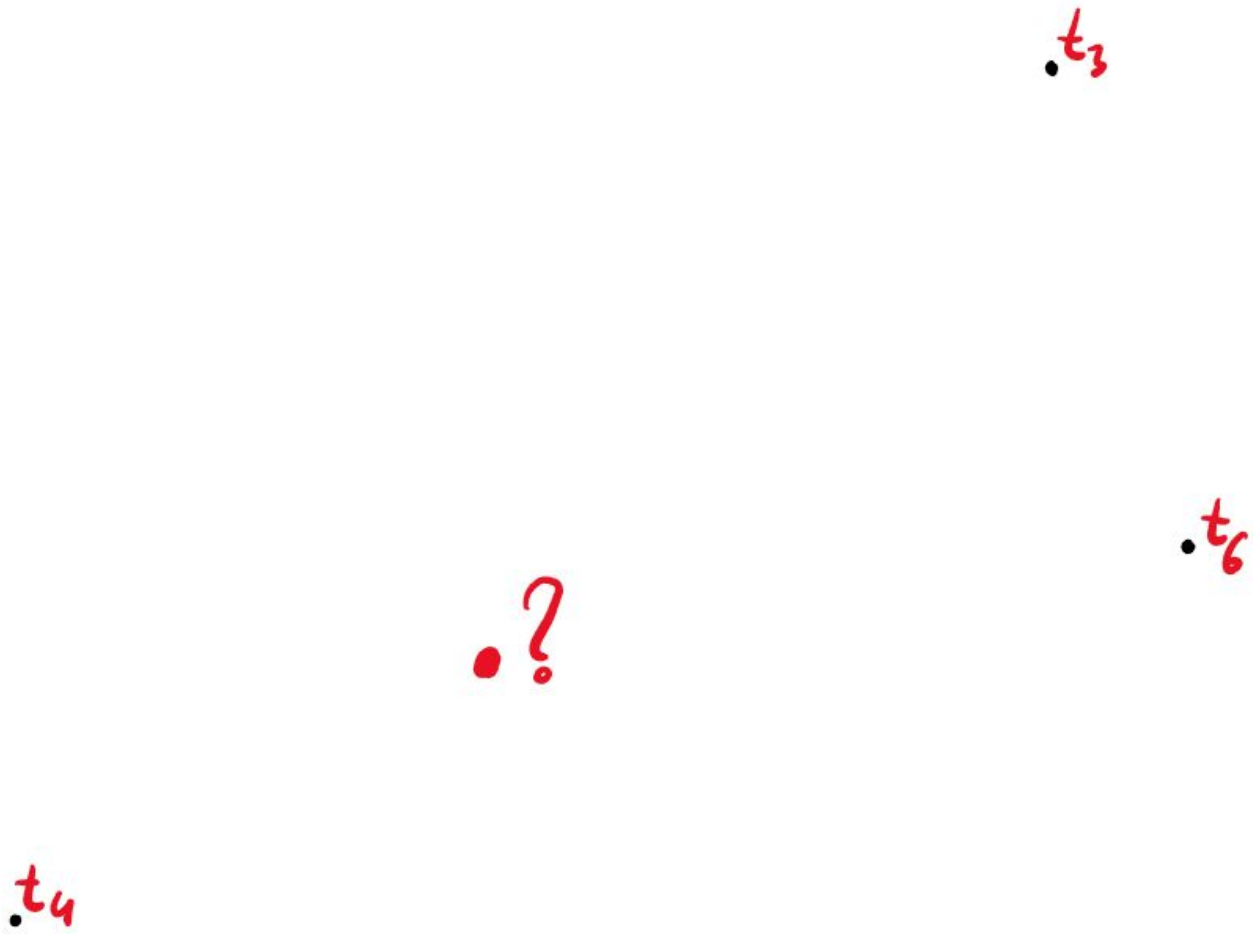
Есть метеовышки, как интерполировать температуру?



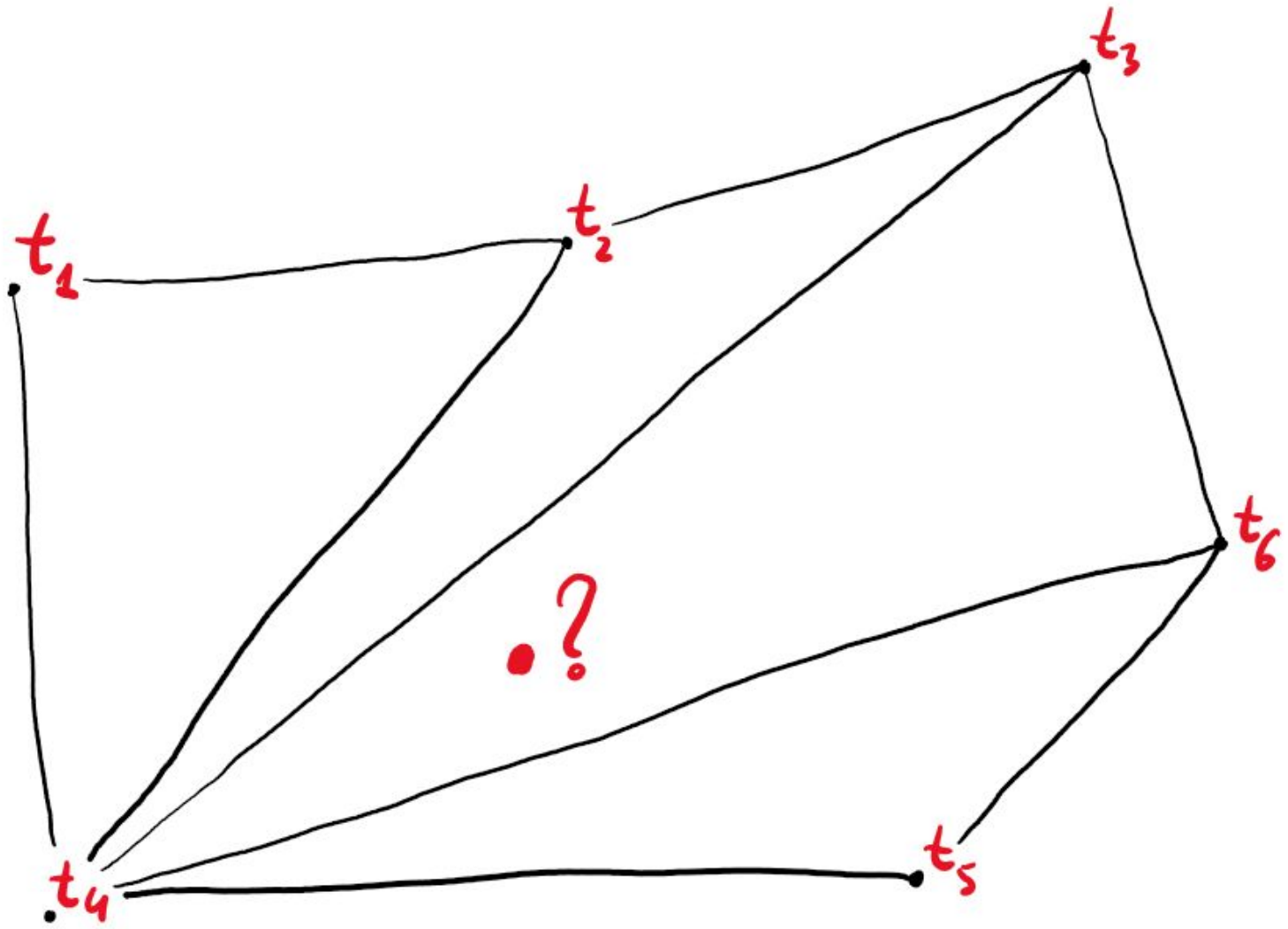
Есть метеовышки, как интерполировать температуру?



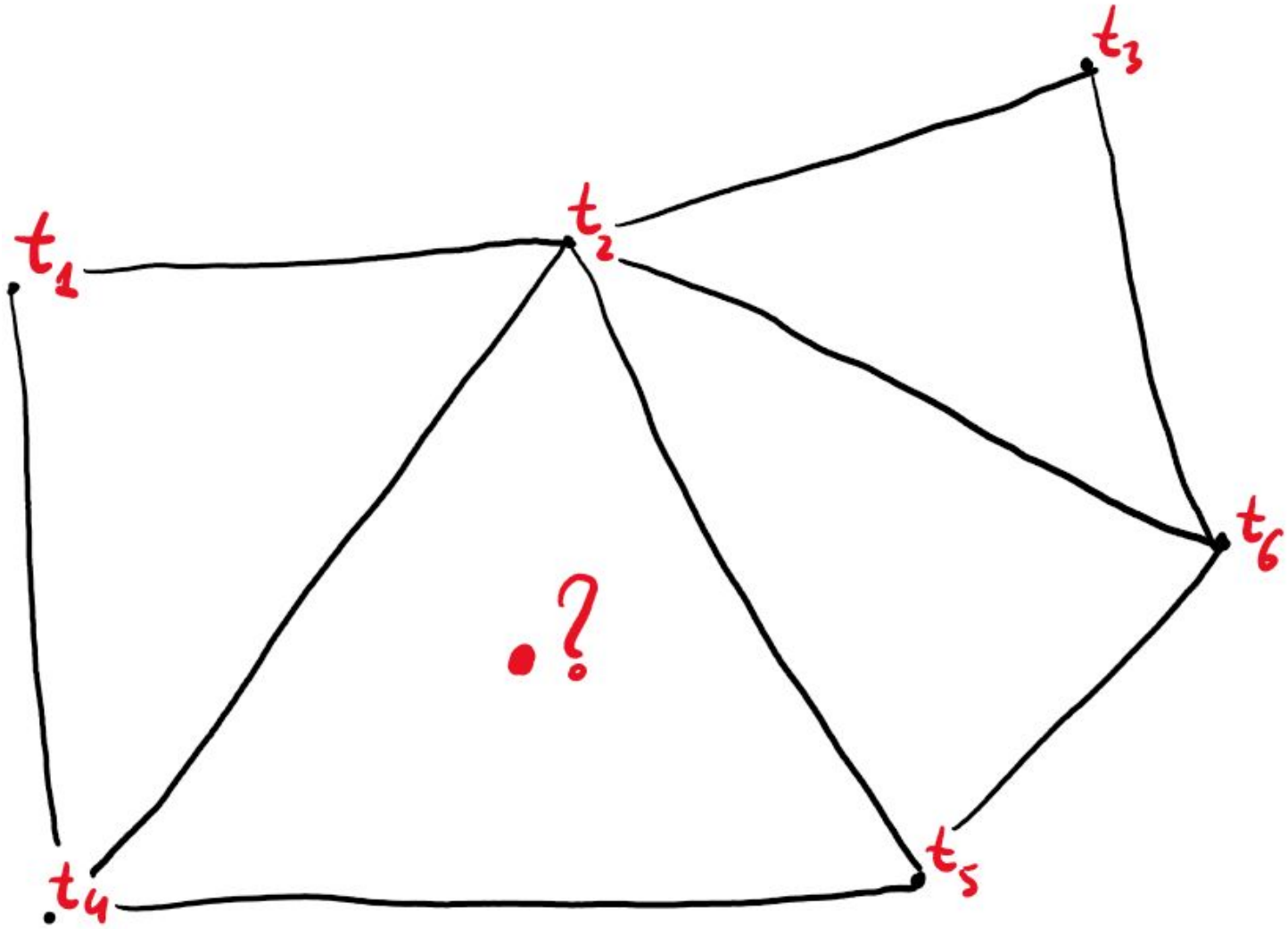
А что если есть всего три метеовышки?



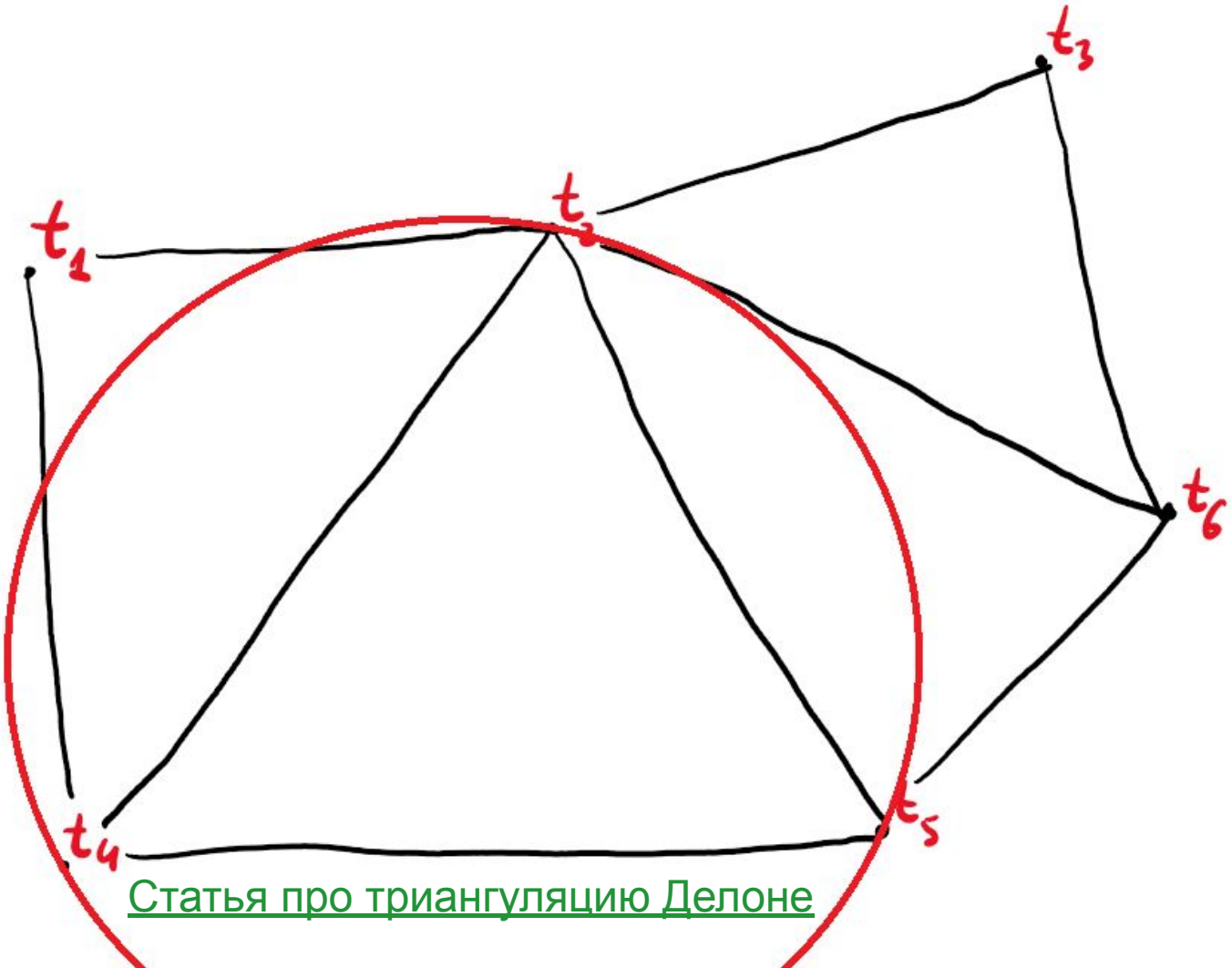
Есть метеовышки, как интерполировать температуру?



Есть метеовышки, как интерполировать температуру?



Триангуляция Делоне/критерий:
внутри каждой описанной окружности нет точек



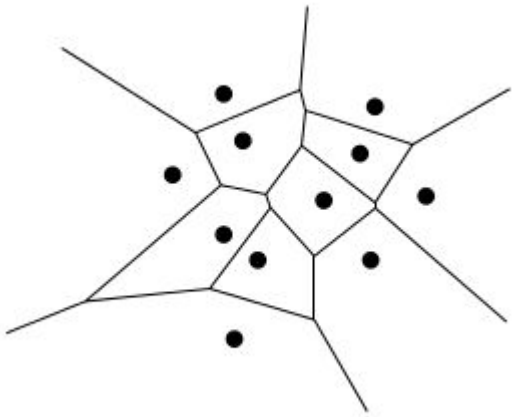
Статья про триангуляцию Делоне

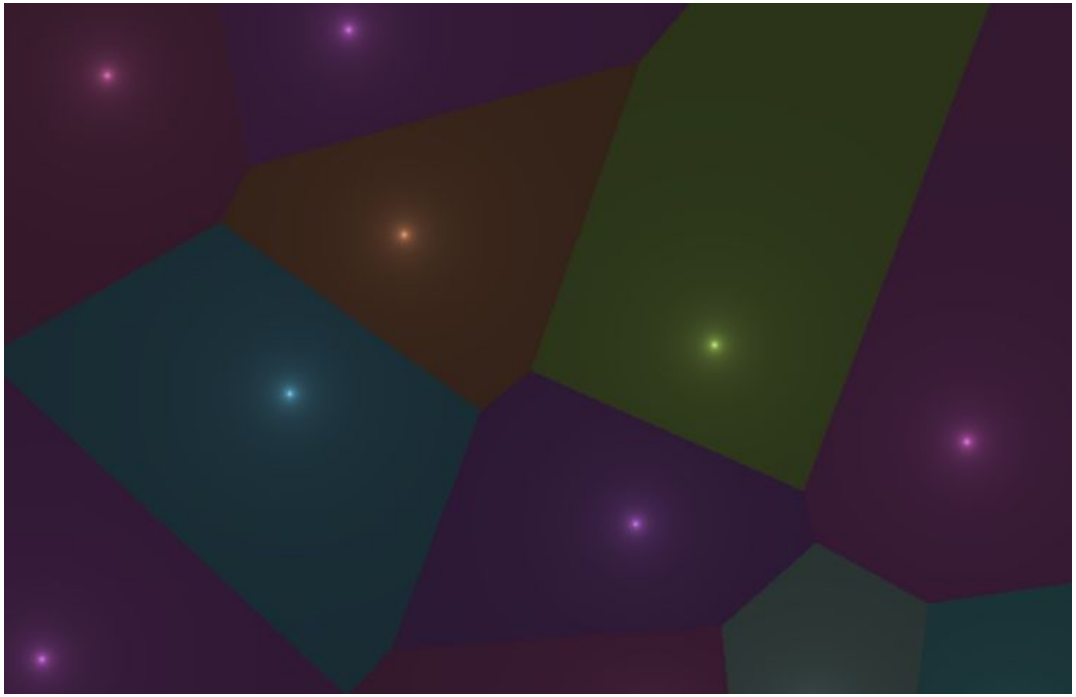
Триангуляция Делоне/критерий:
внутри каждой описанной окружности нет точек

Статья про триангуляцию Делоне

Диаграмма Вороного

Есть точки p_i на плоскости. *i*-ая Ячейка диаграммы Вороного - такой кусок пространства, для любой точки которого ближайшей точкой изначального множества является p_i.

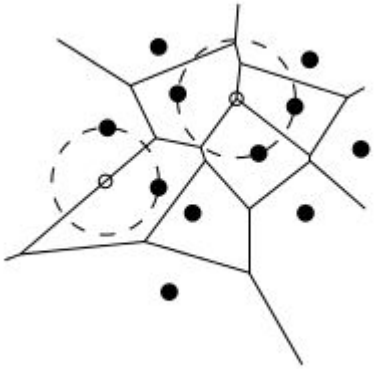


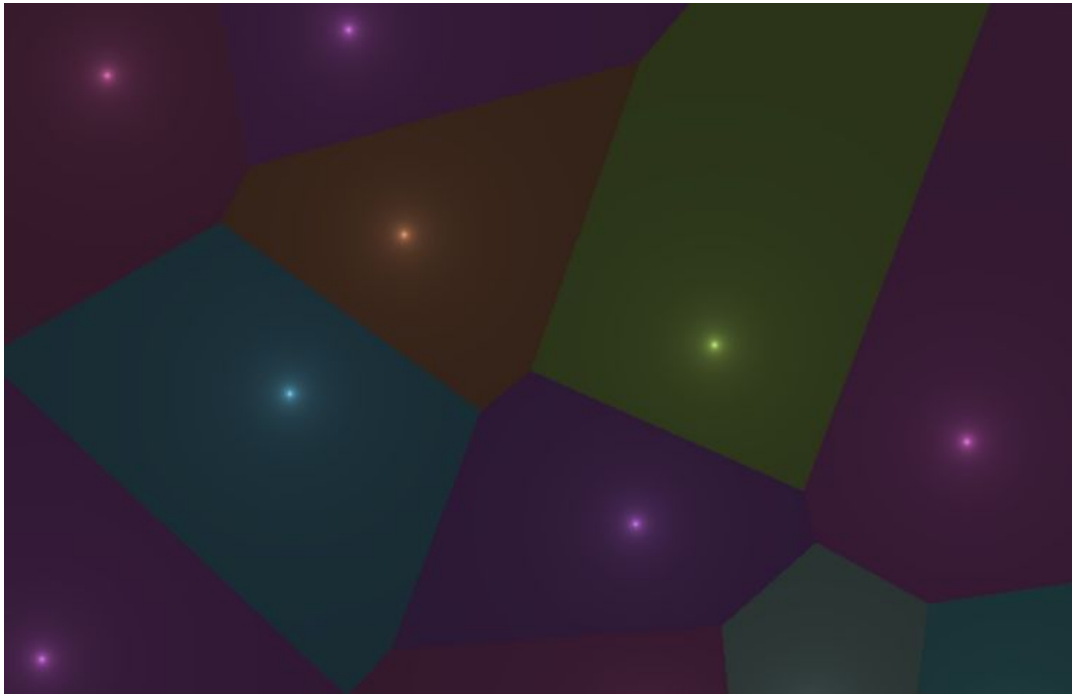


[Статья про диаграмму Вороного]

Диаграмма Вороного

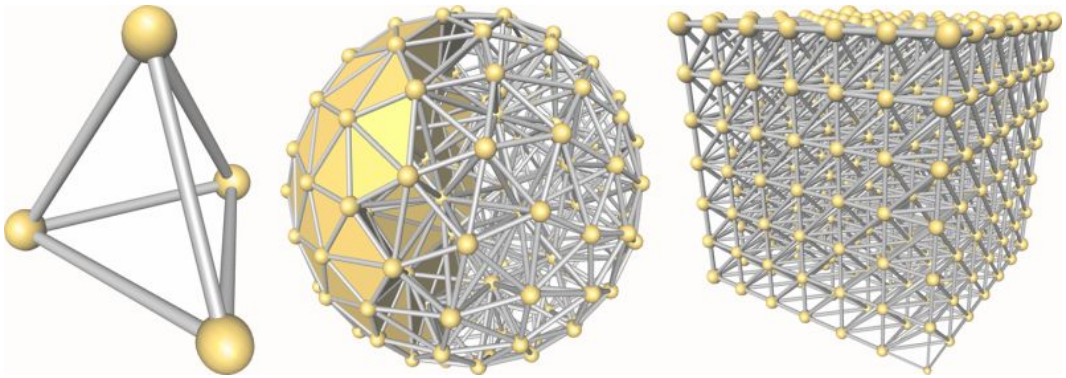
Есть точки p_i на плоскости. *i*-ая Ячейка диаграммы Вороного - такой кусок пространства, для любой точки которого ближайшей точкой изначального множества является p_i.

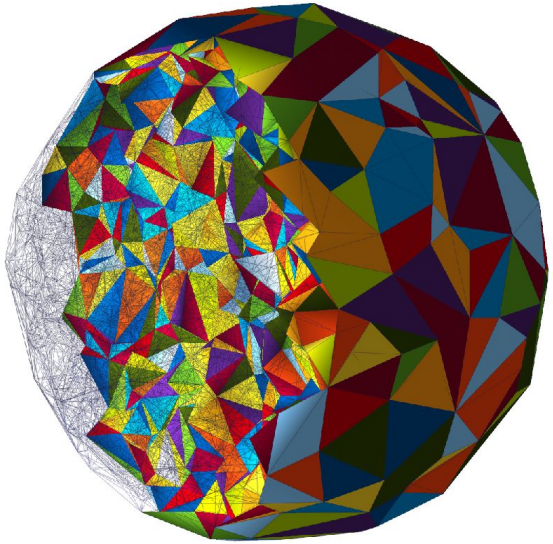




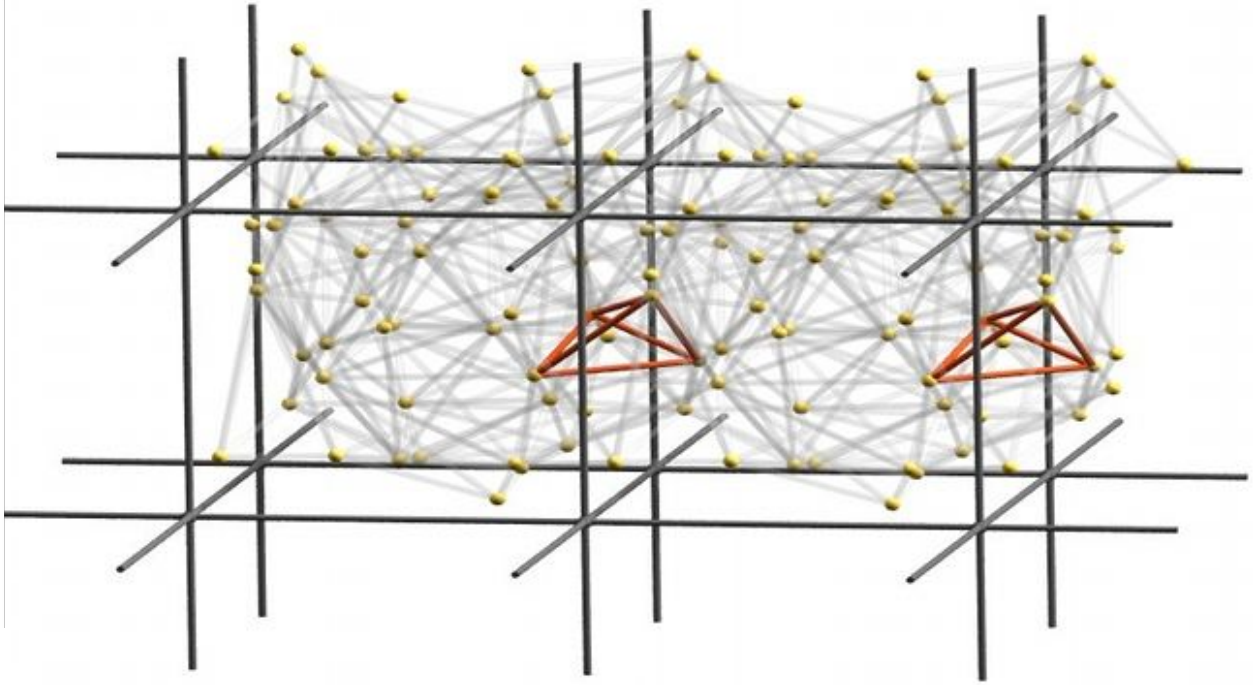
[Статья про диаграмму Вороного]

Триангуляция Делоне в 3D



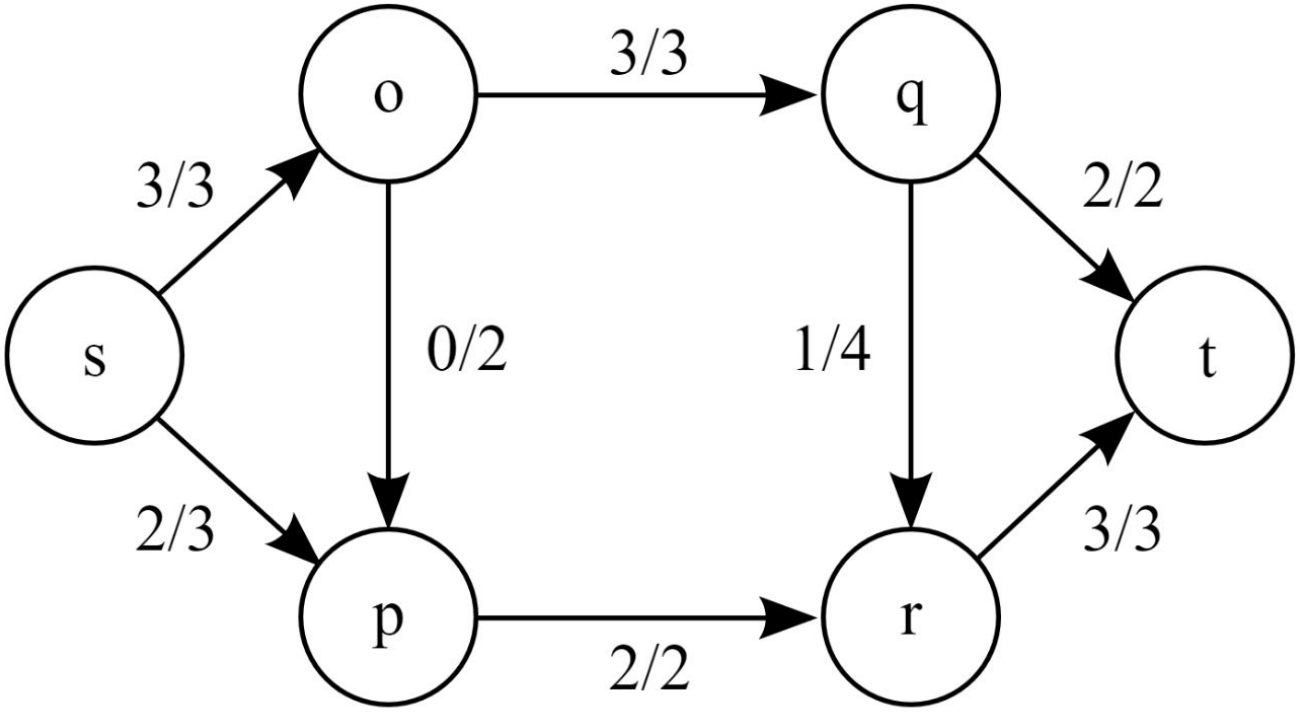


CGAL: 3D triangulation



Максимальный поток в графе (Graph Max-Flow)

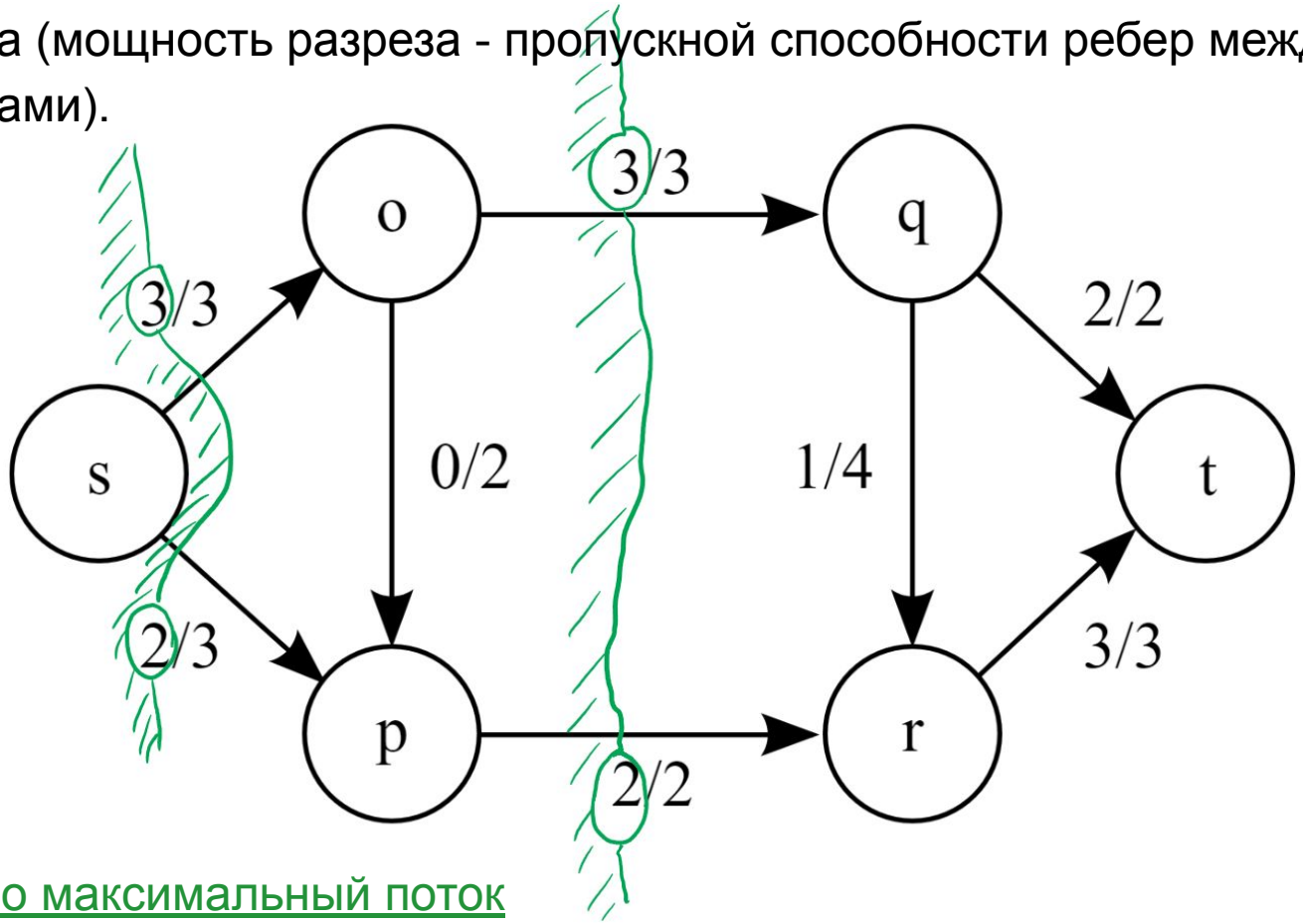
Есть ориентированный граф, на ребрах указана максимальная пропускная способность. Найти **максимальный поток** - максимальную пропускную способность сети из истока ***s*** в сток ***t***.



Статья про максимальный поток

Минимальный разрез в графе (Graph Min-Cut)

Есть ориентированный граф, на ребрах указана максимальная пропускная способность. Найти **минимальный разрез** - разбиение вершин на два множества (мощность разреза - пропускной способности ребер между множествами).



Статья про максимальный поток

Graph Max-Flow/Min-Cut: алгоритмы

Boykov/Kolmogorov:

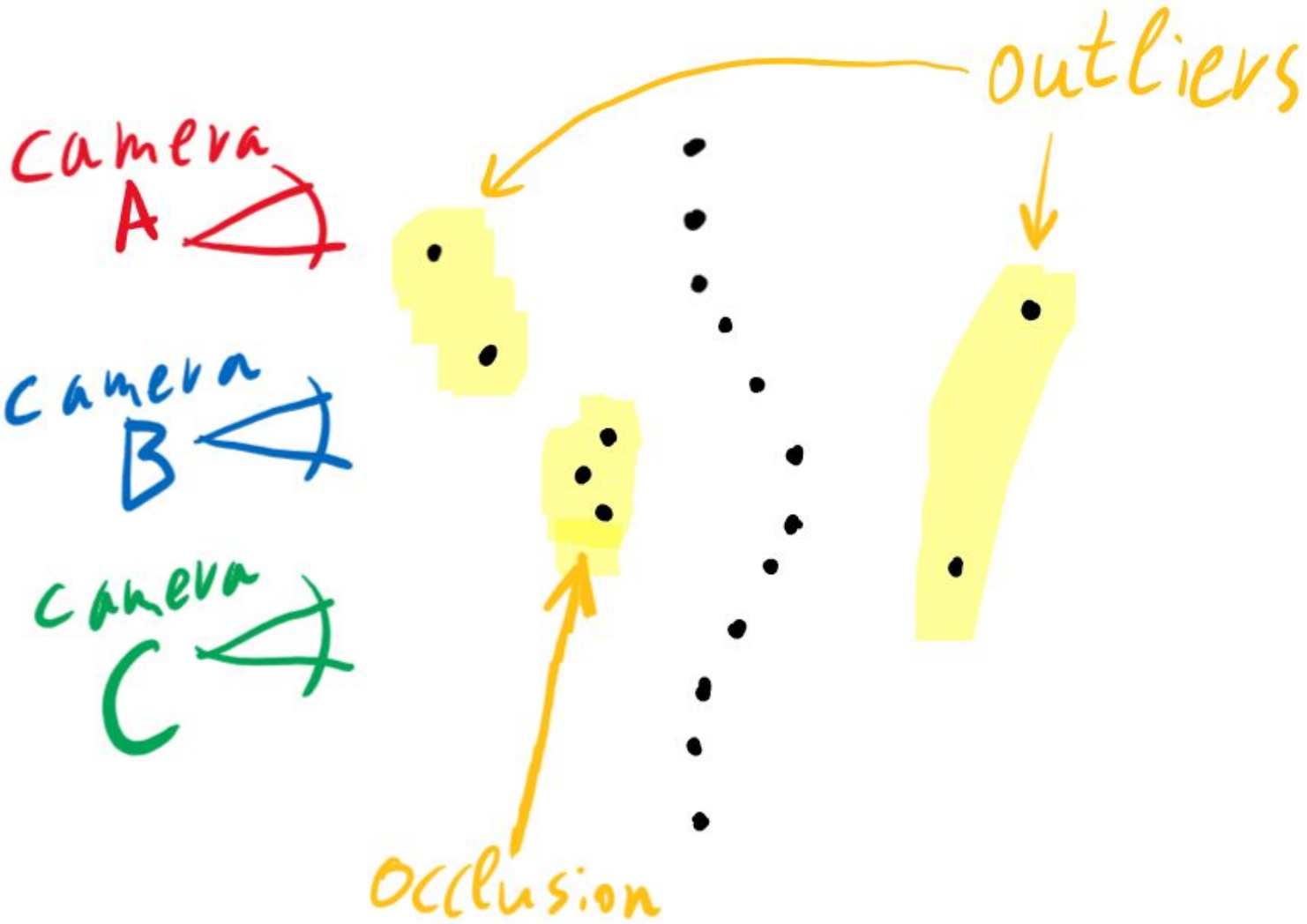
- An Experimental Comparison of Min-Cut/Max-Flow Algorithms for Energy Minimization in Vision, Boykov and Kolmogorov, 2004
- http://pub.ist.ac.at/~vnk/software.html
- https://xip.uclb.com/i/software/maxflow_computervision.html

IBFS:

- Faster and More Dynamic Maximum Flow by Incremental Breadth-First Search, Goldberg et. al., 2015
- http://web.archive.org/web/20170703042834/http://www.cs.tau.ac.il/~sagihed/ibfs/
- https://github.com/PolarNick239/IBFS/tree/53a0a3a974b7b5f2c495b81d003c87cb5e87d42a

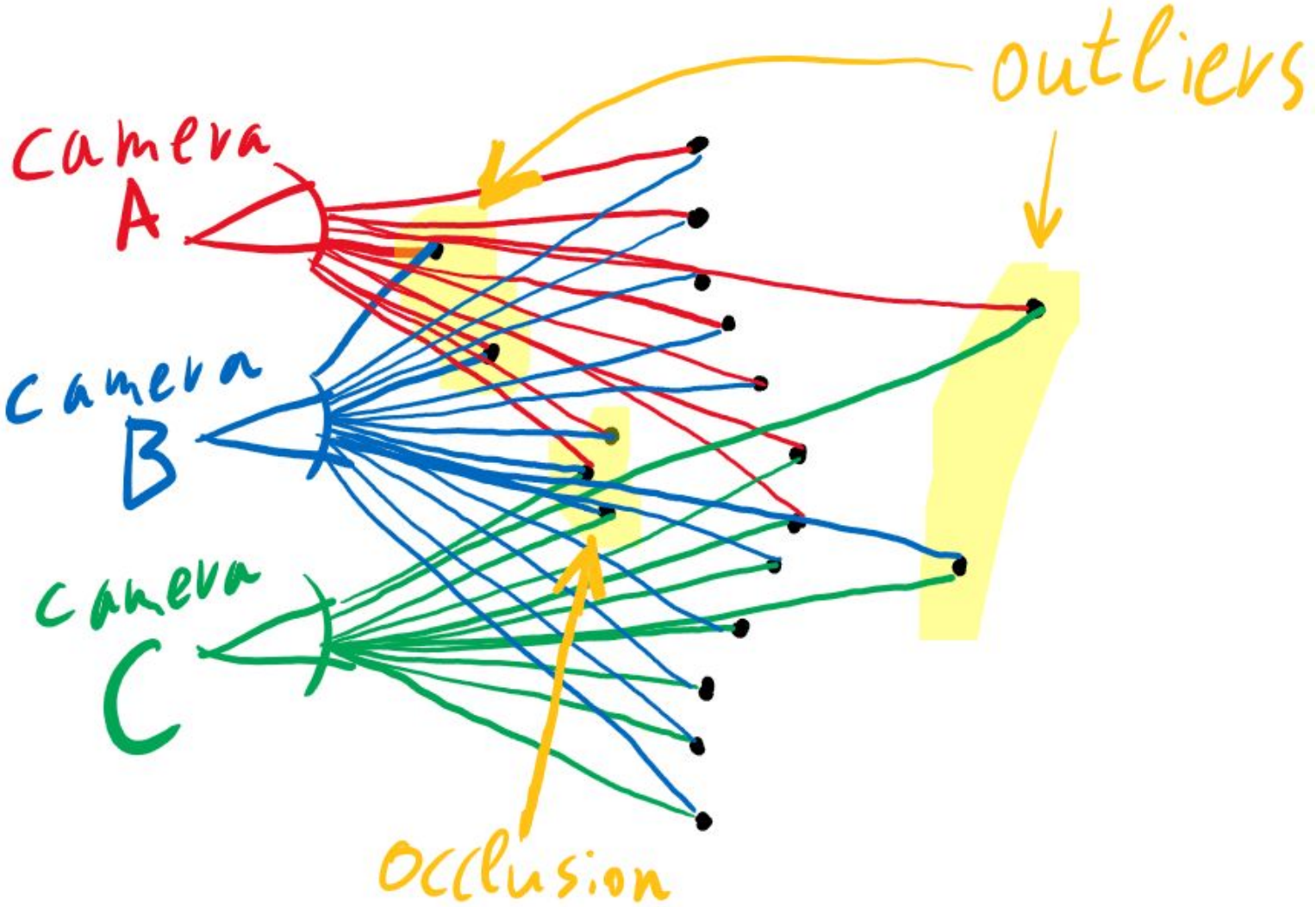
Алгоритм построения 3D модели

На вход даны 3D точки порожденные картами глубины:



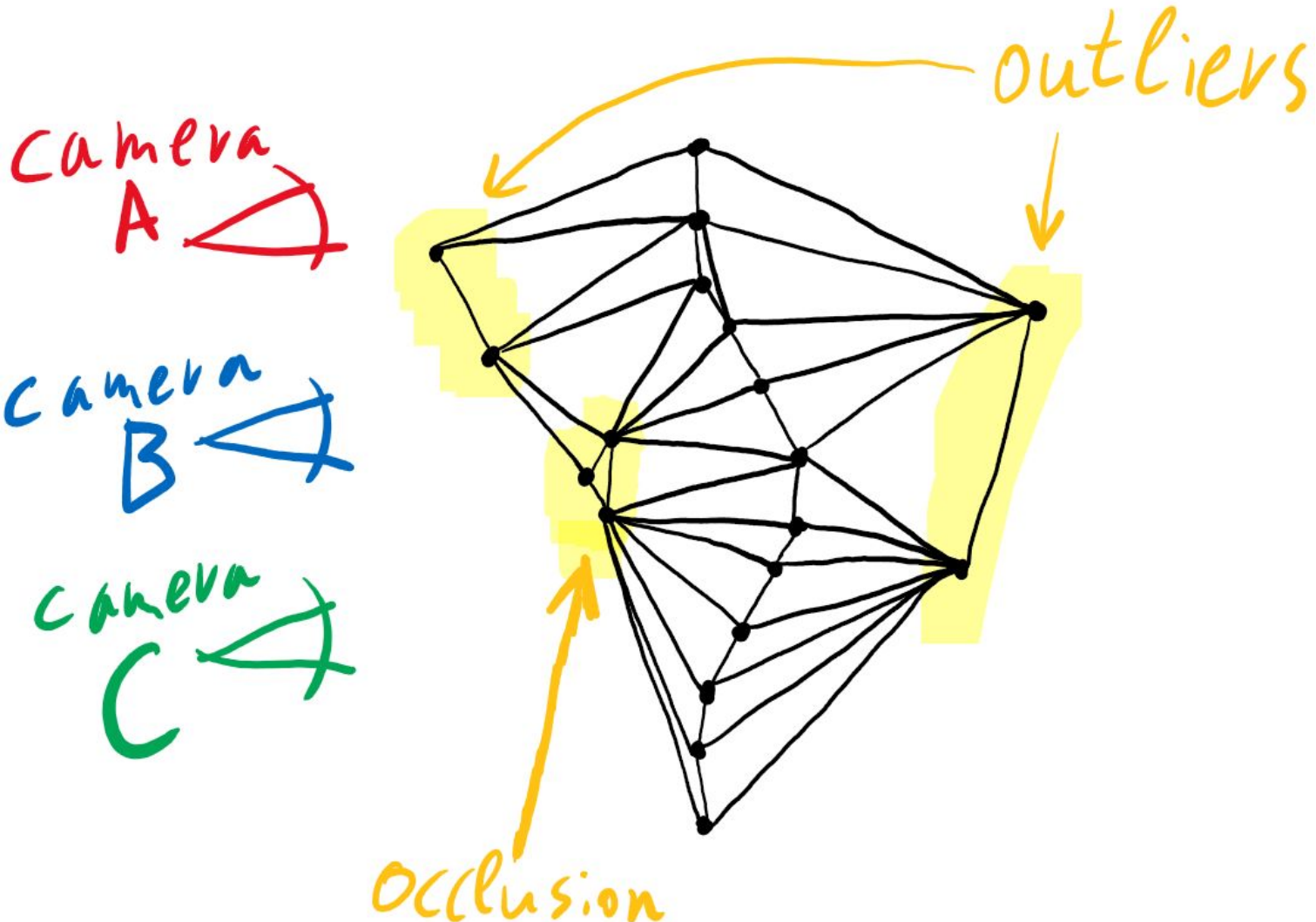
Алгоритм построения 3D модели

У каждой точки есть множество камер которые ее породили, т.е. у нас есть пучки отрезков видимости **(visibility rays)**:



Алгоритм построения 3D модели

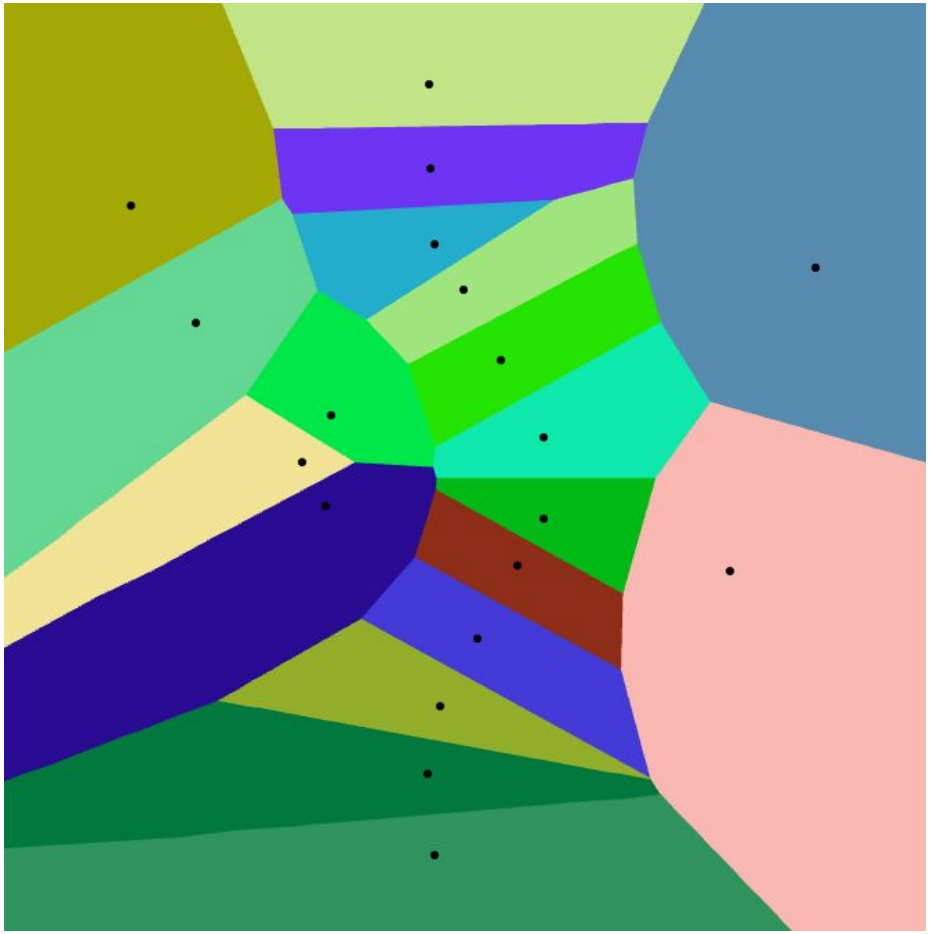
Построим триангуляцию Делоне (иллюстрация в 2D с треугольниками, но нас интересует 3D с тетрагедрончиками):



Алгоритм построения 3D модели

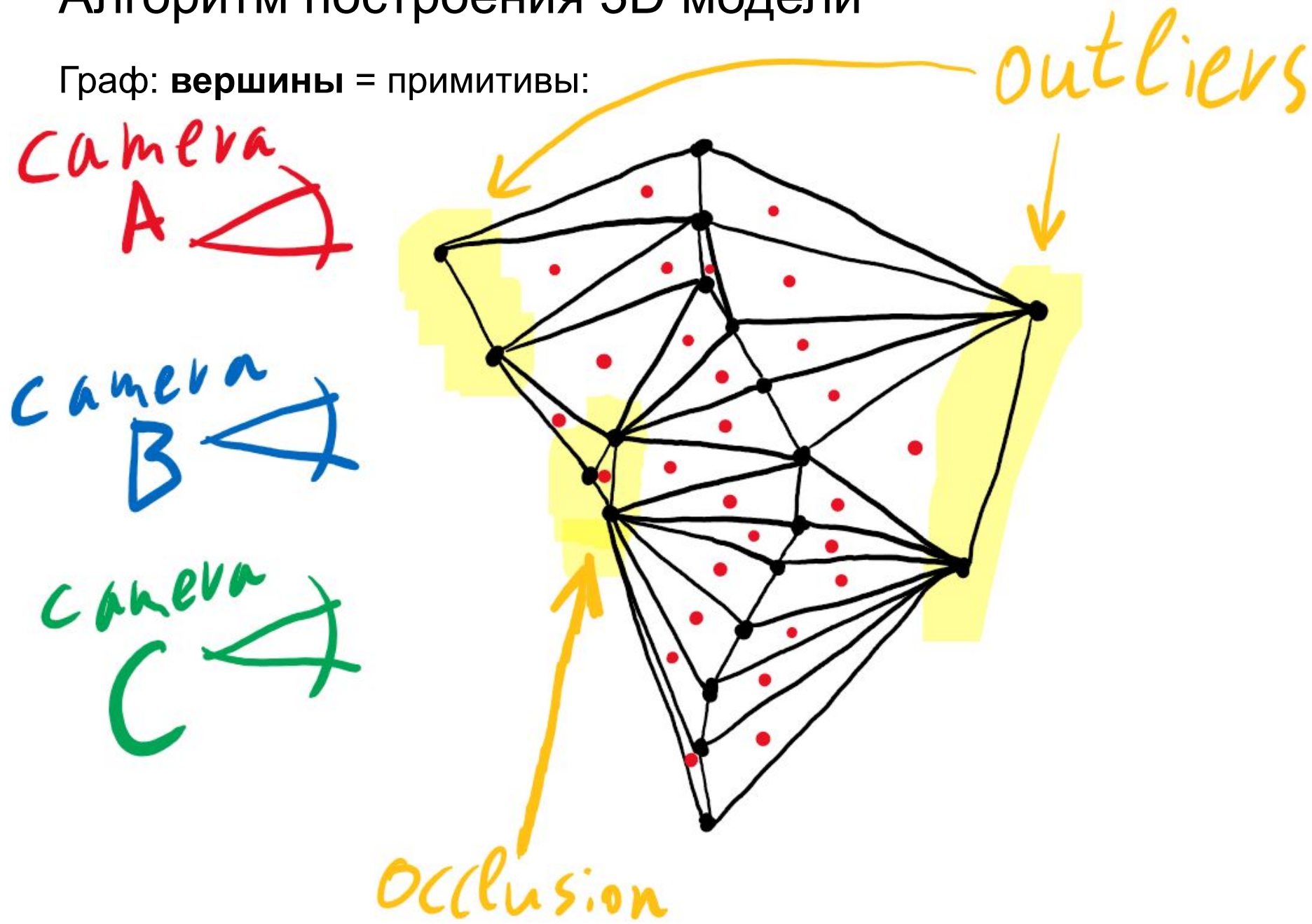
Посмотрим на двойственную диаграмму Вороного (потому что красивая):

http://alexbeutel.com/webgl/voronoi.html



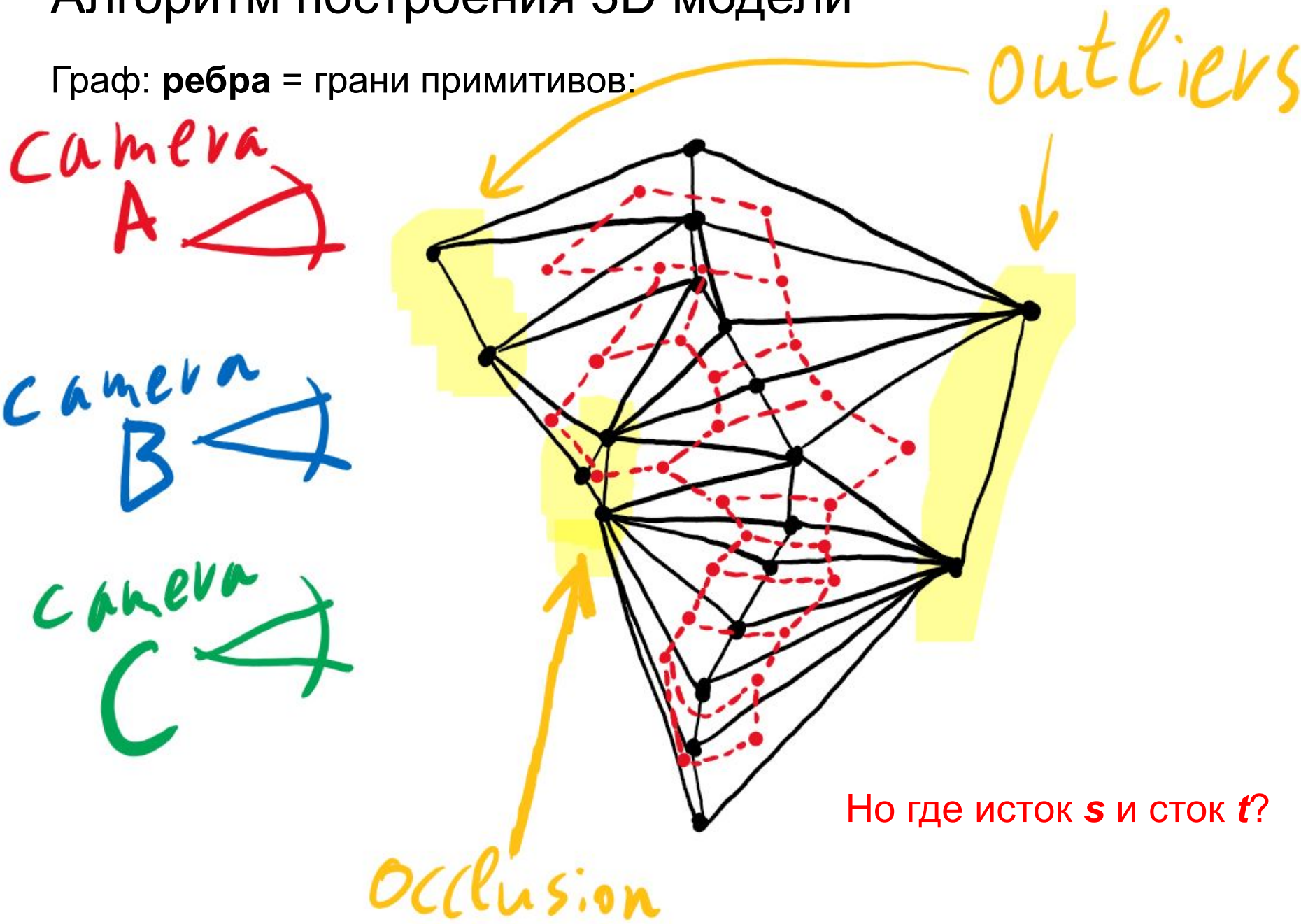
Алгоритм построения 3D модели

Граф: **вершины** = примитивы:



Алгоритм построения 3D модели

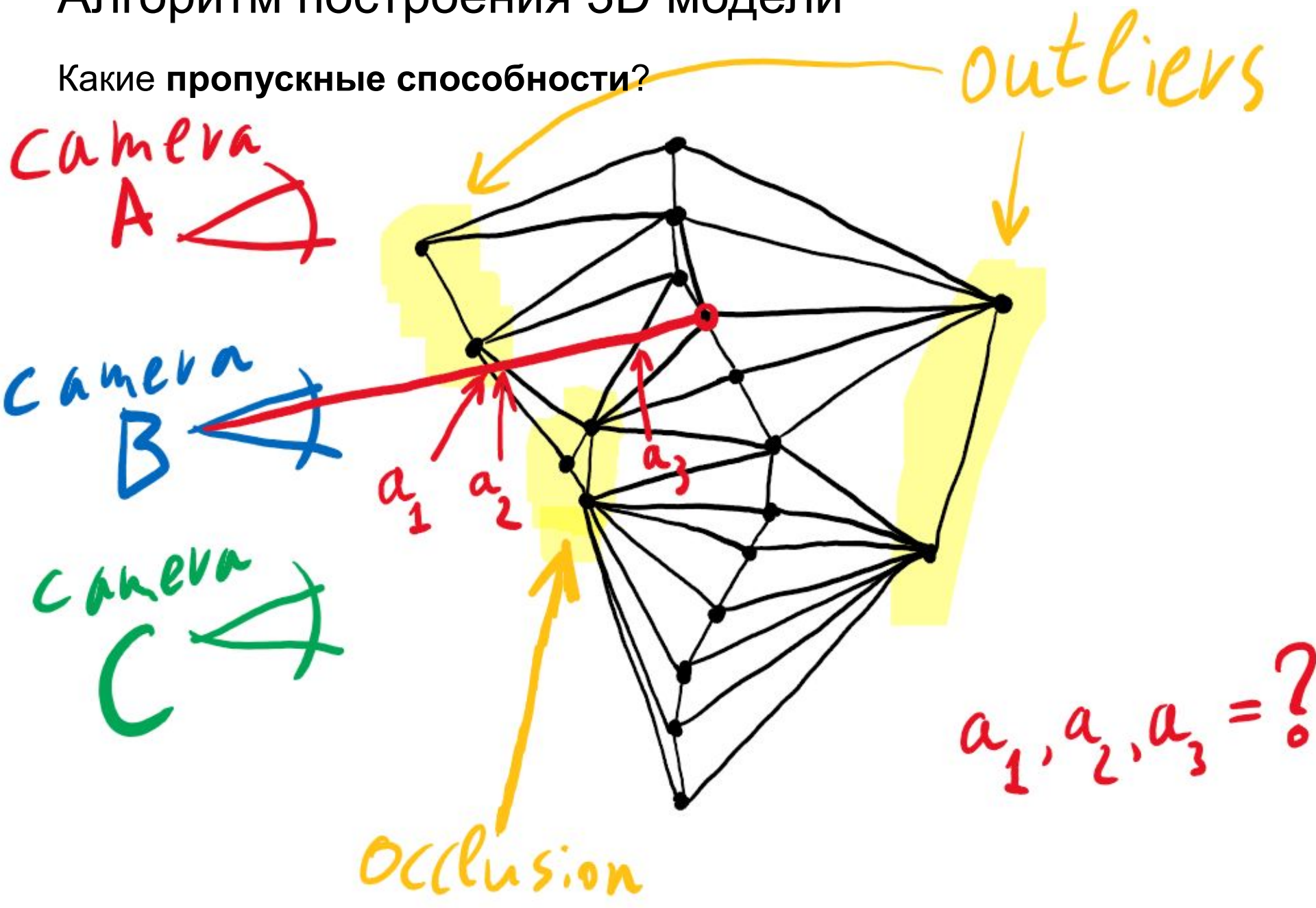
Граф: **ребра** = грани примитивов:



Но где исток ***s*** и сток ***t***?

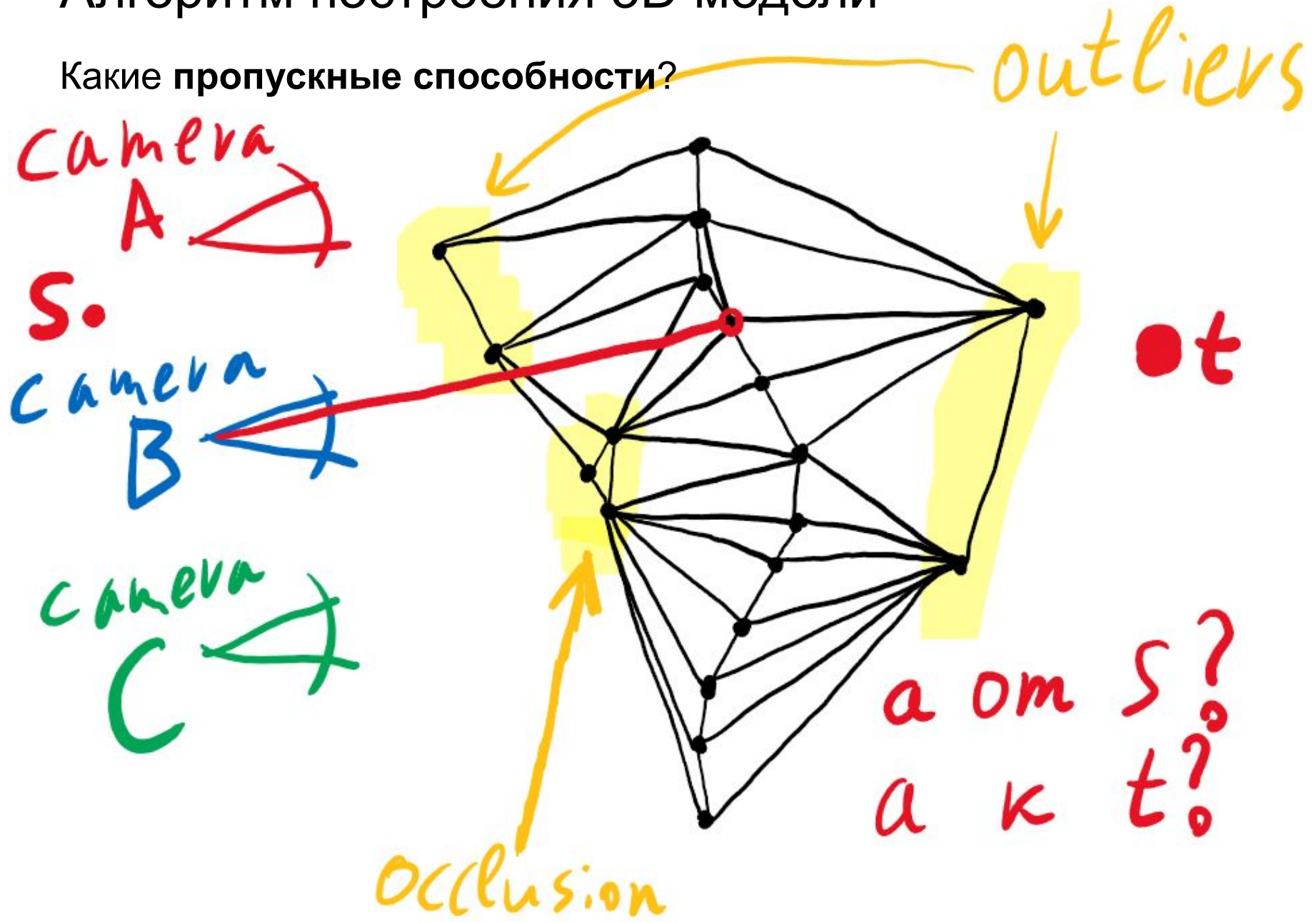
Алгоритм построения 3D модели

Какие **пропускные способности**?



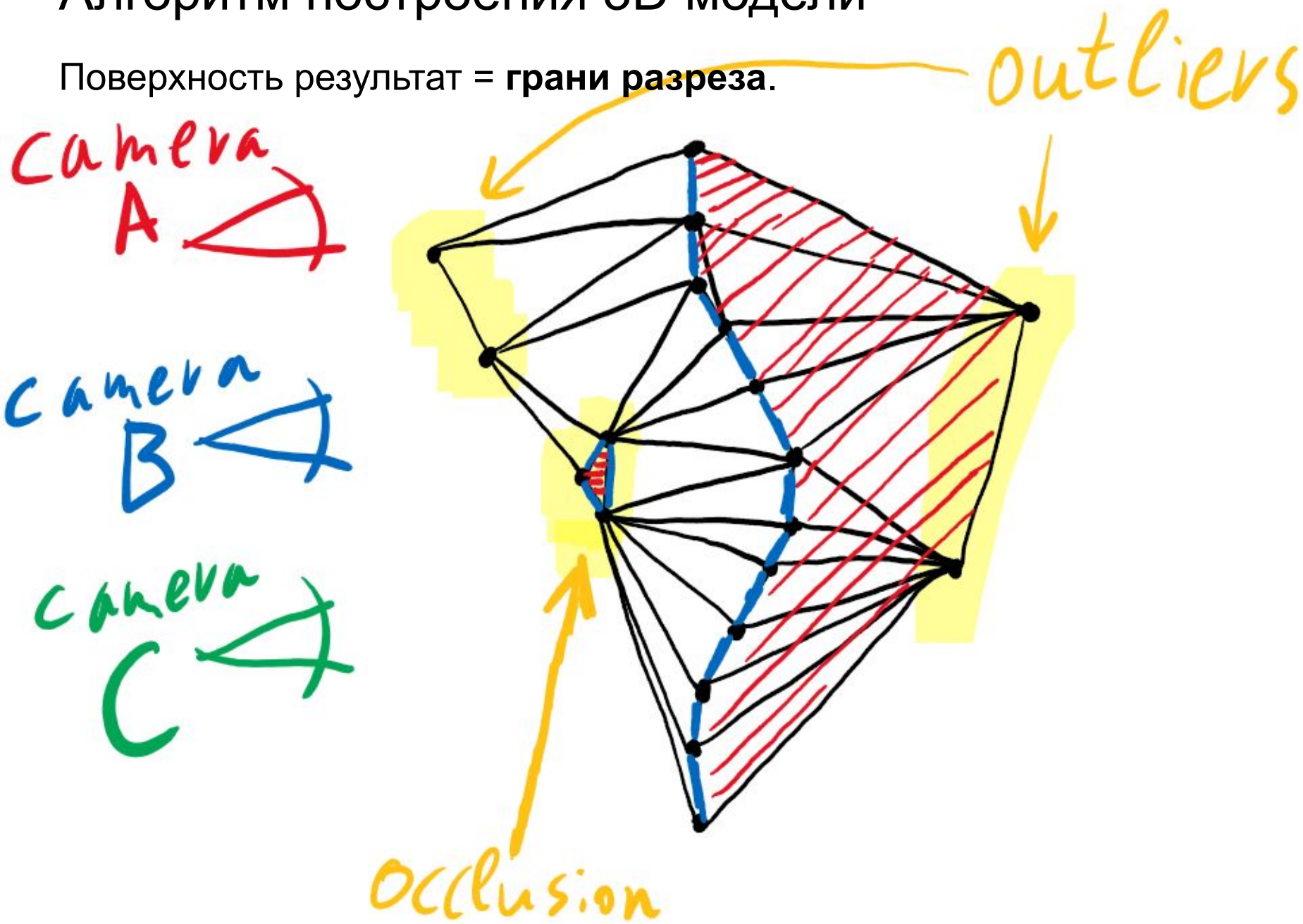
Алгоритм построения 3D модели

Какие **пропускные способности**?



Алгоритм построения 3D модели

Поверхность результат = **грани разреза**.



1) [Efficient Multi-View Reconstruction of Large-Scale Scenes using Interest Points, Delaunay Triangulation and Graph Cuts, Labatut et. al., 2007]

1) Изначальные точки - ключевые SIFT-точки со списком проекций
2) Точки объединяются если максимальная ошибка репроекции мала
3) Ради более плотного облака точек - убрали K-ratio test и добавляют большее число сопоставлений (но близкие к эпиполярной прямой).
4) Минимизируют: $\mathrm{E}(\mathcal{S}) = \mathrm{E}_{\mathrm{vis}}(\mathcal{S}) + \lambda_{\mathrm{photo}}\,\mathrm{E}_{\mathrm{photo}}(\mathcal{S}) + \lambda_{\mathrm{area}}\,\mathrm{E}_{\mathrm{area}}(\mathcal{S})$

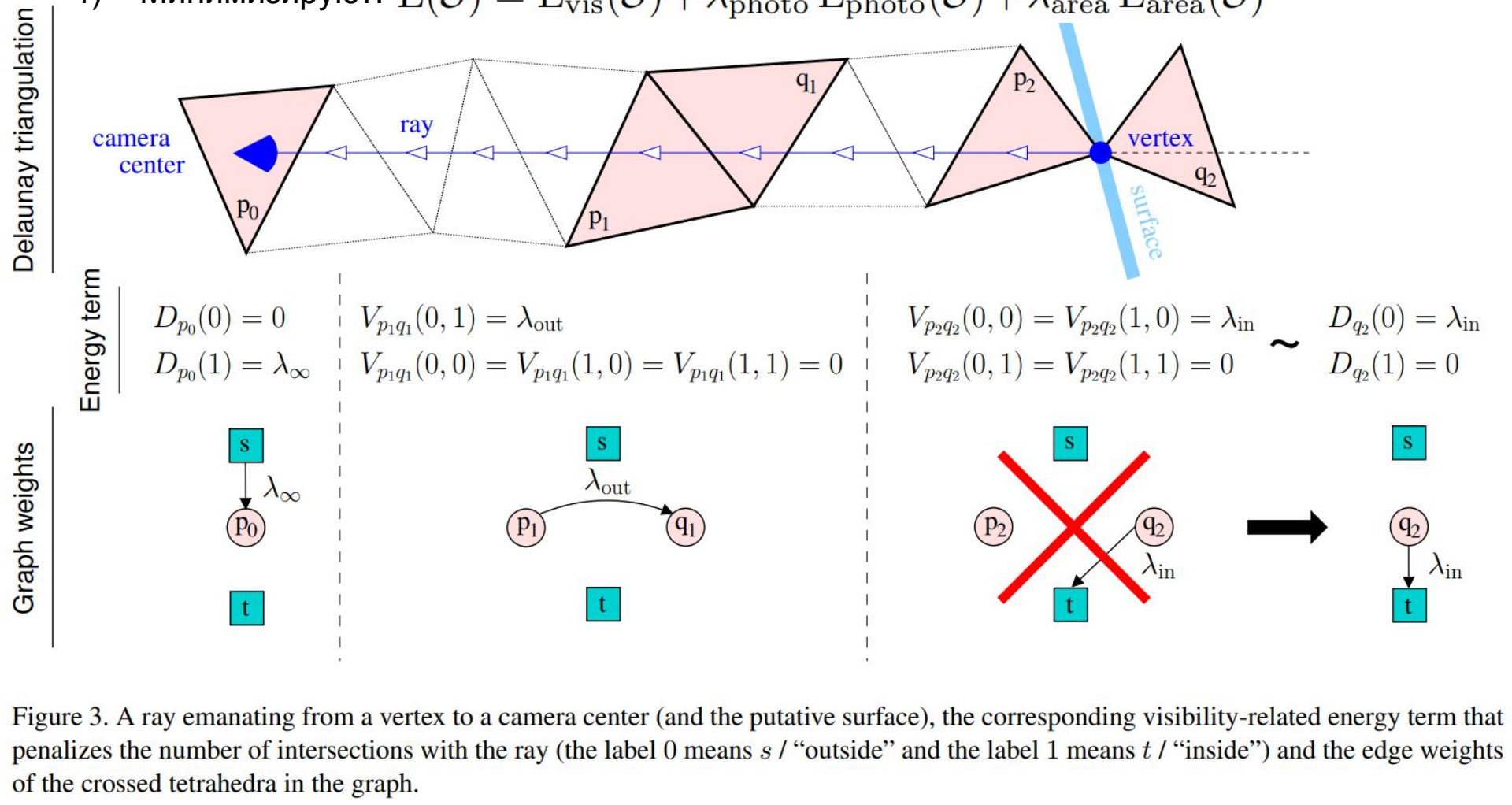
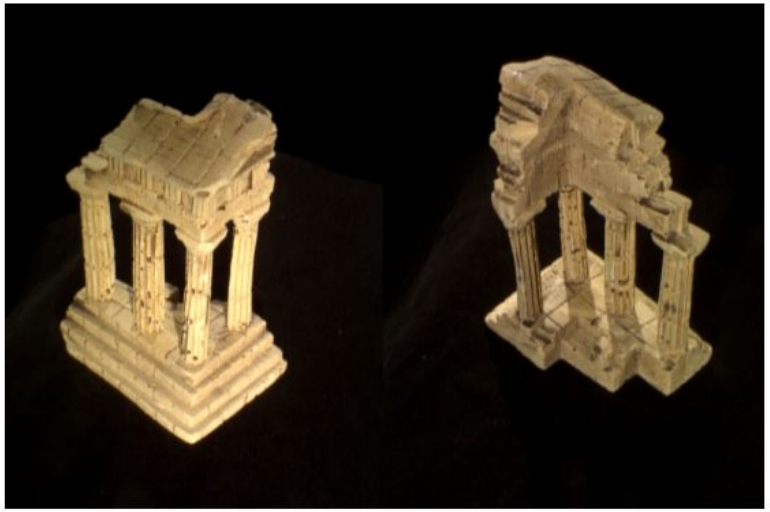
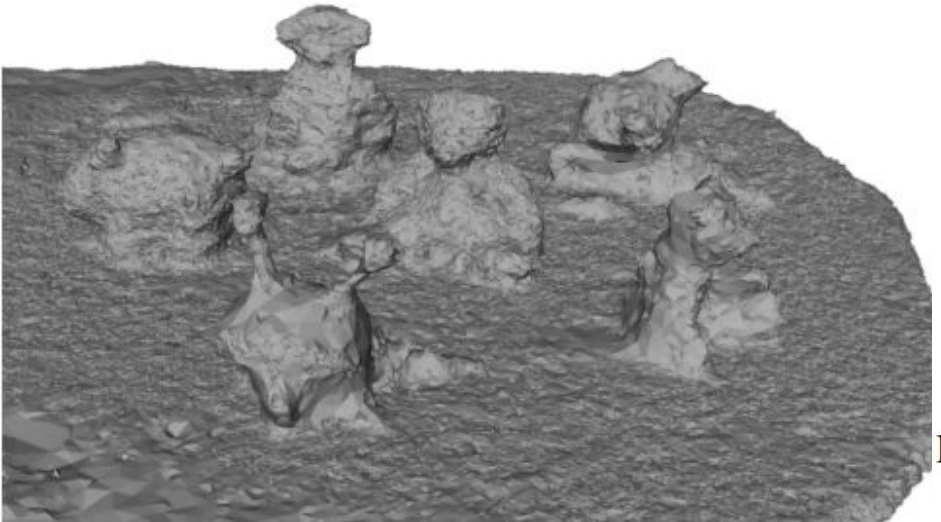


Figure 3. A ray emanating from a vertex to a camera center (and the putative surface), the corresponding visibility-related energy term that penalizes the number of intersections with the ray (the label 0 means s / "outside" and the label 1 means t / "inside") and the edge weights of the crossed tetrahedra in the graph.

1) Efficient Multi-View Reconstruction of Large-Scale Scenes using Interest Points, Delaunay Triangulation and Graph Cuts, Labatut et. al., 2007





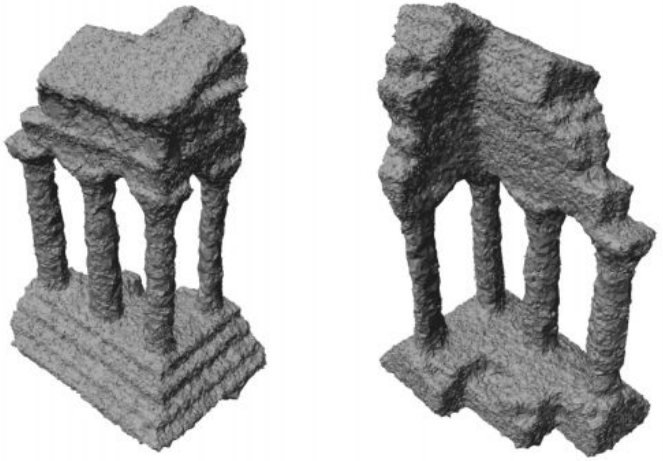
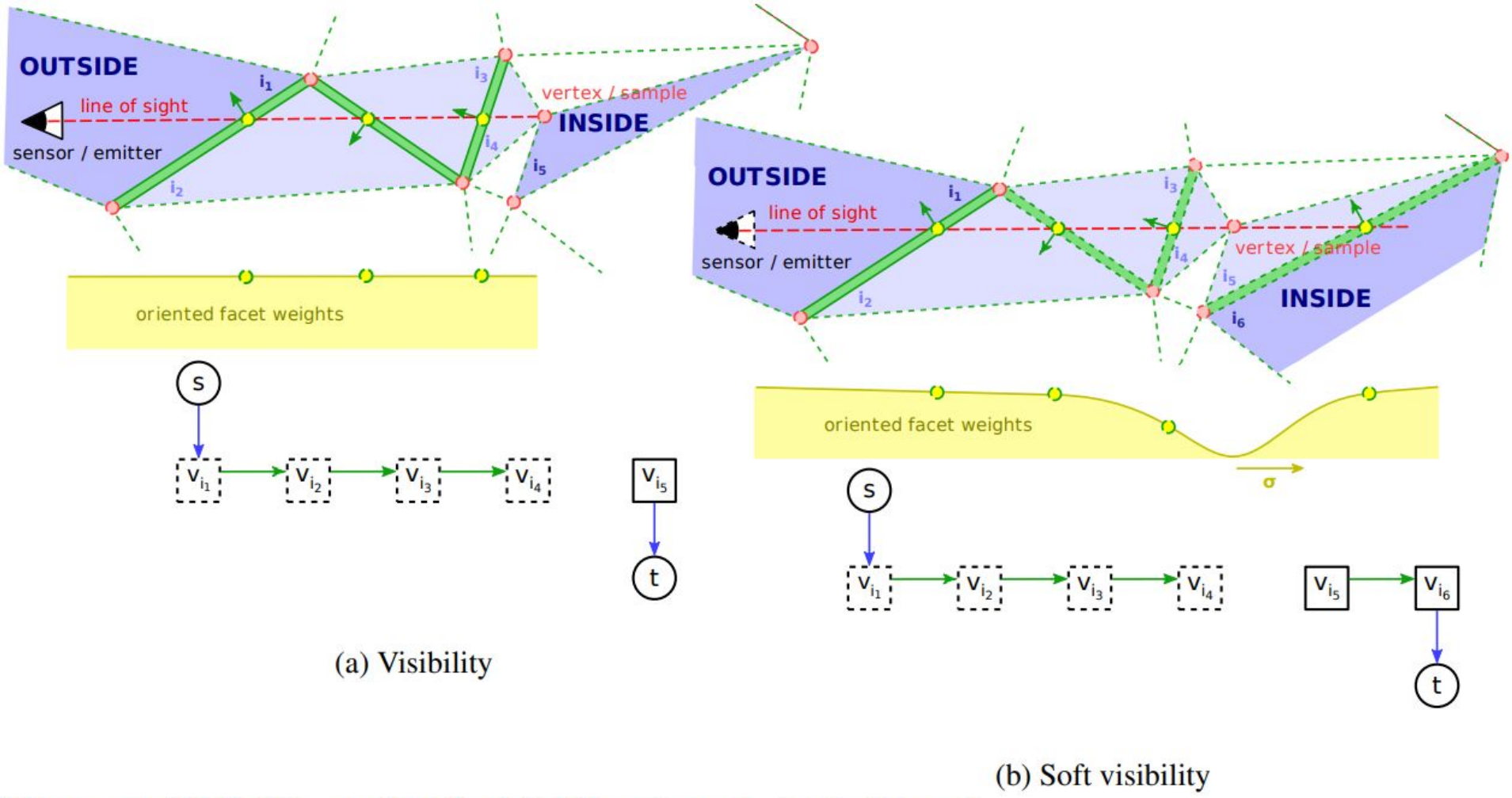


Figure 4. Some images of the temple dataset and our results

2) [Robust and efficient surface reconstruction from range data, Labatut et. al., 2009]



(a) Visibility

(b) Soft visibility

Figure 4: ***Visibility and soft visibility.*** How a single line of sight (pink) of a vertex of the triangulation (red) from a sample point to a laser (or to a sensor) contributes to the weights assigned to the origin tetrahedron, to the facets it crosses and to the final tetrahedron (blue).

2) Robust and efficient surface reconstruction from range data, Labatut et. al., 2009

1) Visibility rays - из любого источника (LIDAR или плотные карты глубины).
2) Более мягкое указание пересечения поверхности (глубже под поверхность заглядываем + пропускная способность тем меньше чем ближе к поверхности).
3) Другой surface term про две смежные описывающие сферы (вместо медленного photoconsistency term и тяготеющего к излишним упрощениям area term).
4) Медленно работает:

	Model	#Points	#Tets	Delaunay	Visibility	Quality	Min. *s-t* cut	Overall	
133M	Sheep (Fig. 8)	153K	966K	2s	3s	1s	3s	10s	$\ll$ 1m
306M	Bunny (Fig. 12)	362K	2,252K	10s	11s	2s	7s	31s	$<$ 1m
1.6G	Dragon (Fig. 6)	1,770K	11,383K	38s	68s	15s	59s	180s	3m
1.7G	Angel (Fig. 1)	2,008K	12,637K	41s	86s	16s	48s	190s	3m 10s
2.0G	Armadillo (Fig. 3)	2,247K	14,519K	47s	58s	13s	177s	295s	4m 55s
2.4G	Buddha (Fig. 7)	2,644K	17,167K	62s	120s	14s	74s	271s	4m 31s
-	Elephant (Fig. 2)	4,413K	27,487K	98s	274s	35s	-	-	
6.5G	Elephant (Fig. 2) / 64-bits	4,413K	27,487K	93s	189s	32s	102s	417s	6m 57s
9.9G	Soufflot (Fig. 14) / 64-bits	6,592K	42,062K	176s	416s	40s	521s	1154s	19m 14s

3) Multi-View Reconstruction Preserving Weakly-Supported Surfaces, Jancosek et. al., 2011

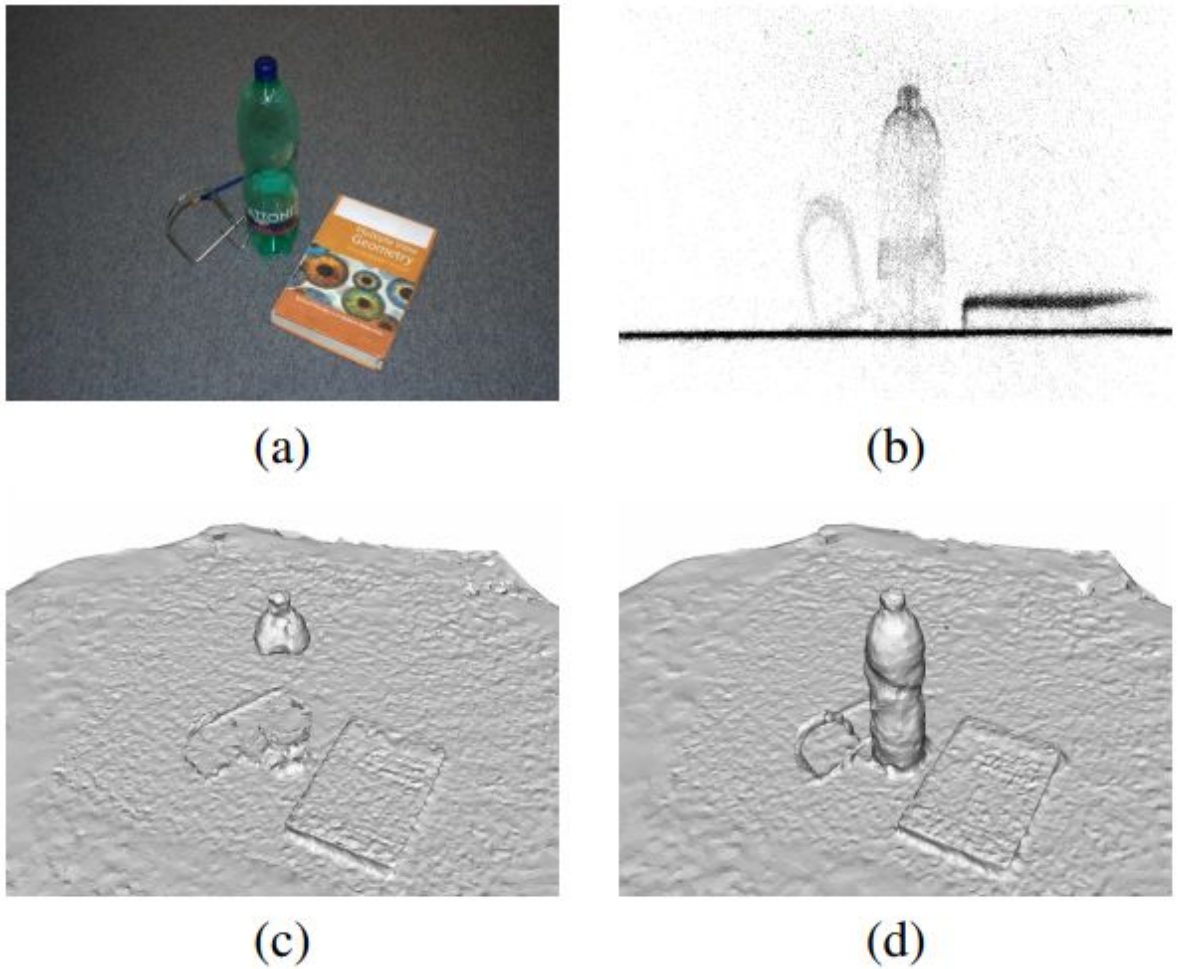


Figure 1. **Results for the 'bottle' data set.** (a) Input image, (b) input 3D point cloud, (c) results using our implementation of [15], (d) the technique presented in this work reconstructs weakly-supported surfaces (the bottle) better.

3) [Multi-View Reconstruction Preserving Weakly-Supported Surfaces, Jancosek et. al., 2011]

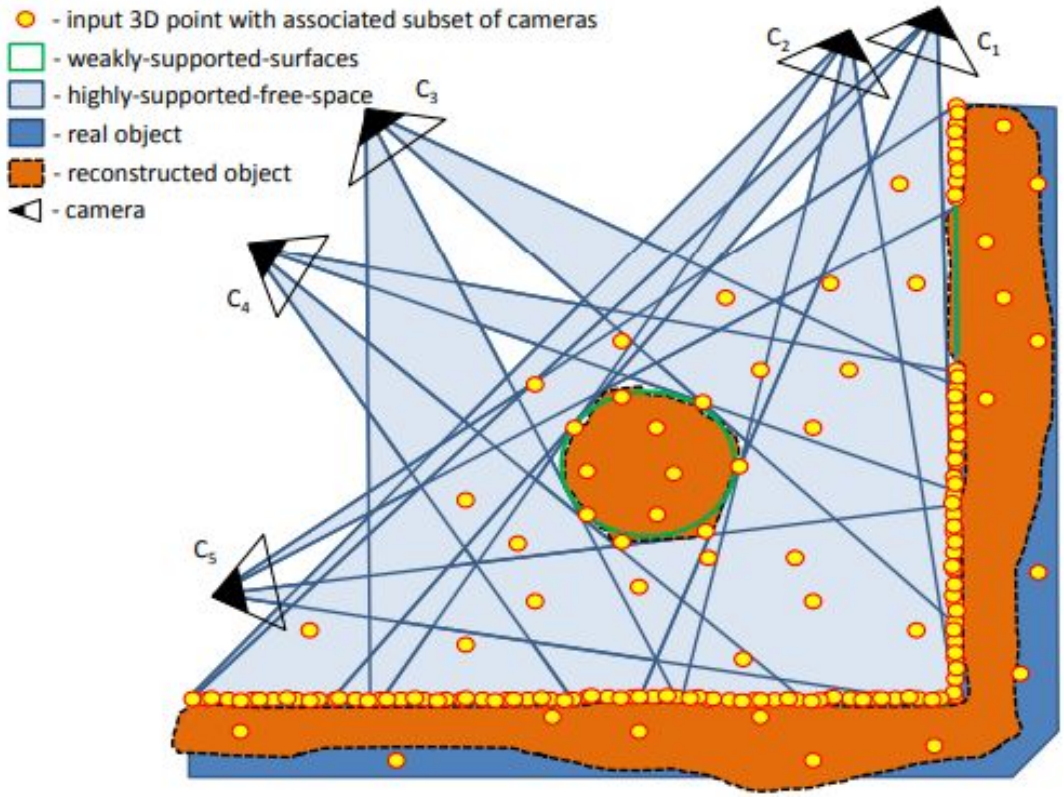


Figure 2. **Weakly-supported surfaces.** A highly-supported-free-space is the part of the space which is between the surfaces densely sampled by the input 3D points and the associated cameras. Weakly-supported surfaces (green) are the surfaces of the real object which are weakly sampled by the input 3D points and are close to the border of the union of the highly-supported-free-space cones.

3) [Multi-View Reconstruction Preserving Weakly-Supported Surfaces, Jancosek et. al., 2011]

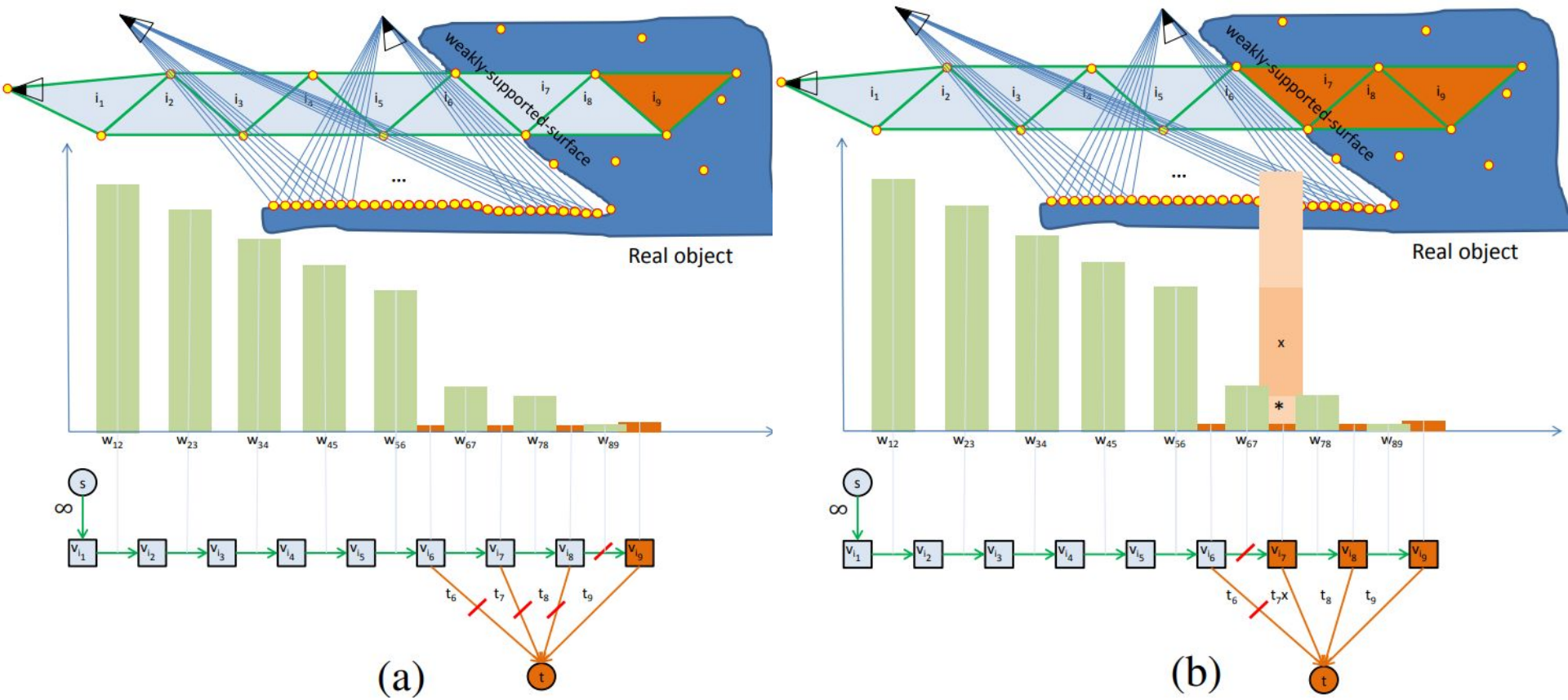


Figure 6. **Representative example.** Light blue: source label (free space), brown: sink label (full space). (a,b) top: a part of the triangulation, bottom: the associated s-t graph, middle: the weights of the associated edges. (a) Minimal cut for weights computed by the base-line approach leads to a wrong solution (b) Multiplying t_7 by $(w_{56} - w_{78})$ leads to the correct solution.

3) Multi-View Reconstruction Preserving Weakly-Supported Surfaces, Jancosek et. al., 2011

Ссылки

Delaunay triangulation + graph min-cut:

- Efficient Multi-View Reconstruction of Large-Scale Scenes using Interest Points, Delaunay Triangulation and Graph Cuts, Labatut et. al., 2007
- Robust and efficient surface reconstruction from range data, Labatut et. al., 2009
- Multi-View Reconstruction Preserving Weakly-Supported Surfaces, Jancosek et. al., 2011

Out-of-core adaptations:

- PHD Thesis: Large-scale and high-quality Multi-view stereo (p. 65 - section 5.3: Merging meshes from partition of bounding box), Vu, 2011
- Scalable Surface Reconstruction from Point Clouds with Extreme Scale and Density Diversity, Mostegel et.al., 2017

Пример реализации в OpenMVS: openMVS/libs/MVS/SceneReconstruct.cpp

Вопросы?

Полярный Николай
polarnick239@gmail.com