

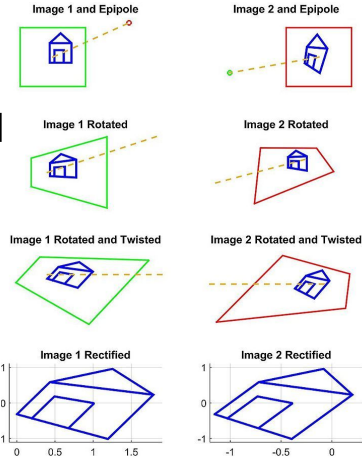
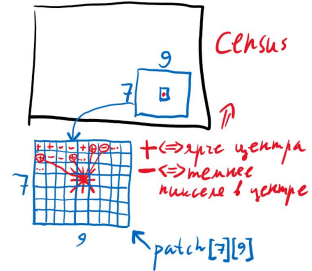
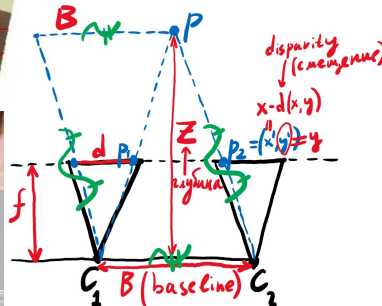
Введение в фотограмметрию

Построение карт глубины

Ректификация стереопар, cost-функции

Фотограмметрия. Лекция 10

- Ректификация стереопар
- Cost-функции похожести патчей (SAD, NCC, ZNCC)



Какой процесс обработки пока что выстраивается?

- 1) Выделили ключевые точки методом **SIFT**

Какой процесс обработки пока что выстраивается?

- 1) Выделили ключевые точки методом **SIFT**
- 2) Сопоставили ключевые точки между фотографиями - **Brute Force/FLANN/...**

Какой процесс обработки пока что выстраивается?

- 1) Выделили ключевые точки методом **SIFT**
- 2) Сопоставили ключевые точки между фотографиями - **Brute Force/FLANN/...**
- 3) Профильтровали сопоставления от шумов - **K-ratio test, Left-Right check, cluster filtering, ...**

Какой процесс обработки пока что выстраивается?

- 1) Выделили ключевые точки методом **SIFT**
- 2) Сопоставили ключевые точки между фотографиями - **Brute Force/FLANN/...**
- 3) Профильтровали сопоставления от шумов - **K-ratio test, Left-Right check, cluster filtering, ...**
- 4) Взаимно выровняли первую пару фотографий - **Essential matrix**

Какой процесс обработки пока что выстраивается?

- 1) Выделили ключевые точки методом **SIFT**
- 2) Сопоставили ключевые точки между фотографиями - **Brute Force/FLANN/...**
- 3) Профильтровали сопоставления от шумов - **K-ratio test, Left-Right check, cluster filtering, ...**
- 4) Взаимно выровняли первую пару фотографий - **Essential matrix**
- 5) Инкрементально выровняли остальные камеры - **триангуляция + Bundle Adjustment + резекция**

Какой процесс обработки пока что выстраивается?

- 1) Выделили ключевые точки методом **SIFT**
- 2) Сопоставили ключевые точки между фотографиями - **Brute Force/FLANN/...**
- 3) Профильтровали сопоставления от шумов - **K-ratio test, Left-Right check, cluster filtering, ...**
- 4) Взаимно выровняли первую пару фотографий - **Essential matrix**
- 5) Инкрементально выровняли остальные камеры - **триангуляция + Bundle Adjustment + резекция**
- 6) ???

Какой процесс обработки пока что выстраивается?

- 1) Выделили ключевые точки методом **SIFT**
- 2) Сопоставили ключевые точки между фотографиями - **Brute Force/FLANN/...**
- 3) Профильтровали сопоставления от шумов - **K-ratio test, Left-Right check, cluster filtering, ...**
- 4) Взаимно выровняли первую пару фотографий - **Essential matrix**
- 5) Инкрементально выровняли остальные камеры - **триангуляция + Bundle Adjustment + резекция**
- 6) Найти детальную геометрию сцены - с детальностью на уровне **пикселей**

Какой процесс обработки пока что выстраивается?

Итого:

- Есть точная калибровка камеры и ракурсов фотографирования (**intrinsics & extrinsics parameters**)
- Есть точное **но разреженное (недетальное)** облако 3D точек

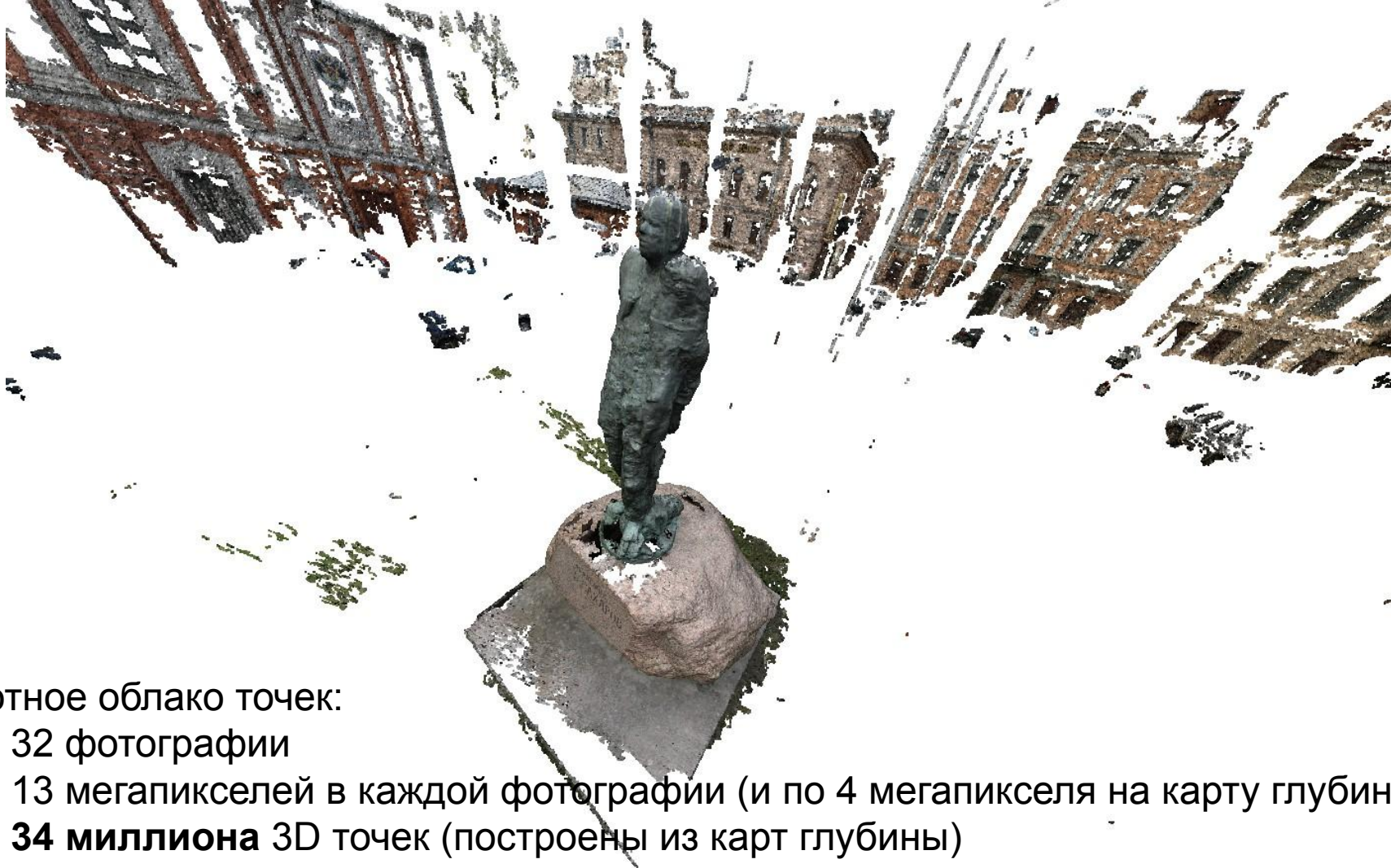
Хотим:

- Детальную геометрию - например плотное облако точек:
один пиксель - одна точка



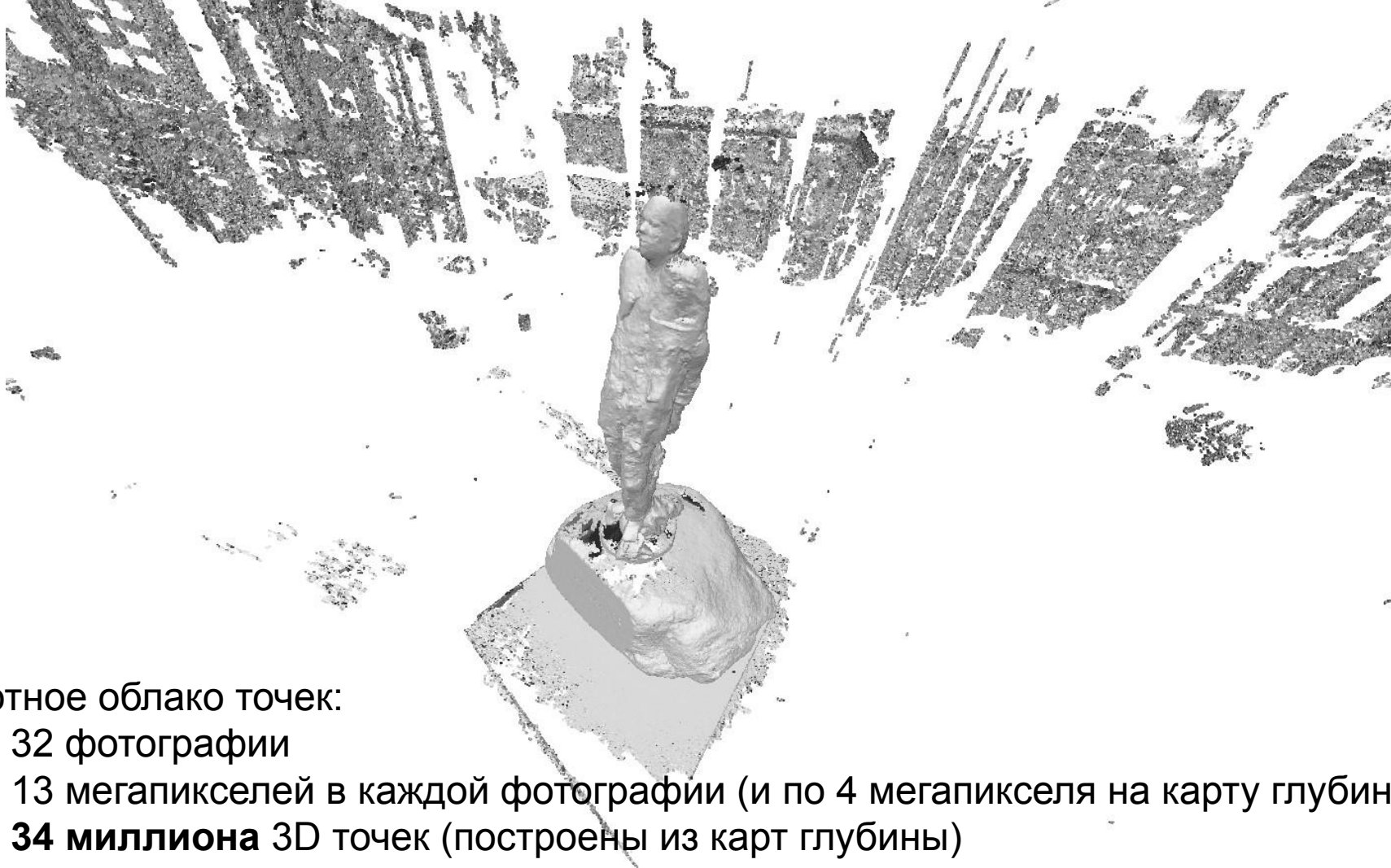
Облако ключевых точек:

- 32 фотографии
- ~8 тысяч ключевых точек на каждой фотографии
- **83 тысячи** 3D точек



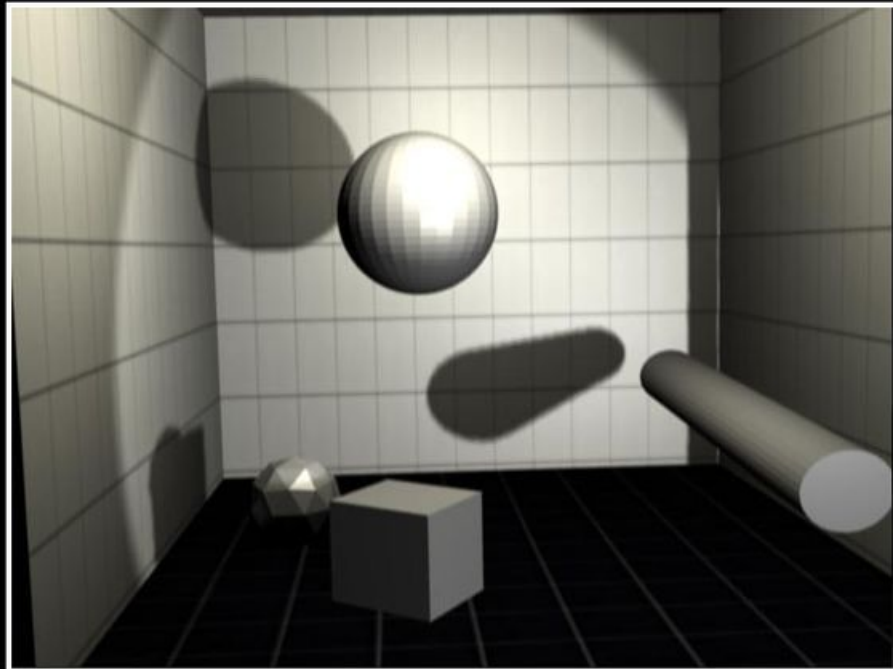
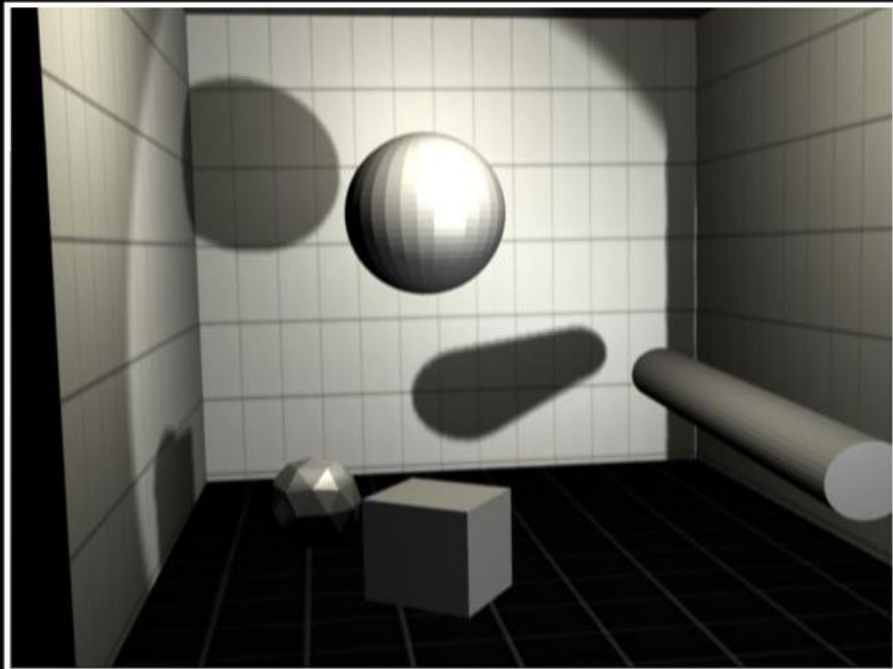
Плотное облако точек:

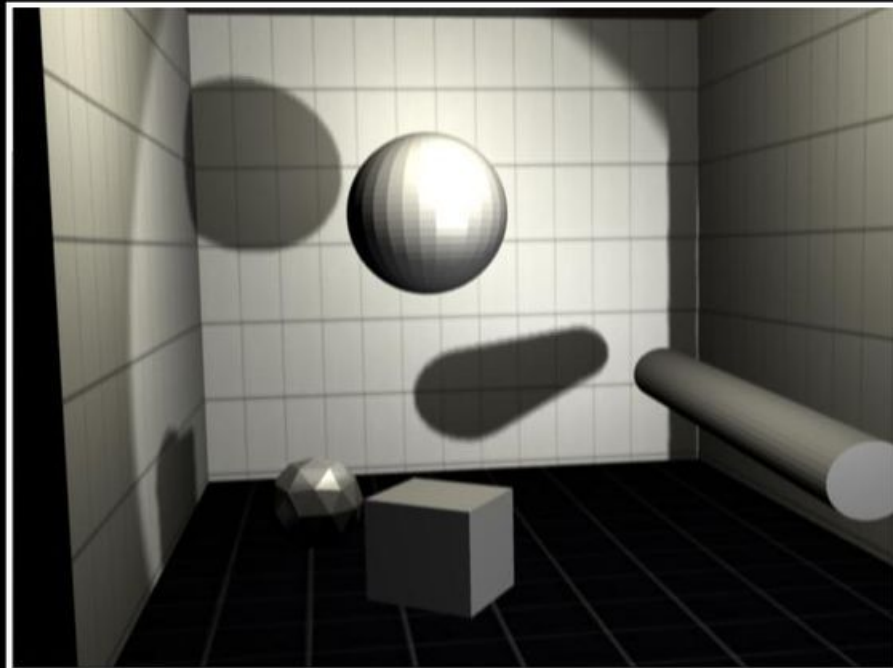
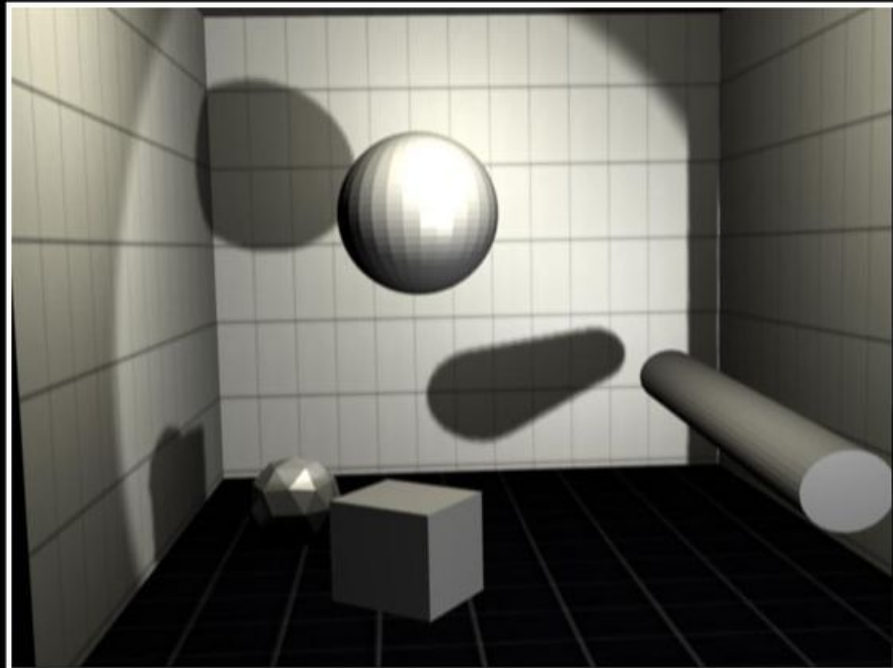
- 32 фотографии
- 13 мегапикселей в каждой фотографии (и по 4 мегапикселя на карту глубины)
- **34 миллиона** 3D точек (построены из карт глубины)

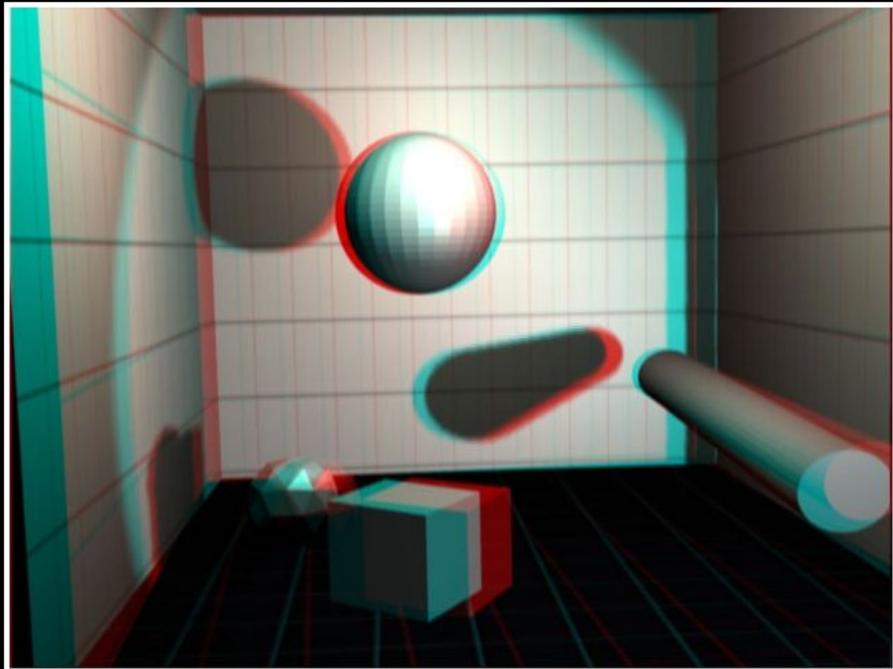


Плотное облако точек:

- 32 фотографии
- 13 мегапикселей в каждой фотографии (и по 4 мегапикселя на карту глубины)
- **34 миллиона** 3D точек (построены из карт глубины)







Стереопара

За счет чего человек “видит” расстояния до объектов и в целом сцену перед ним?

Стереопара

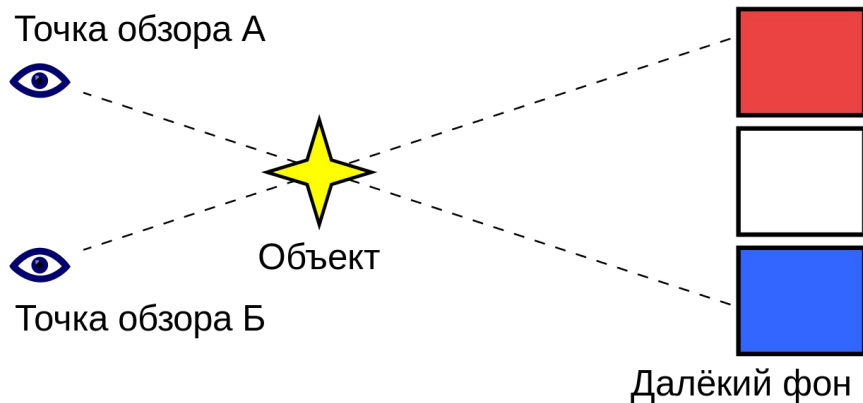
За счет чего человек “видит” расстояния до объектов и в целом сцену перед ним?

За счет пары глаз (две картинки - **стереопара**) и смещения (**параллакса**):

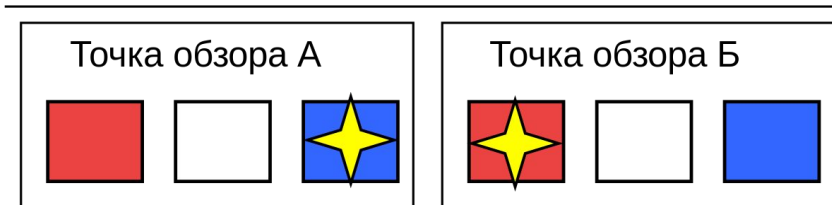
Стереопара

За счет чего человек “видит” расстояния до объектов и в целом сцену перед ним?

За счет пары глаз (две картинки - **стереопара**) и смещения (**параллакса**):



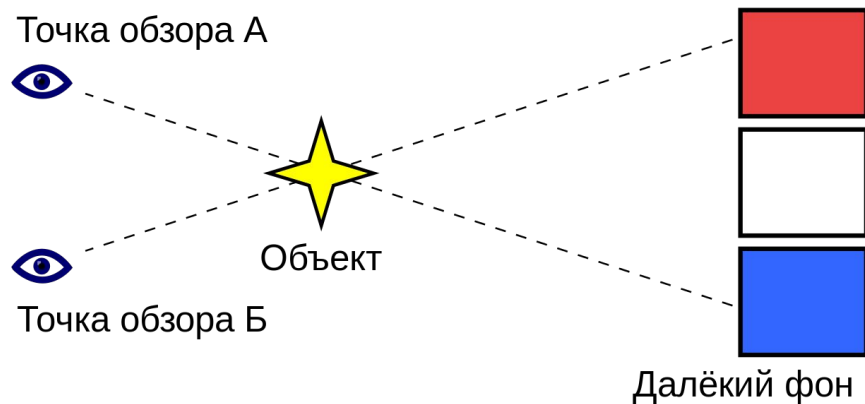
Почему смещение по горизонтали?



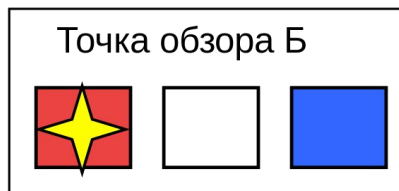
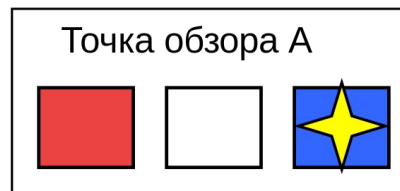
Стереопара

За счет чего человек “видит” расстояния до объектов и в целом сцену перед ним?

За счет пары глаз (две картинки - **стереопара**) и смещения (**параллакса**):



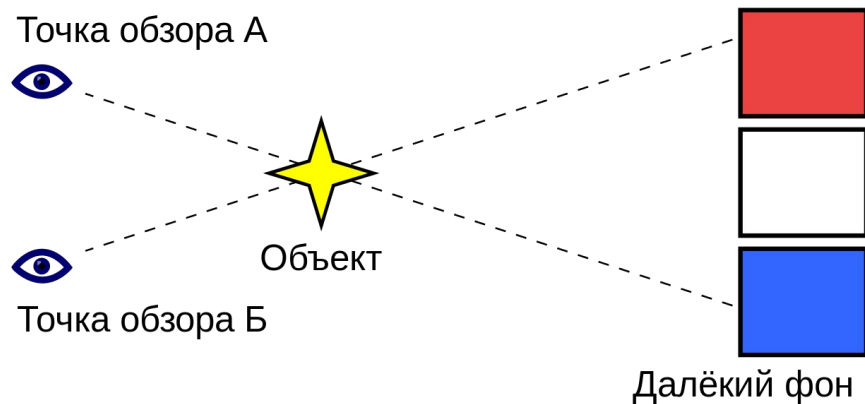
Почему смещение по горизонтали?



Стереопара

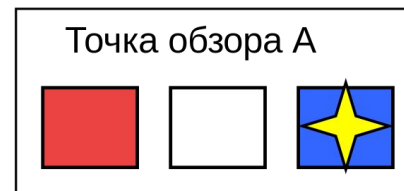
За счет чего человек “видит” расстояния до объектов и в целом сцену перед ним?

За счет пары глаз (две картинки - **стереопара**) и смещения (**параллакса**):



Почему смещение по горизонтали?

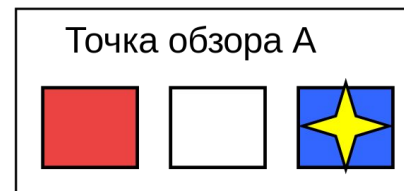
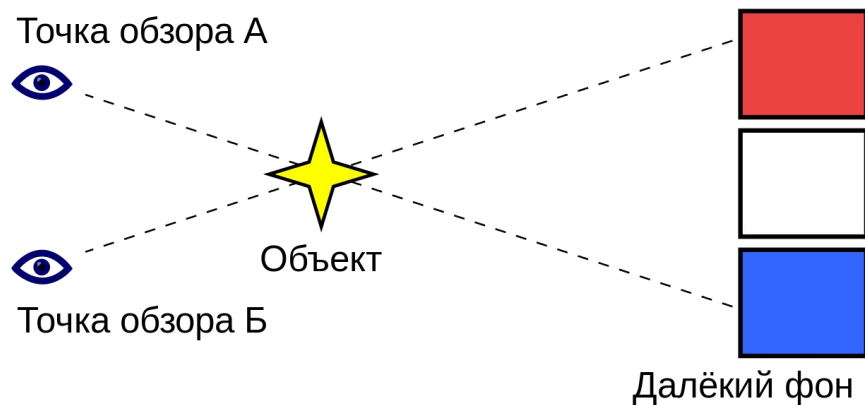
Что это за горизонтальная прямая?



Стереопара

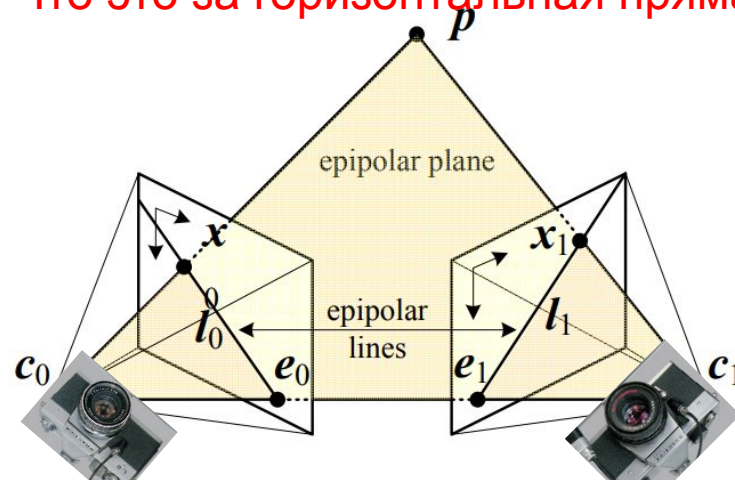
За счет чего человек “видит” расстояния до объектов и в целом сцену перед ним?

За счет пары глаз (две картинки - **стереопара**) и смещения (**параллакса**):



Почему смещение по горизонтали?

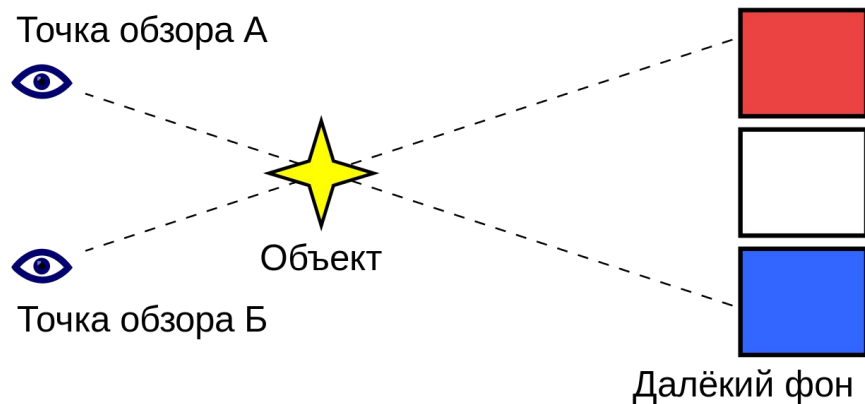
Что это за горизонтальная прямая?



Стереопара

За счет чего человек “видит” расстояния до объектов и в целом сцену перед ним?

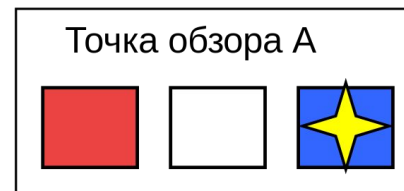
За счет пары глаз (две картинки - **стереопара**) и смещения (**параллакса**):



Почему смещение по горизонтали?

Что это за горизонтальная прямая?

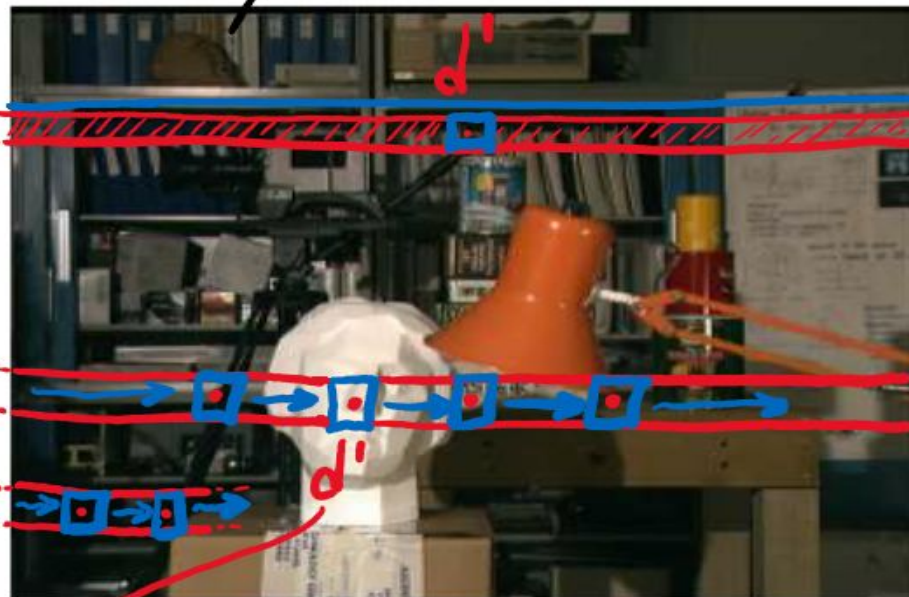
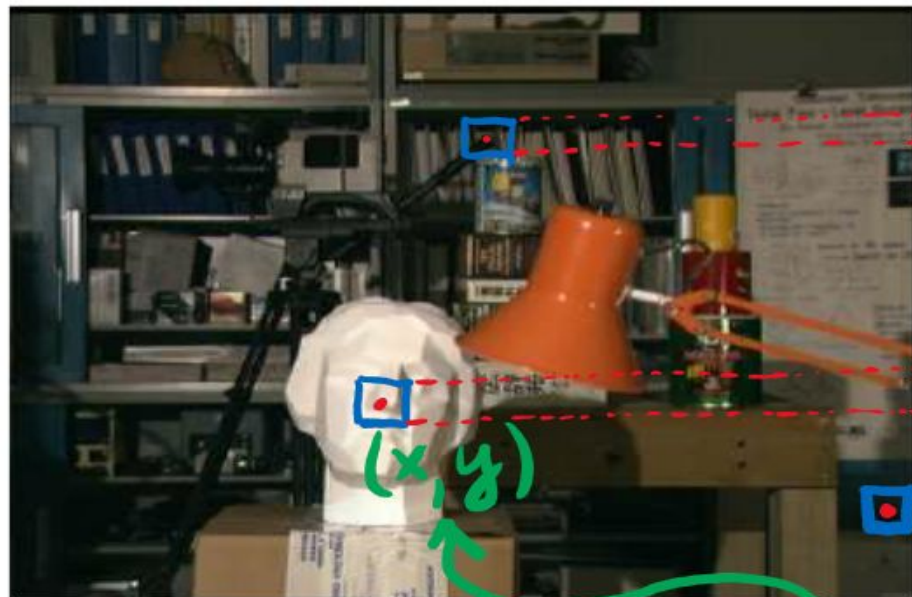
Какая зависимость расстояния до объекта от смещения в стереопаре?



Горизонтальное смещение = параллакс = диспаритет = d

левая

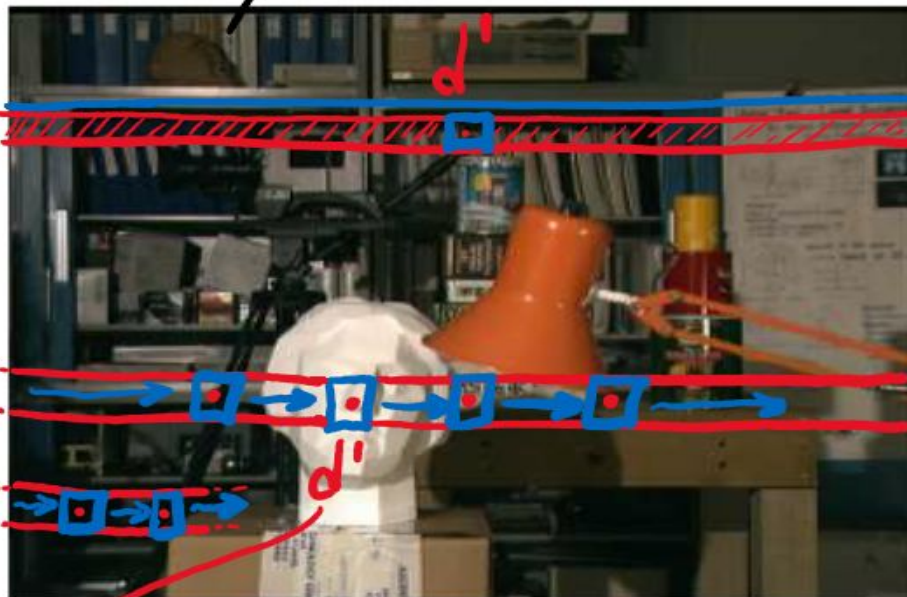
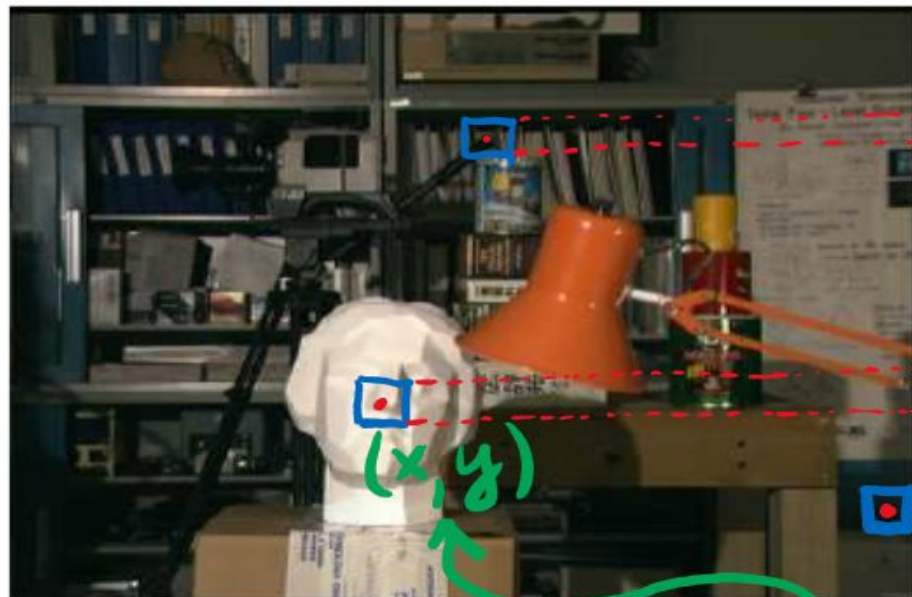
правая



Горизонтальное смещение = параллакс = диспаритет = d

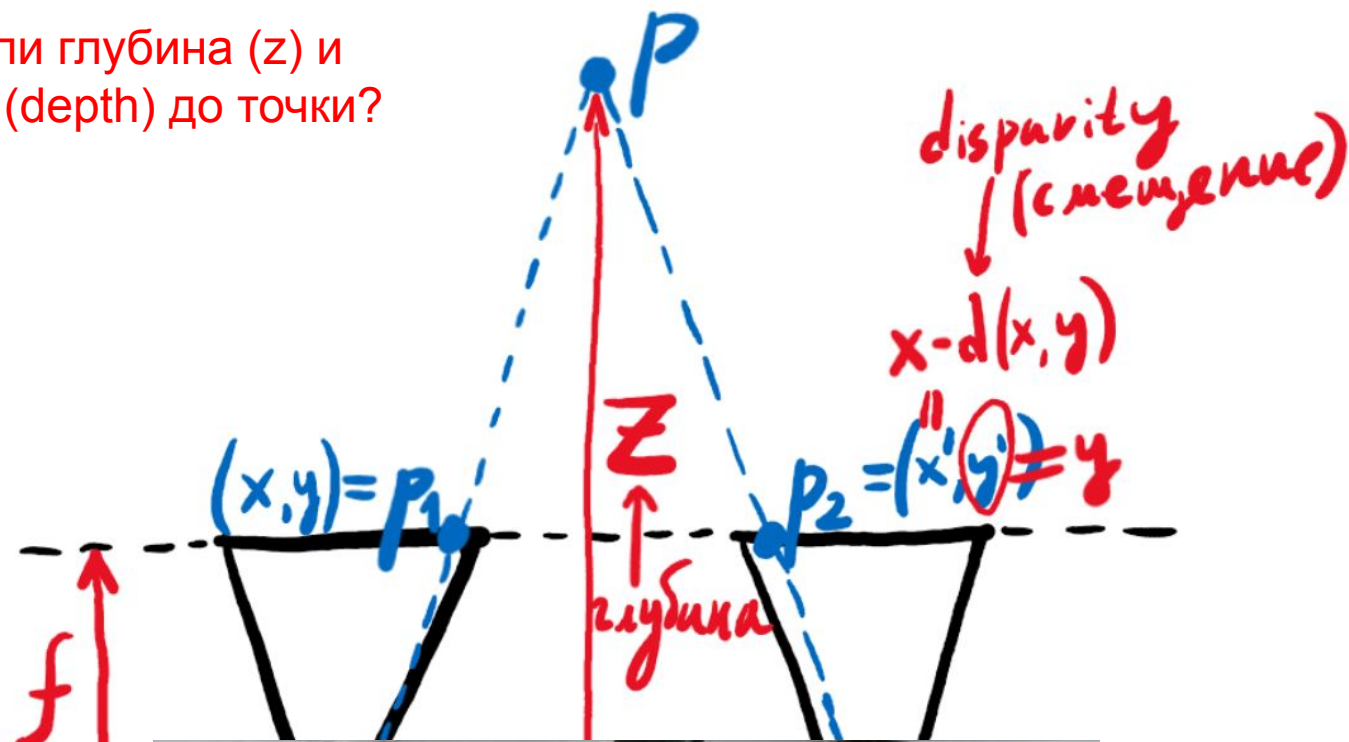
левая

правая



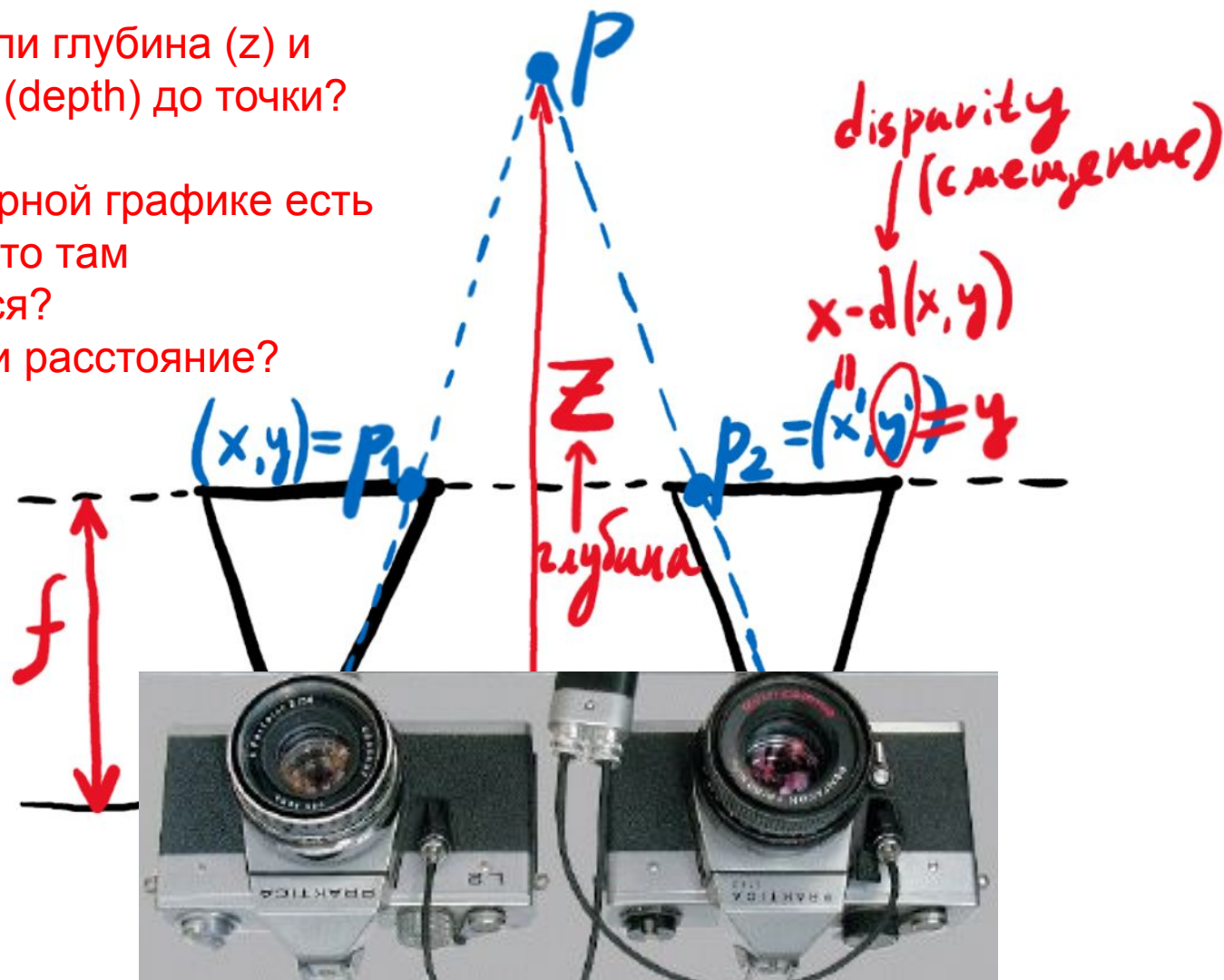
Как соотносятся координаты сопоставлений? (по оси x/y)

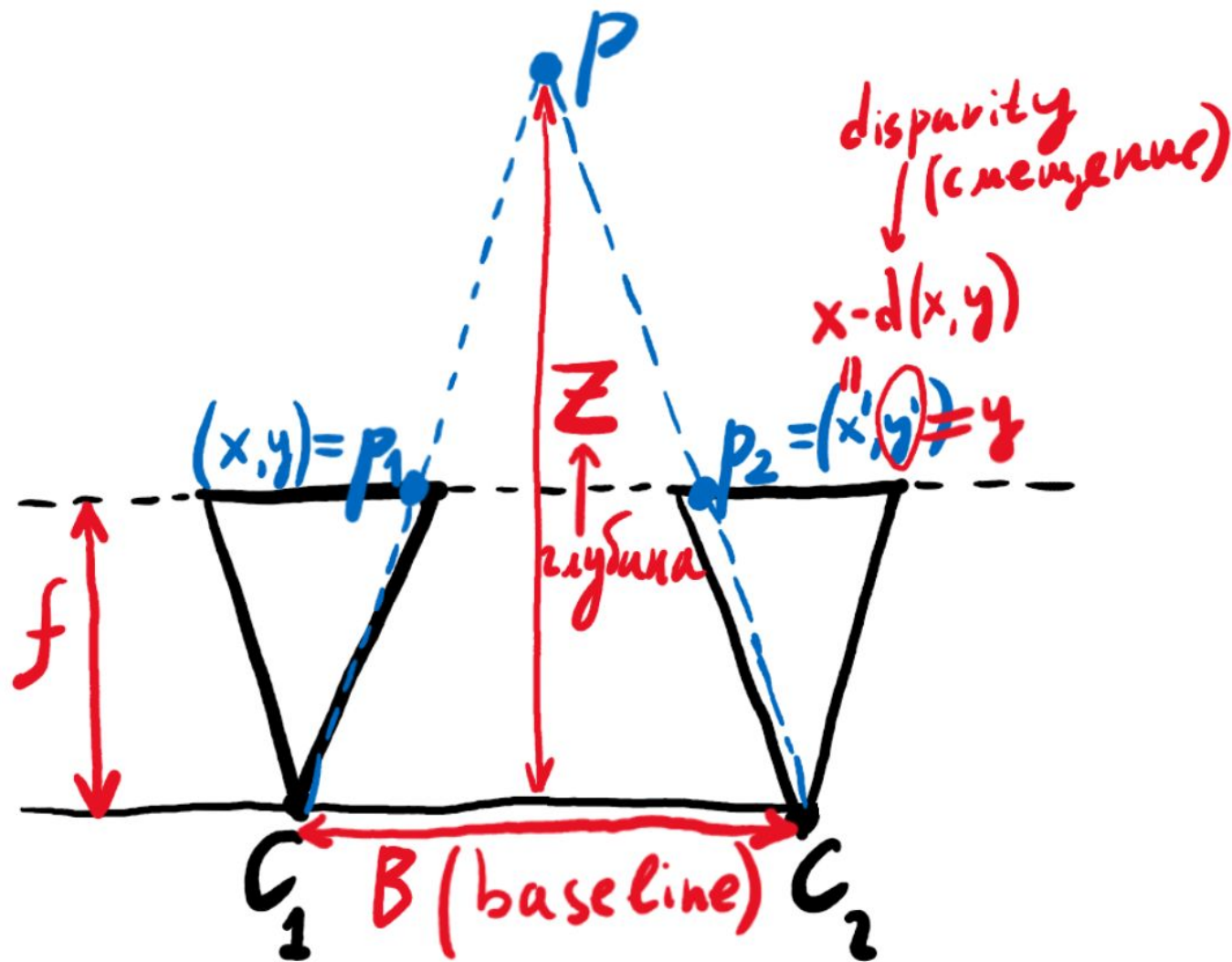
Совпадает ли глубина (z) и
расстояние (depth) до точки?

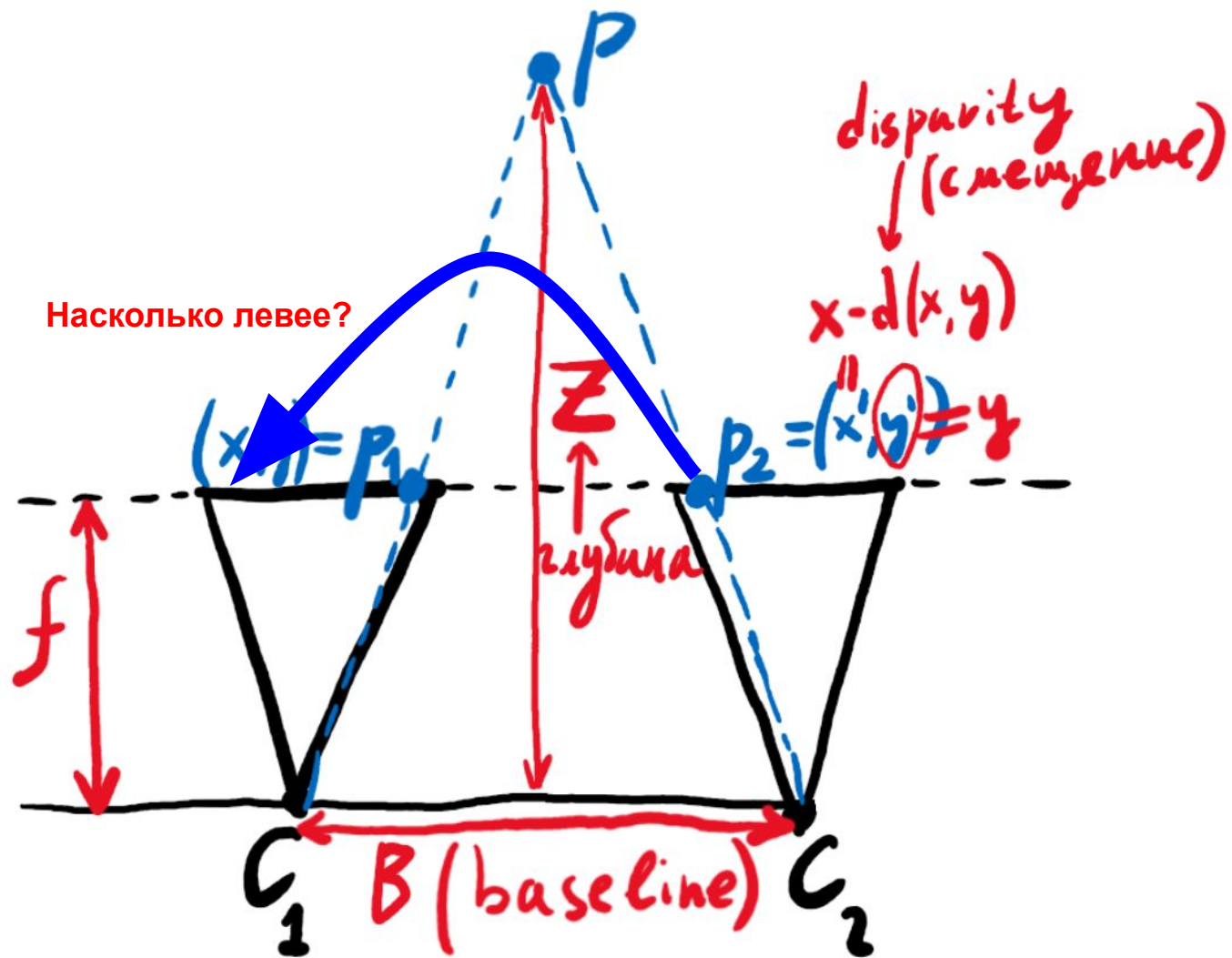


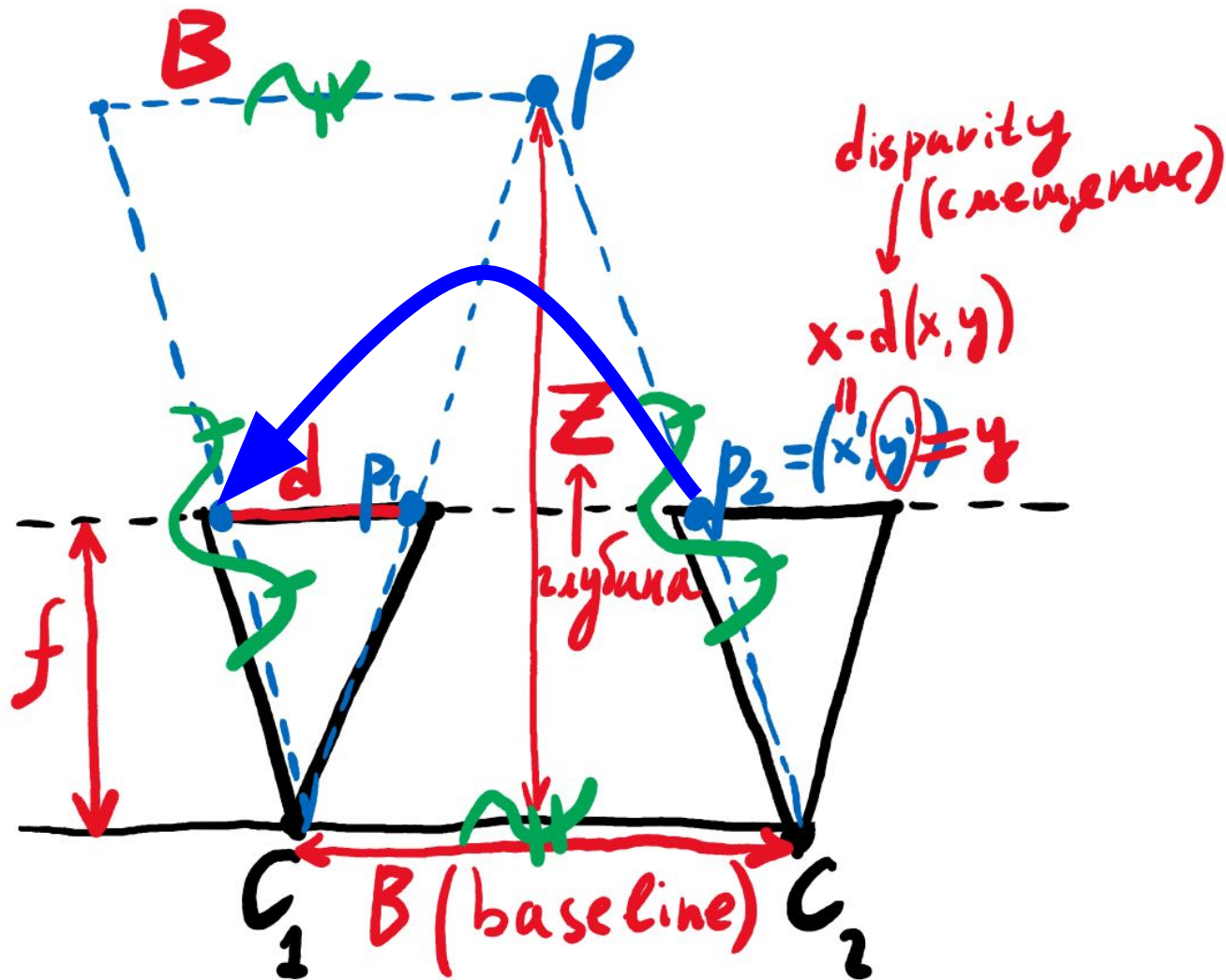
Совпадает ли глубина (z) и расстояние (depth) до точки?

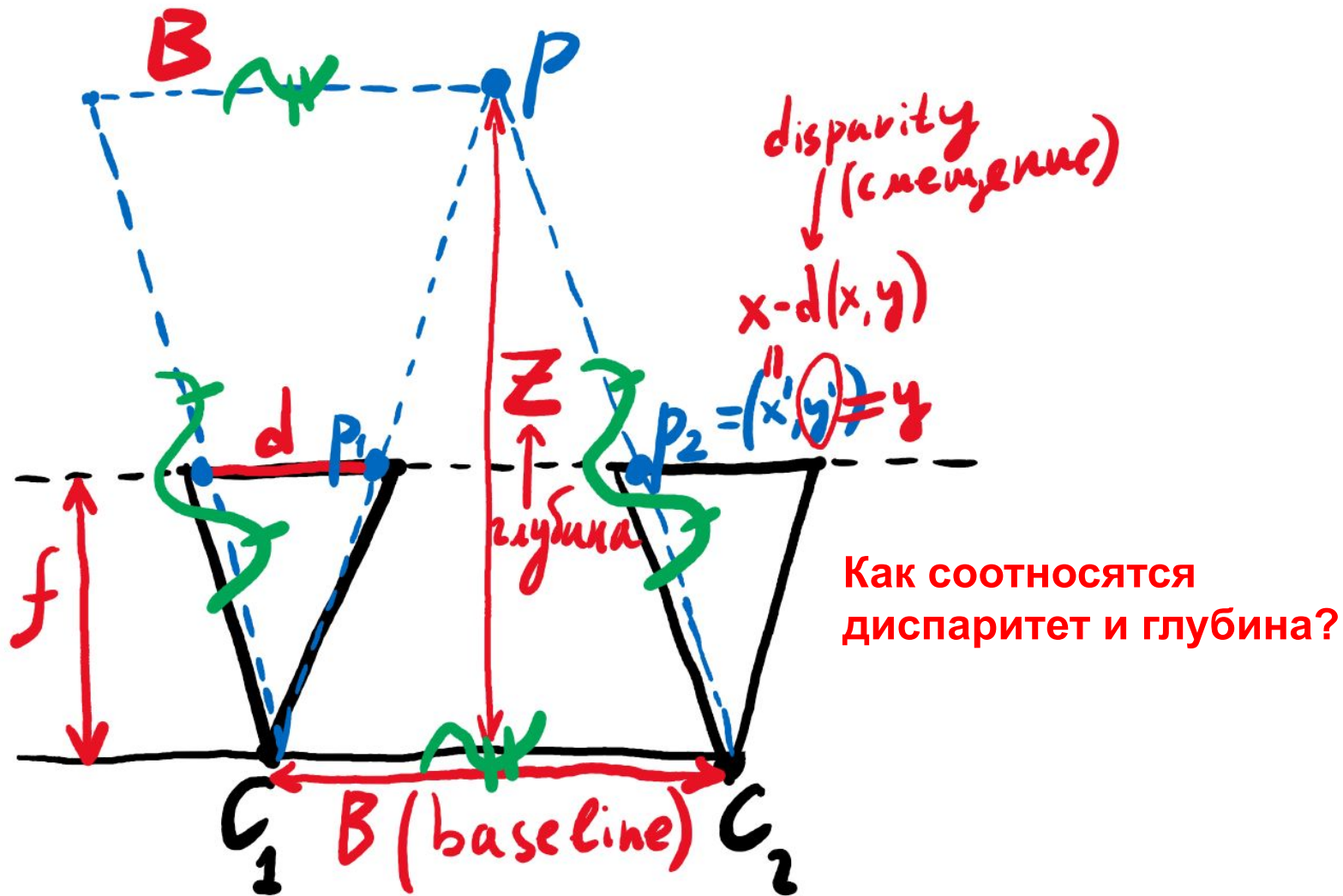
В компьютерной графике есть depth-test, что там используется?
Глубина или расстояние?

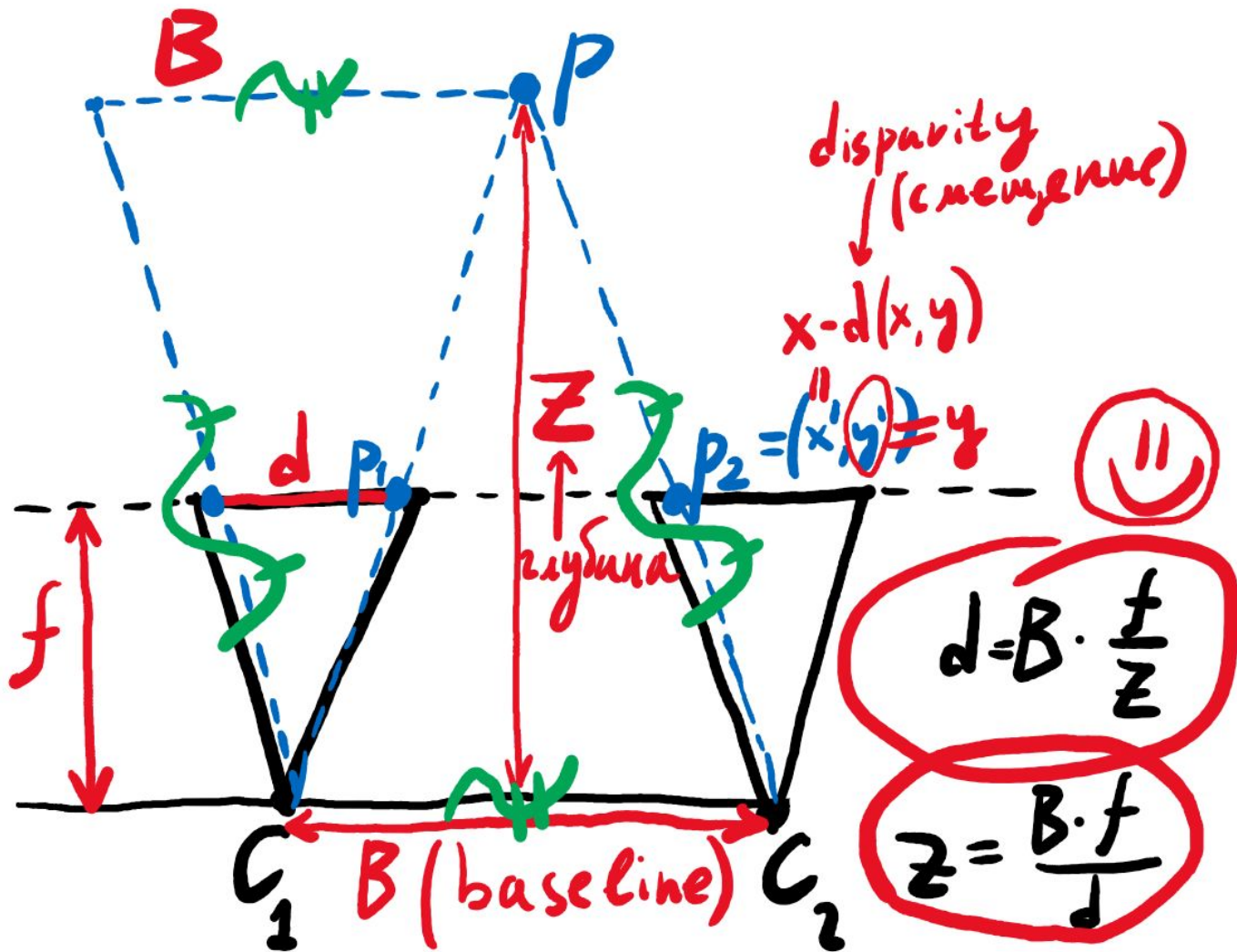


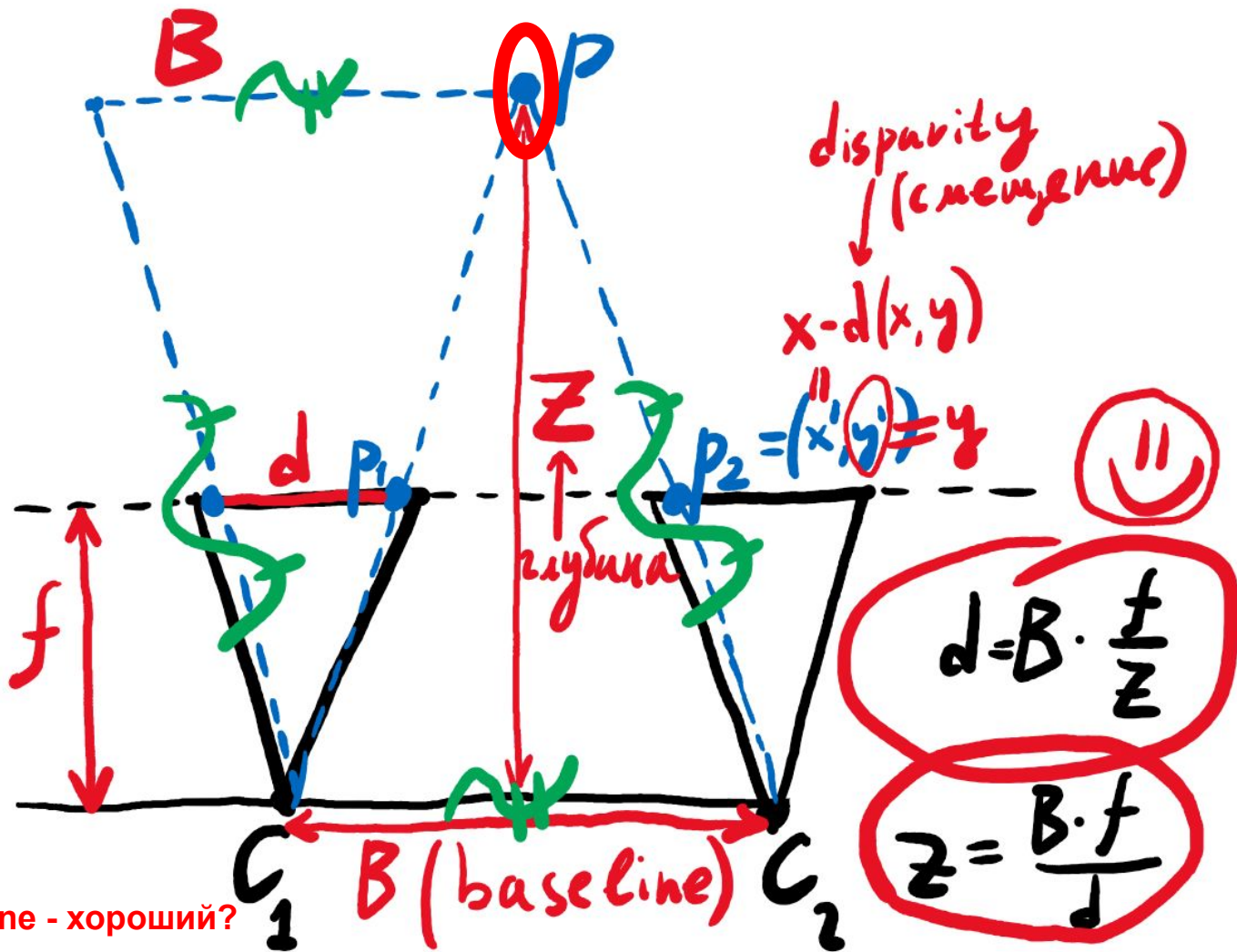










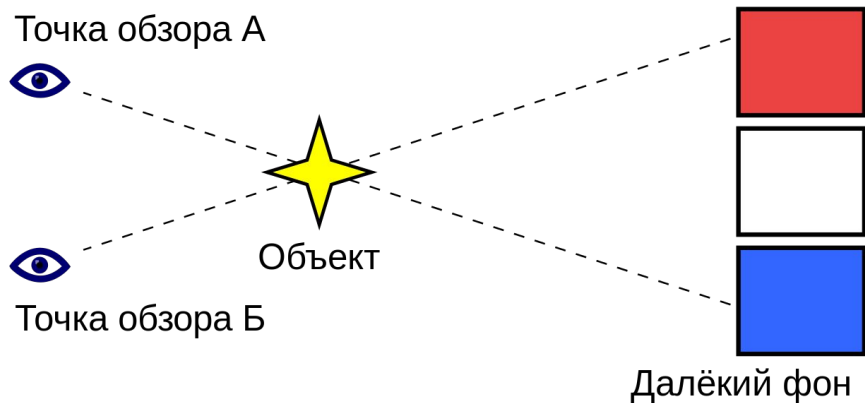


А какой baseline - хороший?

Стереопара

За счет чего человек “видит” расстояния до объектов и в целом сцену перед ним?

За счет пары глаз (две картинки - **стереопара**) и смещения (**параллакса**):

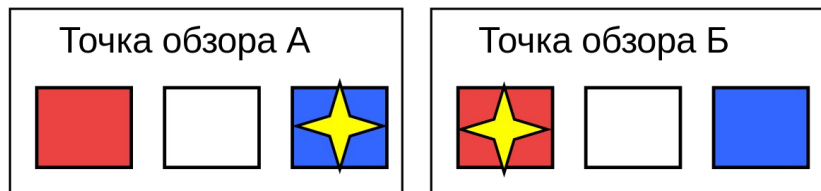


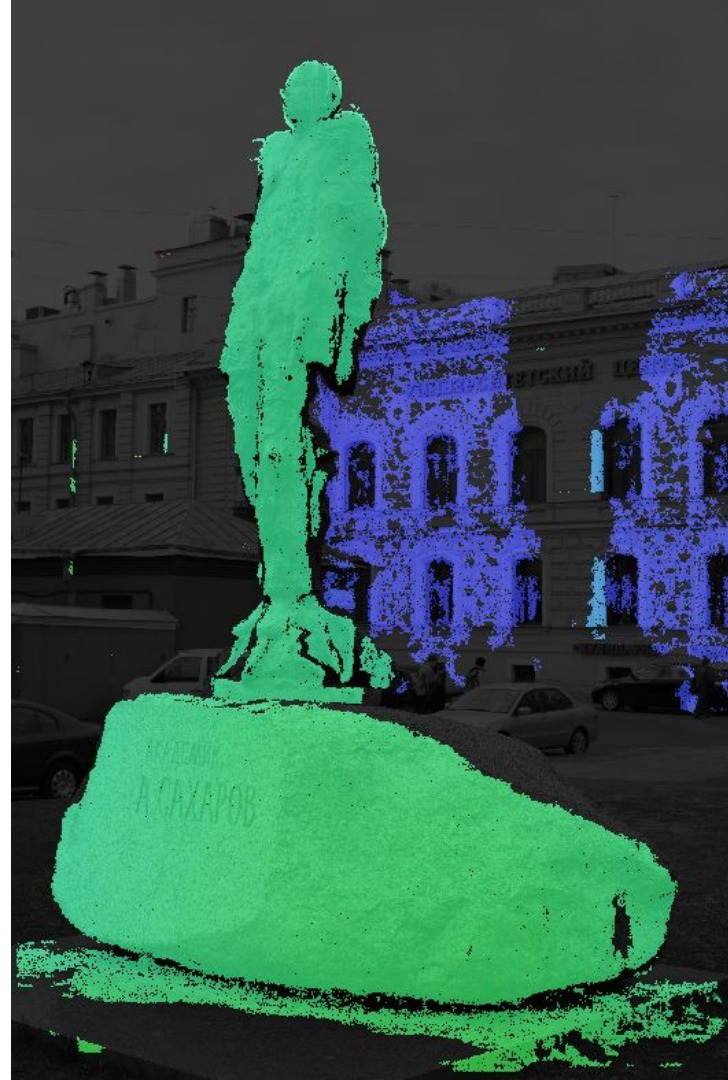
Почему смещение по горизонтали?

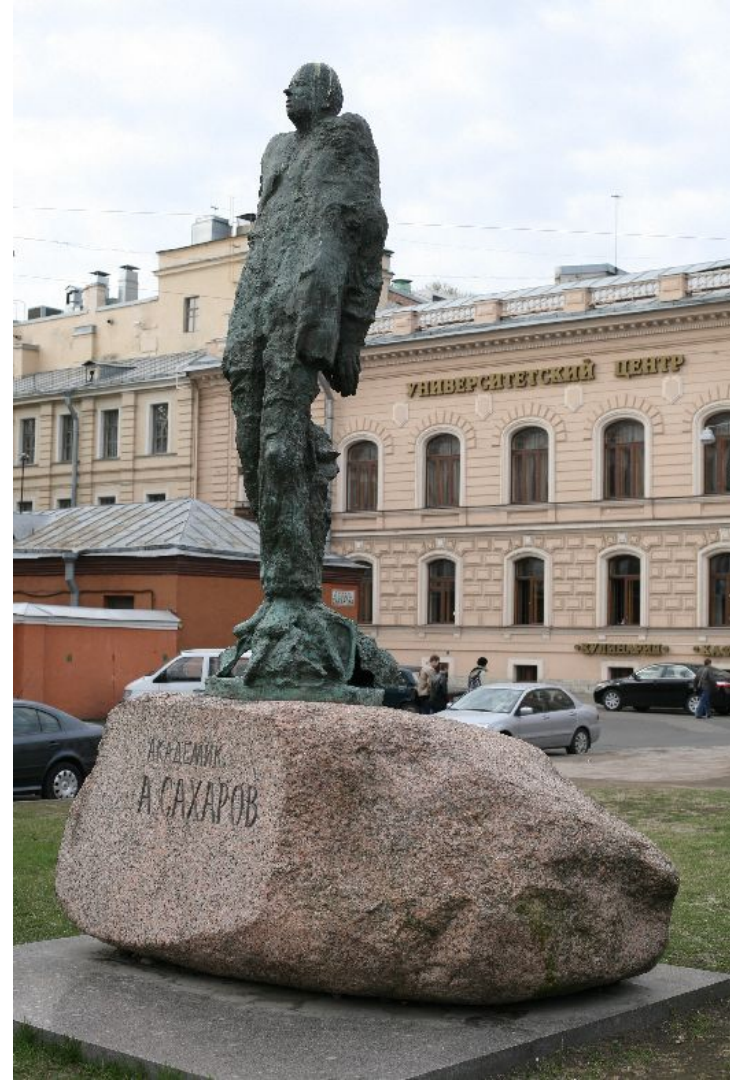
Что это за горизонтальная прямая?

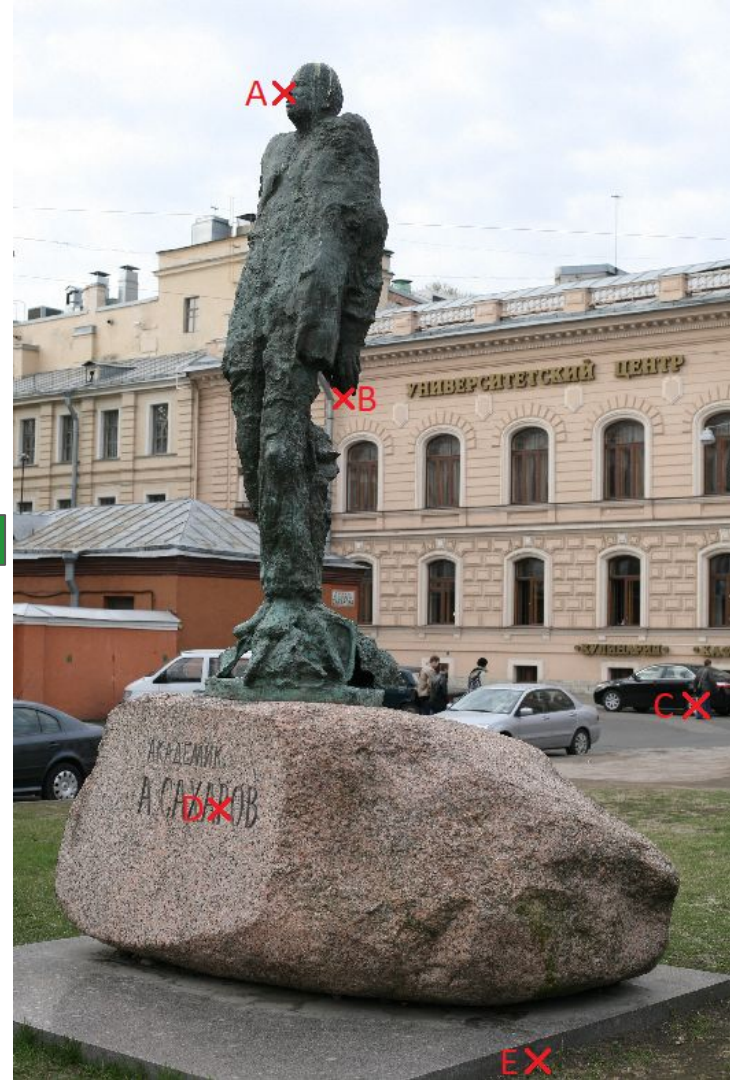
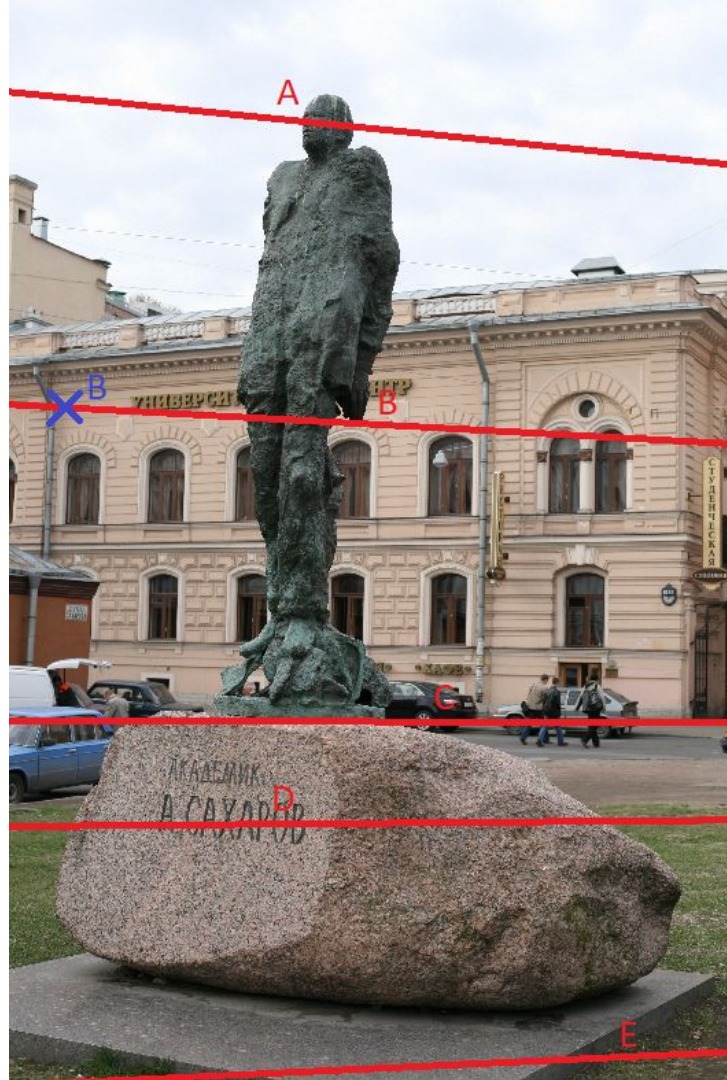
Какая зависимость расстояния до объекта от смещения в стереопаре?

Как выглядит сопоставление для произвольной пары фотографий?









Эпиполярная геометрия

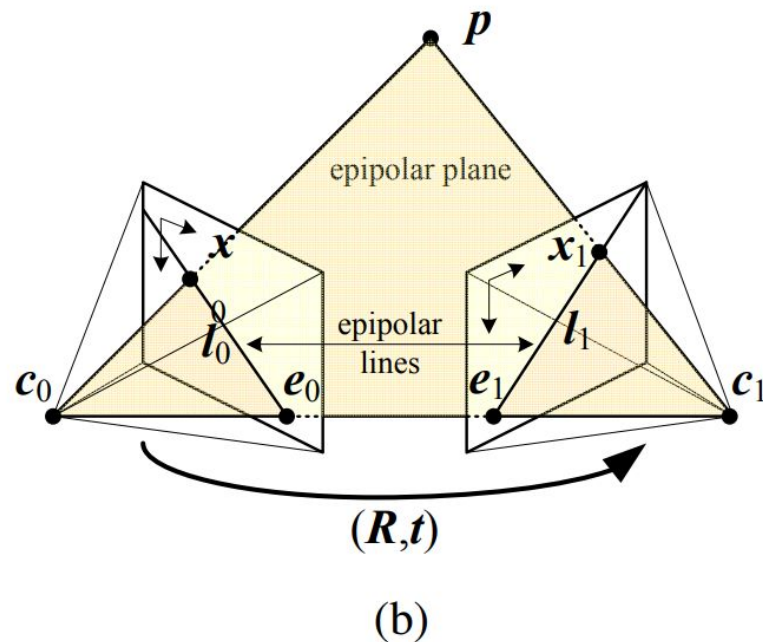
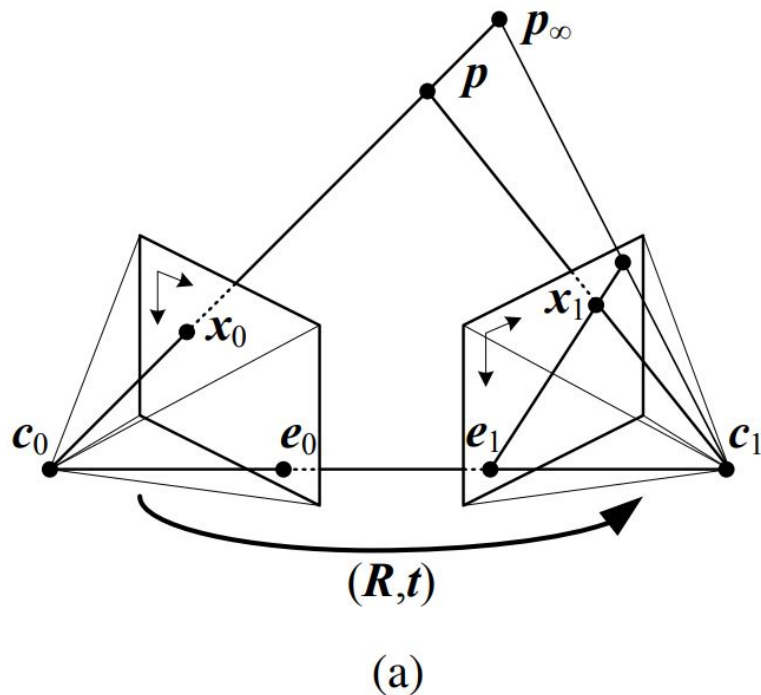


Figure 12.3 Epipolar geometry: (a) epipolar line segment corresponding to one ray; (b) corresponding set of epipolar lines and their epipolar plane.

Эпиполярная геометрия

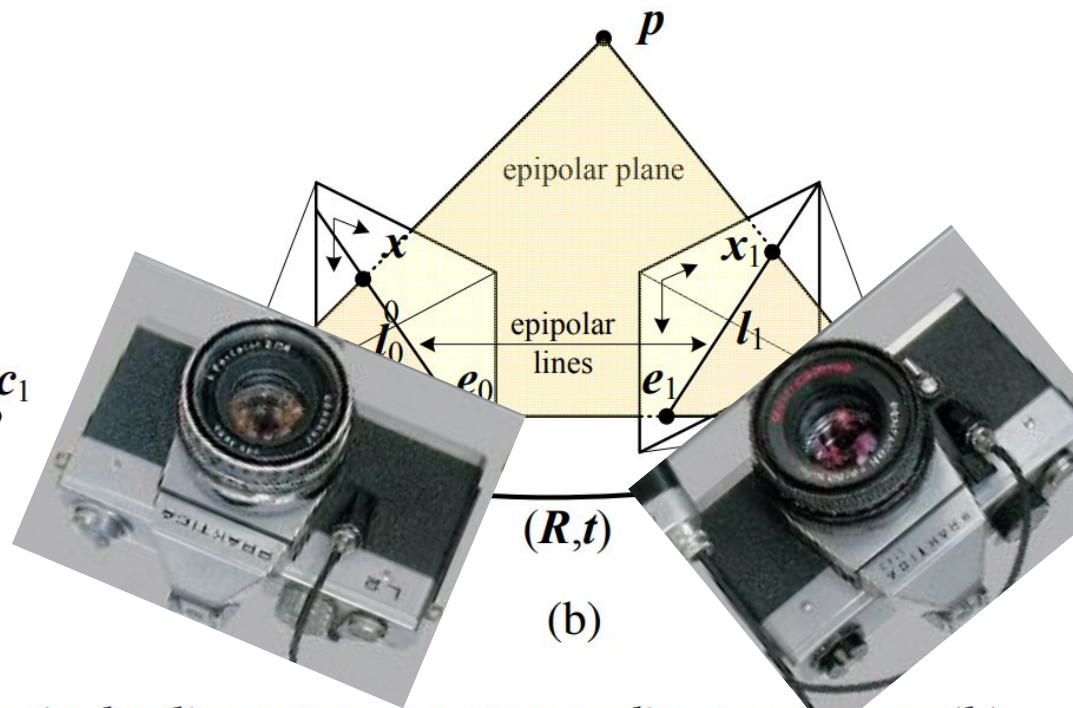
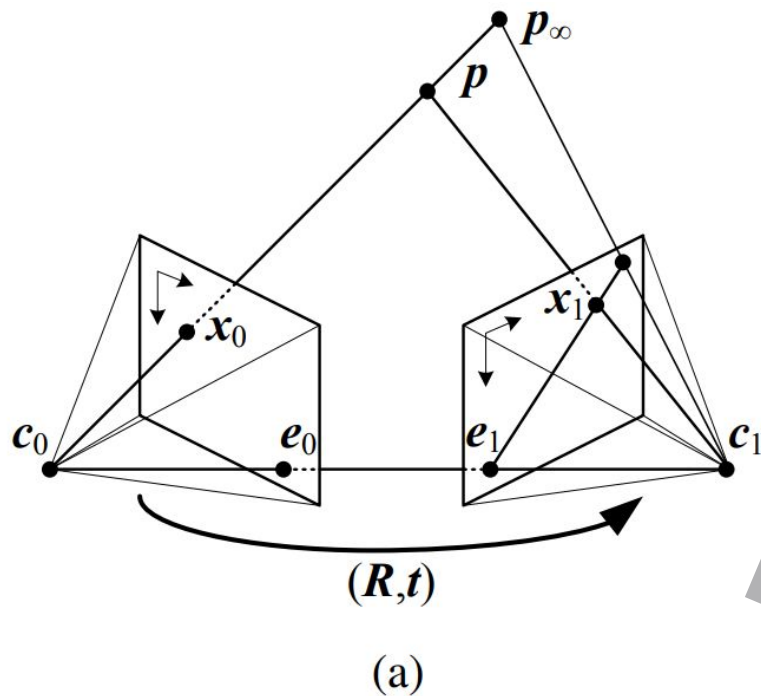
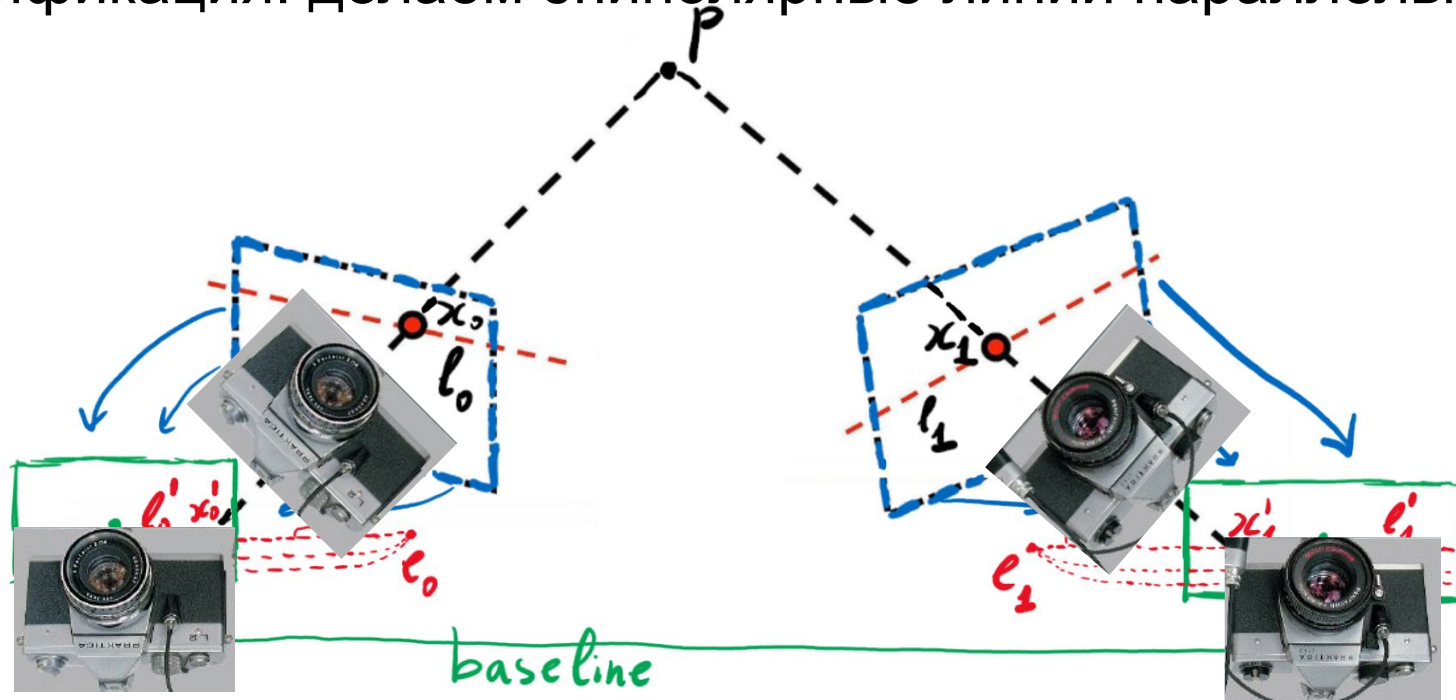


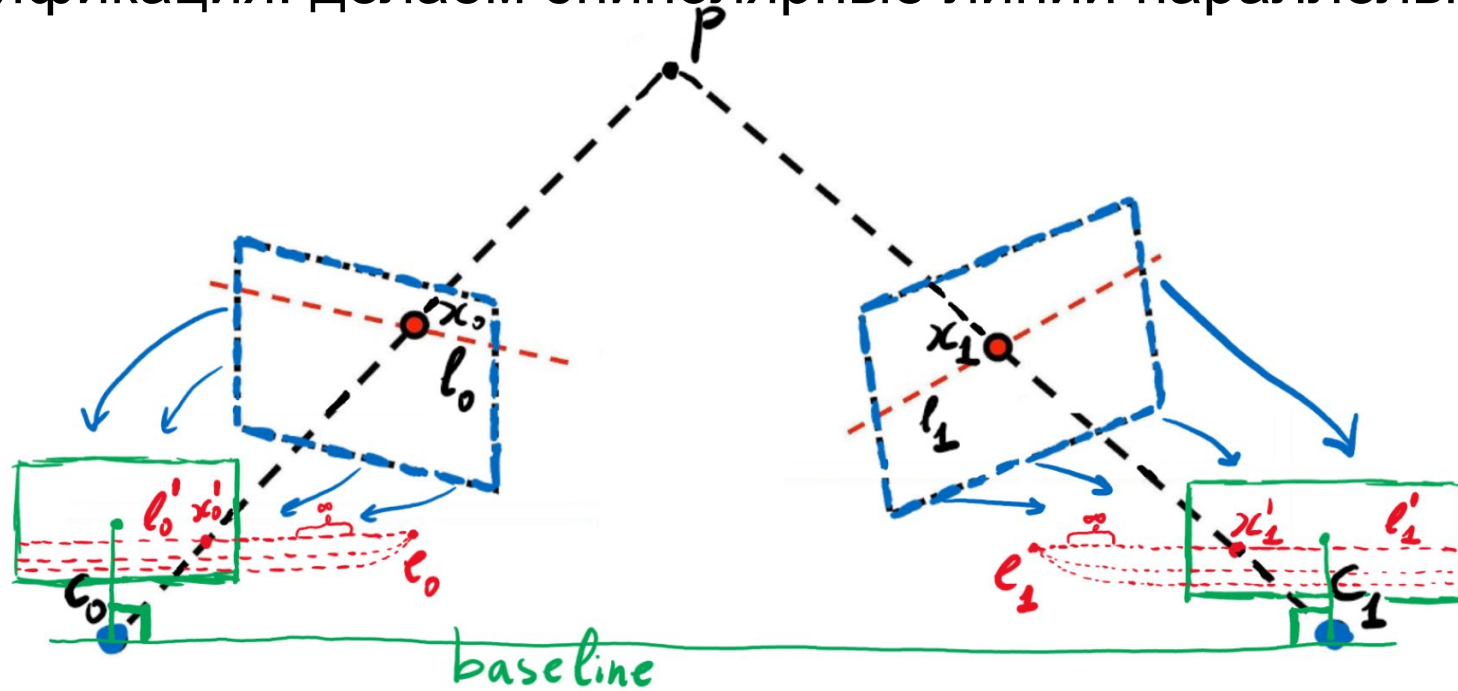
Figure 12.3 Epipolar geometry: (a) epipolar line segment corresponding to one ray; (b) corresponding set of epipolar lines and their epipolar plane.

Ректификация: делаем эпиполярные линии параллельными



Рассмотрим **воображаемую камеру** чья оптическая ось перпендикулярна прямой (**baseline**) между центрами камер. Иначе говоря выкинули эпиполисы на бесконечность (**epipole**: проекция одной камеры в другую - т.е. точка пересечения всех эпиполярных линий). Оставшаяся степень свободы - поворот вокруг оптической оси (**twist**) - делает эпиполярные линии горизонтальными.

Ректификация: делаем эпиполярные линии параллельными



Рассмотрим **воображаемую камеру** чья оптическая ось перпендикулярна прямой (**baseline**) между центрами камер. Иначе говоря выкинули эпиполисы на бесконечность (**epipole**: проекция одной камеры в другую - т.е. точка пересечения всех эпиполярных линий). Оставшаяся степень свободы - поворот вокруг оптической оси (**twist**) - делает эпиполярные линии горизонтальными.

Ректификация: делаем эпиполярные линии параллельными

- 1) Поворачиваем так чтобы оптическая ось стала перпендикулярна прямой baseline, т. е. делаем эпиполярные линии параллельными.

Image 1 and Epipole

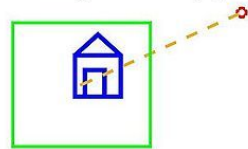


Image 2 and Epipole

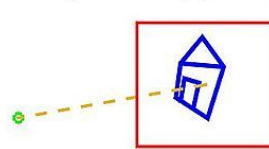


Image 1 Rotated

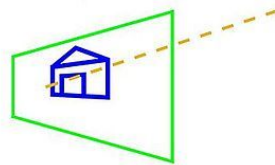
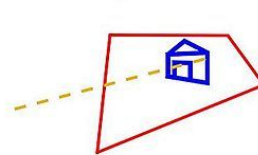
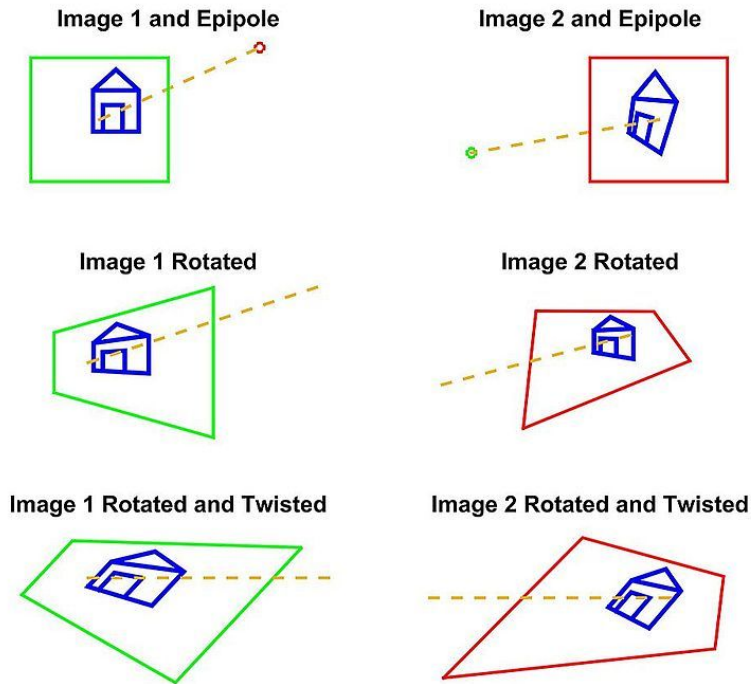


Image 2 Rotated



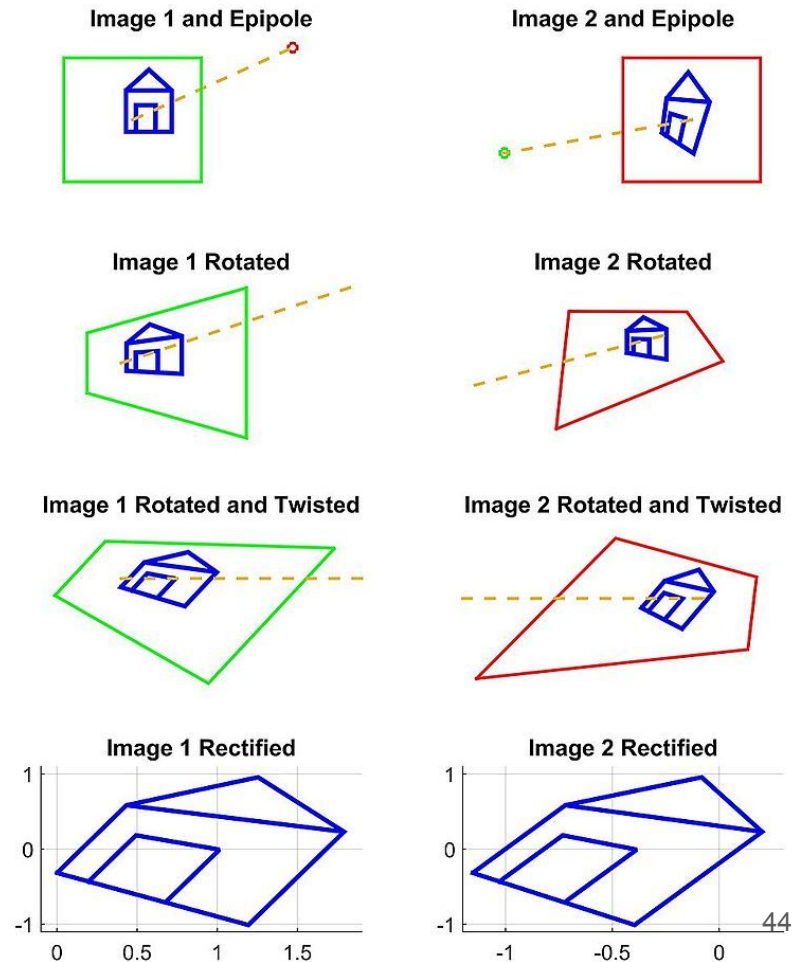
Ректификация: делаем эпиполярные линии параллельными

- 1) Поворачиваем так чтобы оптическая ось стала перпендикулярна прямой baseline, т.е. делаем эпиполярные линии параллельными.
- 2) Поворачиваем вокруг оптической оси (twist), т.е. делаем их горизонтальными.



Ректификация: делаем эпиполярные линии параллельными

- 1) Поворачиваем так чтобы оптическая ось стала перпендикулярна прямой baseline, т.е. делаем эпиполярные линии параллельными.
- 2) Поворачиваем вокруг оптической оси (twist), т.е. делаем их горизонтальными.
- 3) Добавляем масштаб:
 - **чтобы** выровнять фокальные длины (если разные)
 - **чтобы** не терять информацию более детальной фотографии
 - **чтобы** ослабить горизонтальные искажения.



Ректификация: делаем эпиполярные линии параллельными

- 1) Поворачиваем так чтобы оптическая ось стала перпендикулярна прямой baseline, т.е. делаем эпиполярные линии параллельными.
- 2) Поворачиваем вокруг оптической оси (twist), т.е. делаем их горизонтальными.
- 3) Добавляем масштаб

Image 1 and Epipole



Image 2 and Epipole

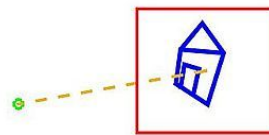


Image 1 Rotated

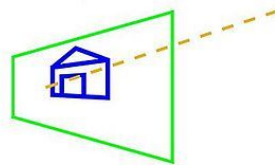


Image 2 Rotated

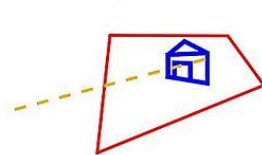


Image 1 Rotated and Twisted

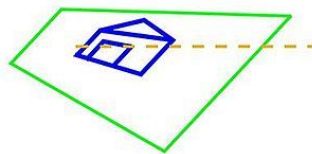


Image 2 Rotated and Twisted

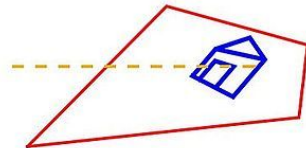


Image 1 Rectified

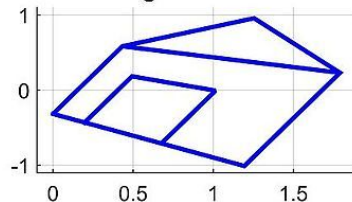
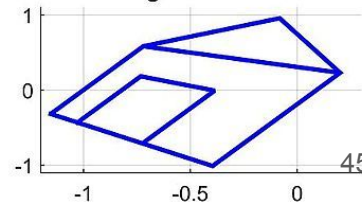


Image 2 Rectified



Но мы сделали 3 преобразования!

А значит 3 интерполяции картинки, получается суммарно потеряли очень много деталей?

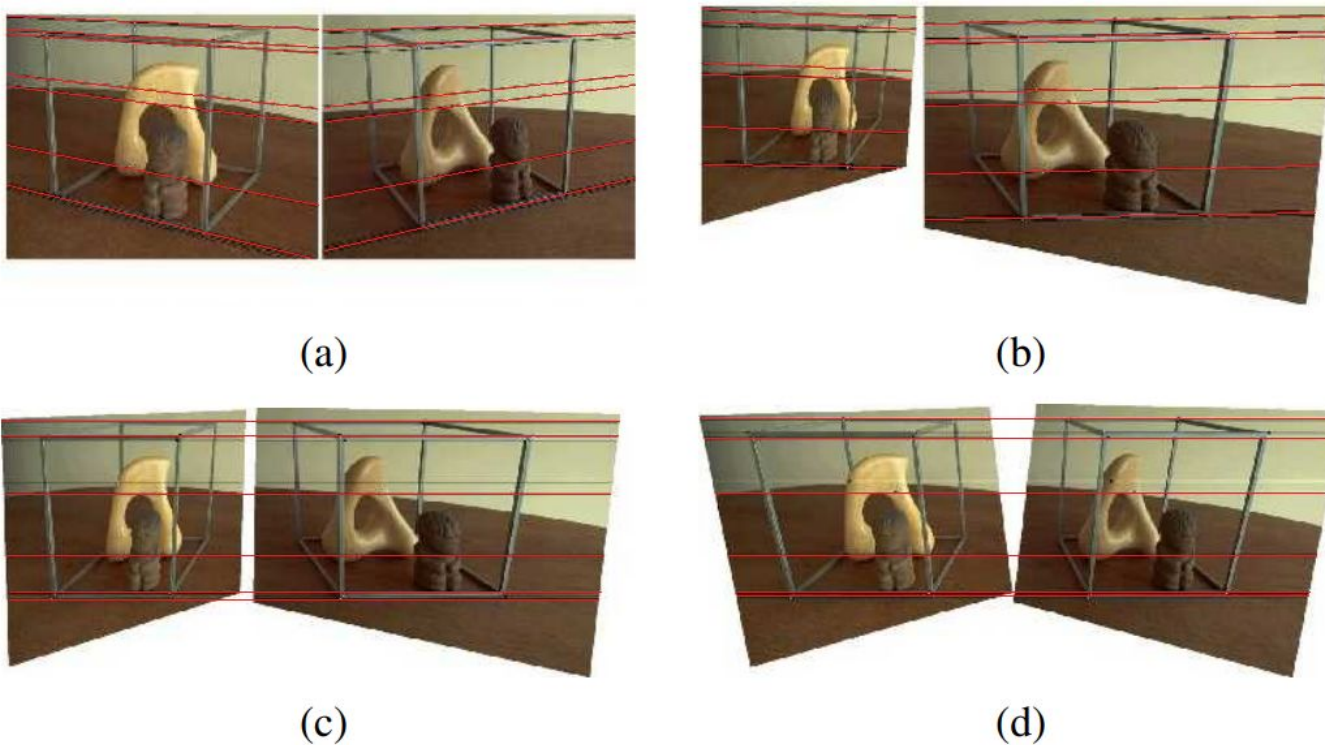
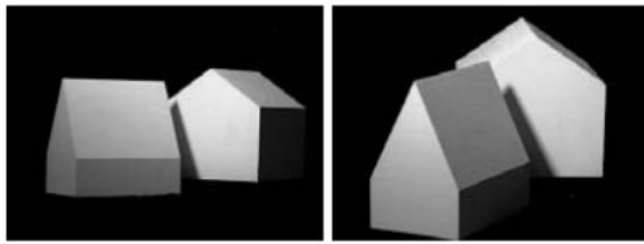
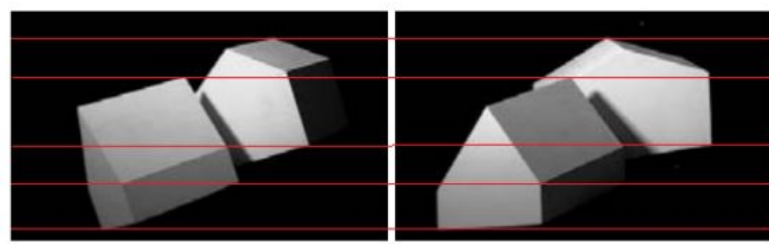


Figure 12.4 *The multi-stage stereo rectification algorithm of Loop and Zhang (1999) © 1999 IEEE. (a) Original image pair overlaid with several epipolar lines; (b) images transformed so that epipolar lines are parallel; (c) images rectified so that epipolar lines are horizontal and in vertical correspondence; (d) final rectification that minimizes horizontal distortions.*



a



b

Fig. 11.11. **Image rectification.** (a) A pair of images of a house. (b) Resampled images computed from (a) using a projective transformation computed from F . Note, corresponding points in (b) match horizontally.



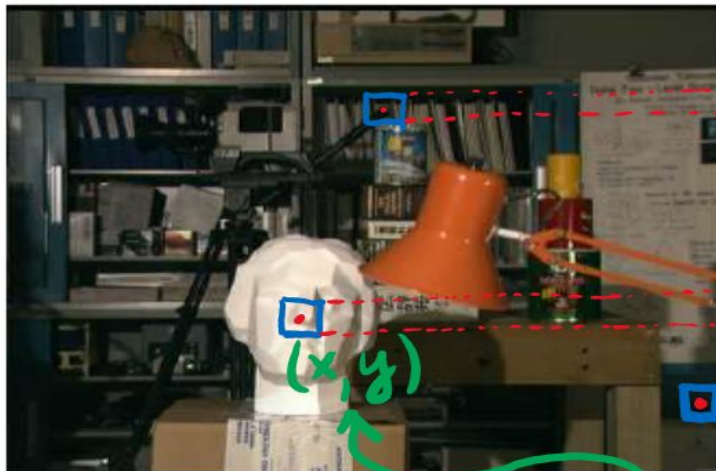
a



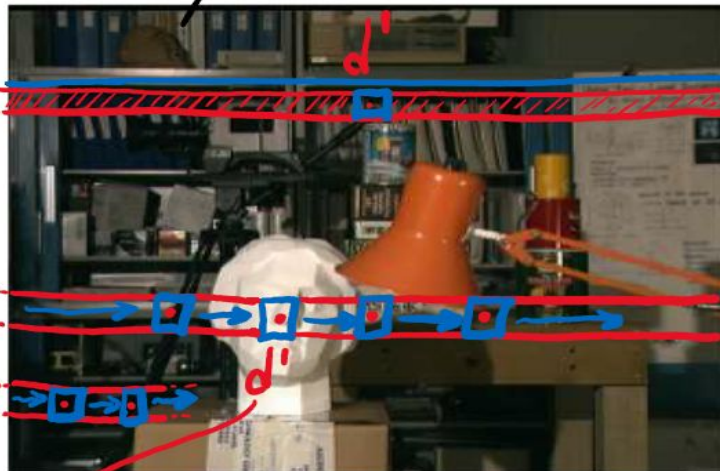
b

Fig. 11.12. **Image rectification using affinities.** (a) A pair of original images and (b) a detail of the images rectified using affine transformations. The average y -disparity after rectification is of the order of 3 pixels in a 512×512 image. (For correctly rectified images the y -disparity should be zero.)

левая

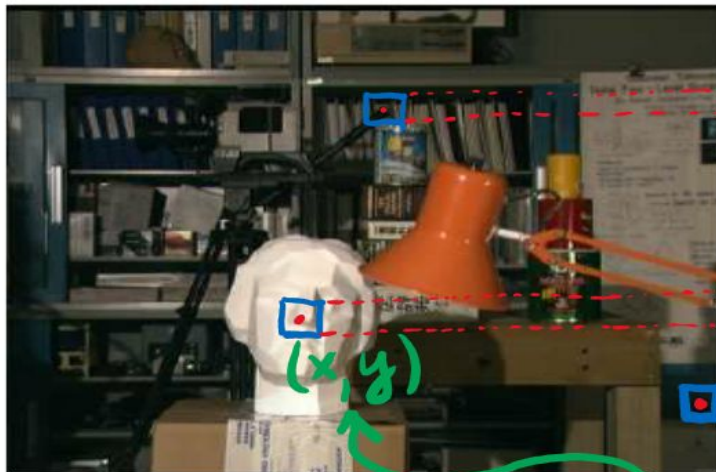


правая

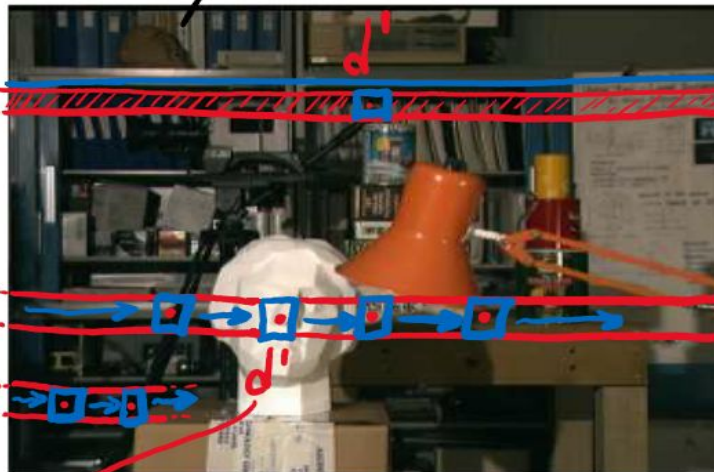


$$d(x, y) = d' = \underset{d}{\operatorname{argmin}} \operatorname{cost}(x, y, d)$$

левая

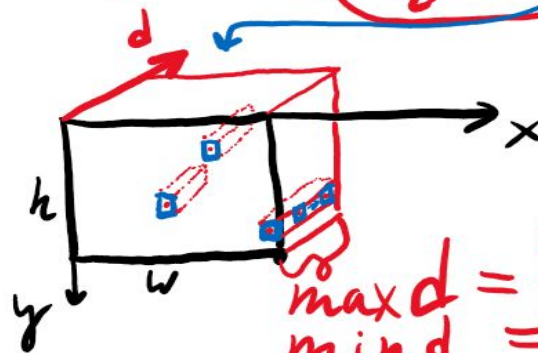


правая



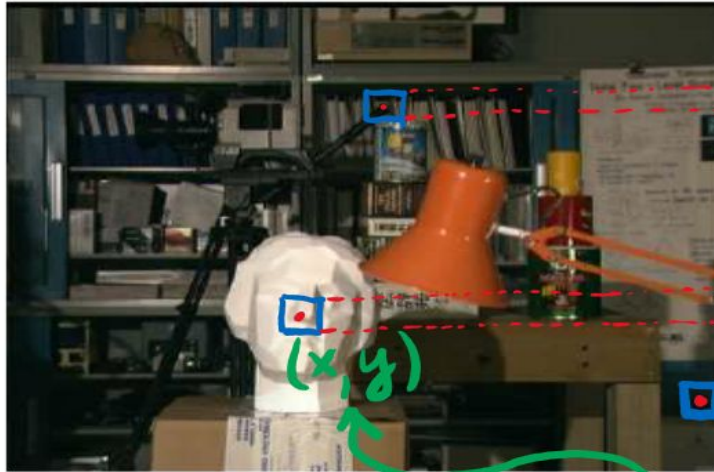
\vec{v}_d

$$d(x, y) = d' = \underset{d}{\operatorname{argmin}} \operatorname{cost}(x, y, d)$$

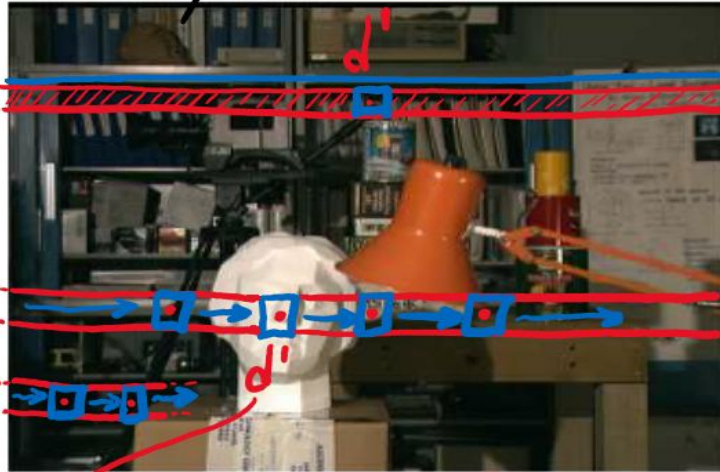


$\max d = ?$
 $\min d = ?$
 $\operatorname{cost} = ?$

левая

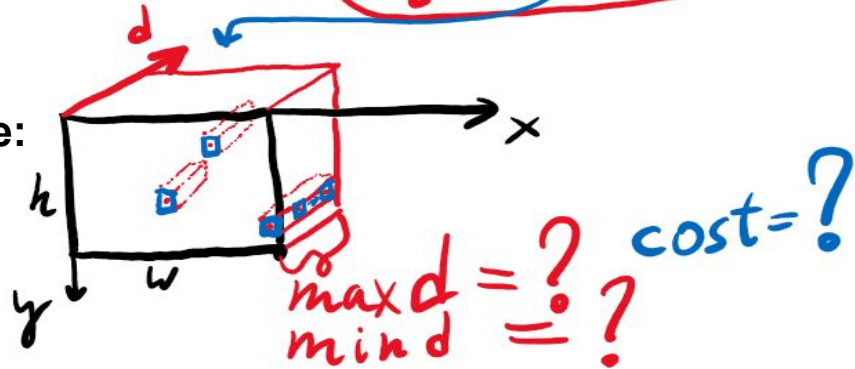


правая



$$d(x, y) = d' = \underset{d}{\operatorname{argmin}} \operatorname{cost}(x, y, d)$$

DSI (disparity space image) volume:



Cost функции

1) Absolute Difference

$$C_{AD}(\mathbf{p}, \mathbf{d}) = |I_L(\mathbf{p}) - I_R(\mathbf{p} - \mathbf{d})|$$

2) Sum of Absolute Differences

$$C_{SAD}(\mathbf{p}, \mathbf{d}) = \sum_{\mathbf{q} \in N_{\mathbf{p}}} |I_L(\mathbf{q}) - I_R(\mathbf{q} - \mathbf{d})|$$

3) Но что если яркость в картинках отличается?

Cost функции

1) Absolute Difference

$$C_{AD}(\mathbf{p}, \mathbf{d}) = |I_L(\mathbf{p}) - I_R(\mathbf{p} - \mathbf{d})|$$

2) Sum of Absolute Differences

$$C_{SAD}(\mathbf{p}, \mathbf{d}) = \sum_{\mathbf{q} \in N_{\mathbf{p}}} |I_L(\mathbf{q}) - I_R(\mathbf{q} - \mathbf{d})|$$

3) Zero-mean SAD

$$C_{ZSAD}(\mathbf{p}, \mathbf{d}) = \sum_{\mathbf{q} \in N_{\mathbf{p}}} |I_L(\mathbf{q}) - \bar{I}_L(\mathbf{p}) - I_R(\mathbf{q} - \mathbf{d}) + \bar{I}_R(\mathbf{p} - \mathbf{d})|$$

$$\bar{I}(\mathbf{p}) = \frac{1}{N_{\mathbf{p}}} \sum_{\mathbf{q} \in N_{\mathbf{p}}} I(\mathbf{q})$$

Cost функции

1) Absolute Difference

$$C_{AD}(\mathbf{p}, \mathbf{d}) = |I_L(\mathbf{p}) - I_R(\mathbf{p} - \mathbf{d})|$$

2) Sum of Absolute Differences

$$C_{SAD}(\mathbf{p}, \mathbf{d}) = \sum_{\mathbf{q} \in N_p} |I_L(\mathbf{q}) - I_R(\mathbf{q} - \mathbf{d})|$$

3) Zero-mean SAD

$$C_{ZSAD}(\mathbf{p}, \mathbf{d}) = \sum_{\mathbf{q} \in N_p} |I_L(\mathbf{q}) - \bar{I}_L(\mathbf{p}) - I_R(\mathbf{q} - \mathbf{d}) + \bar{I}_R(\mathbf{p} - \mathbf{d})|$$

$$\bar{I}(\mathbf{p}) = \frac{1}{N_p} \sum_{\mathbf{q} \in N_p} I(\mathbf{q})$$

4) Normalized Cross-Correlation

$$C_{NCC}(\mathbf{p}, \mathbf{d}) = \frac{\sum_{\mathbf{q} \in N_p} I_L(\mathbf{q}) I_R(\mathbf{q} - \mathbf{d})}{\sqrt{\sum_{\mathbf{q} \in N_p} I_L(\mathbf{q})^2 \sum_{\mathbf{q} \in N_p} I_R(\mathbf{q} - \mathbf{d})^2}}$$

- Стат. оптимальна при гауссовом шуме
- Устойчива при разном значении gain (“масштаб” яркости) за счет деления на среднеквадратичное отклонение
- Чувствительна к выбросам (например на границе объекта, т.е. рядом с заслонением - с occlusion), какой выбирать размер патча?
- И что если яркость в картинках отличается?

Cost функции

1) Absolute Difference

$$C_{AD}(\mathbf{p}, \mathbf{d}) = |I_L(\mathbf{p}) - I_R(\mathbf{p} - \mathbf{d})|$$

2) Sum of Absolute Differences

$$C_{SAD}(\mathbf{p}, \mathbf{d}) = \sum_{\mathbf{q} \in N_{\mathbf{p}}} |I_L(\mathbf{q}) - I_R(\mathbf{q} - \mathbf{d})|$$

3) Zero-mean SAD

$$C_{ZSAD}(\mathbf{p}, \mathbf{d}) = \sum_{\mathbf{q} \in N_{\mathbf{p}}} |I_L(\mathbf{q}) - \bar{I}_L(\mathbf{p}) - I_R(\mathbf{q} - \mathbf{d}) + \bar{I}_R(\mathbf{p} - \mathbf{d})|$$

$$\bar{I}(\mathbf{p}) = \frac{1}{N_{\mathbf{p}}} \sum_{\mathbf{q} \in N_{\mathbf{p}}} I(\mathbf{q})$$

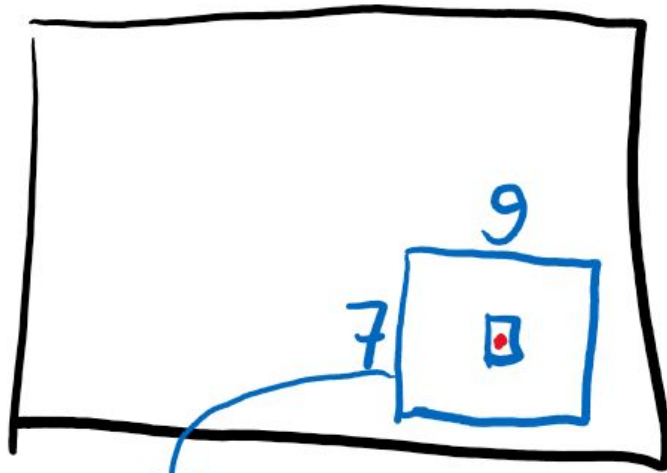
4) Normalized Cross-Correlation

$$C_{NCC}(\mathbf{p}, \mathbf{d}) = \frac{\sum_{\mathbf{q} \in N_{\mathbf{p}}} I_L(\mathbf{q}) I_R(\mathbf{q} - \mathbf{d})}{\sqrt{\sum_{\mathbf{q} \in N_{\mathbf{p}}} I_L(\mathbf{q})^2 \sum_{\mathbf{q} \in N_{\mathbf{p}}} I_R(\mathbf{q} - \mathbf{d})^2}}$$

5) Zero-mean NCC

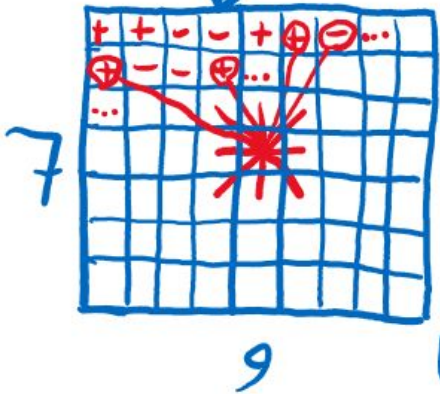
$$C_{ZNCC}(\mathbf{p}, \mathbf{d}) = \frac{\sum_{\mathbf{q} \in N_{\mathbf{p}}} (I_L(\mathbf{q}) - \bar{I}_L(\mathbf{p})) (I_R(\mathbf{q} - \mathbf{d}) - \bar{I}_R(\mathbf{p} - \mathbf{d}))}{\sqrt{\sum_{\mathbf{q} \in N_{\mathbf{p}}} (I_L(\mathbf{q}) - \bar{I}_L(\mathbf{p}))^2 \sum_{\mathbf{q} \in N_{\mathbf{p}}} (I_R(\mathbf{q} - \mathbf{d}) - \bar{I}_R(\mathbf{p} - \mathbf{d}))^2}}$$

Cost функции: Census



Census

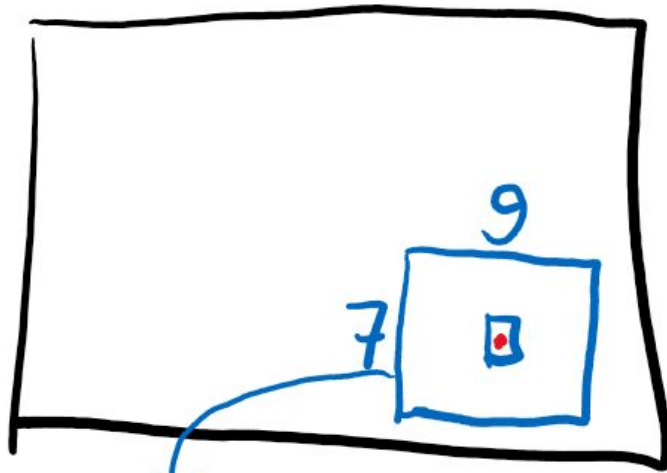
- битовый вектор описывающий патч вокруг пикселя и состоящий из $7 \cdot 9 - 1 = 62$ бит его можно преподсчитать для обеих картинок стереопары.



+ (\Rightarrow) ярче центра
- (\Rightarrow) темнее
пикселе в центре

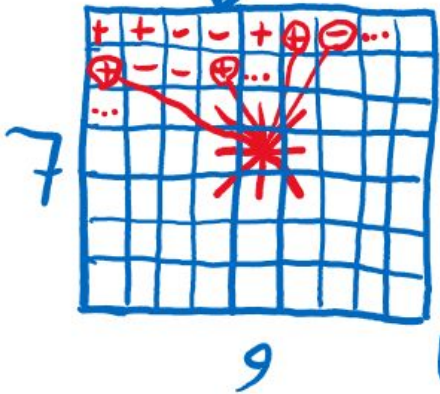
patch[7][9]

Cost функции: Census



Census

- битовый вектор описывающий патч вокруг пикселя и состоящий из $7 \cdot 9 - 1 = 62$ бит его можно преподсчитать для обеих картинок стереопары.

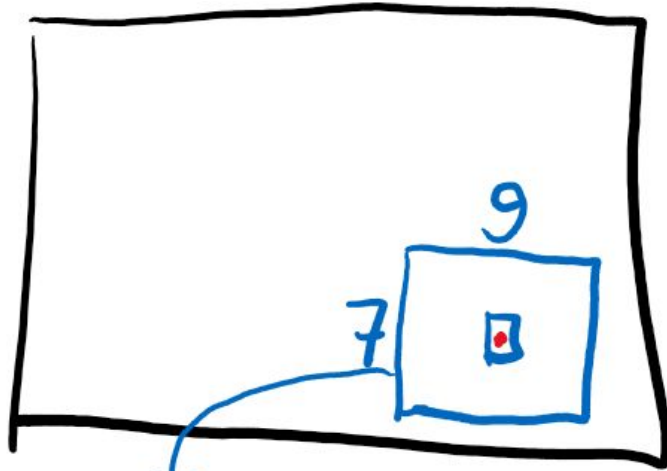


+ (\Leftrightarrow) ярче центра
- (\Leftrightarrow) темнее
пикселе в центре

patch[7][9]

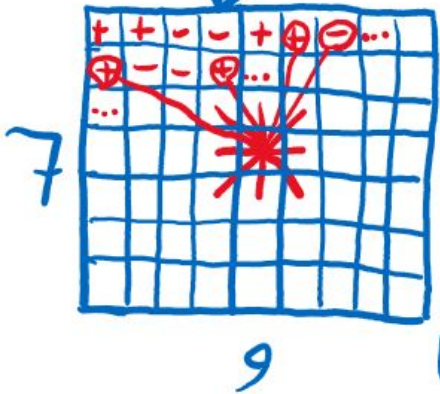
Какая метрика похожести двух Census-ов?

Cost функции: Census



Census

- битовый вектор описывающий патч вокруг пикселя и состоящий из $7 \cdot 9 - 1 = 62$ бит его можно преподсчитать для обеих картинок стереопары.



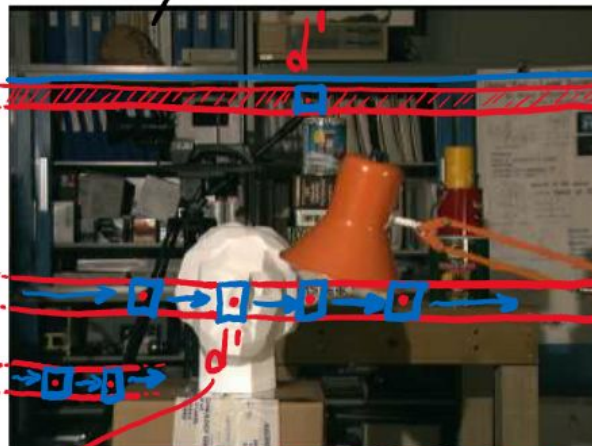
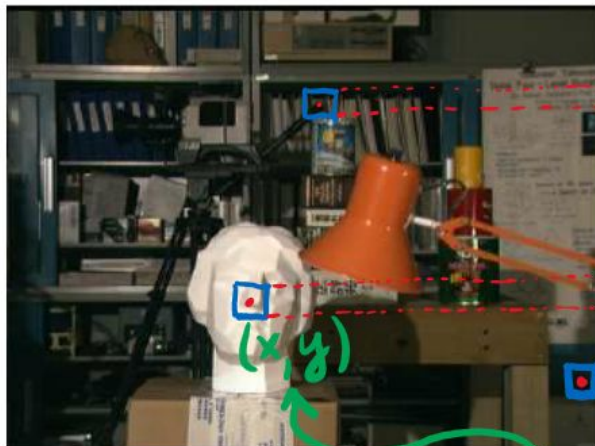
+ (\Rightarrow) ярче центра
- (\Rightarrow) темнее пикселе в центре

patch[7][9]

Метрика похожести двух Census:
Hamming distance, т.е. просто
xor + popcount!

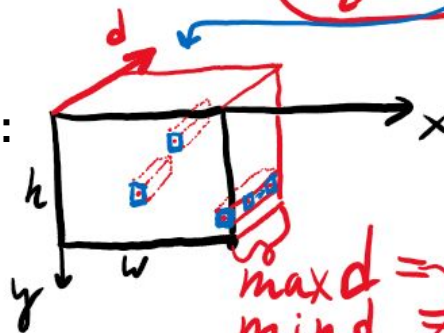
левая

правая



$$d(x, y) = d' = \underset{d}{\operatorname{argmin}} \operatorname{cost}(x, y, d)$$

DSI (disparity space image) volume:



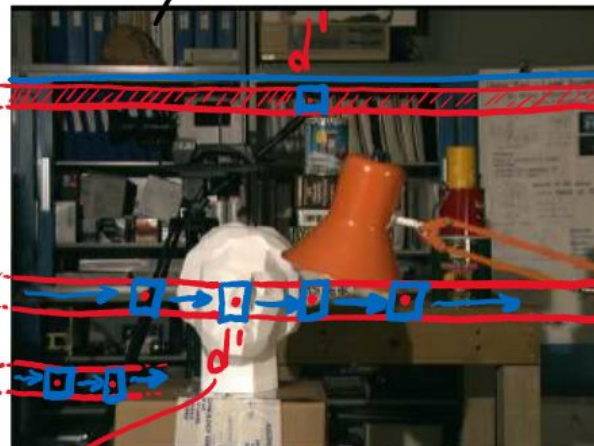
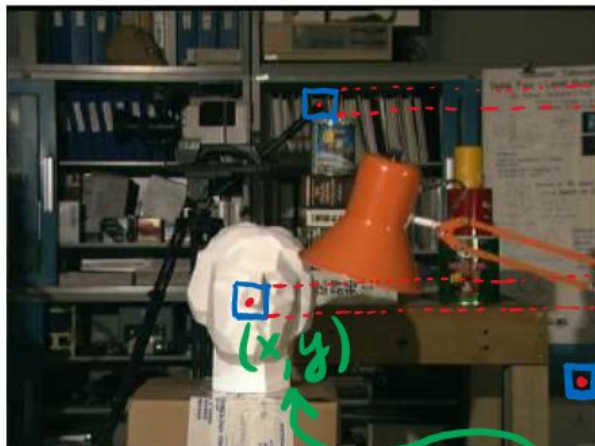
\max_d
 \min_d

$\operatorname{cost} = \text{😊}$

какая
одна из 3D
каждых точек

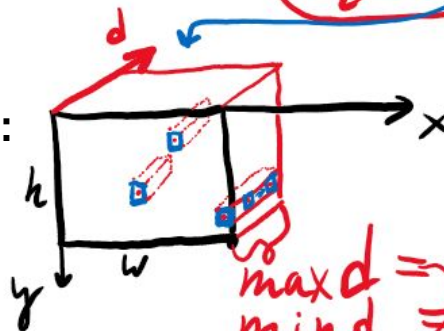
левая

правая



$$d(x, y) = d' = \underset{d}{\operatorname{argmin}} \operatorname{cost}(x, y, d)$$

DSI (disparity space image) volume:



\max_d
 \min_d

$\operatorname{cost} = \text{😊}$

Как добиться субпиксельной точности?

какая
общих 3D
ключевых точек

Как добиться субпиксельной точности?

- При ректификации делать небольшое увеличение картинки
- Parabola fitting

Ссылки

Книга (про ректификацию, SGM):

- [Computer Vision: Algorithms and Applications. Richard Szeliski](#)

Cost functions:

- [Evaluation of Stereo Matching Costs on Images with Radiometric Differences, Hirschmuller, 2008](#)

SGM:

- [Accurate and Efficient Stereo Processing by Semi-Global Matching and Mutual Information, Hirschmuller, 2005](#)
- [Stereo Processing by Semi-Global Matching and Mutual Information, Hirschmuller, 2008](#)

SURE tSGM:

- [SURE: PHOTOGRAMMETRIC SURFACE RECONSTRUCTION FROM IMAGERY, Rothmel et. al., 2013](#)

Вопросы?

