Курс «Трёхмерное компьютерное зрение»

Тема №7

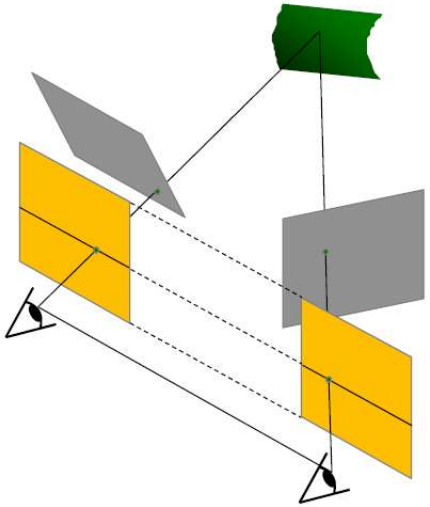
«Реконструкция карты глубины»

Антон Конушин

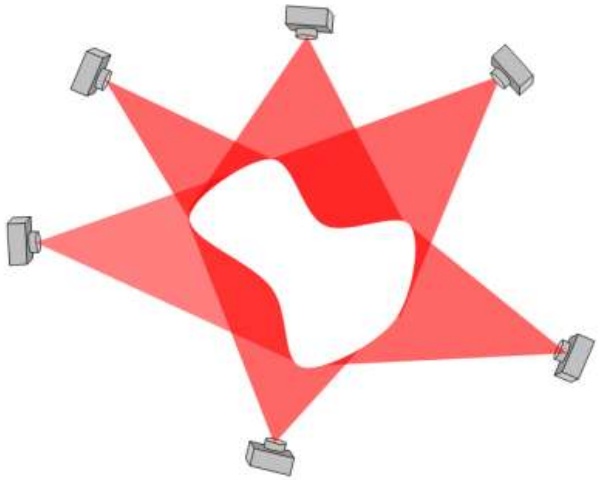
Разреженная и плотная 3D реконструкция

Карта глубины

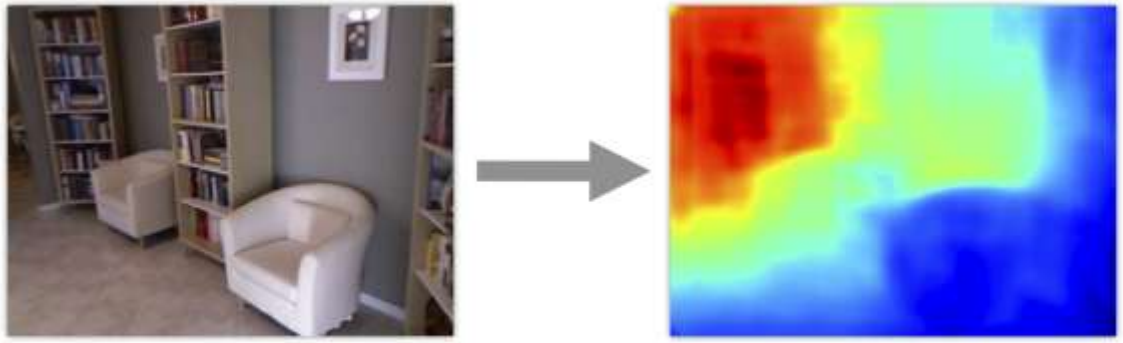
Как получить карту глубины?



Бинокулярное стерео



Многовидовое стерео

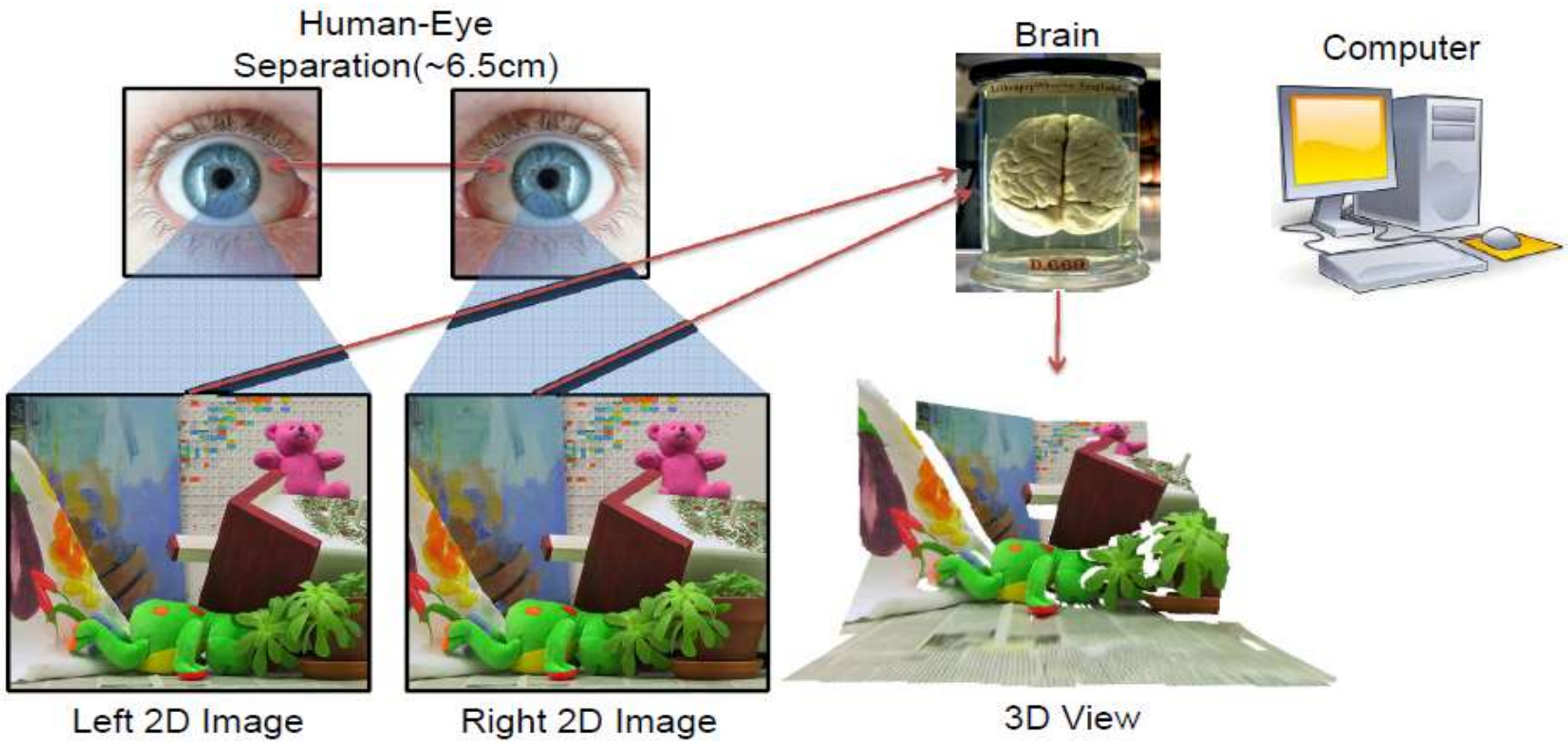


Предсказание глубины

Сенсор глубины

Бинокулярное стерео

Плотное бинокулярное стерео



- Задача плотного бинокулярного стерео: восстановить в 3D все видимые на 2-х изображениях точки
- Калибровку камер будем считать известной

Слайд: M. Bleyer

Параллакс

- Параллакс - видимое смещение объекта в зависимости от точки обзора
- Чем объект ближе, тем смещение больше

Стереопсис

«**Стереопсис** (англ. stereopsis) - сенсорный процесс, возникающий при бинокулярном зрении как психофизическая реакция на сетчаточную горизонтальную диспаратность. В результате С. субъект переживает специфическое ощущение глубины. <…> ».
(Психологическая энциклопедия)

Левое изображение

Правое изображение

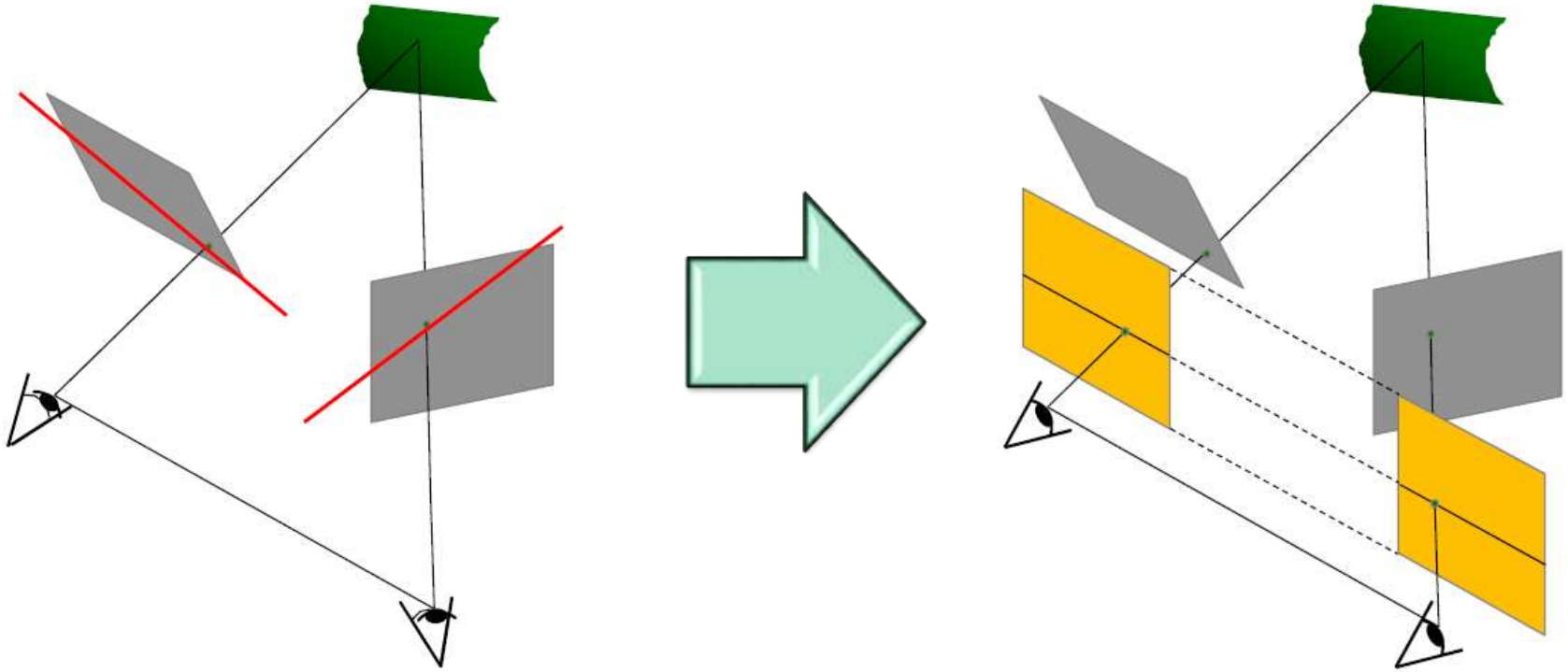
Общая схема

Ректификация изображений

Вычисление соответствующих точек

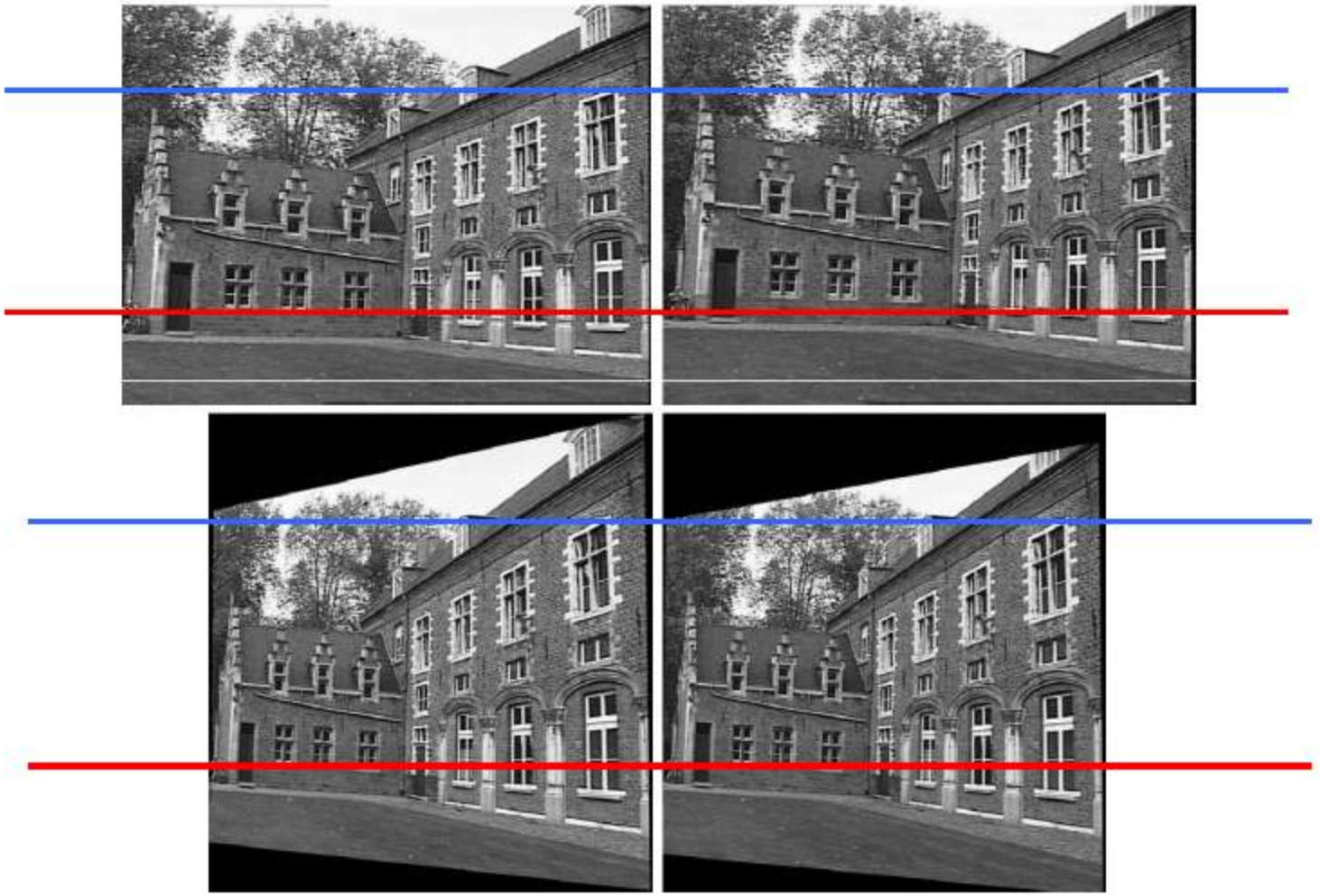
Восстановление 3D путем триангуляции

Ректификация



- «Ректификация» – преобразование стереопары в изображения, в которых соответствующие эпиполярные линии лежат на одной и той же горизонтальной строке
- Первый способ: проецирование на общую плоскость с помощью гомографии

Ректификация через гомографию

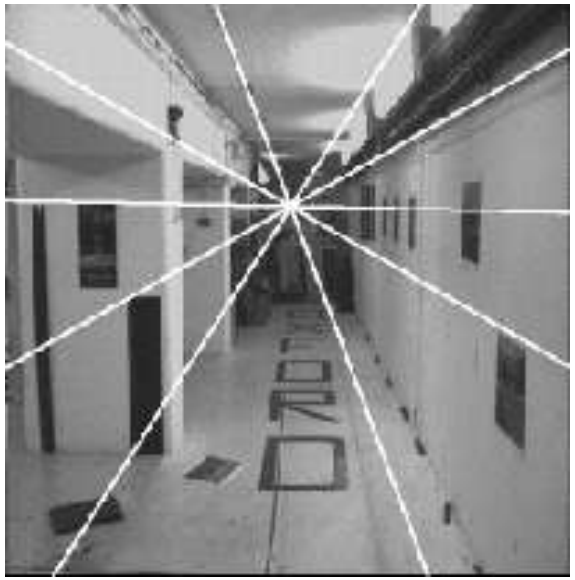


Недостатки:

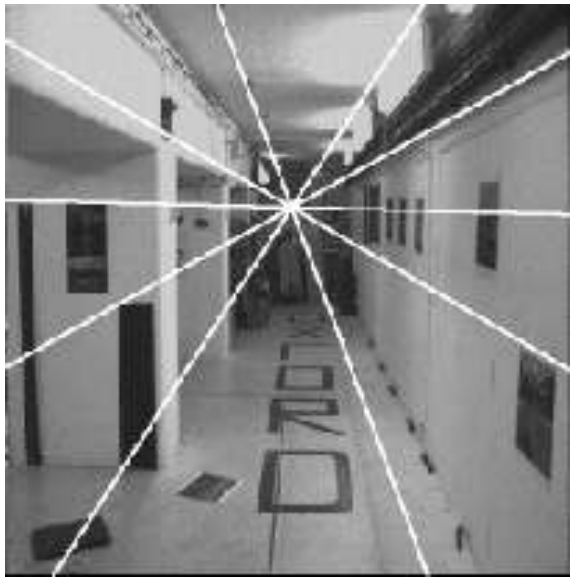
- Не применим, когда камера движется вперёд или назад
- Сильные искажения в некоторых случаях

Радиальная развёртка

- Polar rectification
  - Индексируем эпиполярные линии углом поворота относительно эпиполи
  - Копируем соответствующие пары эпиполярных линии последовательно в соответствующие горизонтали ректифицированных изображений



- Работает в тех случаях, когда ошибается метод проецирования на общую плоскость



Пример

Проецирование

Полярная ректификация

Пример

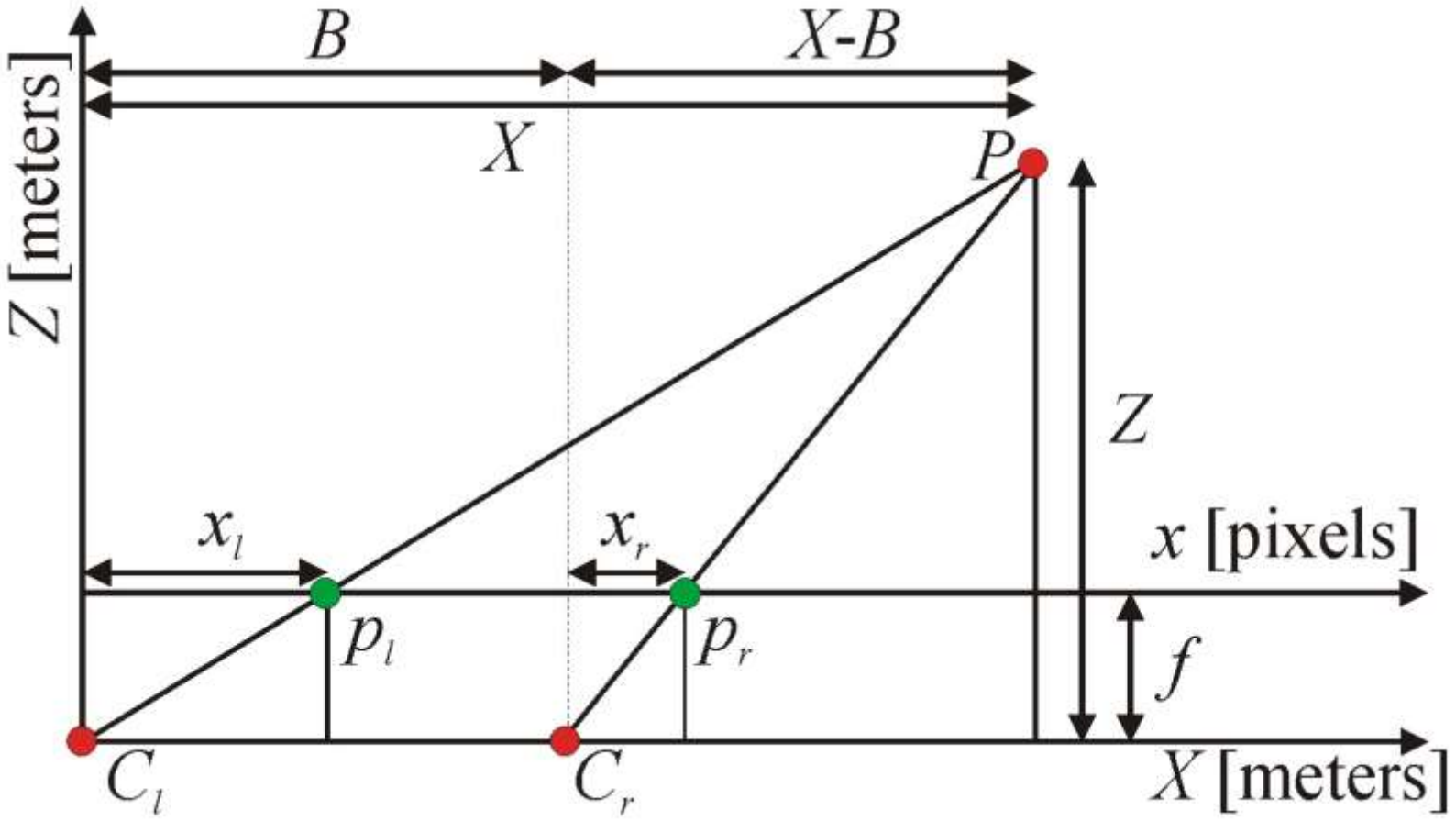
Общая схема

Ректификация изображений

Вычисление соответствующих точек

Восстановление 3D путем триангуляции

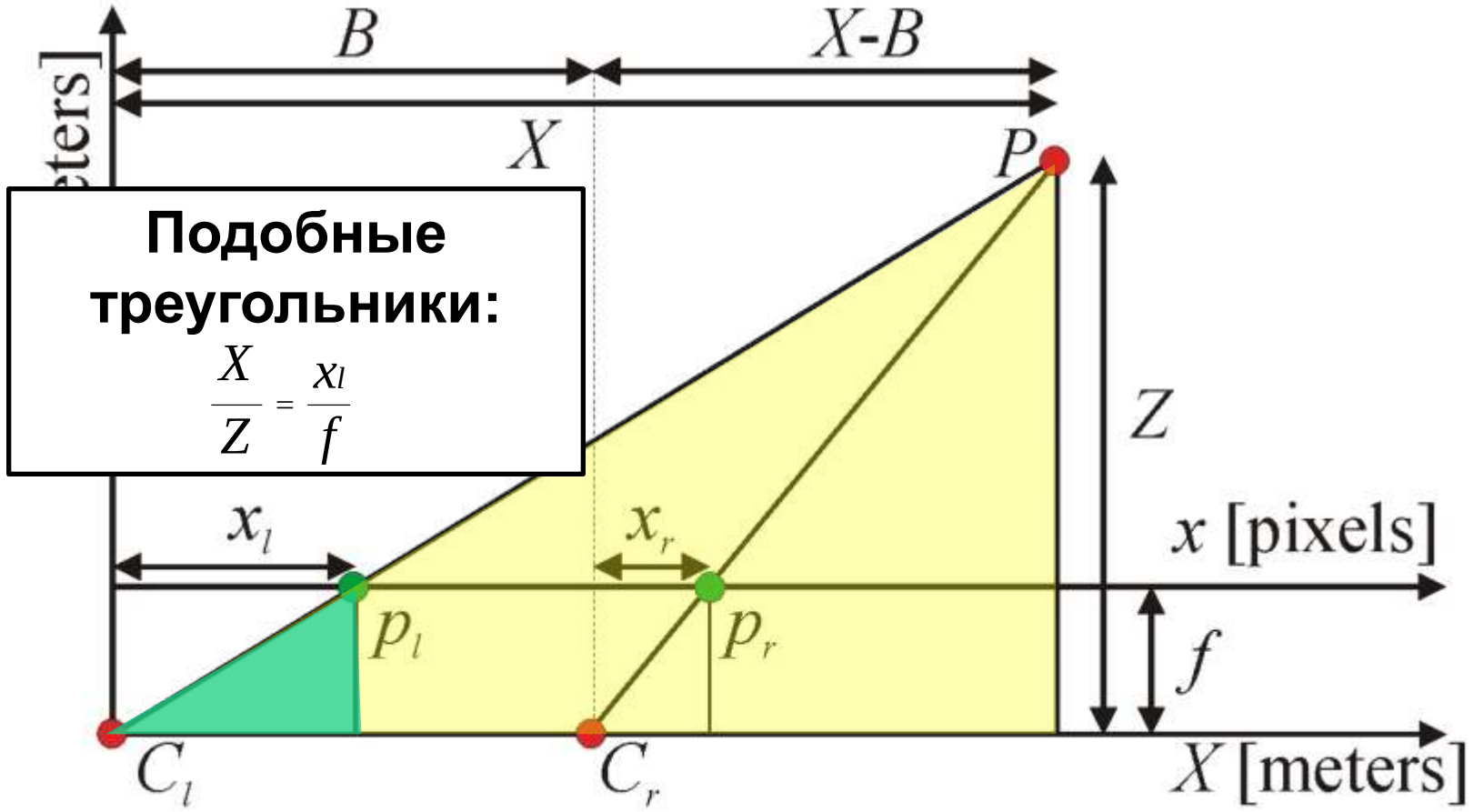
Триангуляция



x_l, x_r - смещения относительно принципиальной точки

Слайд: M. Bleyer

Триангуляция

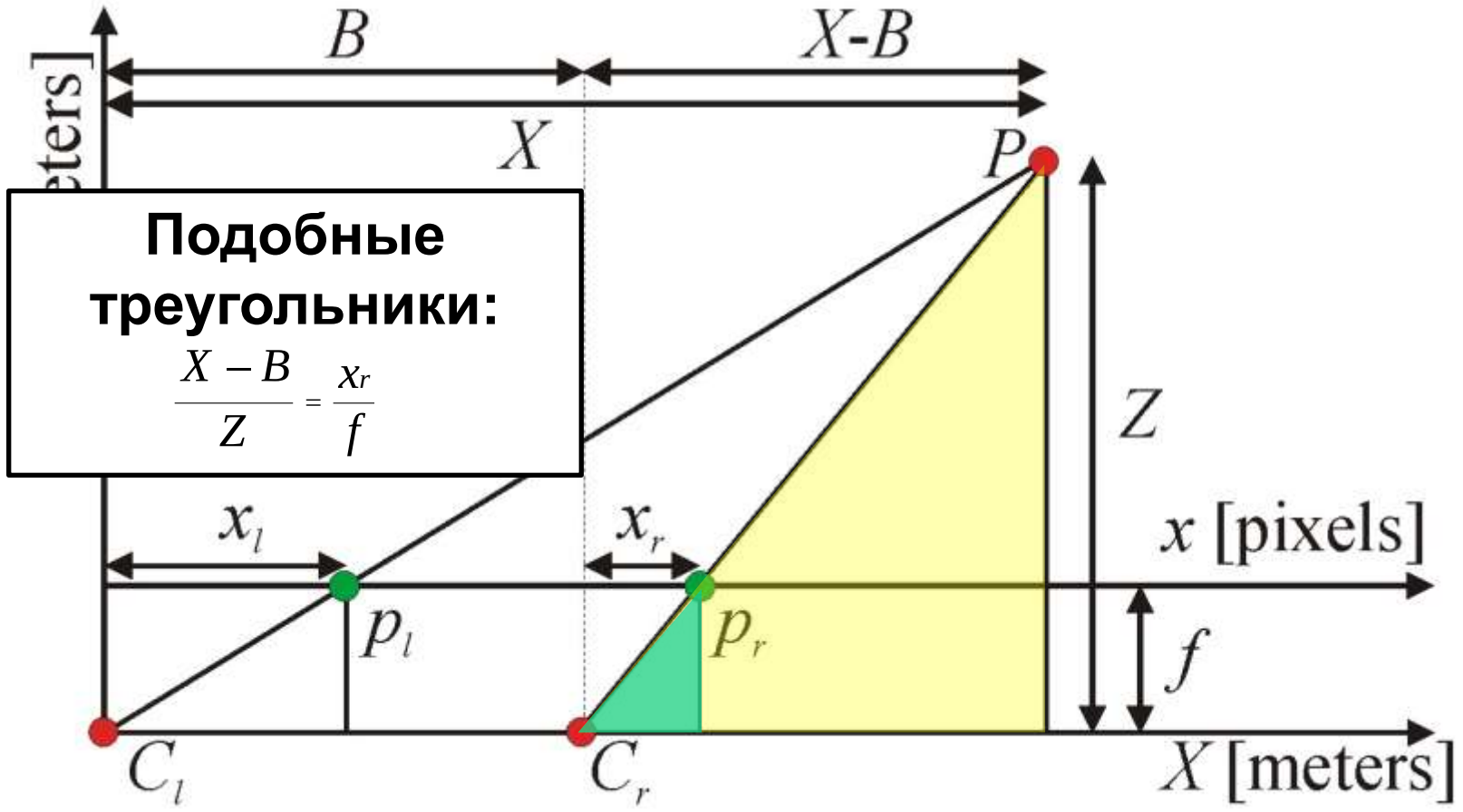


Подобные треугольники:

$$\frac{X}{Z} = \frac{x_l}{f}$$

Слайд: M. Bleyer

Триангуляция



Подобные треугольники:

$$\frac{X - B}{Z} = \frac{x_r}{f}$$

Слайд: M. Bleyer

Триангуляция

Из подобия треугольников:

$$\frac{X}{Z} = \frac{x_l}{f} \qquad \frac{X - B}{Z} = \frac{x_r}{f}$$

Исключаем *X* и выражаем *Z*:

$$Z = \frac{B \cdot f}{x_l - x_r} = \frac{B \cdot f}{d}$$, где $d = x_l - x_r$ - ***диспаритет***

Итоговые координаты 3D-точки:

$$X = \frac{Z \cdot x_l}{f} \qquad Y = \frac{Z \cdot y_l}{f} \qquad Z = \frac{B \cdot f}{d}$$

x_l, x_r - смещения относительно принципиальной точки

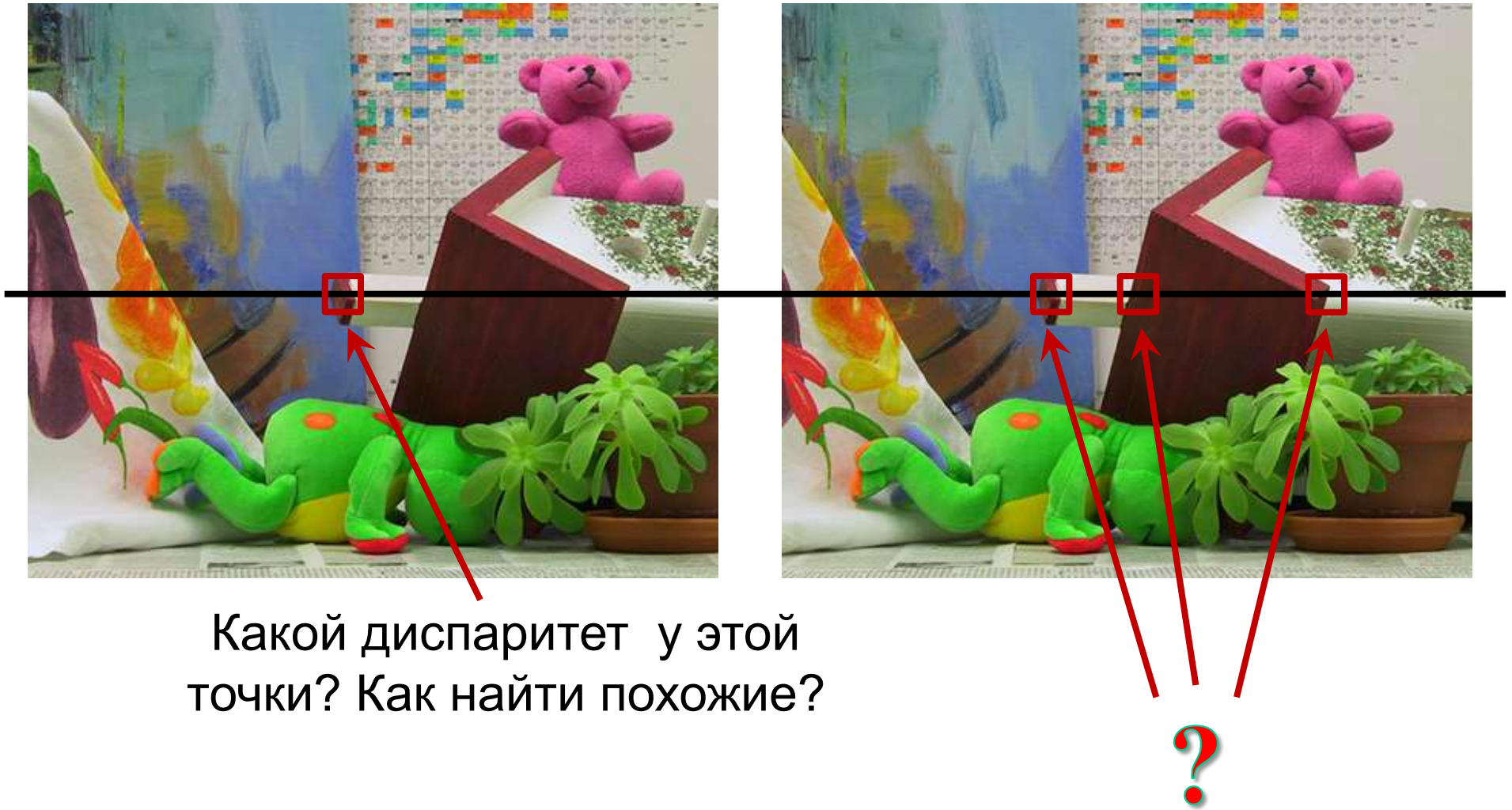
Общая схема

Ректификация и триангуляция – технические задачи, основной интерес это стереосопоставление

Локальные методы оценки диспаритета

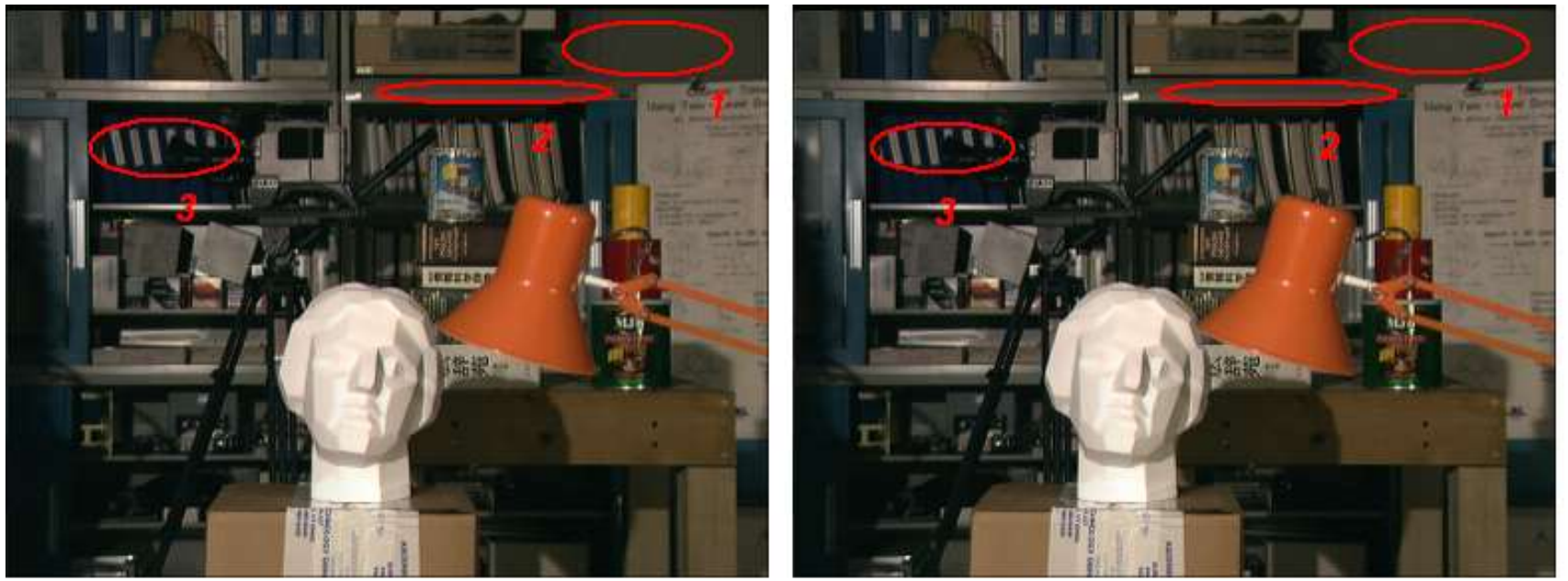
Стереосопоставление через вычисление диспаритета

Эпиполярное ограничение:
«Соответствующие точки лежат на одной строке»



Основные трудности

«Плохо» текстурированные области



1 – Однородная область
2 – Текстура неизменна в горизонтальном направлении
3 – Повторяющаяся текстура

Основные трудности

Изменение цвета точки между ракурсами

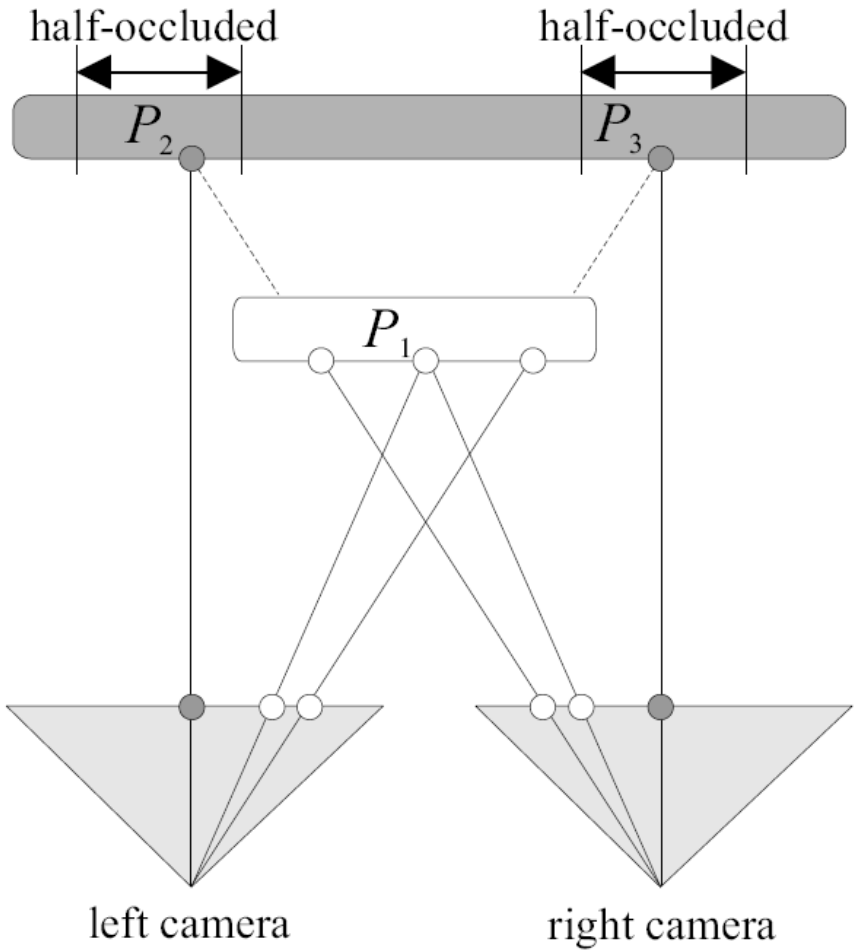
Шум камеры, изменение освещенности, погрешности сэмплирования и т.д.

Основные трудности

Перекрытия

Некоторые пиксели одного изображения могут быть не видны (перекрыты) на другом.

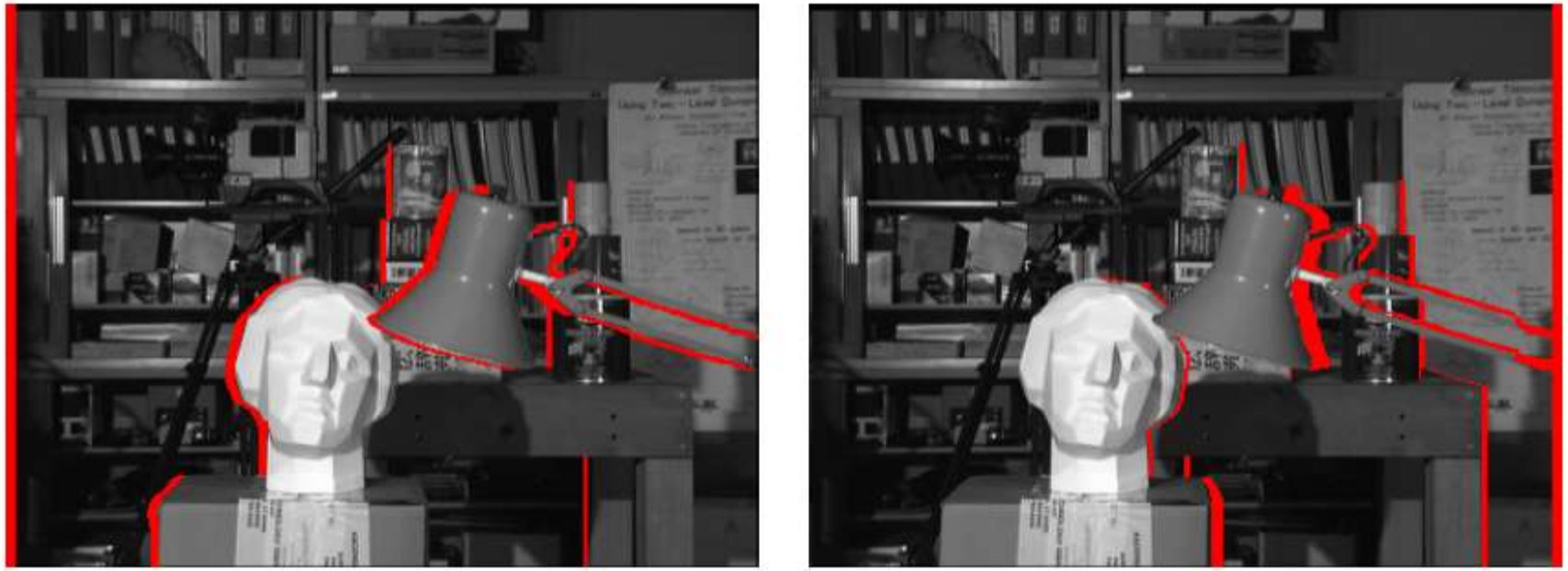
«Невидимые» области называют областями перекрытия



Слайд: M. Bleyer

Основные трудности

Перекрытия



Пример областей перекрытия (выделены красным)

Рисунок: M. Bleyer

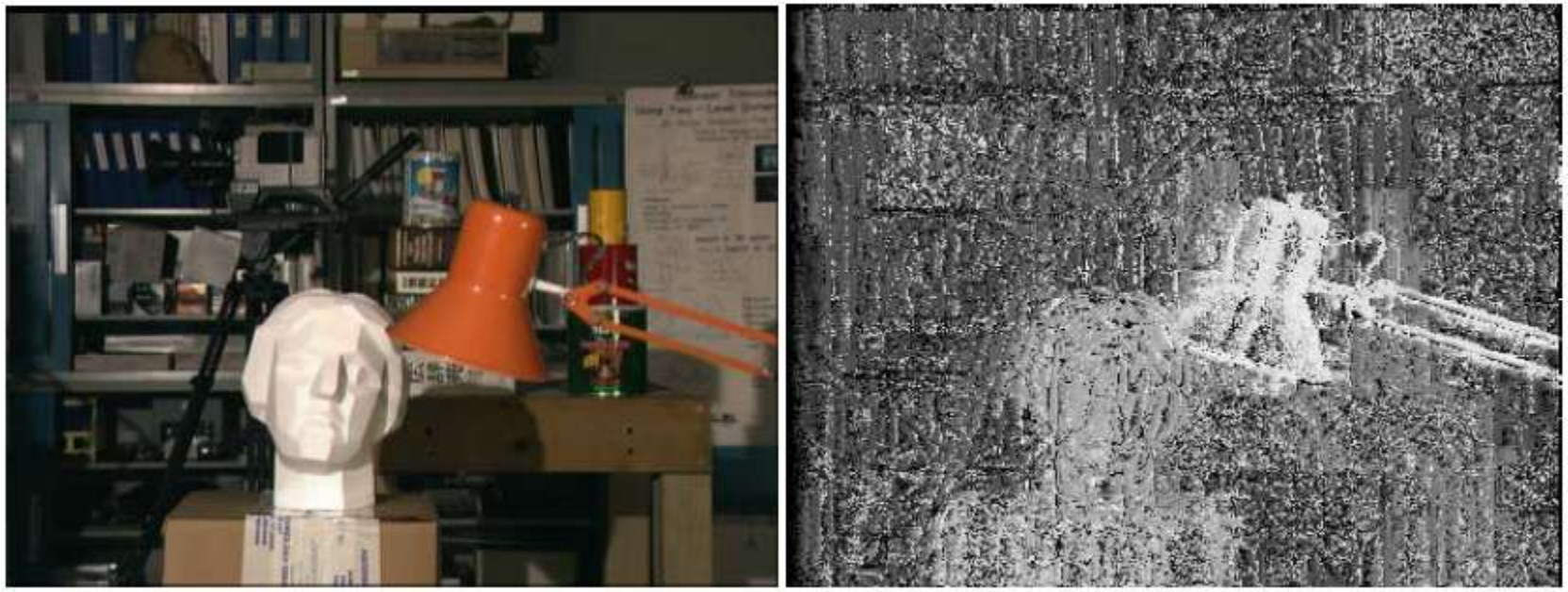
Локальные методы бинокулярного стерео

- Локальные, т.к. диспаритет в каждой точке зависит только от ее локальной окрестности

- Используют стратегию WTA (Winner-Take-All), т.е. в каждой точке «побеждает» соответствие с наименьшей «стоимостью»

Наивный алгоритм

- Стоимость соответствия – разность интенсивностей пикселей
- Результат – слишком много шума



Пример работы наивного алгоритма

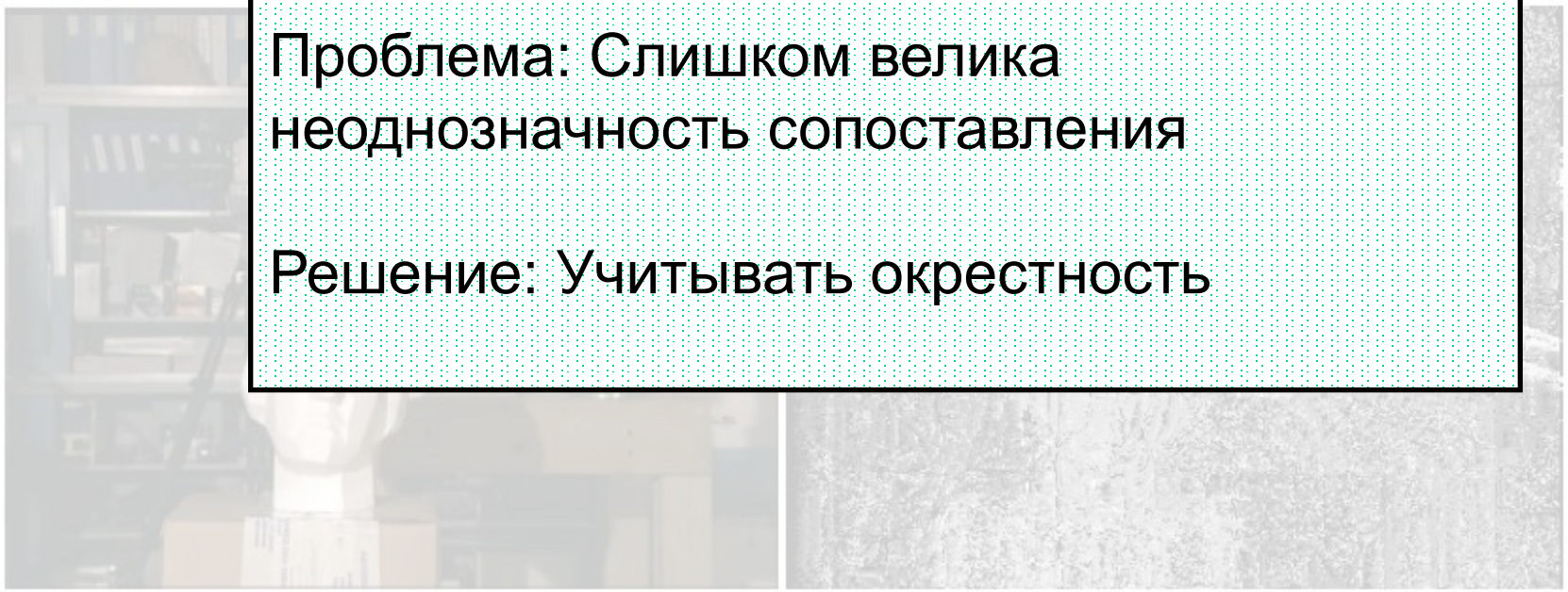
Пример: M. Bleyer

Наивный алгоритм

- Стоимость соответствия – разность интенсивностей пикселей
- Результат – слишком много шума

Проблема: Слишком велика неоднозначность сопоставления

Решение: Учитывать окрестность



Пример работы наивного алгоритма

Пример: M. Bleyer

Учет окрестности

- Стоимость соответствия – SAD, SSD, NCC по окну вокруг пикселя

$$d_p = \arg\min_{0 \le d \le d\max} \sum_{q \in Wp} c(q, q-d)$$

где:

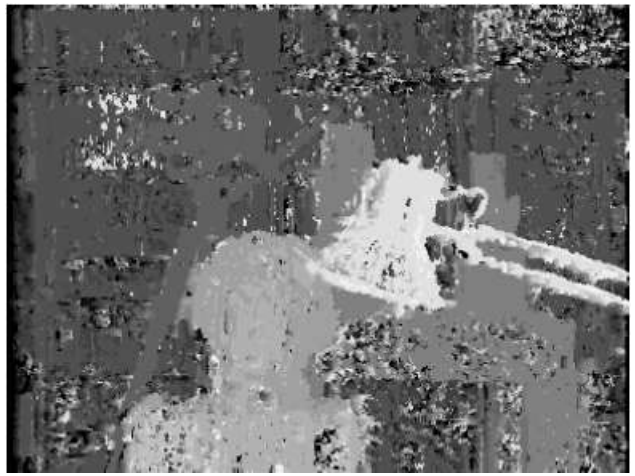
d_p – искомый диспаритет в пикселе p,
$c(p,q)$ – функция стоимости,
W_p – окно вокруг пикселя p,
d_{max} – максимально возможный диспаритет

- Возможна очень эффективная реализация
  - Метод «скользящего окна»

Проблема выбора размера окна

- Маленькие окна – недостаточно текстуры
- Большие окна – эффект «раздувания» объектов переднего плана (foreground fattening)

Ground truth

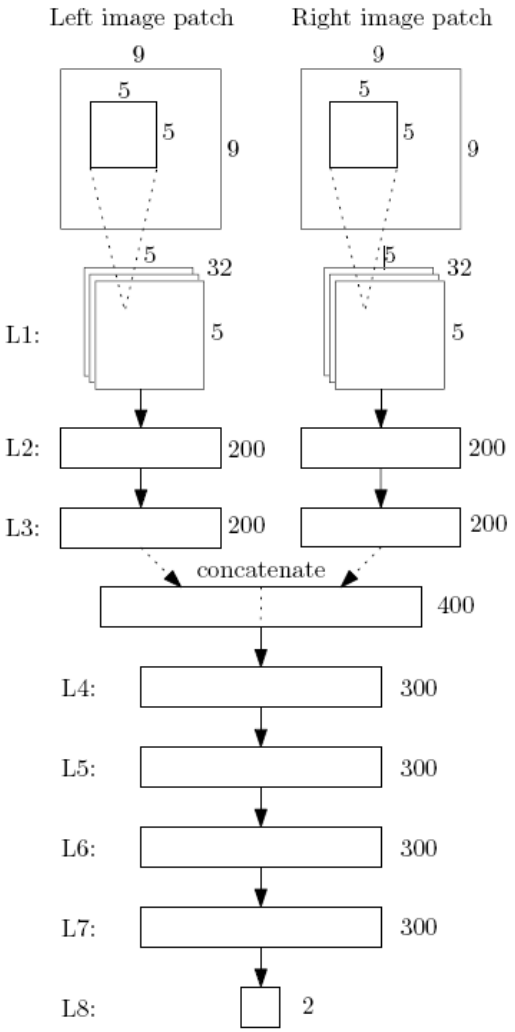


Окно 3x3

Окно 21x21

Примеры: M. Bleyer

Обучение с помощью CNN



- Собираем обучающую выборку $< \mathcal{P}^L_{9\times9}(\mathbf{p}), \mathcal{P}^R_{9\times9}(\mathbf{q}) >$
- 1 положительный и 1 отрицательный пример на 1 пиксель с известным диспаритетом
- Отрицательные примеры:

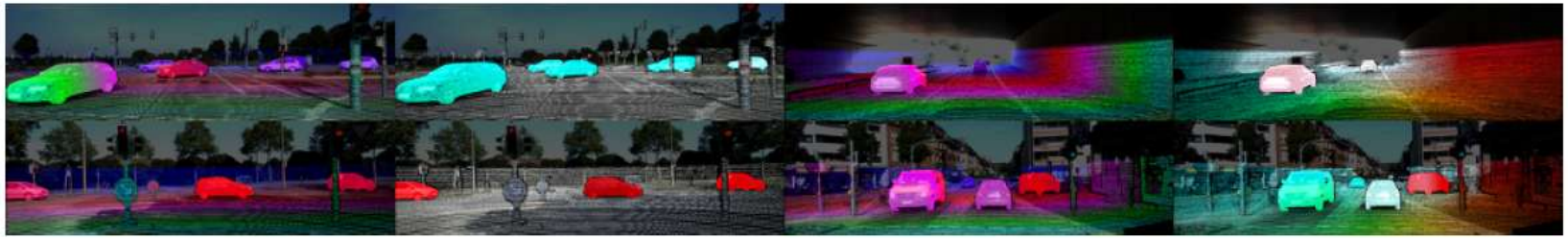
$$\mathbf{q} = (x - d + o_{\text{neg}}, y) \quad \{-N_{\text{hi}}, \ldots, -N_{\text{lo}}, N_{\text{lo}}, \ldots, N_{\text{hi}}\}$$

- Положительные примеры:

$$\mathbf{q} = (x - d + o_{\text{pos}}, y) \quad \{-P_{\text{hi}}, \ldots, P_{\text{hi}}\}$$

J. Zbontar and Y. LeCun: Computing the Stereo Matching Cost with a Convolutional Neural Network. CVPR 2015

Датасет KITTI



400 отобранных кадров

20 000 исходных кадров

стереопары

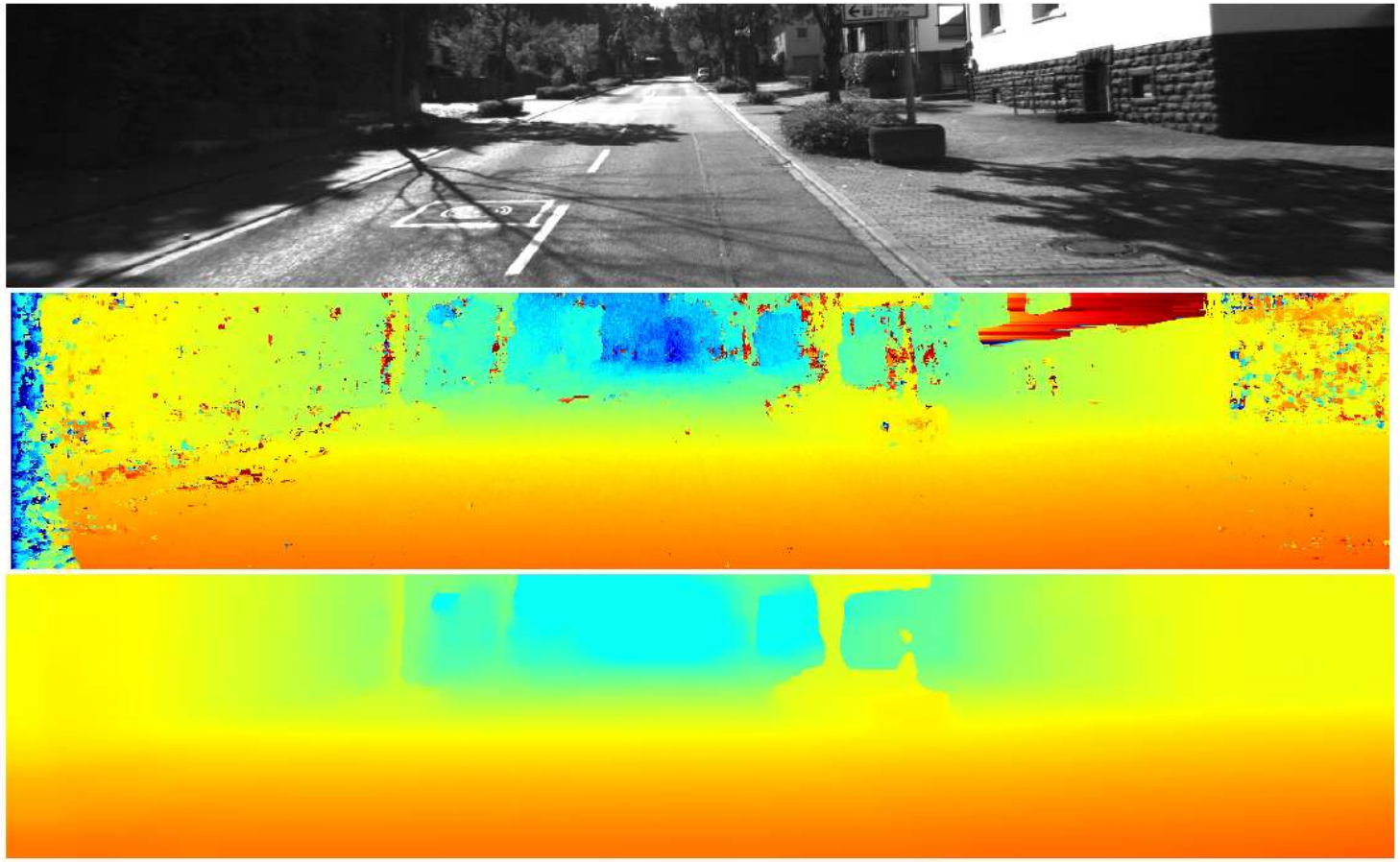
облако точек с LIDAR

оптический поток

одометрия

Geiger et al. Vision meets Robotics: The KITTI Dataset. IJRR 2013

Пример результата на KITTI 2015



Ошибка 14.7% для winner-take-all стратегии

Ошибка 2.61% при использовании метрики внутри semi-global стратегии (идею рассмотрим дальше)

Метрики – диспаритет в пикселе верный, если ошибка <3px (или 5%)

https://www.cvlibs.net/datasets/kitti/index.php

Оценка на датасете KITTI 2015

Stereo Evaluation

Rank	Method	Setting	Code	Out-Noc	Out-All	Avg-Noc	Avg-All	Density	Runtime	Environment	Compare
1	Displets		code	**2.47 %**	3.27 %	0.7 px	0.9 px	100.00 %	265 s	>8 cores @ 3.0 Ghz (Matlab + C/C++)	
F. Guney and A. Geiger: Displets: Resolving Stereo Ambiguities using Object Knowledge. Conference on Computer Vision and Pattern Recognition (CVPR) 2015.											
2	MC-CNN			2.61 %	3.84 %	0.8 px	1.0 px	100.00 %	100 s	Nvidia GTX Titan (CUDA, Lua/Torch7)	
J. Zbontar and Y. LeCun: Computing the Stereo Matching Cost with a Convolutional Neural Network. Conference on Computer Vision and Pattern Recognition (CVPR) 2015.											
3	PRSM		code	2.78 %	**3.00 %**	**0.7 px**	0.7 px	100.00 %	300 s	1 core @ 2.5 Ghz (C/C++)	
C. Vogel, K. Schindler and S. Roth: 3D Scene Flow Estimation with a Piecewise Rigid Scene Model. Ijcv 2015.											
4	SPS-StFl			2.83 %	3.64 %	0.8 px	0.9 px	100.00 %	35 s	1 core @ 3.5 Ghz (C/C++)	
K. Yamaguchi, D. McAllester and R. Urtasun: Efficient Joint Segmentation, Occlusion Labeling, Stereo and Flow Estimation. ECCV 2014.											
5	VC-SF			3.05 %	3.31 %	0.8 px	0.8 px	100.00 %	300 s	1 core @ 2.5 Ghz (C/C++)	
C. Vogel, S. Roth and K. Schindler: View-Consistent 3D Scene Flow Estimation over Multiple Frames. Proceedings of European Conference on Computer Vision. Lecture Notes in, Computer Science 2014.											
6	OSF		code	3.28 %	4.07 %	0.8 px	0.9 px	99.98 %	50 min	1 core @ 3.0 Ghz (Matlab + C/C++)	
M. Menze and A. Geiger: Object Scene Flow for Autonomous Vehicles. Conference on Computer Vision and Pattern Recognition (CVPR) 2015.											
7	CoR		code	3.30 %	4.10 %	0.8 px	0.9 px	100.00 %	6 s	6 cores @ 3.3 Ghz (Matlab + C/C++)	
A. Chakrabarti, Y. Xiong, S. Gortler and T. Zickler: Low level Vision by Consensus in a Spatial Hierarchy of Regions. CVPR 2015.											
8	SPS-St		code	3.39 %	4.41 %	0.9 px	1.0 px	100.00 %	2 s	1 core @ 3.5 Ghz (C/C++)	
K. Yamaguchi, D. McAllester and R. Urtasun: Efficient Joint Segmentation, Occlusion Labeling, Stereo and Flow Estimation. ECCV 2014.											
9	PCBP-SS			3.40 %	4.72 %	0.8 px	1.0 px	100.00 %	5 min	4 cores @ 2.5 Ghz (Matlab + C/C++)	
K. Yamaguchi, D. McAllester and R. Urtasun: Robust Monocular Epipolar Flow Estimation. CVPR 2013.											
10	DDS-SS			3.83 %	4.59 %	0.9 px	1.0 px	100.00 %	1 min	1 core @ 2.5 Ghz (Matlab + C/C++)	
D. Wei, C. Liu and W. Freeman: A Data-driven Regularization Model for Stereo and Flow. 3DTV-Conference, 2014 International Conference on 2014.											
11	StereoSLIC			3.92 %	5.11 %	0.9 px	1.0 px	99.89 %	2.3 s	1 core @ 3.0 Ghz (C/C++)	
K. Yamaguchi, D. McAllester and R. Urtasun: Robust Monocular Epipolar Flow Estimation. CVPR 2013.											
12	PR-Sf+E			4.02 %	4.87 %	0.9 px	1.0 px	100.00 %	200 s	4 cores @ 3.0 Ghz (Matlab + C/C++)	
C. Vogel, S. Roth and K. Schindler: Piecewise Rigid Scene Flow. International Conference on Computer Vision (ICCV) 2013.											
13	PCBP			4.04 %	5.37 %	0.9 px	1.1 px	100.00 %	5 min	4 cores @ 2.5 Ghz (Matlab + C/C++)	
K. Yamaguchi, T. Hazan, D. McAllester and R. Urtasun: Continuous Markov Random Fields for Robust Stereo Estimation. ECCV 2012.											
14	CSPMS			4.13 %	5.92 %	1.2 px	1.6 px	100.00 %	6 s	4 cores @ 2.5 Ghz (C/C++)	
Anonymous submission											
15	MBM			4.35 %	5.43 %	1.0 px	1.1 px	100.00 %	0.2 s	1 core @ 3.0 Ghz (C/C++)	
Anonymous submission											
16	PR-Sceneflow			4.36 %	5.22 %	0.9 px	1.1 px	100.00 %	150 sec	4 core @ 3.0 Ghz (Matlab - C/C++)	
C. Vogel, S. Roth and K. Schindler: Piecewise Rigid Scene Flow. International Conference on Computer Vision (ICCV) 2013.											

The KITTI Vision Benchmark Suite

A project of Karlsruhe Institute of Technology and Toyota Technological Institute at Chicago

https://www.cvlibs.net/datasets/kitti/index.php

Глобальные методы оценки диспаритета

Глобальные методы

- Глобальные, т.к. диспаритет в каждой точке вычисляется при помощи некоторой глобальной процедуры оптимизации, т.е. зависит не только от локальной окрестности

- В до-нейросетевых методах формулируются в терминах разметки графа и минимизации энергии

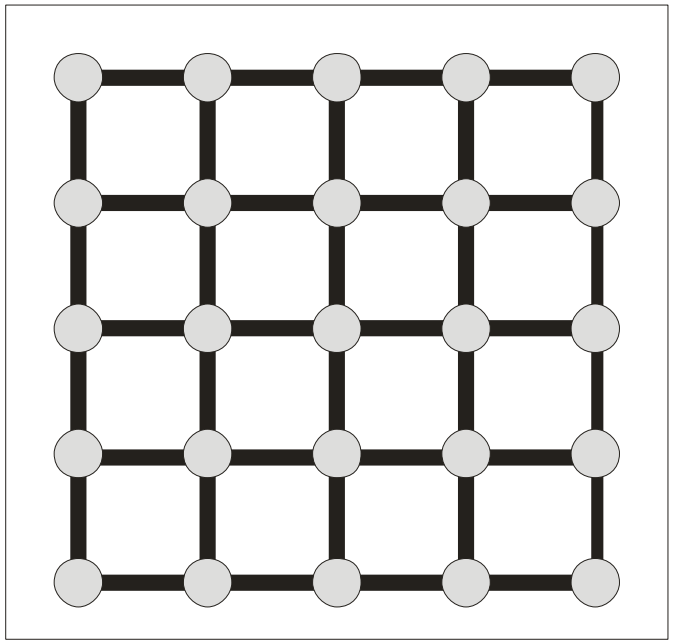
- В нейросетевых методах – одновременное предсказание всей карты глубины

Глобальные методы

Необходимо найти разметку *D*, минимизирующую функцию энергии *E(D)*.

Метки – значения диспаритета.

Граф – решетка, узлы – пиксели.



$$E(D) = E_{data}(D) + E_{smooth}(D)$$

Соответствие цветов

Гладкость

Рисунок: M. Bleyer

Проблема сохранения границ

Предположим, необходимо восстановить диспаритет в области границы между объектами, как показано на рисунке

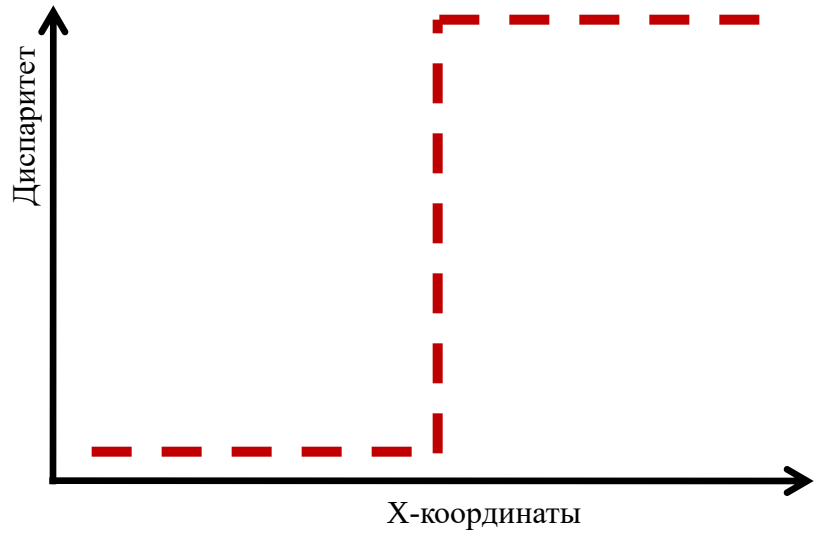
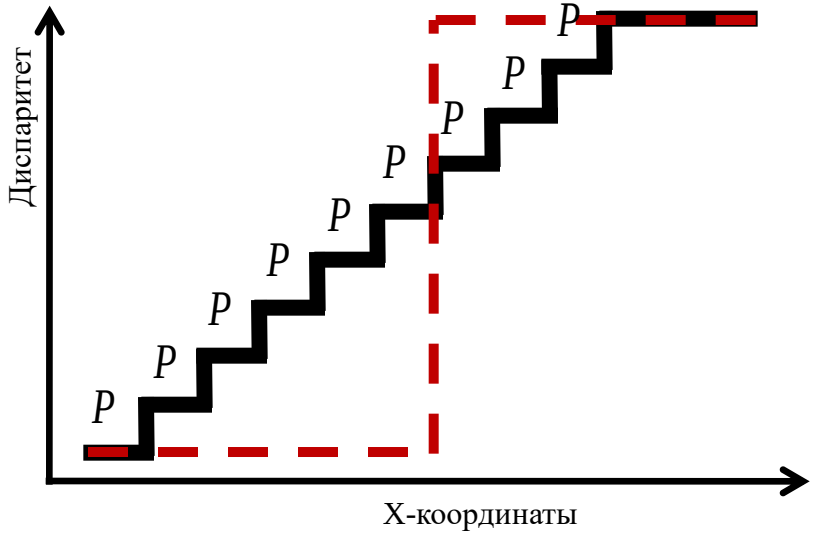


Рисунок: M. Bleyer

Проблема сохранения границ

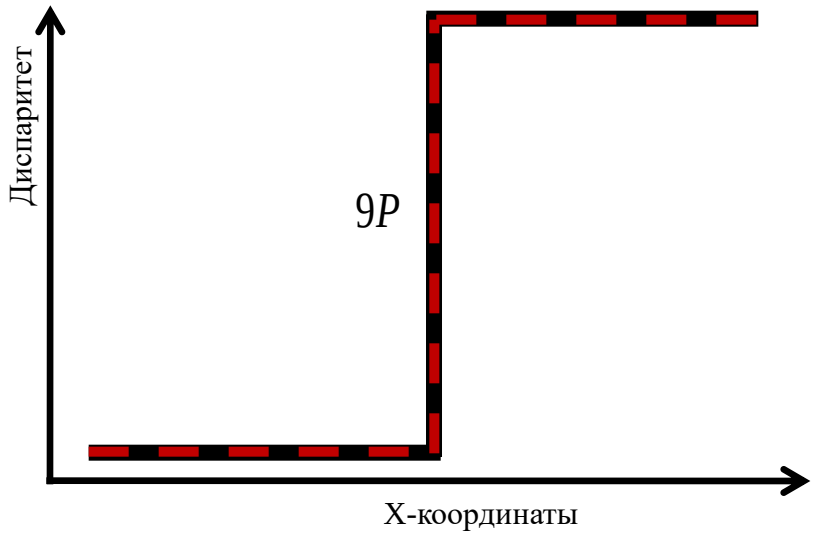
Случай линейной модели: $s(d_p, d_q) = |d_p - d_q| \cdot P$

Вклад в энергию:
9P



(неверное решение)

Вклад в энергию:
9P



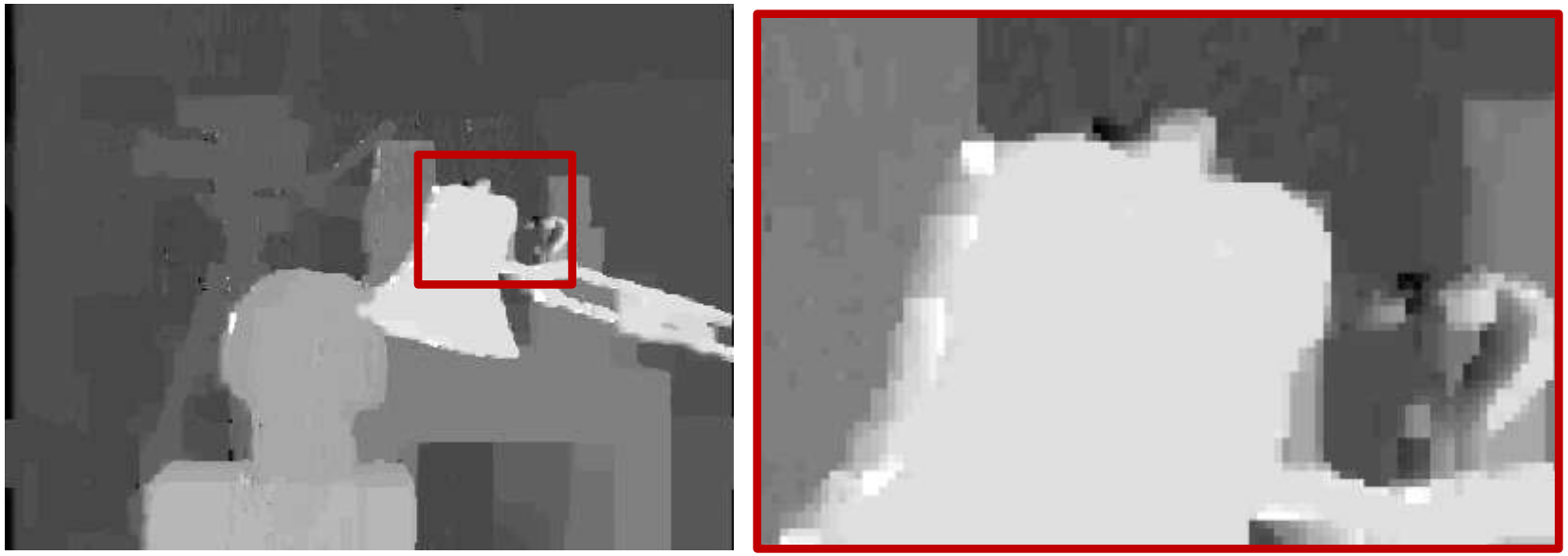
(верное решение)

Линейная модель не поощряет резких разрывов диспаритета.
Она чересчур сглаживает решение.

Рисунок: M. Bleyer

Проблема сохранения границ

Случай линейной модели: $s(d_p, d_q) = |d_p - d_q| \cdot P$



Пример результатов для линейной модели. Справа показан увеличенный фрагмент

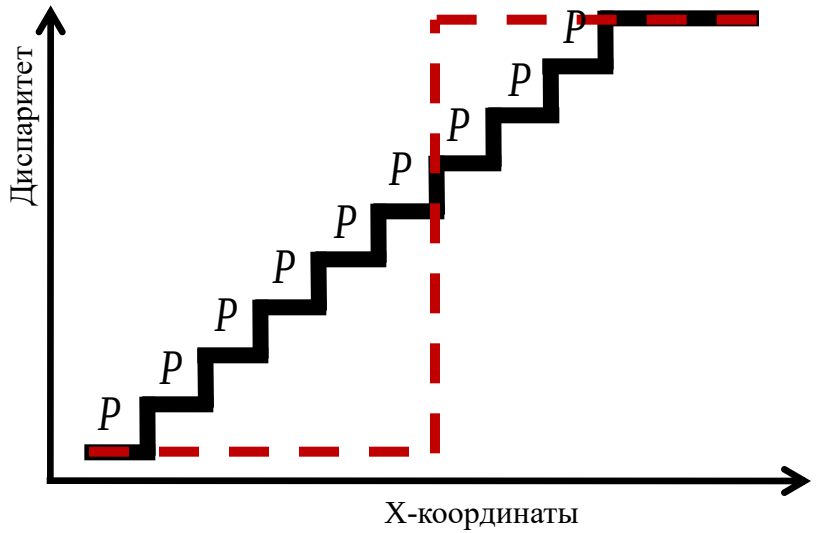
Примеры: M. Bleyer

Проблема сохранения границ

Случай модели Поттса:

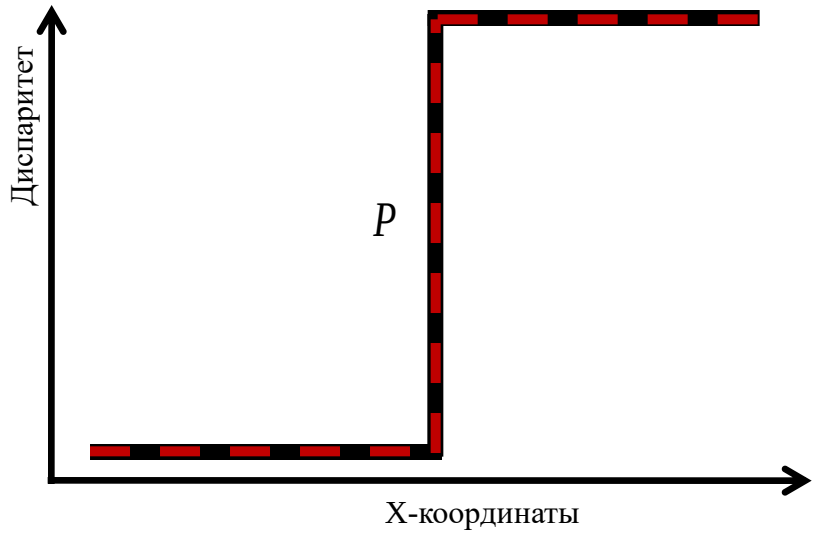
$$s(d_p, d_q) = \begin{cases} 0, d_p = d_q \\ P, d_p \neq d_q \end{cases}$$

Вклад в энергию:
9P



(неверное решение)

Вклад в энергию:
P



(верное решение)

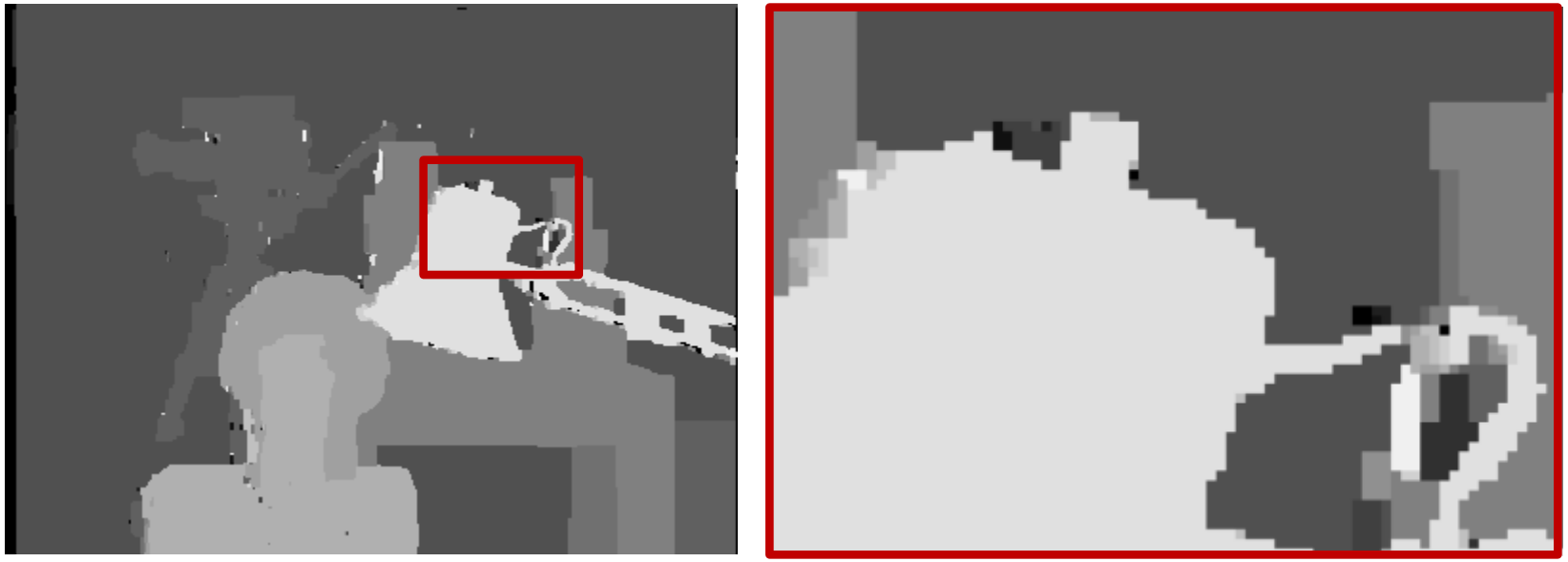
- Модель Поттса не препятствует резким разрывам диспаритета.
- Такие модели называют сохраняющими разрывы (discontinuity preserving) – как и в оптическом потоке

Рисунки: M. Bleyer

Проблема сохранения границ

Случай модели Поттса:

$$s(d_p, d_q) = \begin{cases} 0, d_p = d_q \\ P, d_p \neq d_q \end{cases}$$



Пример результатов для модели Поттса. Справа показан увеличенный фрагмент

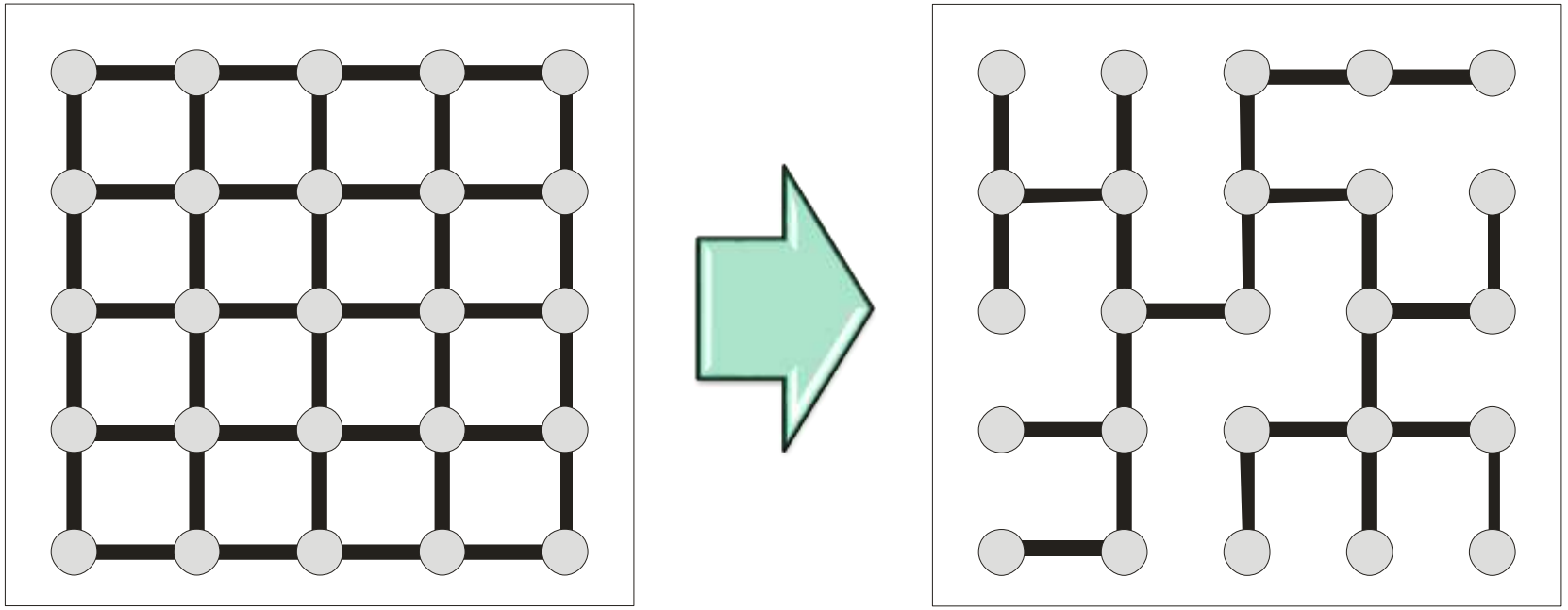
Примеры: M. Bleyer

Минимизация энергии

- Плотное стерео – задача многоклассовой разметки
- Эффективное решение на графе общего вида существует лишь для выпуклых относительно $|d_p - d_q|$ парных потенциалов [Ishikava, 2003]
- Но необходимо использовать модели, сохраняющие границы, а они невыпуклы относительно $|d_p - d_q|$
- Задача становится NP-полной
- Необходимы приближённые алгоритмы
  - Fusion move, Loopy belief propagation, TRW
- Альтернативный вариант – уход от графов общего вида к деревьям

H. Ishikawa. Exact Optimization for Markov Random Fields with Convex Priors. *PAMI*, 2003.

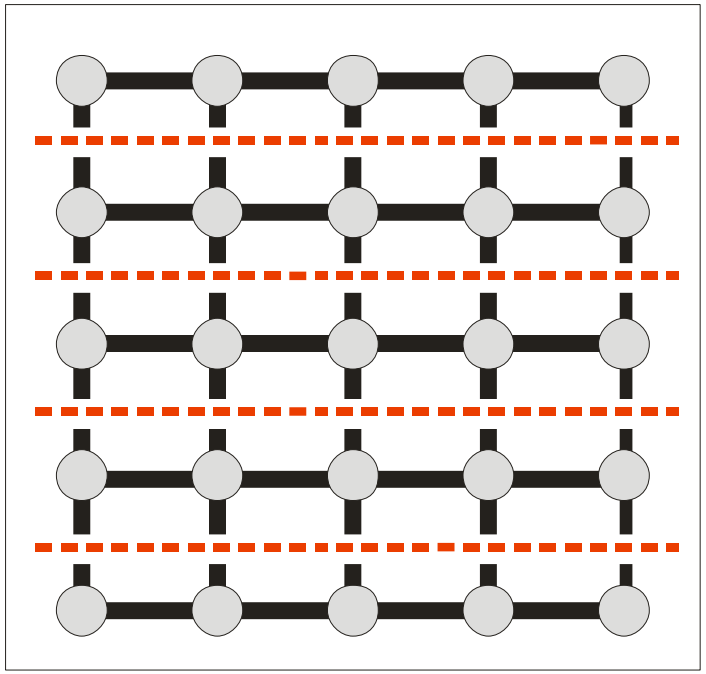
Переход к деревьям

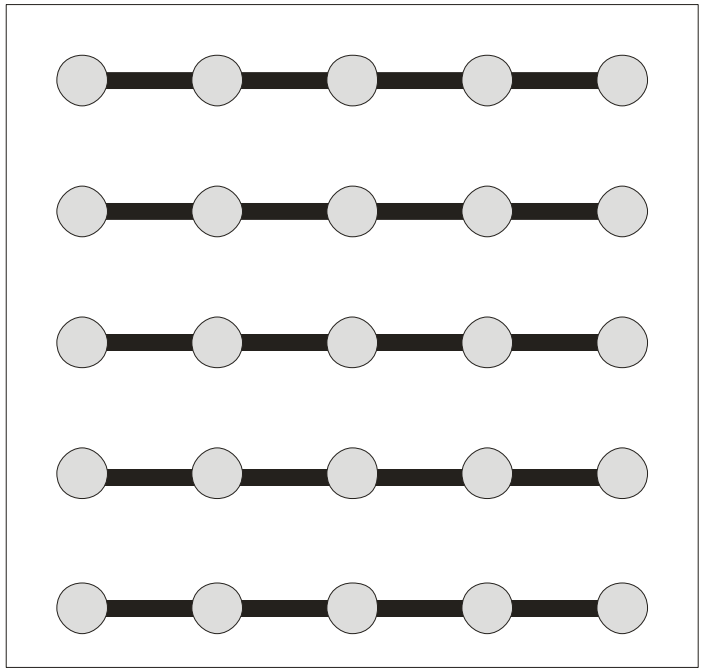


- Отсутствие циклов позволяет использовать метод динамического программирования
  - Глобальный минимум, произвольная энергия, высокая скорость работы
- Главный вопрос – какие ребра убирать?

Рисунки: M. Bleyer

Алгоритм Scanline Optimization





- Удаляются все вертикальные ребра
- Так поступали в первых подобных алгоритмах

Рисунки: M. Bleyer

Алгоритм Scanline Optimization

Очевидная проблема – рассогласованность строк между собой (horizontal streaking)

Пример: M. Bleyer

Алгоритм на основе MST

Идея – не форсировать гладкость между пикселями сильно разного цвета

Каждому ребру (паре пикселей p и q) присваивается вес:

$$w(p,q) = |I(p) - I(q)|$$

Строится минимальное покрывающее дерево (Minimum Spanning Tree, MST)

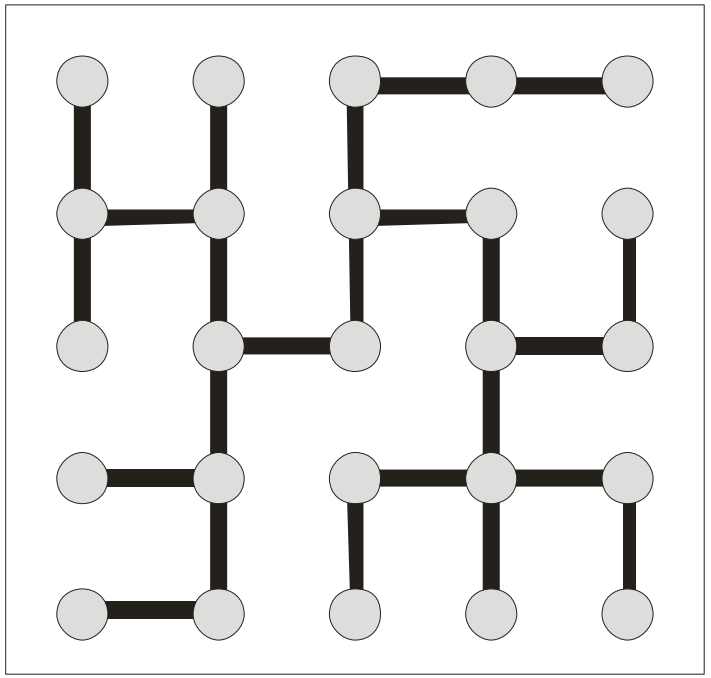


Рисунок: M. Bleyer

O. Veksler. Stereo Correspondence by Dynamic Programming on a Tree. *CVPR*, 2005.

Алгоритм на основе MST

Лучше, чем scanline optimization, но некоторая рассогласованность остается

Алгоритм Semi-Global Matching

В каждом пикселе строится свое дерево

Оптимизация производится вдоль лучей, исходящих из пикселя

Подход не совсем глобальный, но и не локальный

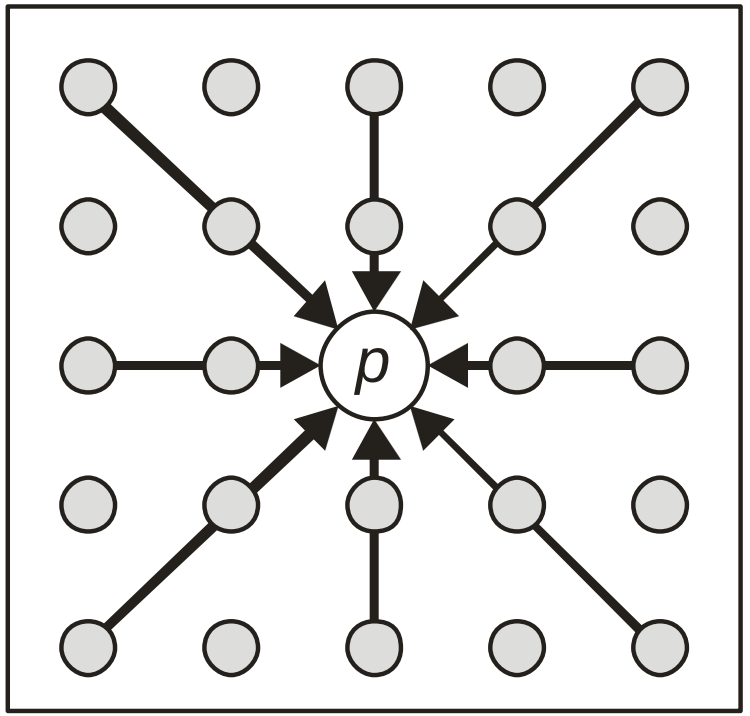


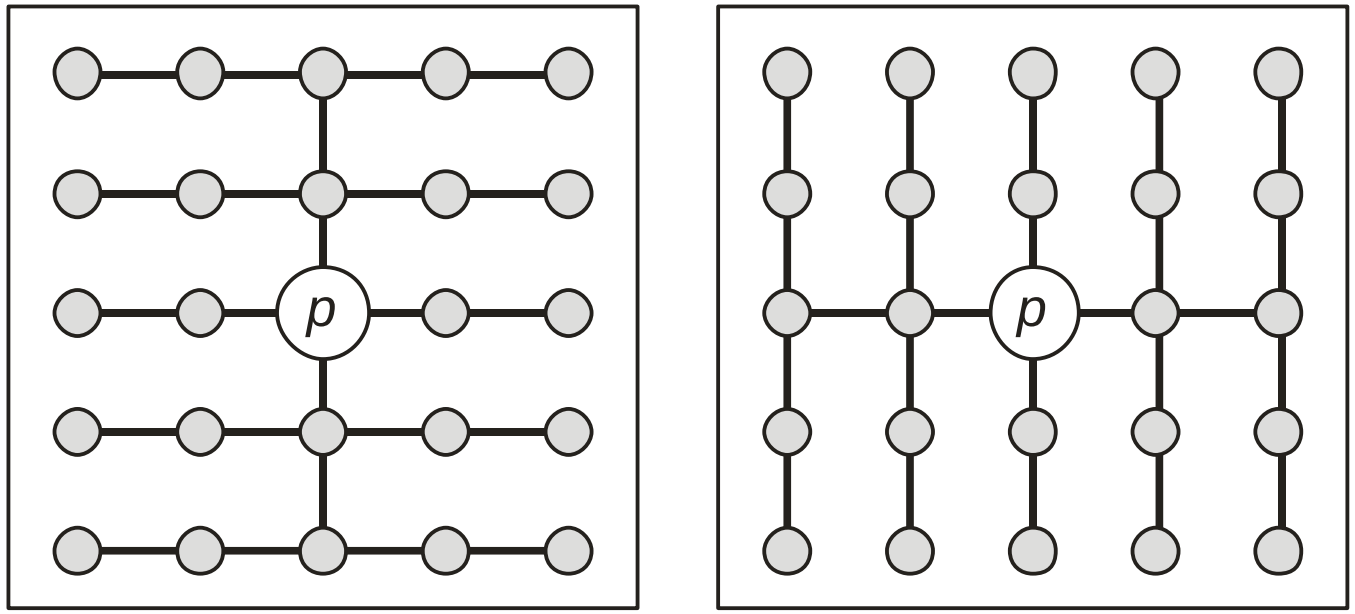
Рисунок: M. Bleyer

H. Hirschmueller. Accurate and Efficient Stereo Processing by Semi-Global Matching and Mutual Information. *CVPR*, 2005.

Алгоритм Semi-Global Matching

Рассогласованности нет, но есть «изолированные» пиксели

Алгоритм Simple Tree



Рисунки: M. Bleyer

В каждом пикселе строятся два дерева, совместно покрывающих все изображение. Алгоритм глобальный и лишен недостатка SGM.

M. Bleyer, M. Gelautz. Simple but Effective Tree Structures for Dynamic Programming-based Stereo Matching. *VISAPP*, 2008.

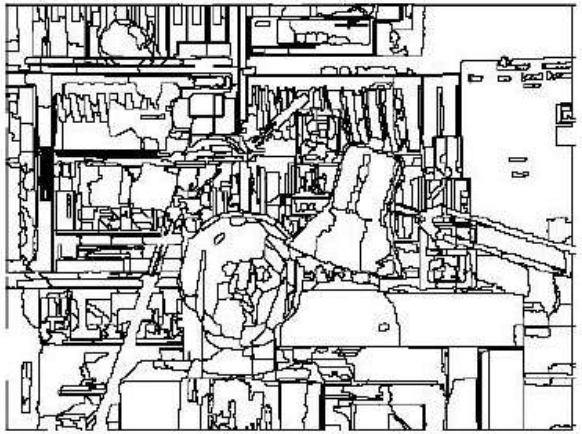
Алгоритм Simple Tree

Пример работы

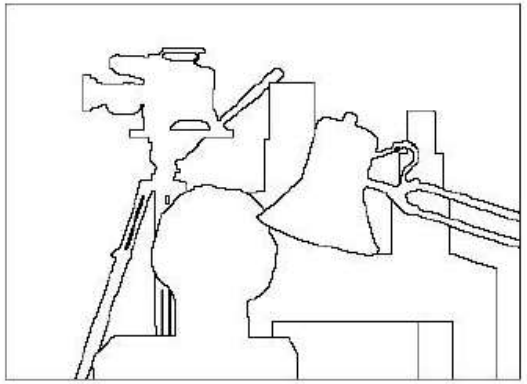
Использование сегментации

- Используется предположение о сегментации («области разрыва диспаритета совпадают с краями на изображении)
- Для его реализации применяют пересегментацию (т.е. сегменты достаточно мелкие, «с запасом»)

Исходное изображение



Результат сегментации



Границы объектов

- Происходит переход из пиксельного пространства в пространство сегментов. Гладкость внутри сегментов форсируется
- Этот подход сейчас показывает наилучшие результаты
  - Возможно, переобучение на Middleburry

Примеры: M. Bleyer

Базовый алгоритм с сегментацией

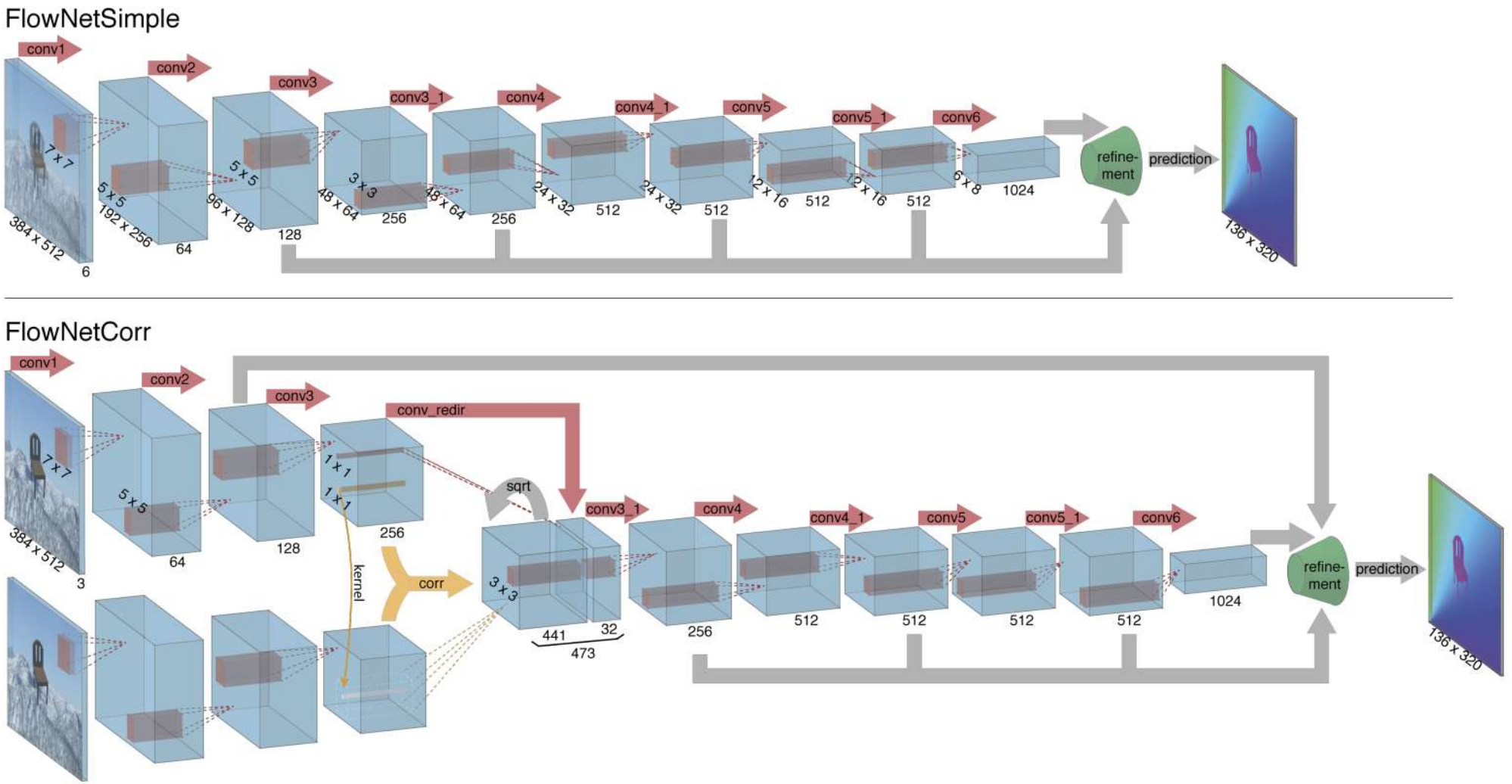
- Пересегментация
- Инициализация решения
  - Любой локальный алгоритм на пикселях
- Аппроксимация сегментов гладкими поверхностями
  - Модель: плоскость, B-сплайн
  - Средство: RANSAC, голосование и т.д.
- Уточнение разметки сегментов
  - Iterated Conditional Modes (ICM), Cooperative Optimization и др.

Использование сегментации

- Преимущества
  - Надежность в областях со слабой текстурой
  - Снижение размерности задачи (оптимизация на уровне сегментов)
- Недостатки
  - Нет защиты от нарушения предположения о сегментации
  - Сложность выбора модели, описывающей изменение диспаритета внутри сегмента
  - Проблему перекрытий все равно необходимо решать на пиксельном уровне

Нейросетевые методы оценки диспаритета

Глобальная оценка нейросетями



Source: https://arxiv.org/abs/1504.06852

A. Dosovitskiy et. al. FlowNet: Learning Optical Flow with Convolutional Networks. 2015

SceneFlow Dataset & DispNet

Monkaa

Flying Things 3D

Driving

- 35000+ пар синтетических кадров
- Сеть DispNet как полный аналог FlowNet, только для расчёта диспартитета
- Базовый подход для глобального стереосопоставления нейросетями

https://arxiv.org/pdf/1512.02134.pdf

Mayer et.al. A Large Dataset to Train Convolutional Networks for Disparity, Optical Flow, and Scene Flow Estimation. 2015

PCW-Net

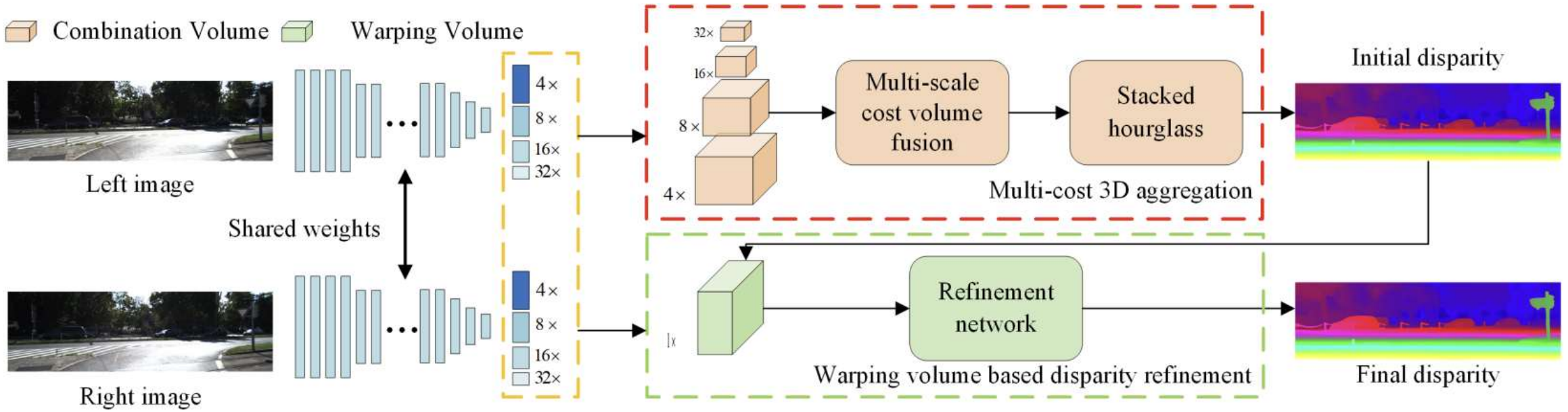


Fig. 2: General Structure of the proposed PCW-Net, which consists of three main modules as multi-scale feature extraction, multi-scale combination volume based cost aggregation, and warping volume based disparity refinement.

- Развитие идей DispNet
- Обучение на SceneFlow + дообучение на реальных данных под датасет

Shen. et.al. PCW-Net: Pyramid Combination and Warping Cost Volume for Stereo Matching. ECCV2022 (https://github.com/gallenszl/PCWNet)

Practical Stereo Matching

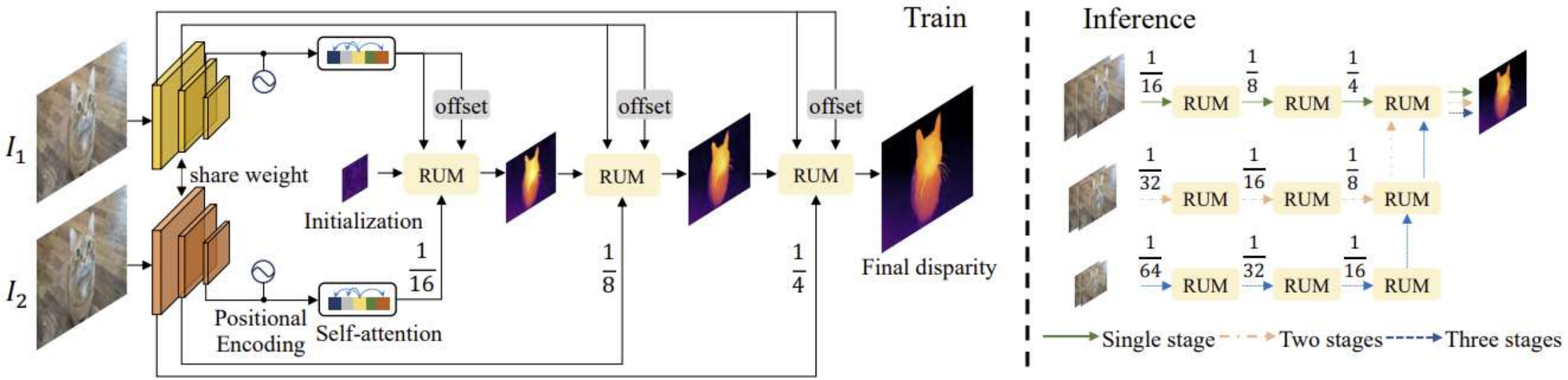


Figure 2. An overview of our proposed network. Left: A pair of stereo images I_1 and I_2 are fed into two shared-weight feature extraction networks to produce a 3-level feature pyramid, which is used to compute different scales of correlations in the 3 stages of cascaded recurrent networks. The feature pyramid of I_1 also provides context information for latter update blocks and offsets computation. In each stage of the cascades, the features and the predicted disparities are refined iteratively using the Recurrent Update Module (RUM, Sec. 3.2), and the final output disparity of the former stage is fed to the next as an initialization. For each iteration in RUM, we apply Adaptive Group Correlation Layer (AGCL, Sec. 3.1) to compute the correlation. Right: Our proposed stacked cascaded architecture in inference phase, which takes an image pyramid as input, taking advantage of multi-level context, as detailed in Sec. 3.3 .

[Practical Stereo Matching via Cascaded Recurrent Network with Adaptive Correlation (CVPR 2022)]

Разные идеи

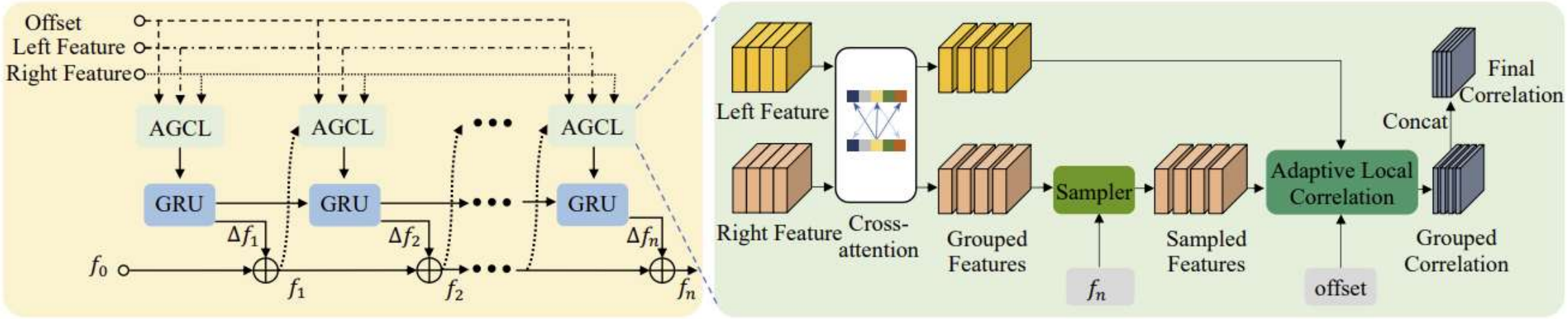


Figure 3. The architecture of proposed modules. Left: Recurrent Update Module (RUM). Right: Adaptive Group Correlation Layer (AGCL). Details are described in Sec. 3.2 and Sec. 3.1, respectively.

- Комбинирование 1D и 2D поиска в окрестностях
  - Ablation study для выбора оптимальной конфигурации
  - Одинаковое число гипотез
- Деформируемые окна поиска для оценки корреляции
- Много элементов из других методов – self-attention, группировка и т.д.

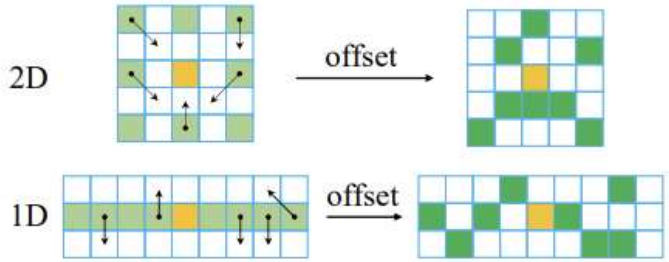


Figure 4. Illustration of the adaptive local correlation. The top and the bottom are 2D and 1D situations respectively, which share the same number of searched neighbors to produce correlation maps in the same shape.

Новый синтетический датасет

Figure 5. Example image-disparity pairs of our synthetic data featuring various shapes and textures (repetitive-texture, reflective non-texture surface, etc.)

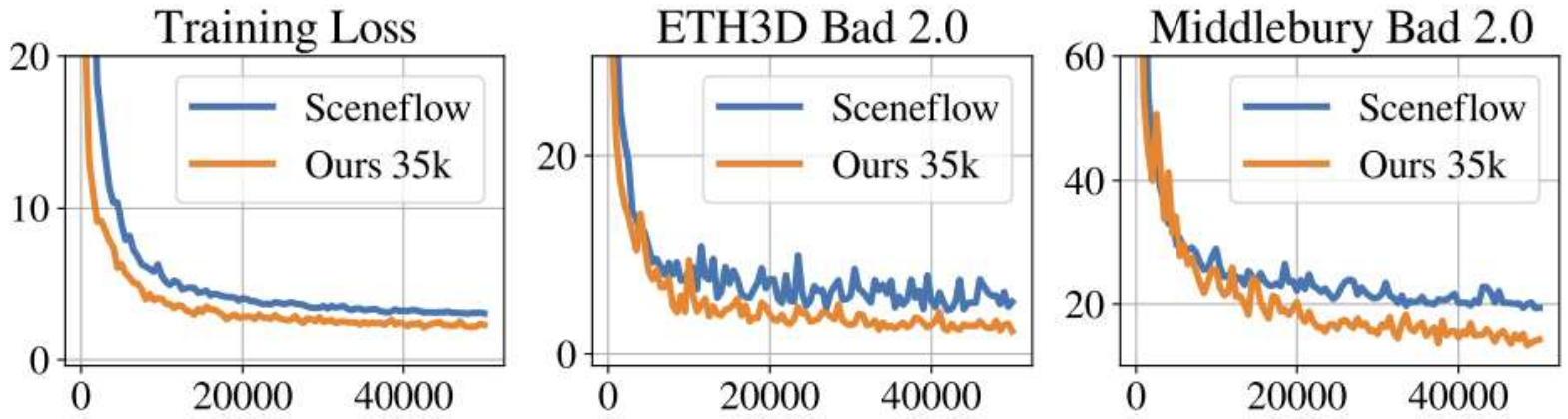
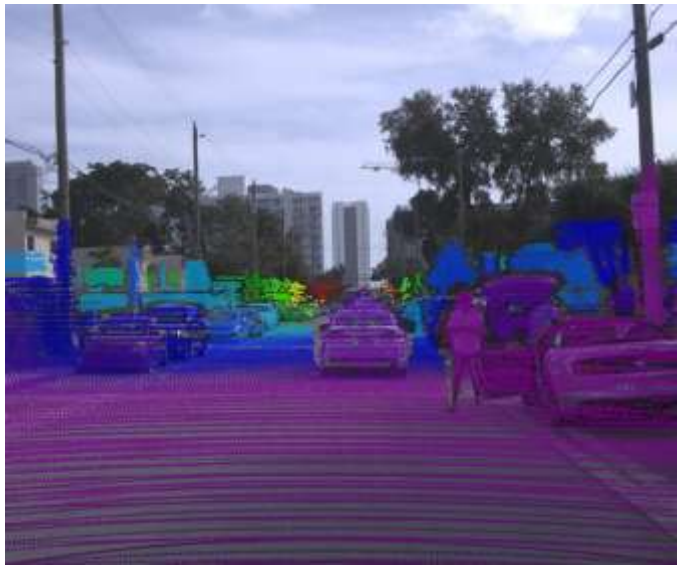
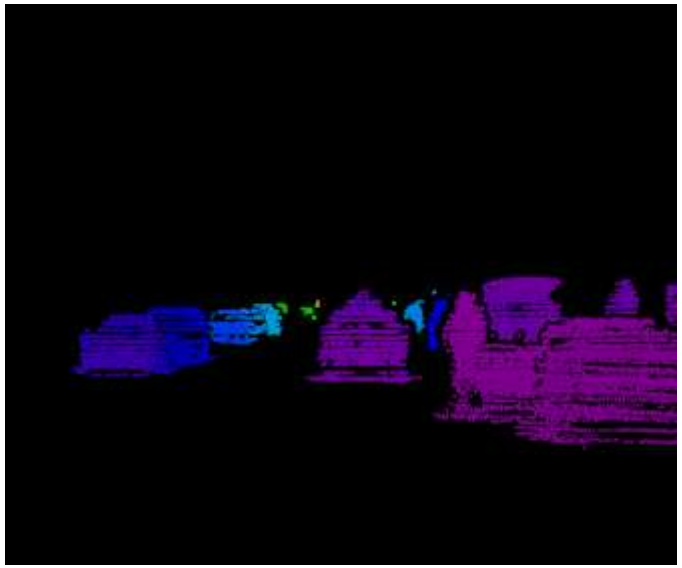


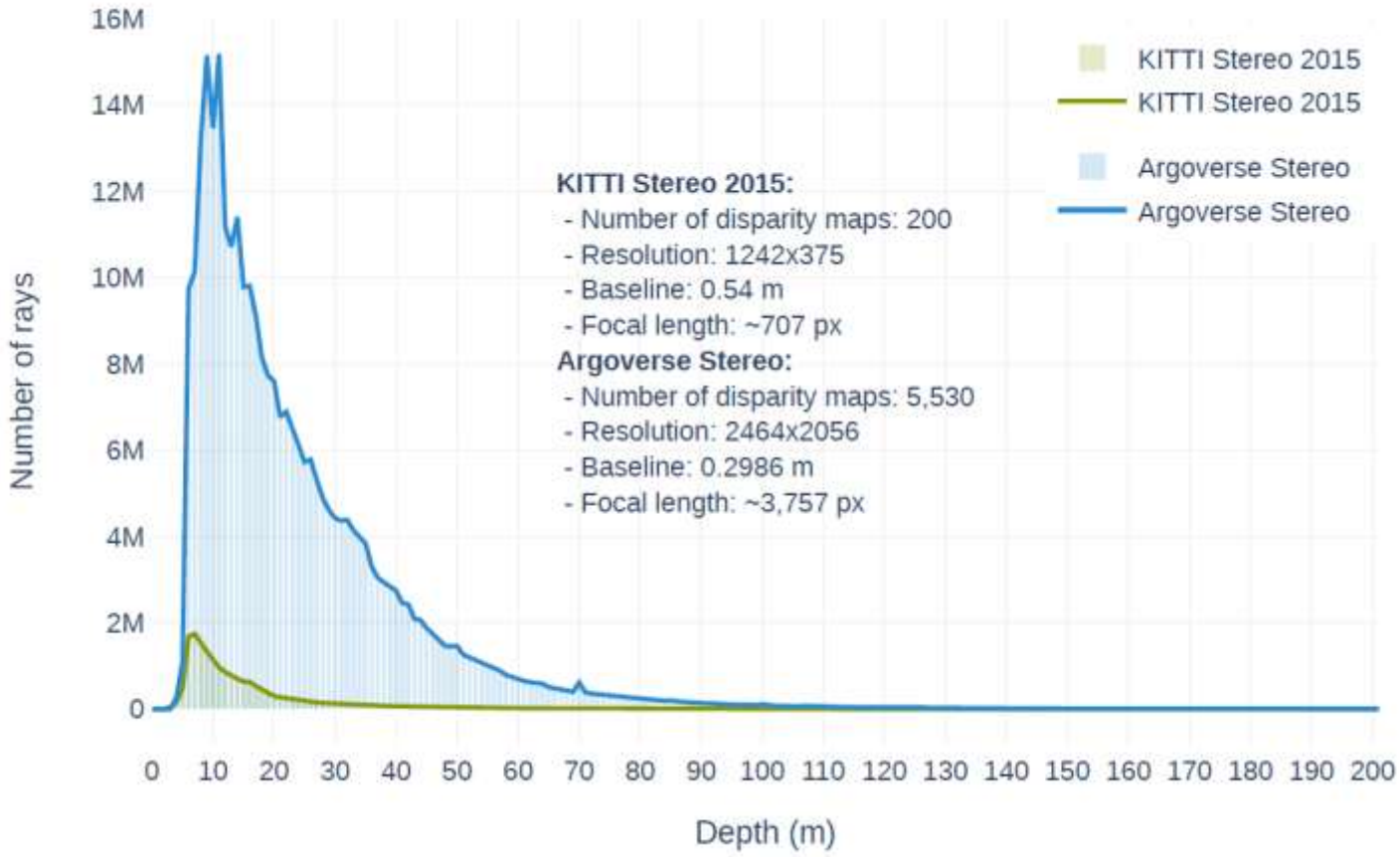
Figure 6. Training loss and ETH3D / Middlebury validation error of models trained with Sceneflow and our synthetic dataset.

Argoverse Dataset





- Новый датасет, вдохновлённый KITTI от стартапа ArgoAI (Carnegie Mellon University & Georgia Institute of Technology)



Willson et.al. Argoverse 2: Next Generation Datasets for Self-Driving Perception and Forecasting. NeurIPS 2021 (link) ArgoAI

KITTI 2015 (февраль 2023)

Evaluation ground truth: All pixels Evaluation area: All pixels

	Method	Setting	Code	D1-bg	D1-fg	D1-all	Density	Runtime	Environment	Compare
1	GANet+ADL			1.38 %	**2.40 %**	**1.55 %**	100.00 %	1.8 s	GPU @ 1.5 Ghz (Python)	
2	IGEV-Stereo			1.38 %	2.67 %	1.59 %	100.00 %	0.18 s	NVIDIA RTX 3090 (PyTorch)	
3	RCA-Stereo			1.40 %	2.71 %	1.62 %	100.00 %	0.40 s	1 core @ 2.5 Ghz (Python)	
4	UPFNet			1.38 %	2.85 %	1.62 %	100.00 %	0.25 s	1 core @ 2.5 Ghz (C/C++)	
5	ERNet			1.36 %	3.09 %	1.65 %	100.00 %	0.2 s	1 core @ 2.5 Ghz (Python)	
ERROR: Wrong syntax in BIBTEX file.										
6	M-FUSE		code	1.40 %	2.91 %	1.65 %	100.00 %	1.3 s	GPU	
L. Mehl, A. Jahedi, J. Schmalfuss and A. Bruhn: M-FUSE: Multi-frame Fusion for Scene Flow Estimation. Proc. Winter Conference on Applications of Computer Vision (WACV) 2023.										
7	SF2SE3		code	1.40 %	2.91 %	1.65 %	100.00 %	2.7 s	GPU @ >3.5 Ghz (Python)	
L. Sommer, P. Schröppel and T. Brox: SF2SE3: Clustering Scene Flow into SE (3)-Motions via Proposal and Selection. DAGM German Conference on Pattern Recognition 2022.										
8	LEAStereo		code	1.40 %	2.91 %	1.65 %	100.00 %	0.30 s	GPU @ 2.5 Ghz (Python)	
X. Cheng, Y. Zhong, M. Harandi, Y. Dai, X. Chang, H. Li, T. Drummond and Z. Ge: Hierarchical Neural Architecture Search for Deep Stereo Matching. Advances in Neural Information Processing Systems 2020.										
9	ACVNet		code	1.37 %	3.07 %	1.65 %	100.00 %	0.2 s	NVIDIA RTX 3090 (PyTorch)	
G. Xu, J. Cheng, P. Guo and X. Yang: Attention Concatenation Volume for Accurate and Efficient Stereo Matching. CVPR 2022.										
10	CGF-ACV			**1.31 %**	3.37 %	1.65 %	100.00 %	0.25 s	1 core @ 2.5 Ghz (Python)	
11	DCANet			1.42 %	2.91 %	1.66 %	100.00 %	0.18 s	3090TI GPU	
12	PCWNet		code	1.37 %	3.16 %	1.67 %	100.00 %	0.44 s	1 core @ 2.5 Ghz (C/C++)	
Z. Shen, Y. Dai, X. Song, Z. Rao, D. Zhou and L. Zhang: PCW-Net: Pyramid Combination and Warping Cost Volume for Stereo Matching. European Conference on Computer Vision(ECCV) 2022.										
13	LaC+GANet		code	1.44 %	2.83 %	1.67 %	100.00 %	1.8 s	GPU @ 2.5 Ghz (Python)	
B. Liu, H. Yu and Y. Long: Local Similarity Pattern and Cost Self- Reassembling for Deep Stereo Matching Networks. Proceedings of the AAAI Conference on Artificial Intelligence 2022.										
14	GwcNet-DCA			1.43 %	2.91 %	1.68 %	100.00 %	0.24 s	GPU @ 2.5 Ghz (Python)	
15	GwcNet+ADL			1.42 %	3.01 %	1.68 %	100.00 %	0.32 s	GPU @ 1.5 Ghz (Python)	
16	CREStereo		code	1.45 %	2.86 %	1.69 %	100.00 %	0.41 s	GPU @ >3.5 Ghz (Python)	
J. Li, P. Wang, P. Xiong, T. Cai, Z. Yan, L. Yang, J. Liu, H. Fan and S. Liu: Practical Stereo Matching via Cascaded Recurrent Network with Adaptive Correlation. 2022.										

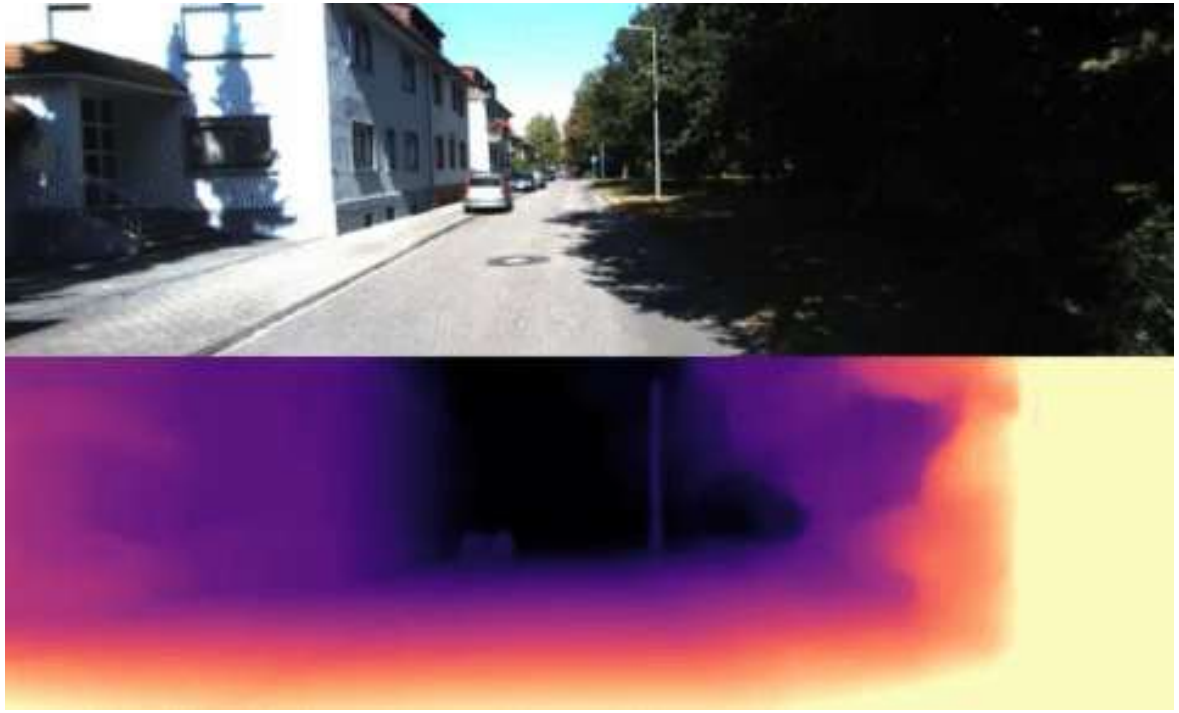
Резюме бинокулярного стерео

- Бинокулярное стерео разбивается на 3 шага – ректификация, стереосопоставление и триангуляция
- Ключевые идеи для стереосопоставления:
  - Локальное сопоставление (для каждого пиксела независимо)
  - Глобальное (для всех пикселов сразу)
  - Полуглобальное (например, для каждого независимо, но почти по всему изображению)
  - Пересегментация (условие связанности в визуальном сегменте)
- Нейросети позволяют решать глобально, с использованием cost volume
- Явная ограниченность датасетов для обучения

Оценка карт глубины

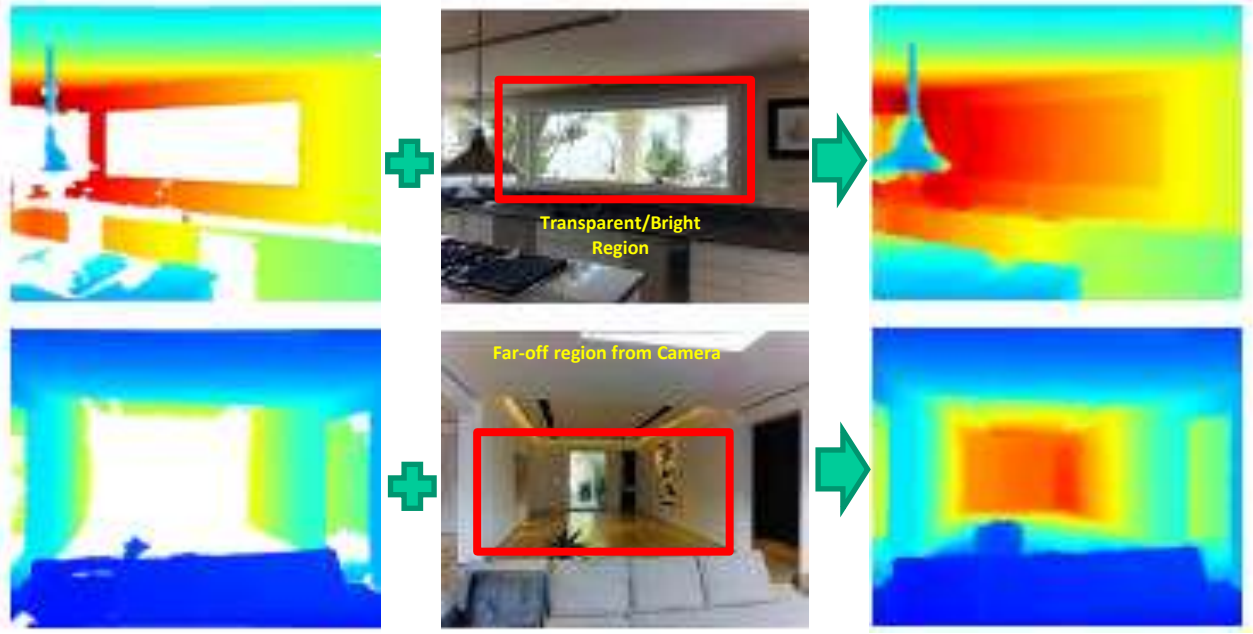
Задача оценки глубины

Single View Depth Estimation (SVDE)



Goddard et. al. Digging into Self-Supervised Monocular Depth Prediction. ICCV2019

Depth Completion (DC)

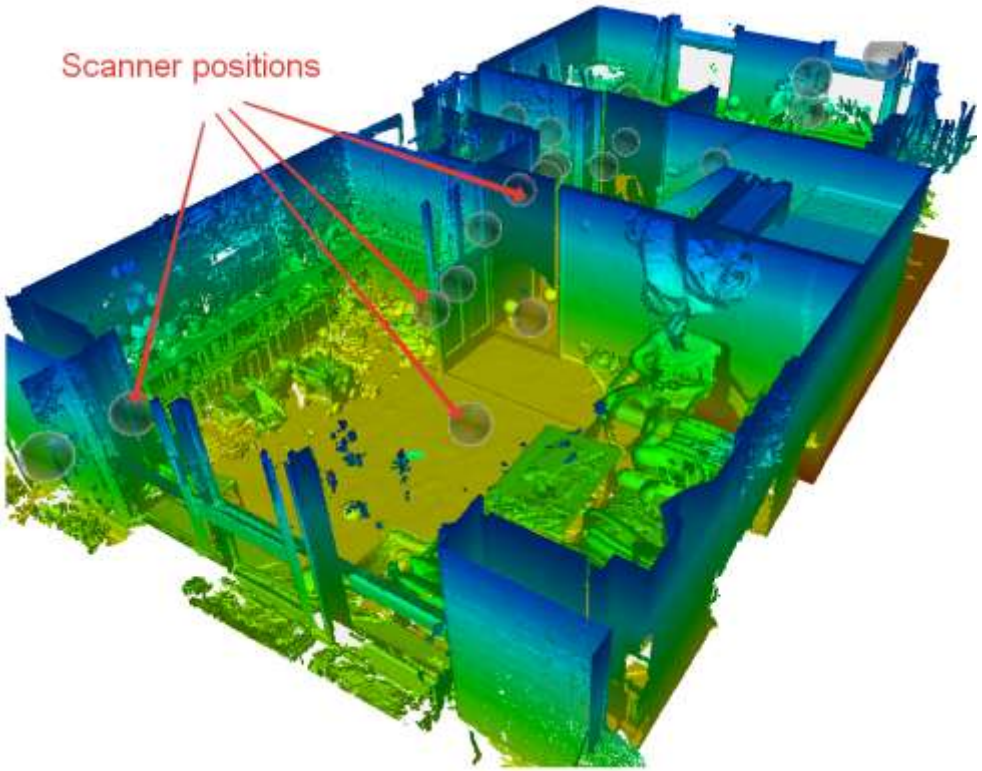


Raw depth from depth sensor

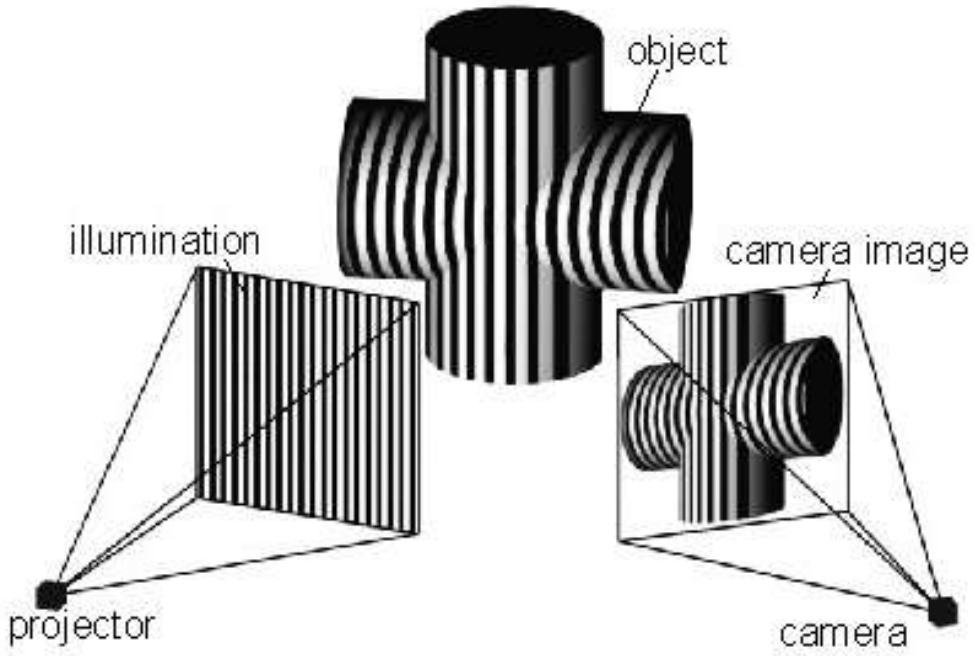
RGB Image

Result

Лазерные сканеры



Камера со структурной подсветкой



Структурная подсветка - текстурируем любую однотонную поверхность уже сегодня

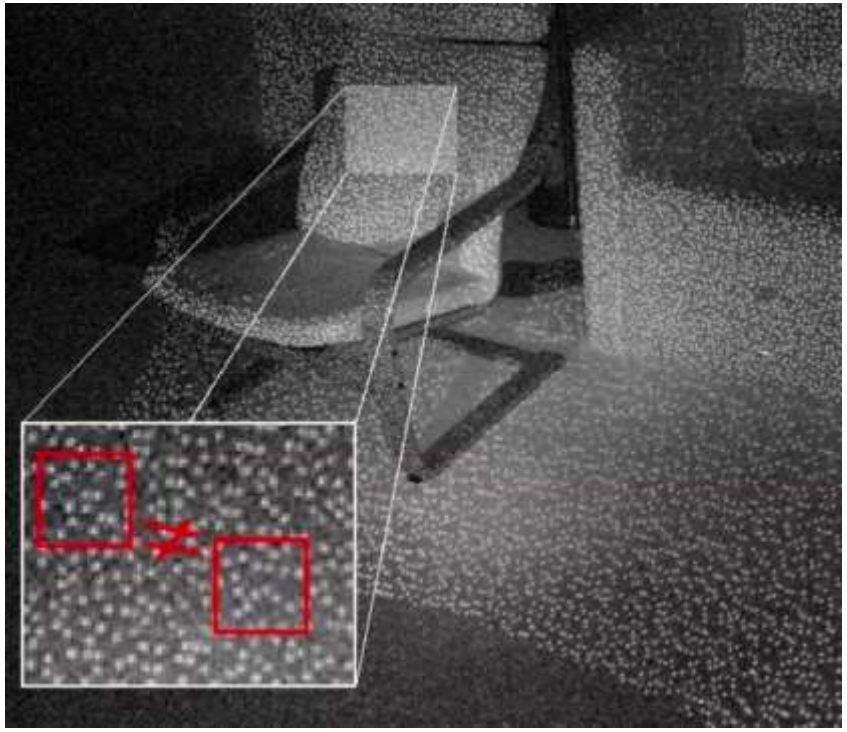
3D сканер ценой в 1 годовой грант РФФИ, да и рабочая глубина ограничена

https://wiki.dfrobot.com/brief_analysis_of_camera_principles

Microsoft Kinect (2009-2023)

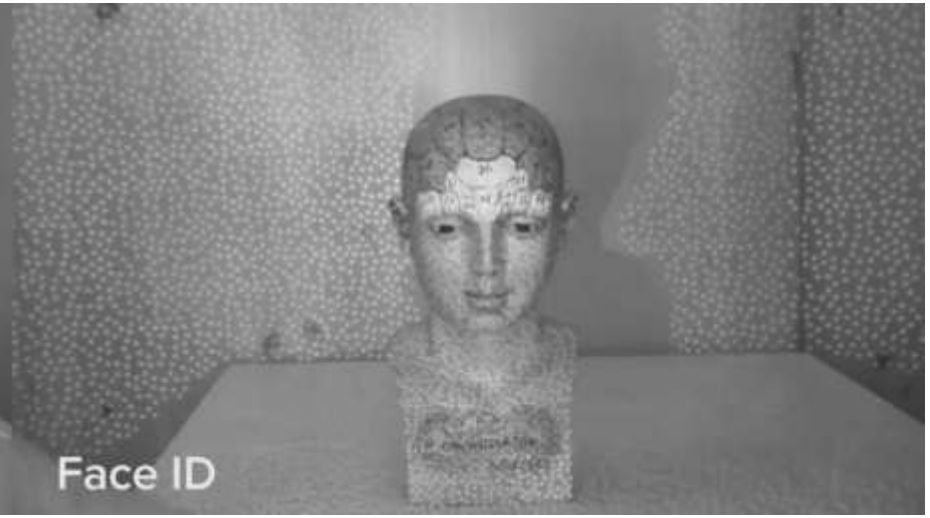
Технология компании PrimeSense, лицензированная Microsoft и реализованная в камере Kinect for Xbox 360 ("Project Natal")

Kinect for Xbox One & Kinect for Windows - Time of Flight камеры

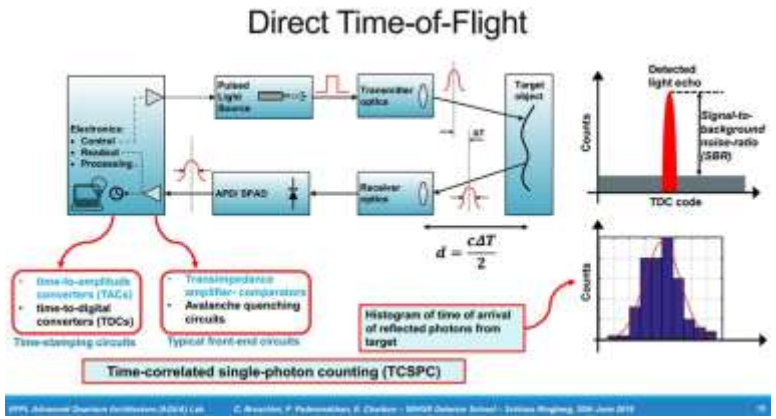


"Умная" структурная подсветка в виде набора пятен по хитрому шаблону. Форма пятна также анализируется для оценки глубины и нормалей.

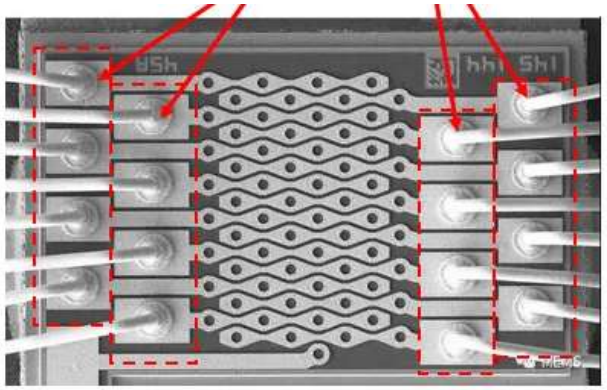
Apple и 3D камеры



- Apple приобрела PrimeSense в 2013 году
- В 2020 году объявила выпуск Ipad Pro с "инновационным" LiDAR Scanner (с технологиями Sony)
- 576 точки в поле зрения, в которых измеряется глубина
- Depth Completion - интеграция всех данных для оценки глубины



Измерение времени возврата отраженного луча



64 лазерных диода, с помощью дифракционной решётки x9

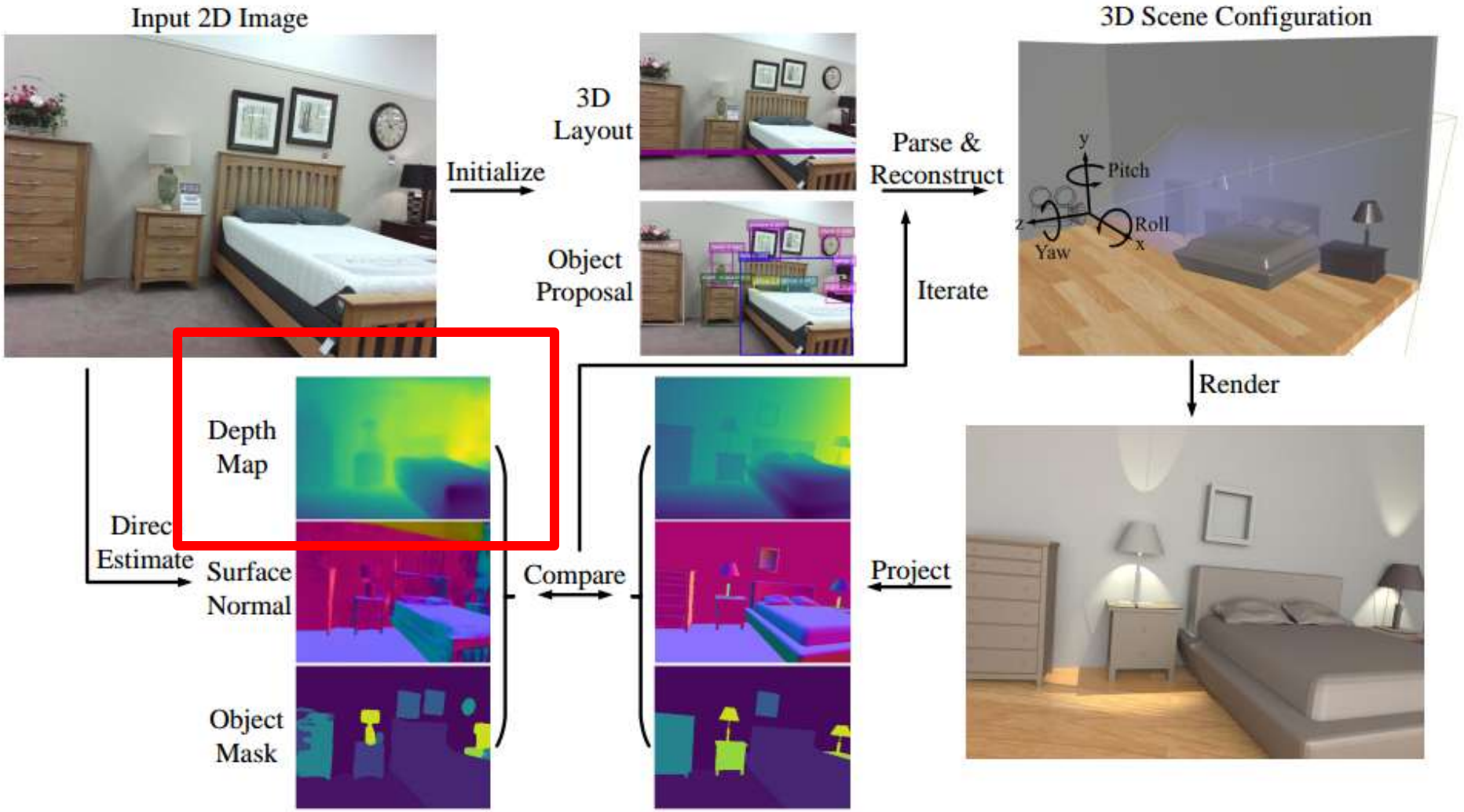
https://4sense.medium.com/apple-lidar-demystified-spad-vcsel-and-fusion-aa9c3519d4cb

dToF и применение карт глубины

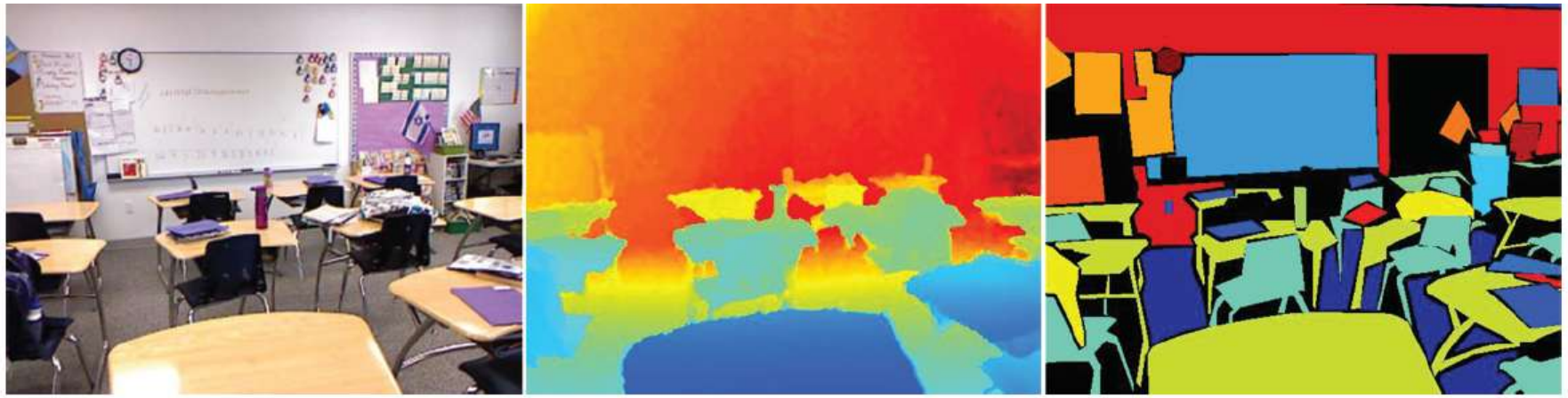
Разреженная карта глубины, напоминающая LIDAR

Эффект боке в ночном режиме

Применение для реконструкции



NYUv2



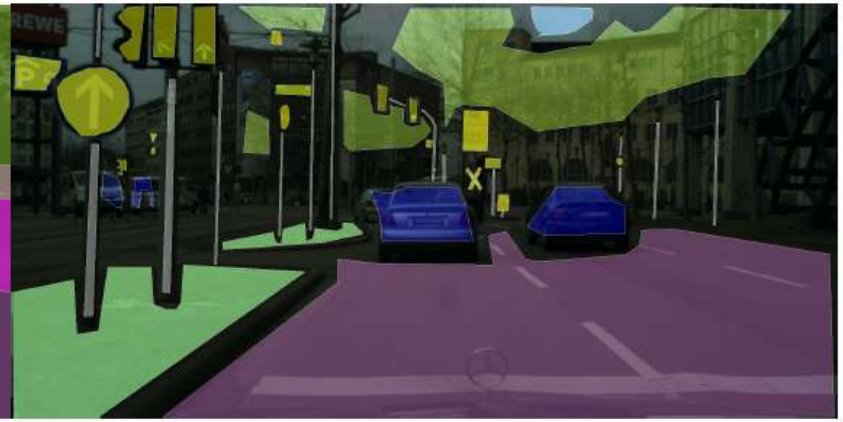
семантическая сегментация для 1 500 кадров

120 000 исходных кадров

карта глубины с Kinect

Silberman et al. Indoor Segmentation and Support Inference from RGBD Images. ECCV 2012

Cityscapes



5 000 кадров

20 000 кадров

50 городов, несколько времен года

30 классов объектов

стереопары

GPS-координаты

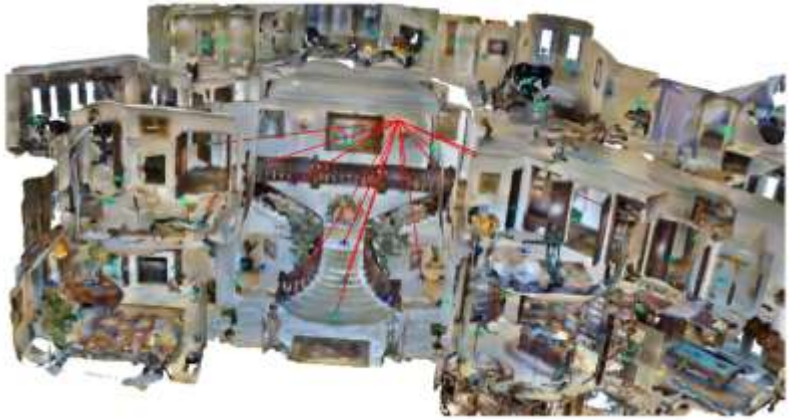
одометрия

Cordts et al. The cityscapes dataset for semantic urban scene understanding. CVPR 2016

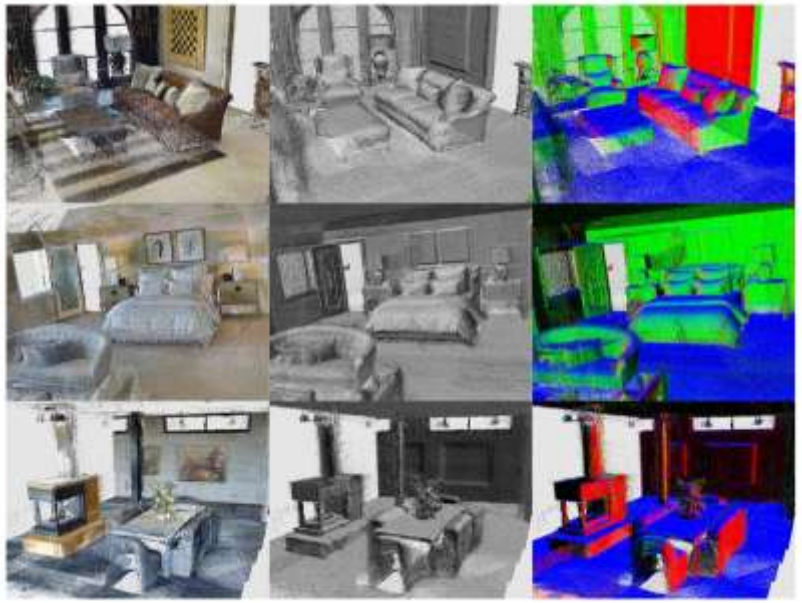
Matterport 3D Dataset

- 3 combined structured light sensors
- Scans are aligned using structure from motion software
- 11k panoramic views from 194k RGB-D images of 90 building-scale scenes
- RGB, depth, normals, surface reconstructions, camera poses, 2D and 3D semantic segmentations

Final building-scale reconstruction



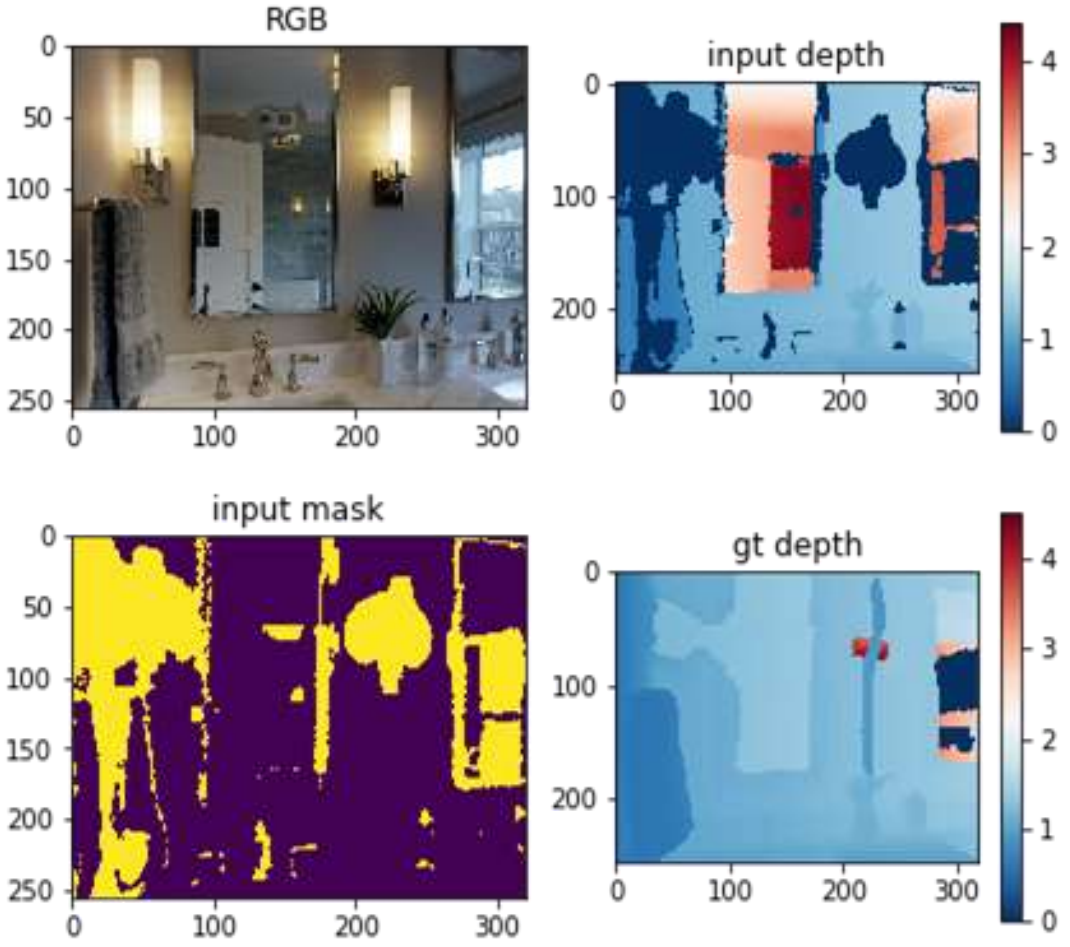
Matterport 3D camera

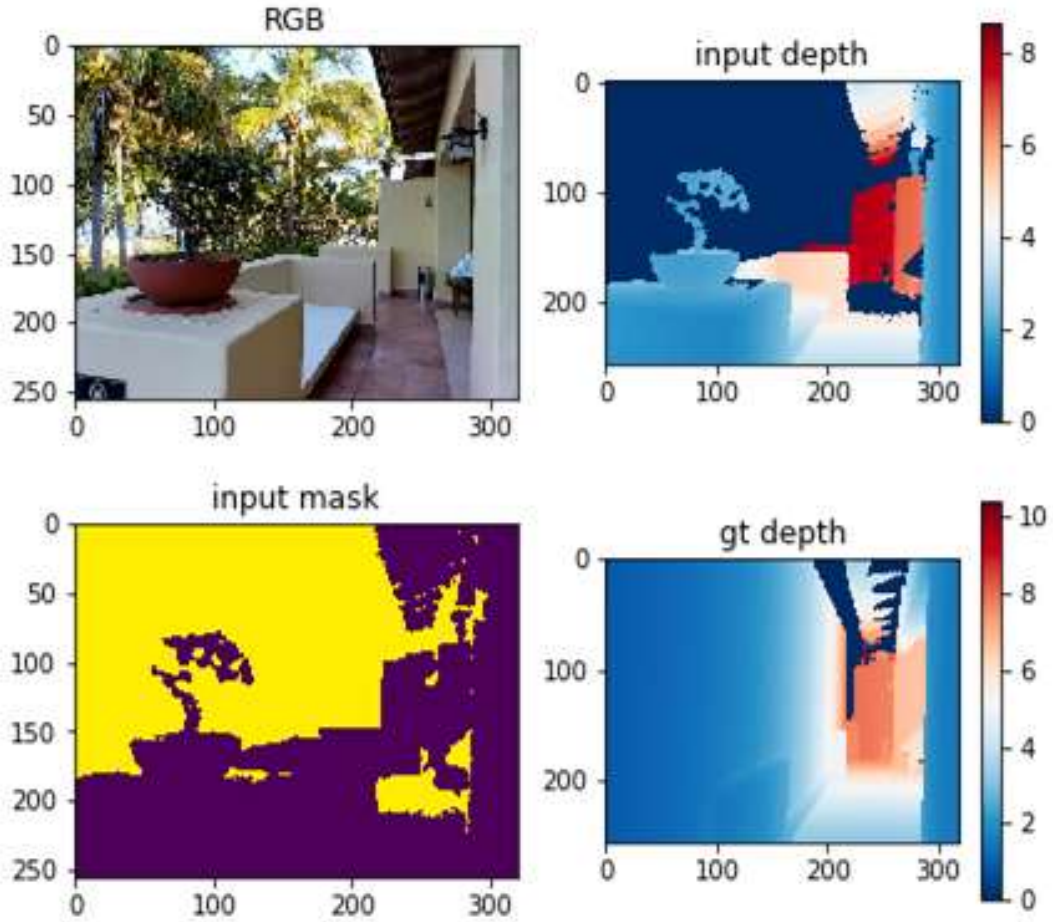


Raw point clouds: color, diffuse shading, normals

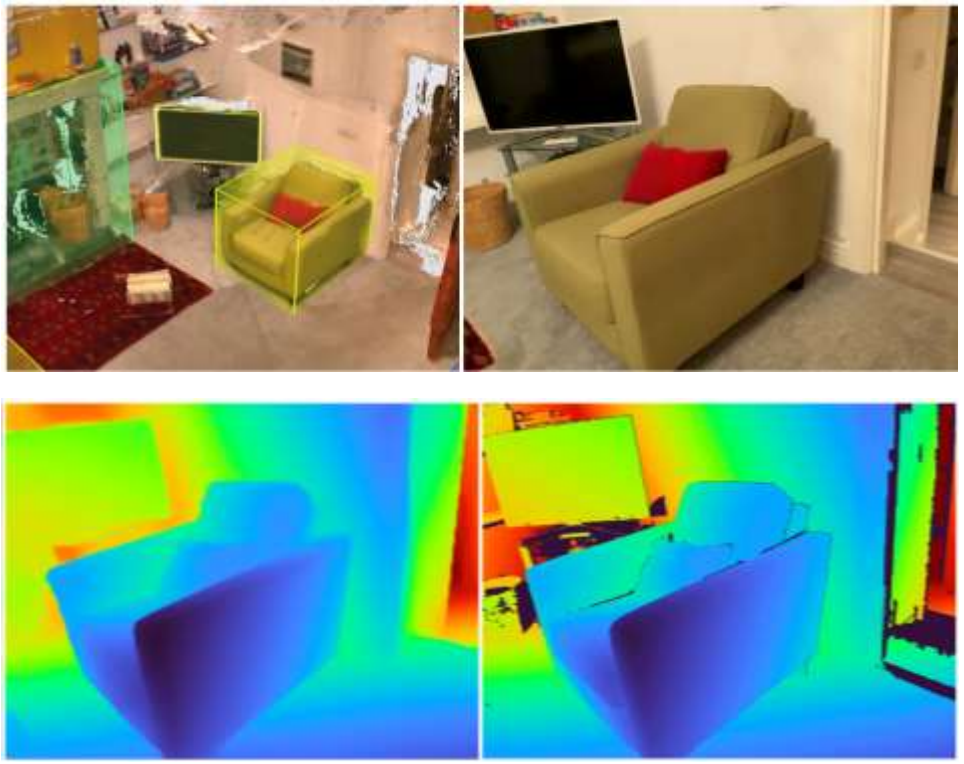
Ограничения на примере Matterport 3D

Зеркала, ограниченная глубина (на улице), тонкие детали и т.д.





ARKit Scenes (2021)



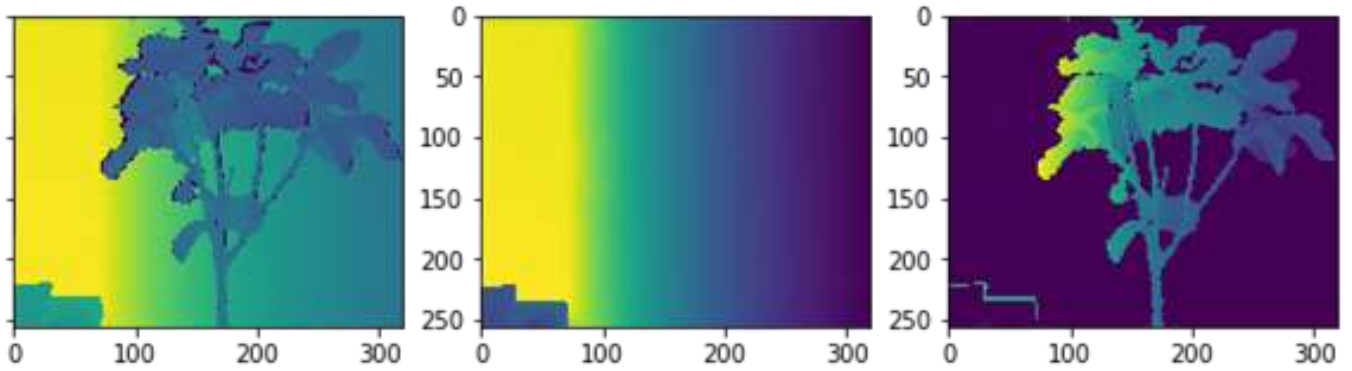
Dataset	Size	3DOD Labels	HR #Frames	HR	LR
MPI-Sintel	35 Scenes	-	1,628	1024×436	-
Middleburry	-	-	34	432×381 to 2964×2000	-
NYU v2	464 Scans	1,449 frames	-	-	-
SUN RGB-D	10K frames	10k frames	-	-	-
SceneNN	100 Scans	100 scans	-	-	-
ScanNet	707 Venues 1,513 Scans	1,513 scans	-	-	-
Matterport3D	2,056 rooms	2056 scans	-	-	-
ARKitScenes	**1,661 venues 5,047 Scans**	**5047 scans**	**450K^2**	**1920×1440 Laser Scanner**	**256×192 ARKit Depth**

- Сканирование комнат Faro Focus S70 со штатива и Ipad Pro 2020 в hand-held режиме
- Бенчмарки 3D object detection & depth map upsampling.

https://github.com/apple/ARKitScenes

Оценка качества

- δ-metric $max(\frac{D(p)}{D_0(p)}, \frac{D_0(p)}{D(p)}) < t$

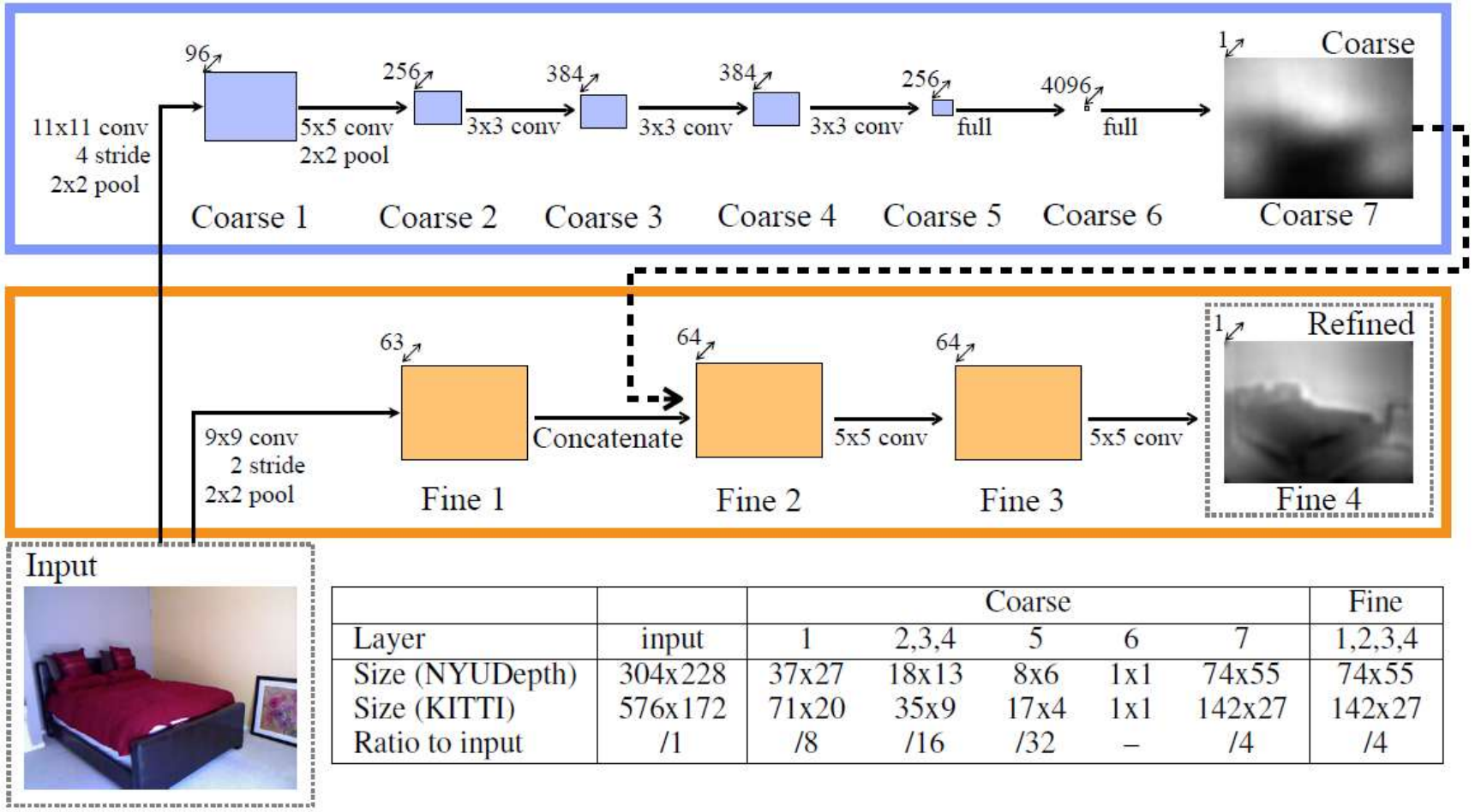


raw depth ground truth error

	δ<1.05	δ<1.10	δ<1.25	δ<1.25^2	δ<1.25^3
Raw depth vs GT depth	0.895	0.912	0.929	0.944	0.953

Оценка карт глубины по изображению

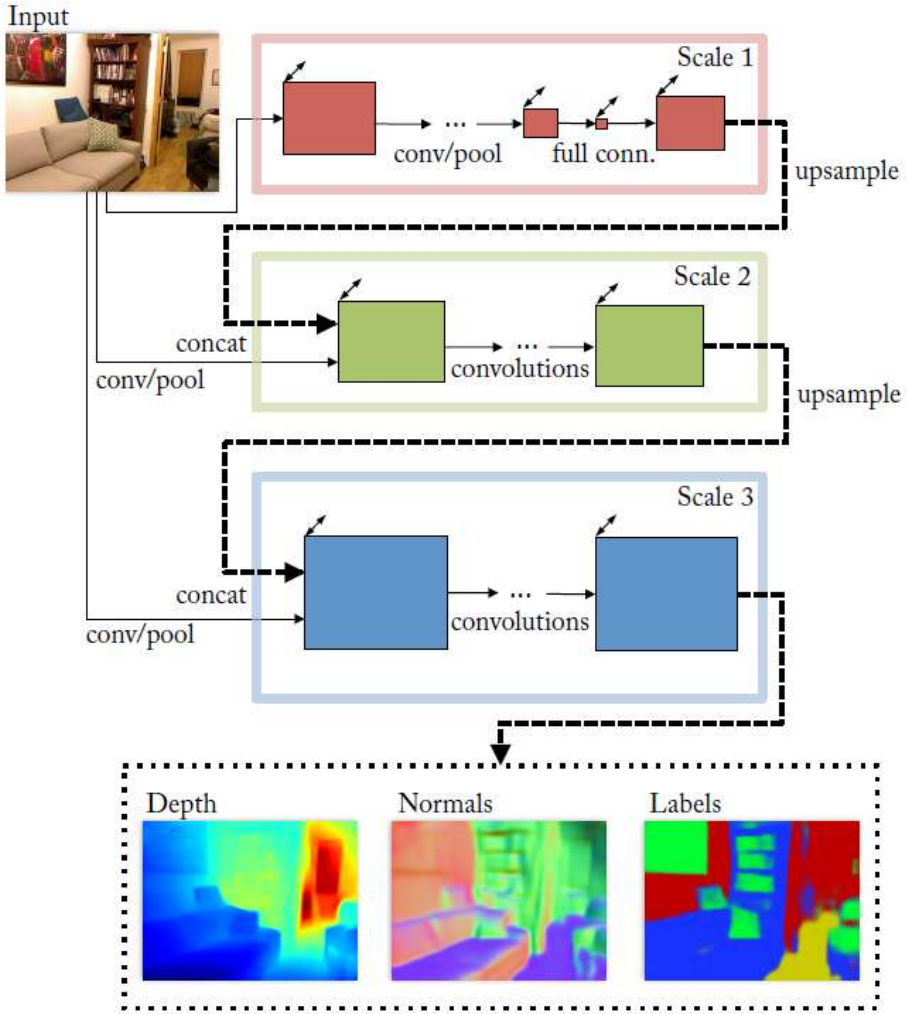
Одна из первых работ по SVDE



		Coarse					Fine
Layer	input	1	2,3,4	5	6	7	1,2,3,4
Size (NYUDepth)	304x228	37x27	18x13	8x6	1x1	74x55	74x55
Size (KITTI)	576x172	71x20	35x9	17x4	1x1	142x27	142x27
Ratio to input	/1	/8	/16	/32	–	/4	/4

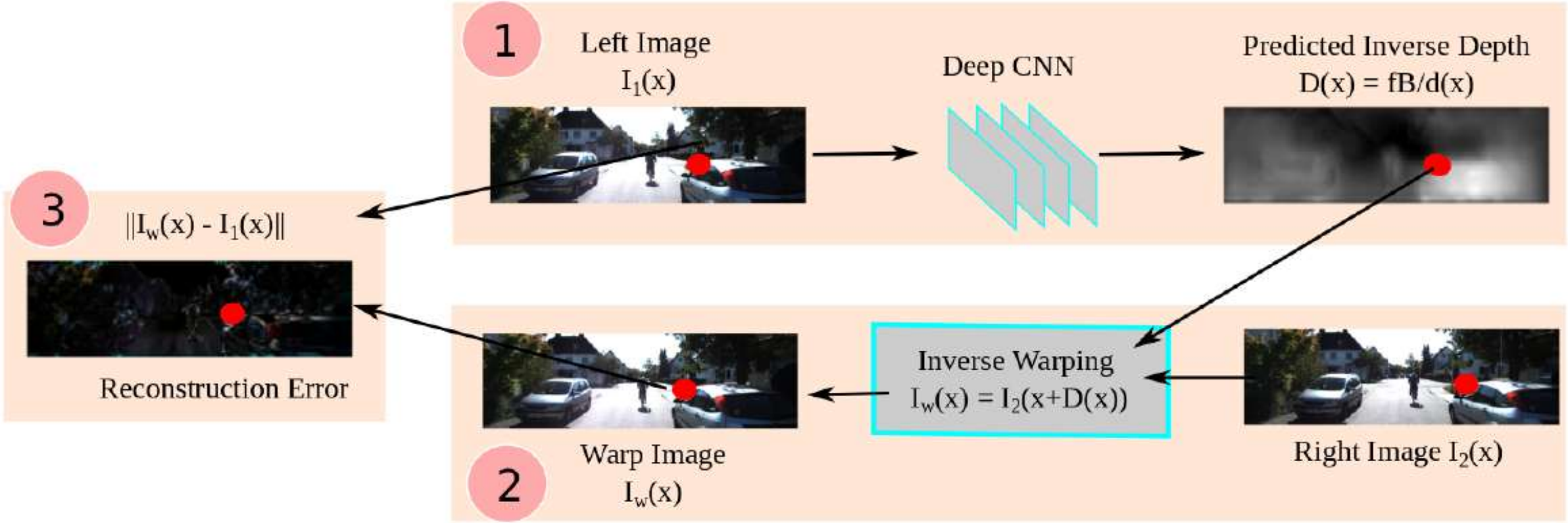
Eigen et al. Depth Map Prediction from a Single Image using a Multi-Scale Deep Network. NIPS 2014

Совместная оценка



Eigen et al. Predicting Depth, Surface Normals and Semantic Labels with a Common Multi-Scale Convolutional Architecture. ICCV 2015

Обучение моделей по стереоданным



Garg et al. Unsupervised CNN for Single View Depth Estimation: Geometry to the Rescue. ECCV 2016

Функция потерь

Функция потерь:

$$E = E_{rec} + \gamma E_{smooth}$$

$$E_{smooth} = \sum_x \|\nabla D(x)\|^2$$

$$E_{rec} = \sum_x \|I_w(x) - I_1(x)\|^2 = \sum_x \|I_2(x + D(x)) - I_1(x)\|^2$$

Датасеты

Dataset	Dense/ sparse	Depth type	#Samples
DIML Indoor [14]	indoor	absolute	220K
MegaDepth [18]	general	UTS	130K
ReDWeb [41]	general	UTSS	3600
3D Movies [27]	general	UTSS	500K
Sintel [2]	general	absolute	1064
NYUv2 Raw [22]	indoor	absolute	407K
TUM-RGBD [34]	indoor	absolute	80K
DIW [4]	general	ordinal	496K

Table 1: Overview of the datasets used in our experiments. **Top**: training datasets, **bottom**: test datasets.

UTS и UTSS

$$\mathbf{d}^{*-1} = \tilde{C}_1 \mathbf{d}^{-1}$$

- Up-to-Scale (UTS), где d – оценённая глубина

$$\mathbf{d}^{*-1} = C_1(\mathbf{D} + C_2)$$

- Up-to-Shift-and-Scale (UTSS), где D - диспаритет

$$\mathcal{L}_{Mixture} = \mathbb{I}_{UTS}\mathcal{L}_{SI} + \mathcal{L}_{SSI}$$

- Где I_{uts} =1, если есть UTS или абсолютные данные

Сравнение результатов

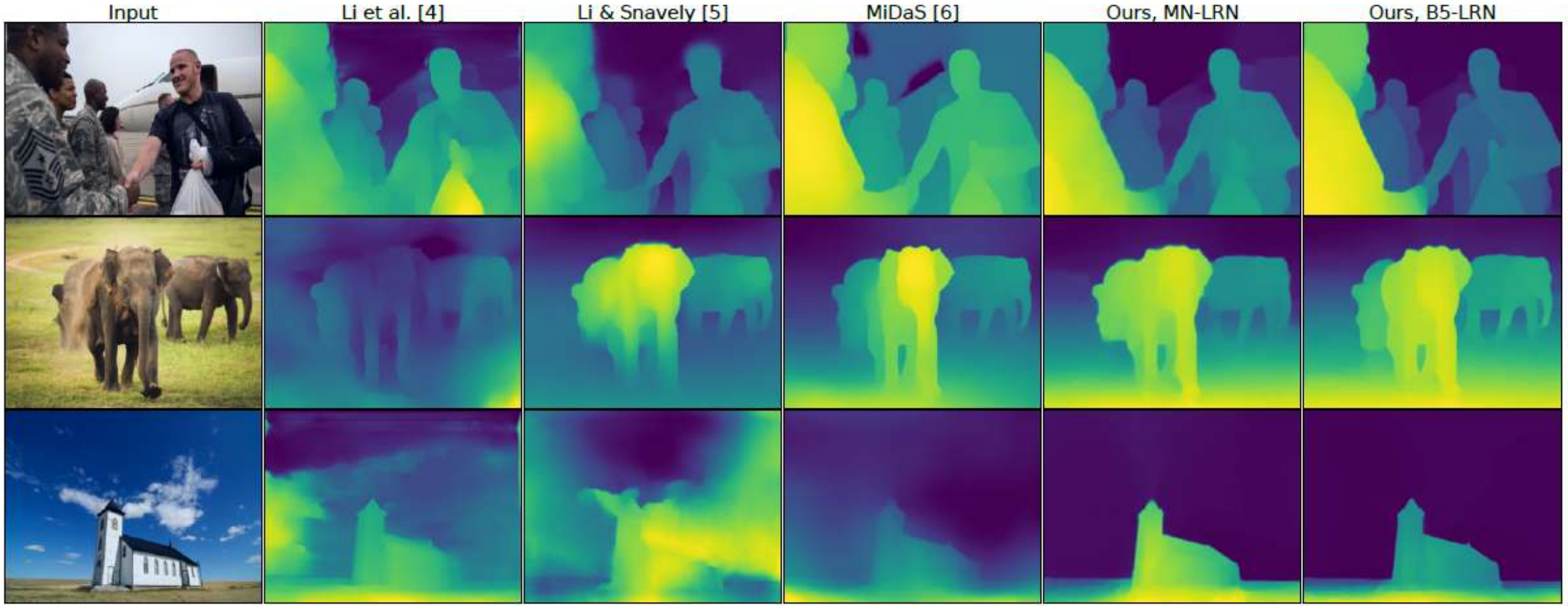


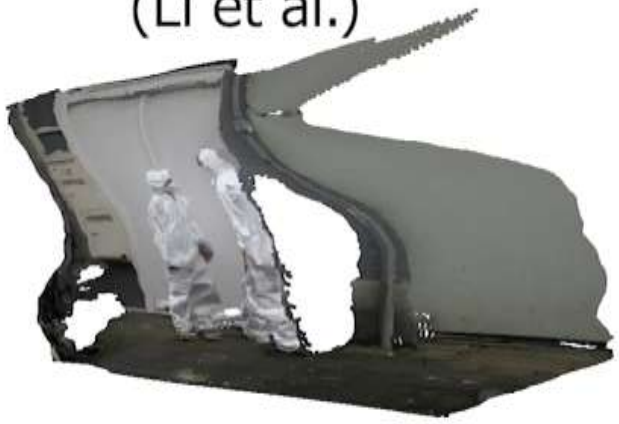
Figure 5: Qualitative comparison of depth maps produced by our models and existing competitors. Images are taken from the DIW dataset and were not seen during training.

Визуализация

Source

Ours, B5-LRN4

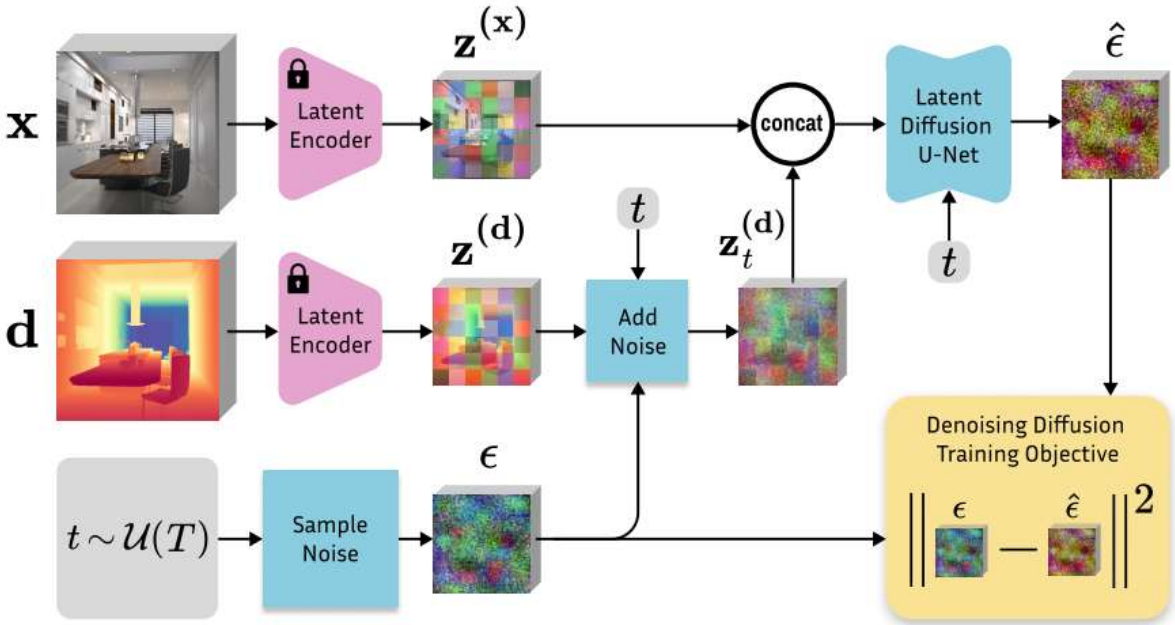
Mannequin Challenge
(Li et al.)



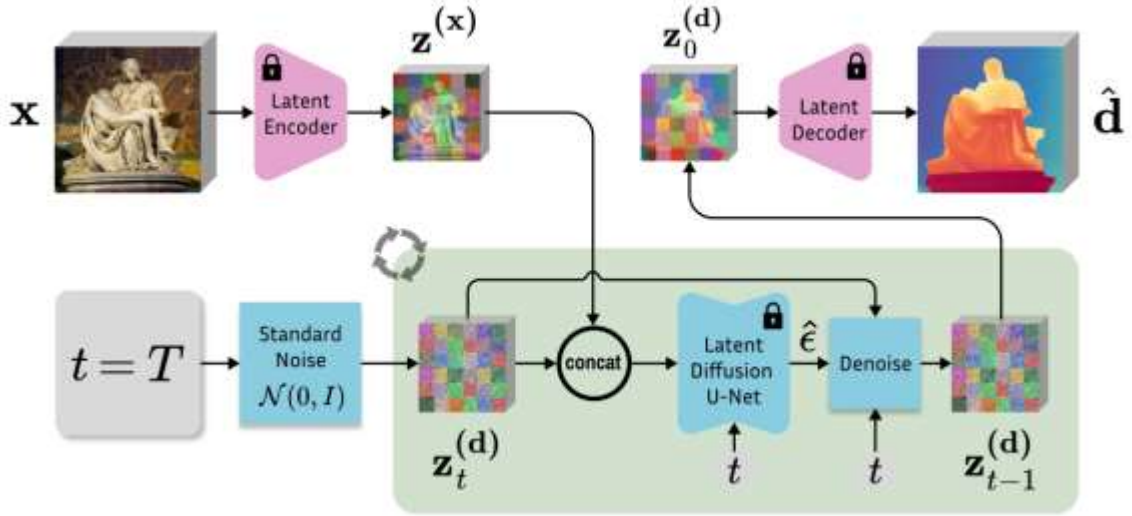
Megadepth

MIDAS
(w/o alignment)

Использование генеративных моделей



Дообучение модели



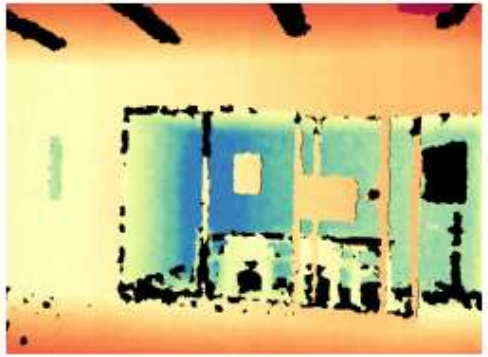
Вывод

https://marigoldmonodepth.github.io/

Depth Anything

https://depth-anything-v2.github.io/

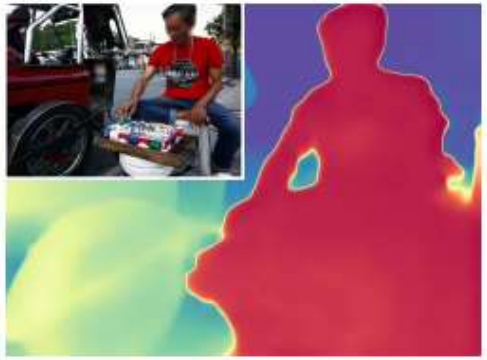
Проблемы в датасетах



(a) Label noise in transparent object (depth sensor)

(b) Label noise in repetitive pattern (stereo matching)

(c) Label noise in dynamic objects (SfM)



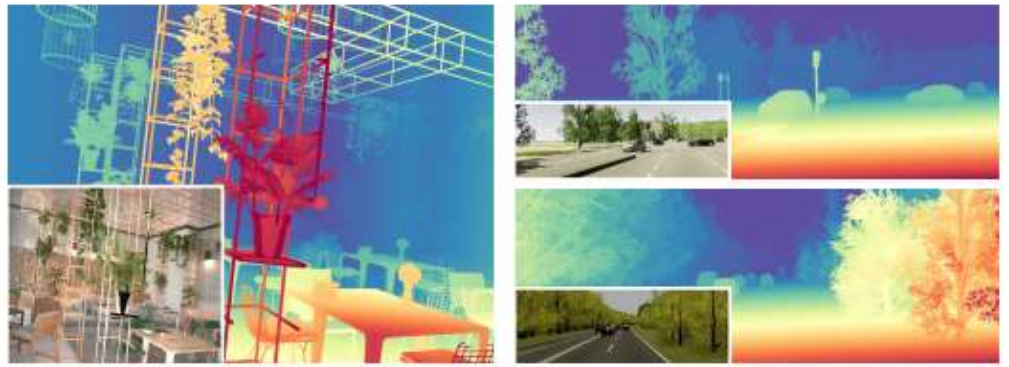
(d) Caused errors in model prediction

Figure 3: Various noise in "GT" depth labels (a: NYU-D [70], b: HRWSI [83], c: MegaDepth [37]) and prediction errors in correspondingly trained models (d). Black regions are ignored during training.

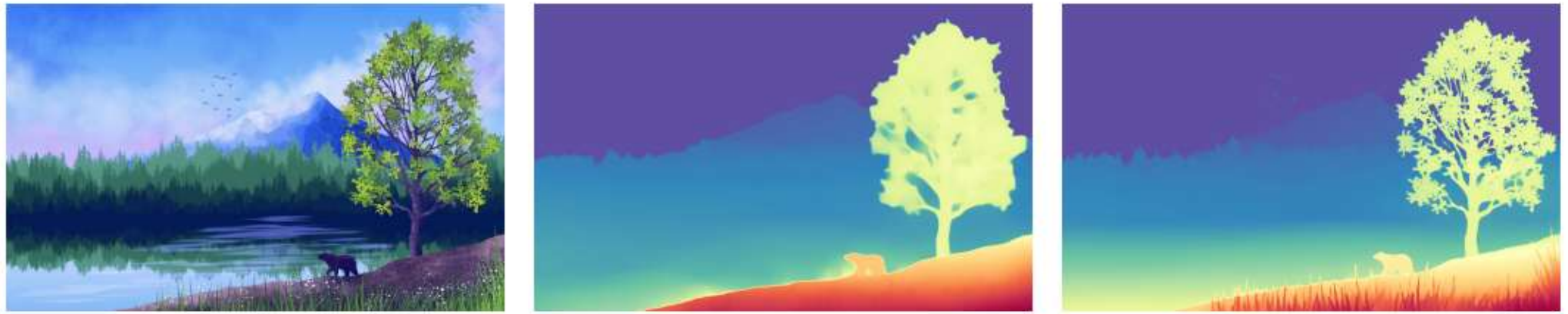
https://depth-anything-v2.github.io/

Синтетические данные

(a) Coarse depth of real data (HRWSI [83], DIML [14])



(b) Depth of synthetic data (Hypersim [58], vKITTI [9])



(c) Predictions of models trained on labeled real images (middle) and synthetic images (right)

Figure 4: Depth labels of real images (a) and synthetic images (b), and the corresponding model predictions (c). The labels of synthetic images are highly precise, and so are their trained models.

https://depth-anything-v2.github.io/

DepthAnything V2

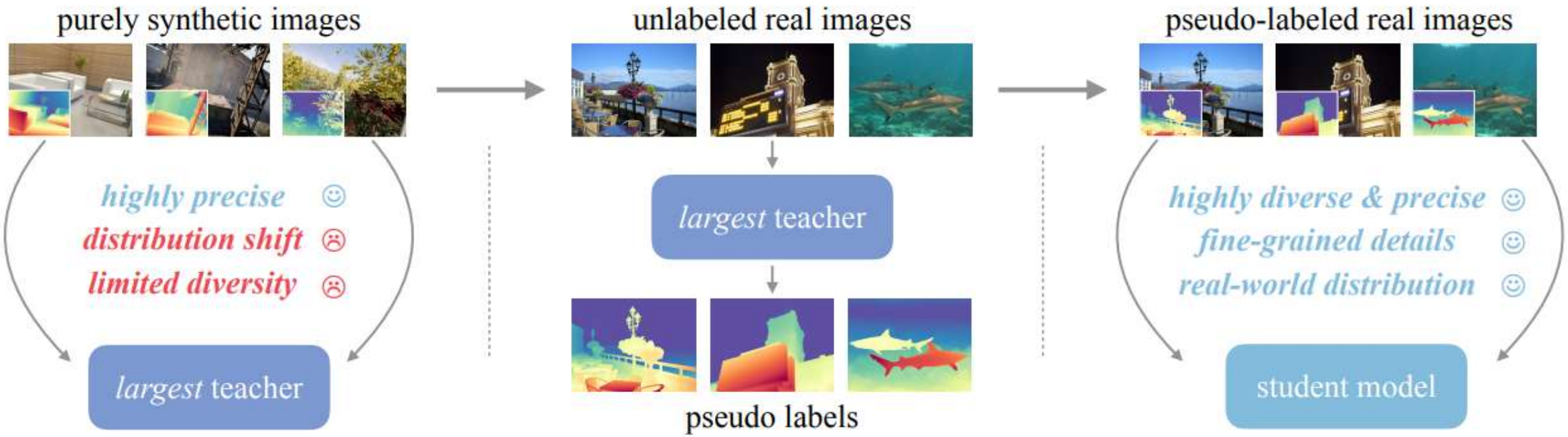
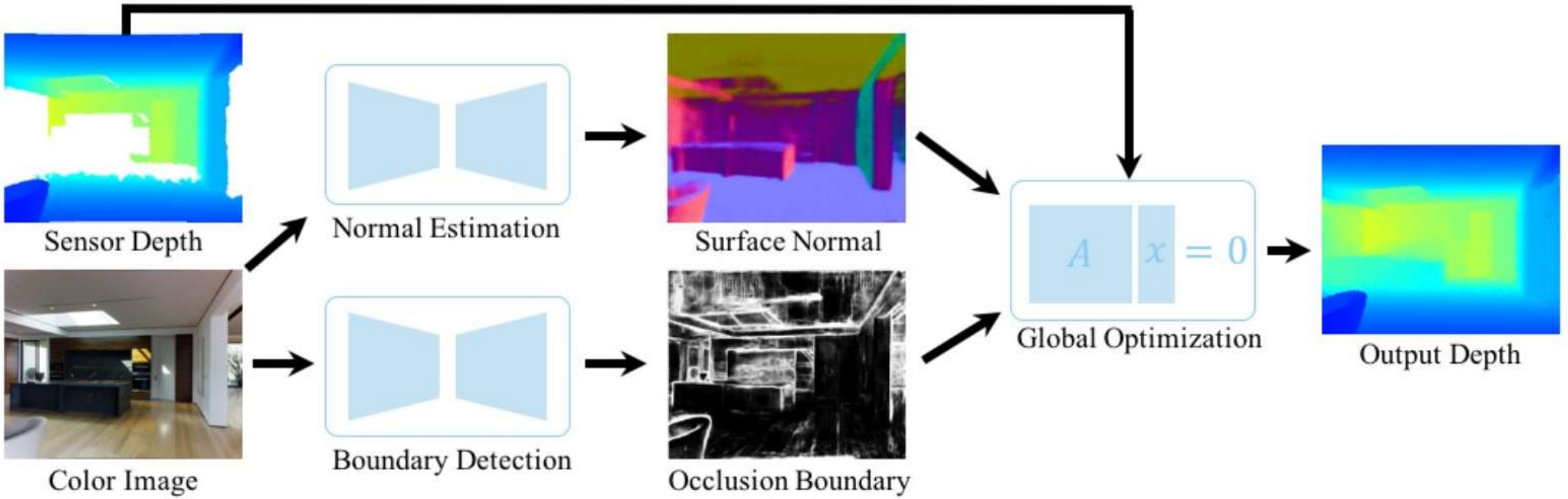


Figure 7: Depth Anything V2. We first train the most capable teacher on precise synthetic images. Then, to mitigate the distribution shift and limited diversity of synthetic data, we annotate unlabeled real images with the teacher. Finally, we train student models on high-quality pseudo-labeled images.

https://depth-anything-v2.github.io/

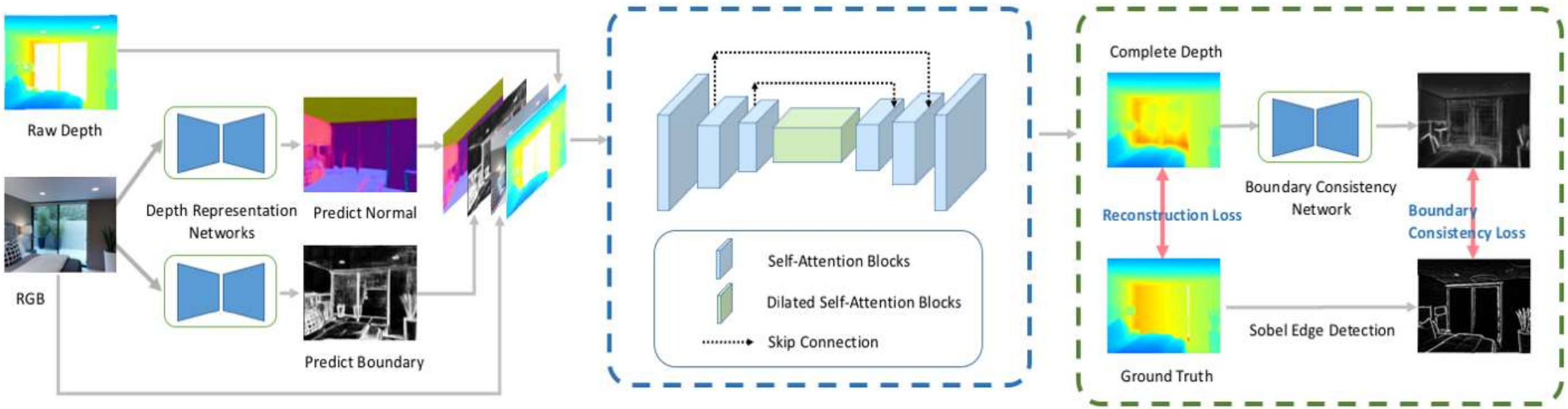
Методы решения задачи depth completion

Пример решения Depth Completion



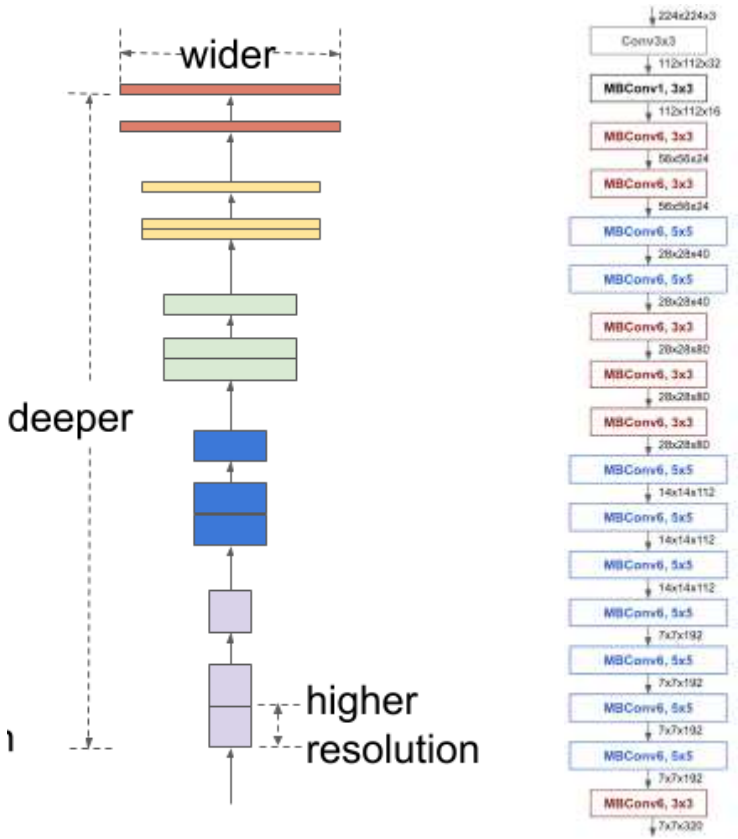
Yinda Zhang, Thomas Funkhouser. Deep Depth Completion of a Single RGB-D Image. CVPR 2018

Предыдущий SOTA метод



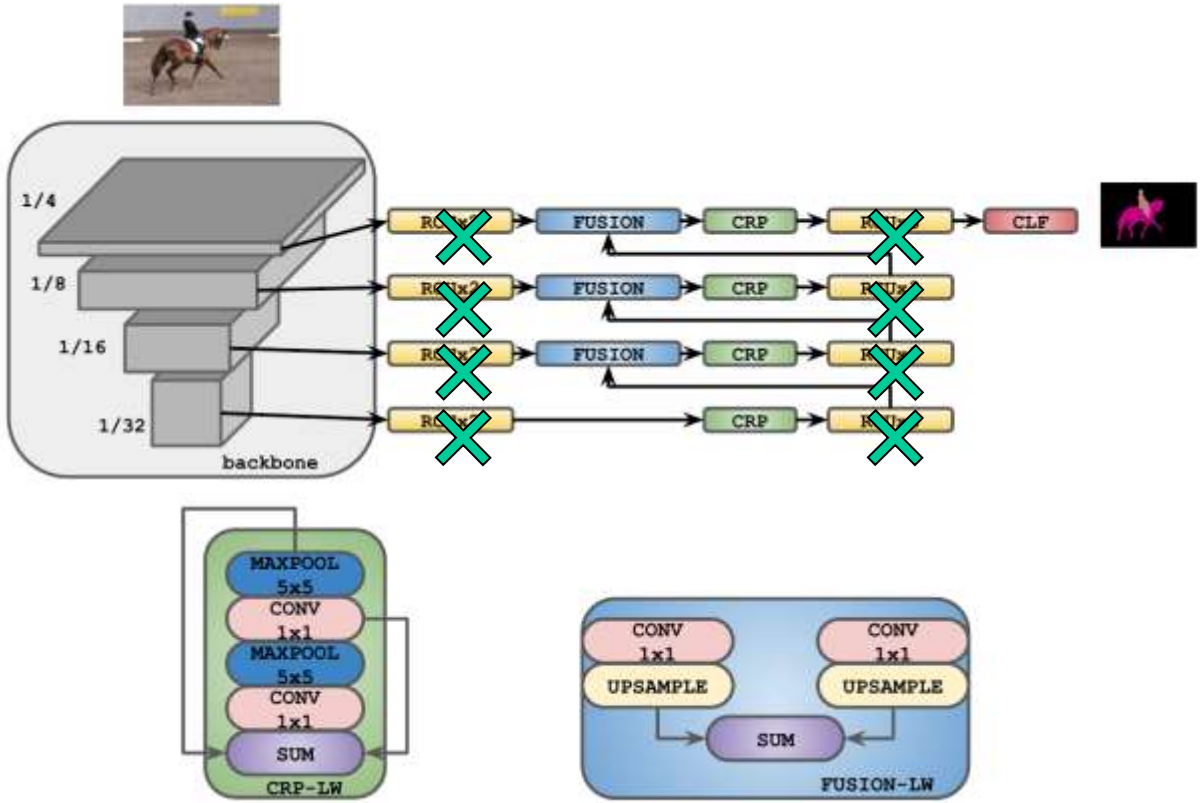
Huang et. al., Indoor depth completion with boundary consistency and self-attention. 2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW), Oct 2019.

Удивительно неплохая базовая модель



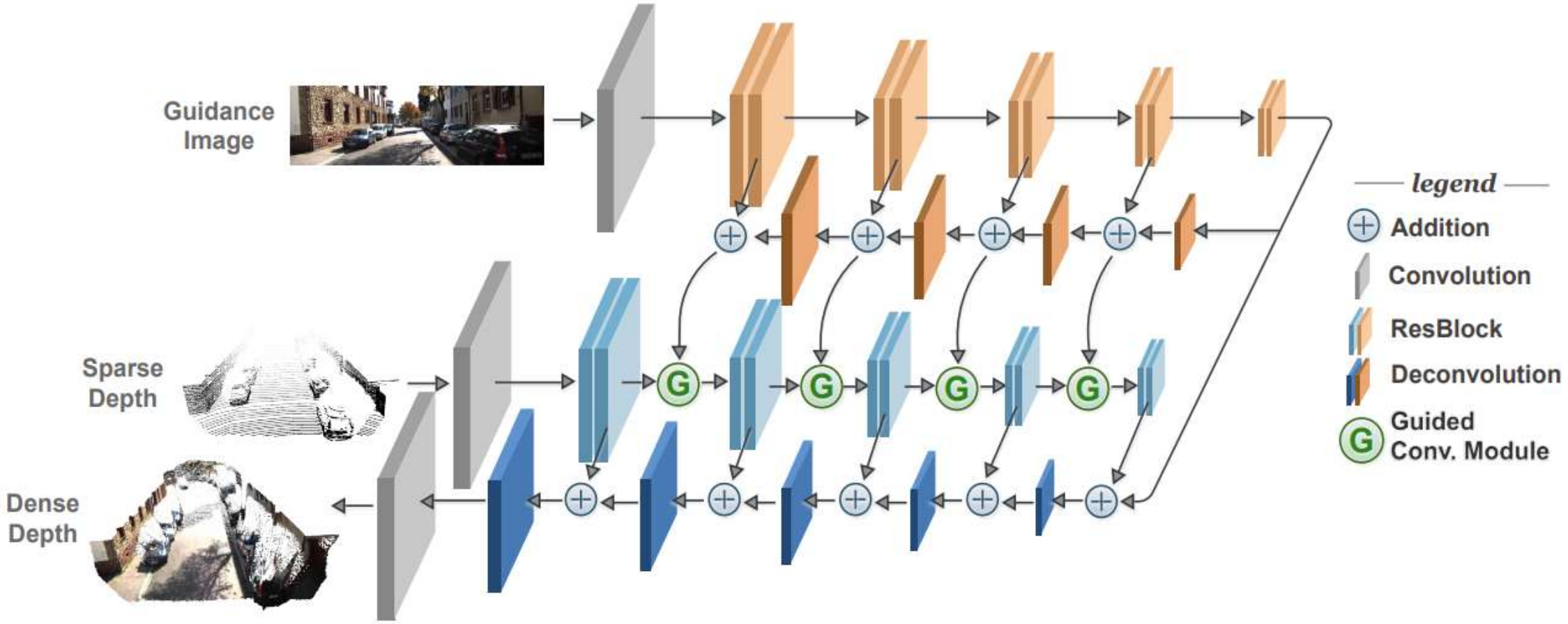
M.Tan, Q.Le. EfficientNet:Rethinking model scaling for convolutional neural networks. ICML2019

Decoder



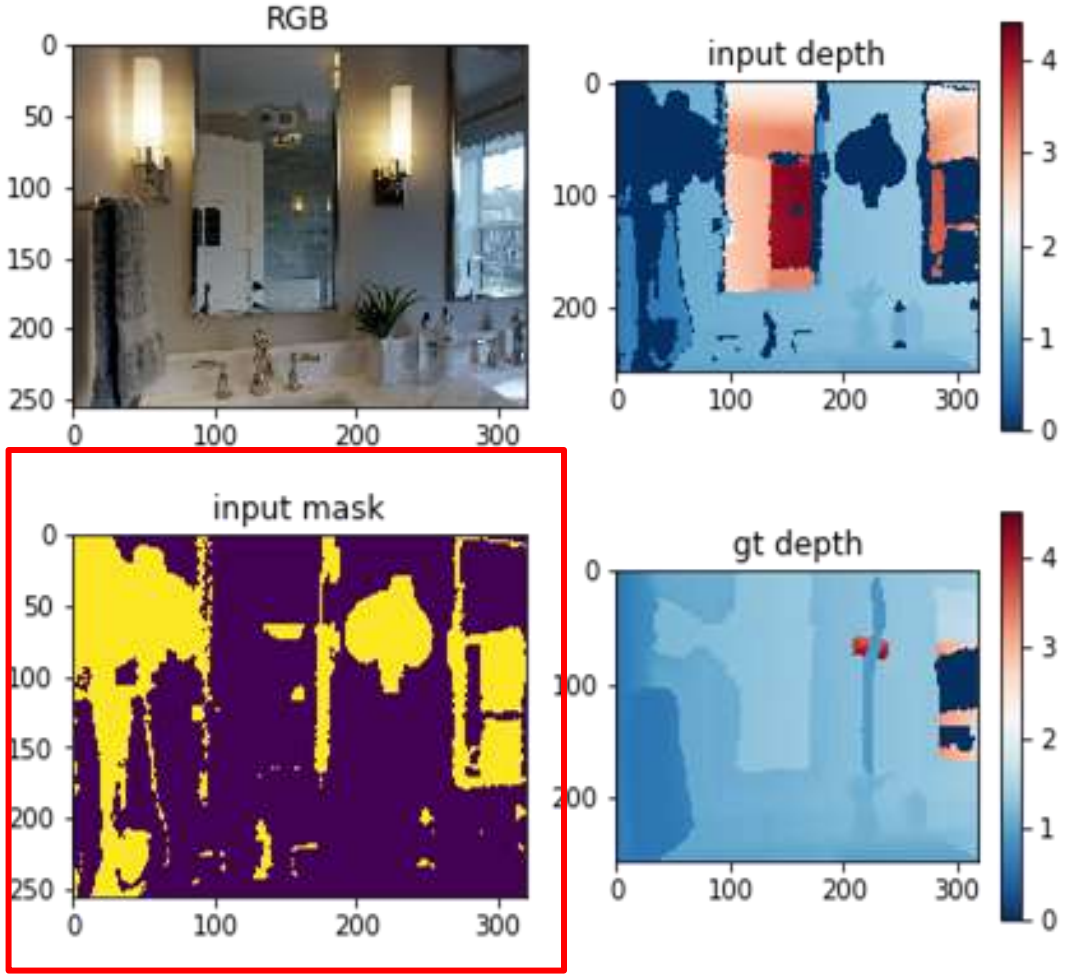
V.Nekrasov, C.Shen, I.Reid. Light-Weight RefineNet for Real-Time Semantic Segmentation. BMVC2018

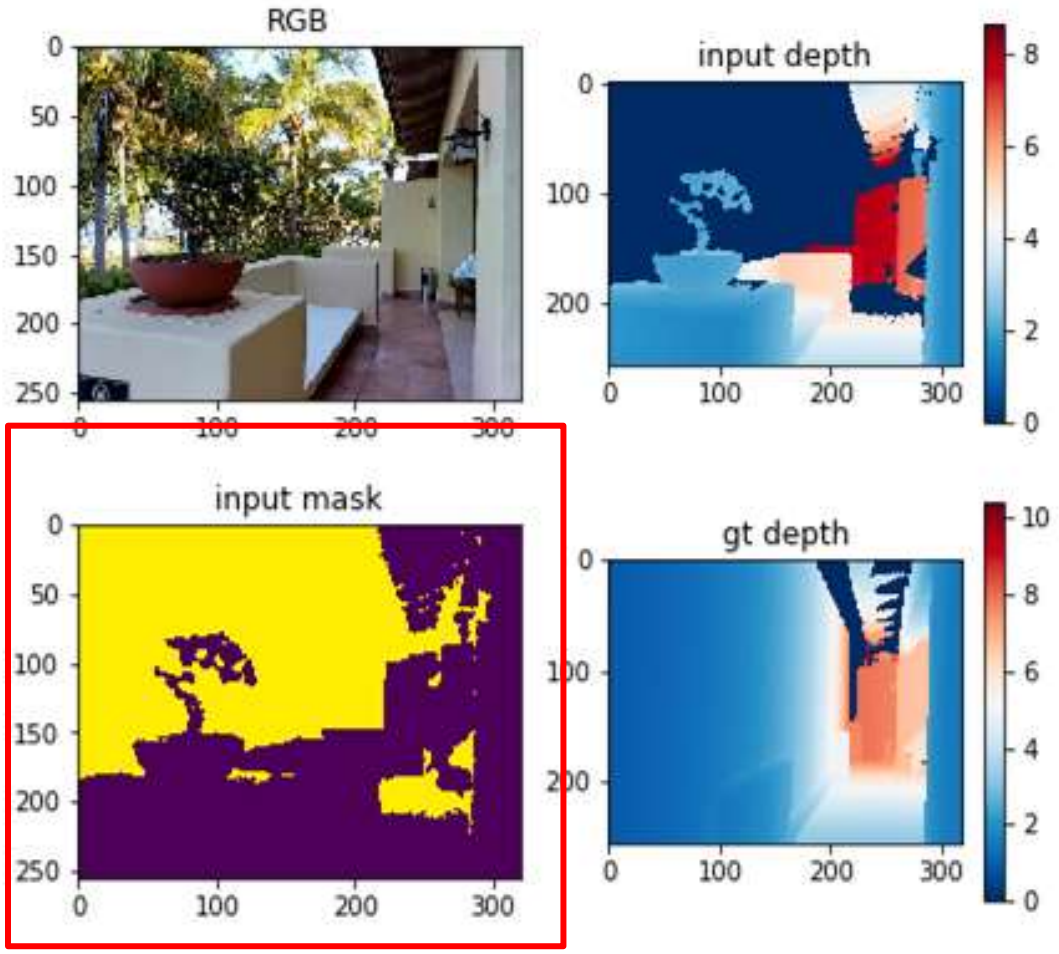
Мотивирующая работа



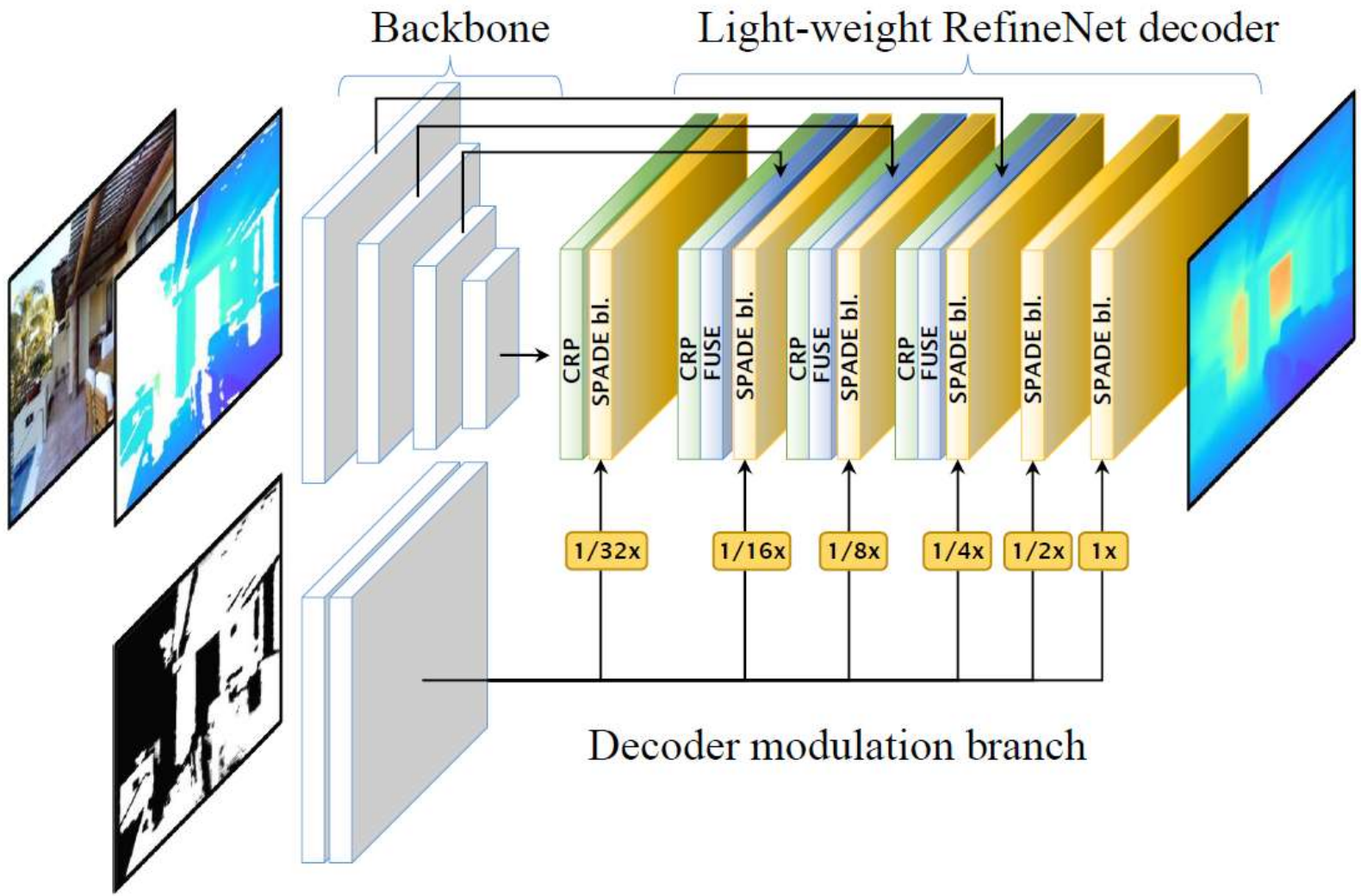
J. Tang. et. al. Learning Guided Convolutional Network for Depth Completion. ArXiv 2019

Маска отсутствующих значений

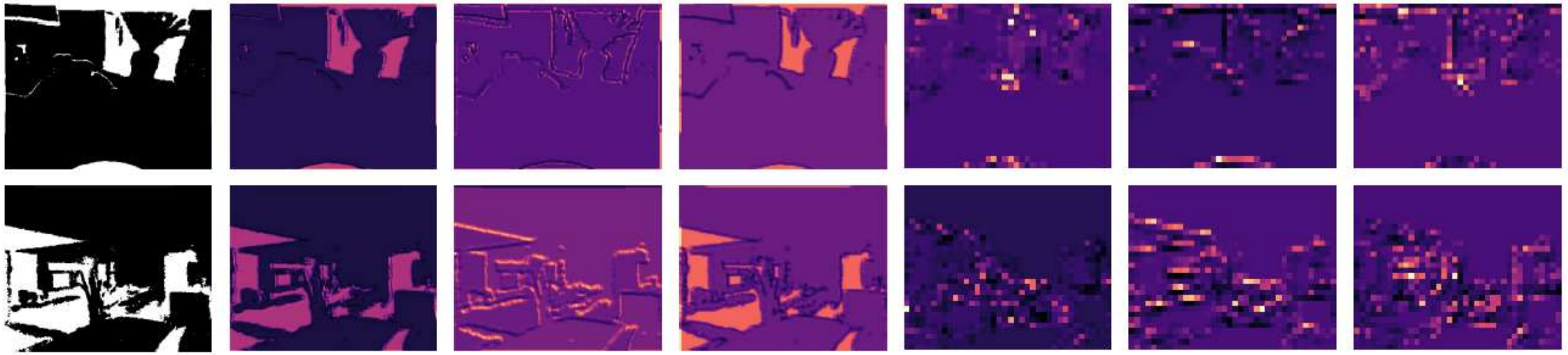




Идея метода


Backbone
Light-weight RefineNet decoder
CRP
SPADE bl.
CRP
FUSE
SPADE bl.
CRP
FUSE
SPADE bl.
CRP
FUSE
SPADE bl.
SPADE bl.
SPADE bl.
1/32x
1/16x
1/8x
1/4x
1/2x
1x
Decoder modulation branch

Маска наличия & признаки



(a) (b) (c) (d) (e) (f) (g)

Figure 2: **Mask features**. (a) input mask , (b)-(d) high resolution features ($\frac{H}{2} \times \frac{W}{2}$), (e)-(f) mid resolution features ($\frac{H}{8} \times \frac{W}{8}$). Large values are highlighted. Features of filled regions tend to be small and constant while for unfilled areas features might take values in a wide range. One can also notice large activation values marking the boundaries of objects that might also be helpful for depth inpainting.

Функция потерь

- Вычисление потерь в логарифмической шкале, следуя практике оценки глубины по RGB изображениям

$$\mathcal{L}(d_i, d_i^*) = \frac{1}{|\mathcal{O}|} \sum_{i \in \mathcal{O}} \left| \log d_i - d_i^* \right|$$

  - d_i - gt depth value
  - d_i^* - predicted depth in log scale

- Напрямую оптимизируем δ-метрики
- Ведёт к повышению точности

Сравнение на Matterport3D

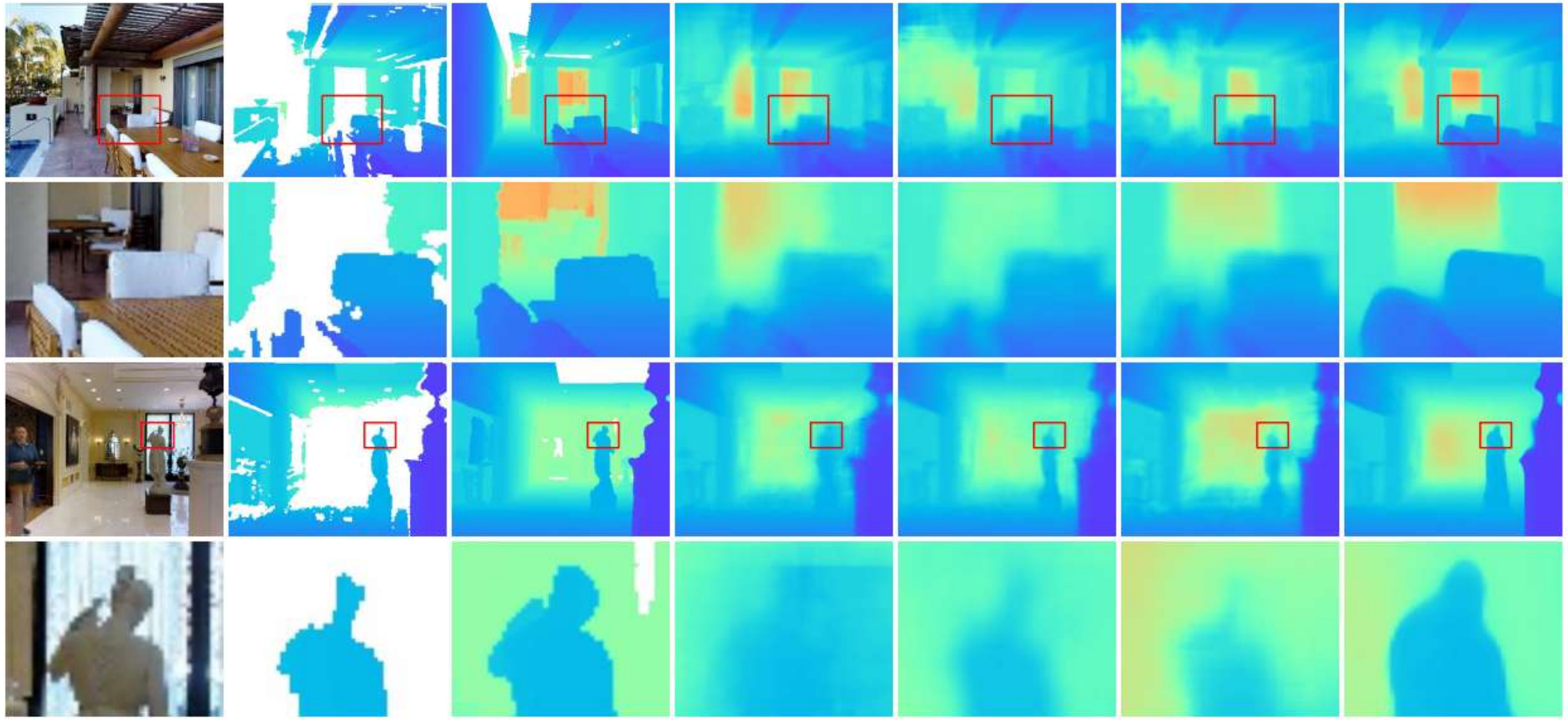


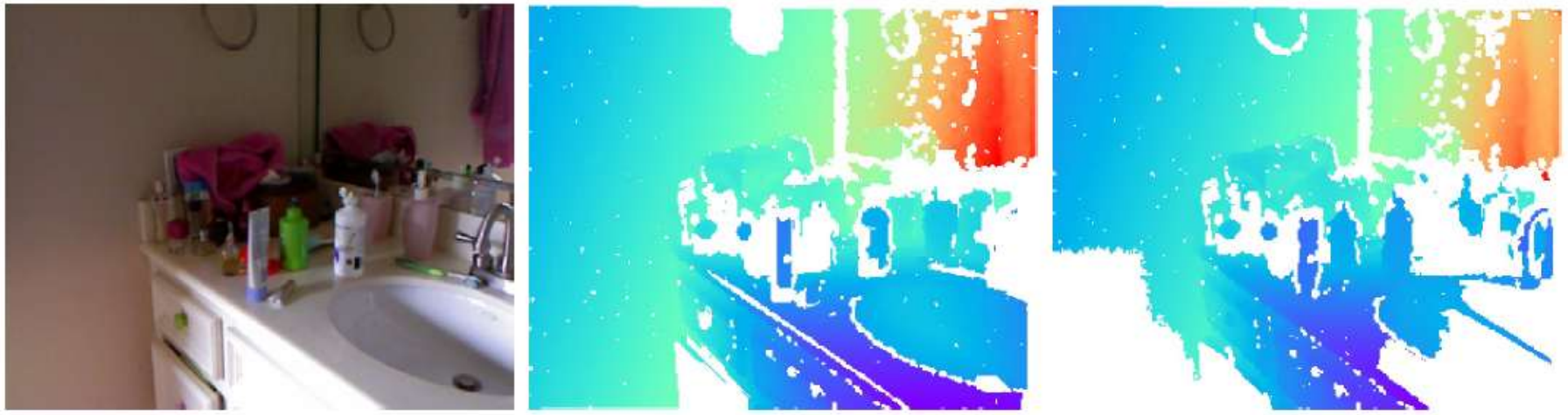
Figure 5: Qualitative comparison with Gansbeke *et al.* [42], Li *et al.* [20], Huang *et al.* [15] on Matterport3D test set. We train [42] and [20] on Matterport3D using the official code of the corresponding approaches, and results for [15] are based on the official pretrained model. Rows 2 and 4 represent zoomed-in fragments from rows 1 and 3, respectively. All images are created using color maps with the same value limits. Our model generates the completed depth map with very sharp boundaries.

Численные оценки

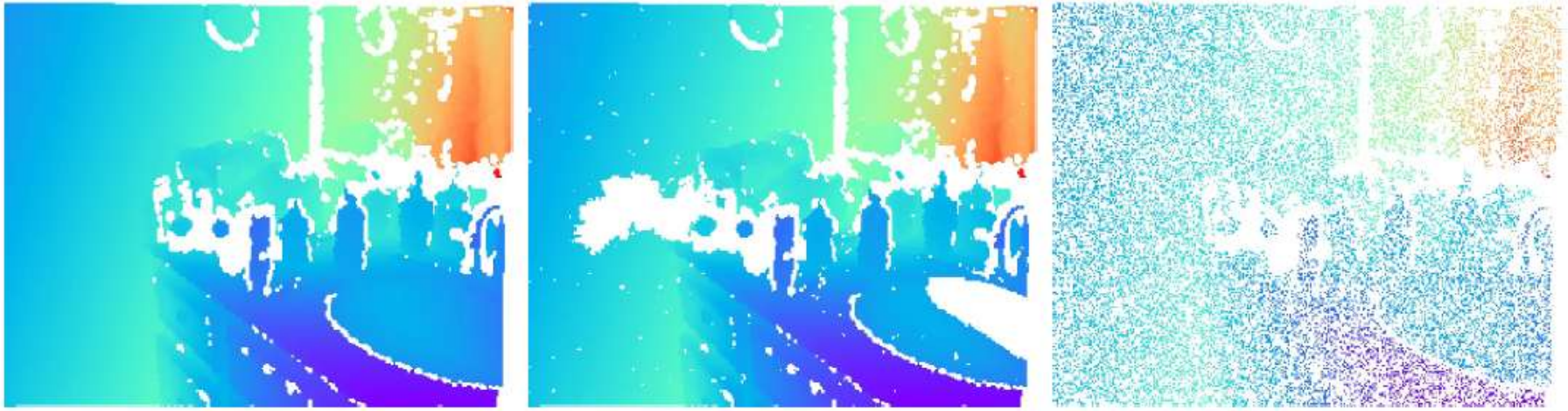
	RMSE ↓	MAE ↓	$\delta_{1.05}$ ↑	$\delta_{1.10}$ ↑	$\delta_{1.25}$ ↑	$\delta_{1.25^2}$ ↑	$\delta_{1.25^3}$ ↑	SSIM ↑
Huang *et al.* [15]	1.092	0.342	0.661	0.750	0.850	0.911	0.936	0.799
Zhang *et al.* [48]	1.316	0.461	0.657	0.708	0.781	0.851	0.888	0.762
Gansbeke *et al.* [42]	1.161	0.395	0.542	0.657	0.799	0.887	0.927	0.700
Li *et al.* [20]	1.054	0.397	0.508	0.631	0.775	0.874	0.920	0.700
Gansbeke *et al.* [42] (ours)	1.264	0.484	0.675	0.741	0.826	0.888	0.920	0.780
Li *et al.* [20] (ours)	1.134	0.426	0.649	0.729	0.0.834	0.899	0.928	0.774
DM-LRN (ours)	**0.961**	**0.285**	0.726	**0.813**	**0.890**	**0.933**	0.949	**0.844**
LRN (ours)	1.028	0.299	0.719	0.805	**0.890**	0.932	**0.950**	0.843
LRN + mask (ours)	1.054	0.298	**0.737**	**0.815**	0.889	**0.933**	**0.950**	**0.844**

Table 1: *Matterport3D TEST*. We use the results for Huang *et al.* [15] and Zhang *et al.* [48] reported in [15]. Gansbeke *et al.* [42] and Li *et al.* [20] are trained on Matterport3D using their official implementations. Models labeled as "ours" are trained using our proposed pipeline. The two bottom rows represent models without the decoder modulation branch, with and without the mask on the input. RMSE and MAE are measured in meters.

Сэмплирование данных для обучения



(a) RGB (c) Graph-based [12] (e) Slic [2]



(b) Initial real sensor (d) Quickshift [43] (f) Uniform [27]

Визуальное сравнение на NYUv2

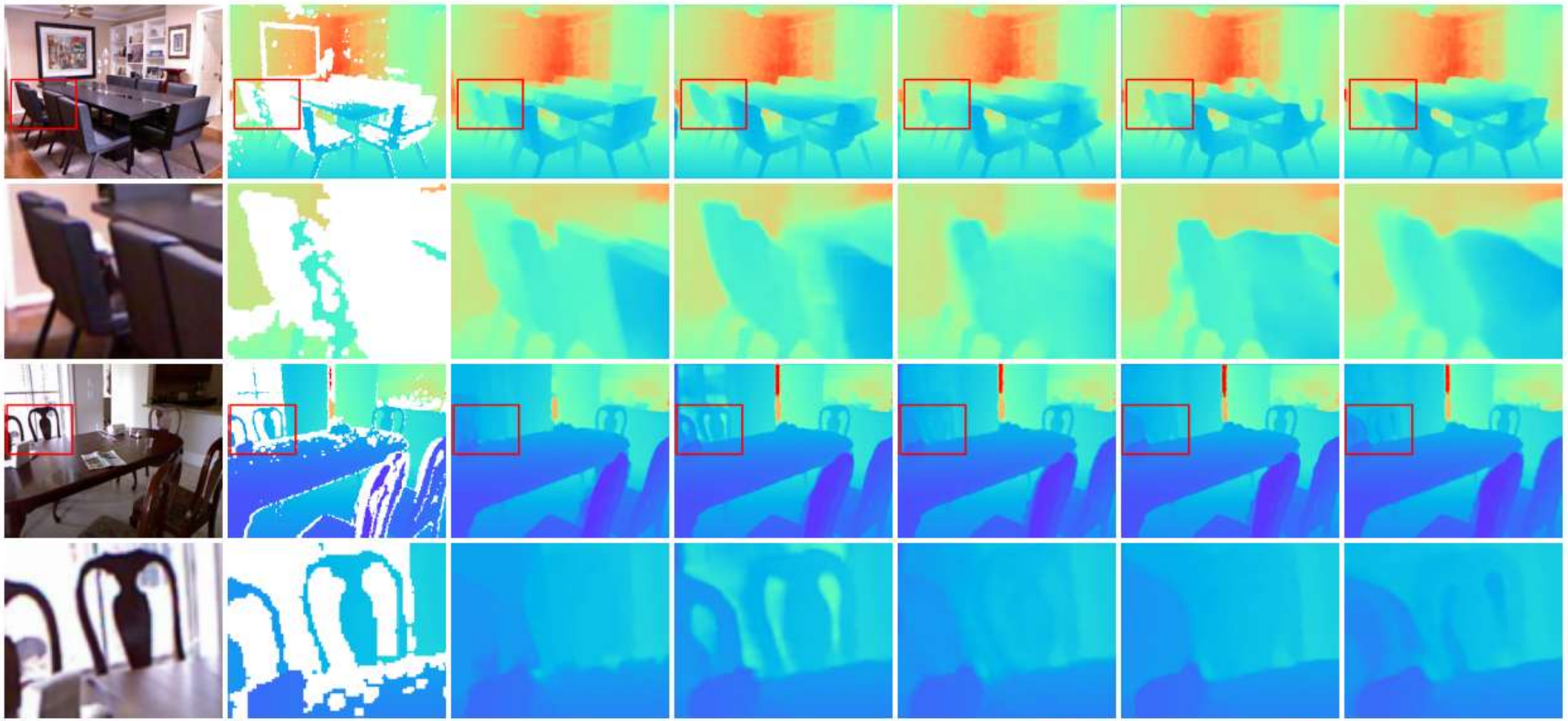


Figure 7: Qualitative comparison with Gansbeke *et al.* [42], Li *et al.* [20], Huang *et al.* [15] on NYUv2 [28] test set. All models are trained using our semi-dense sampling strategy. The third and fourth raws present a hard example.

Численные оценки

	semi-dense					sparse (500 points)				
	RMSE ↓	rel ↓	$\delta_{1.25}$ ↑	$\delta_{1.25^2}$ ↑	$\delta_{1.25^3}$ ↑	RMSE ↓	rel ↓	$\delta_{1.25}$ ↑	$\delta_{1.25^2}$ ↑	$\delta_{1.25^3}$ ↑
Huang *et al.* [15]	0.271	0.016	98.1	99.1	99.4	–	–	–	–	–
Gansbeke *et al.* [42]	0.260	0.017	97.9	99.3	99.7	0.344	0.042	96.1	98.5	99.5
Li *et al.* [20]	**0.190**	0.018	**98.8**	**99.7**	**99.9**	0.272	**0.034**	97.3	99.2	99.7
DM-LRN (ours)	0.205	**0.014**	**98.8**	99.6	**99.9**	**0.263**	0.035	**97.5**	**99.3**	**99.8**

Table 3: *NYUv2 TEST*. Quantitative comparison of training setups for different models. Semi-dense sampling preserves more accurate information that leads to better results. Although our approach is not intended to be applied to sparse depth sensors, it demonstrates strong results in the sparse training setting in indoor environments. We do not use any densification scheme for target depth reconstruction. Pseudo-sensor data is directly sampled from real sensor data.

Резюме

- Сенсоры глубины позволяют оценить глубину по 1 ракурсу, без стерео. Но имеют ограничения
- С помощью DL мы можем оценивать глубину без сенсоров и без стерео, но пока не так хорошо
- Сенсоры глубины позволяют получить много обучающих данных, но неидеальных
- Стерео данные разнообразны, но для них нет эталонных решений
- Все общие наработки моделей плотной разметки изображений перетекают в задачи оценки глубины