

Введение в разработку приложений дополненной и виртуальной реальностей

Альхеев Геннадий Евгеньевич
Главный инженер, Интер РАО

Организационная часть

Контекст

- 1) Вас резко стало больше
- 2) Я бы не хотел менять требования по сдаче

Изменение в процессе

1) Каждая команда готовит

- a) Доклад
- b) Практику (Воркшоп)
- c) Статью по докладу другой команды

2) Оценивание

- a) Доклад – 1 балл (всей команде)
- b) Практика – 1 балл (всей команде)
- c) Статья – 1 балл (всей команде)
- d) Отчеты 1 и 2 – 2 балла
- e) Капитанство – 5 баллов, если команда сдает доклад, практику и статью, иначе на общих условиях

Новое сопоставление команд и работ

Команда	Вид работы	Тема	Название задачи
Команда 1	Доклад	Тема 1	SIFT vs ORB: сравнительный анализ для реальных проектов
	Практика	Тема 2	Сравнение карт глубины: MDS vs традиционные методы
	Статья	Тема 3	Статья по докладу "PCL (Point Cloud Library): программируемая обработка облаков точек"
Команда 2	Доклад	Тема 1	Bundle Adjustment: математические основы и методы ускорения
	Практика	Тема 3	Создание и калибровка маркерной AR-сцены
	Статья	Тема 2	Статья по докладу "NeRF: от research прорыва к практическому применению"
Команда 3	Доклад	Тема 4	Проблема дрейфа в SLAM и методы борьбы с ней
	Практика	Тема 1	Создание 3D-модели архитектурного объекта по 40+ фото
	Статья	Тема 5	Статья по докладу "WebXR Device API: работа с устройствами VR/AR в браузере"
Команда 4	Доклад	Тема 2	MDS: универсальная оценка глубины на одном изображении
	Практика	Тема 3	Сегментация облака точек на объекты с помощью PCL
	Статья	Тема 1	Статья по докладу "Bundle Adjustment: математические основы и методы ускорения"
Команда 5	Доклад	Тема 2	NeRF: от research прорыва к практическому применению
	Практика	Тема 4	Тестирование устойчивости трекинга в разных условиях освещения
	Статья	Тема 4	Статья по докладу "Визуально-инерциальный SLAM: ORB-SLAM3 vs VINS-Mono"
Команда 6	Доклад	Тема 2	Depth from Video: методы оценки глубины из видео
	Практика	Тема 5	Оптимизация 3D-модели для веб-загрузки
	Статья	Тема 3	Статья по докладу "Сравнение алгоритмов ретопологии"
Команда 7	Доклад	Тема 3	CloudCompare: продвинутые техники фильтрации облаков точек
	Практика	Тема 2	Создание 3D-сцены с помощью NeRF из набора изображений
	Статья	Тема 2	Статья по докладу "Depth from Video: методы оценки глубины из видео"
Команда 8	Доклад	Тема 3	PCL (Point Cloud Library): программируемая обработка облаков точек
	Практика	Тема 4	Настройка и запуск ORB-SLAM3 на собственных видео
	Статья	Тема 1	Статья по докладу "SIFT vs ORB: сравнительный анализ для реальных проектов"
Команда 9	Доклад	Тема 3	Сравнение алгоритмов ретопологии
	Практика	Тема 5	Создание интерактивного WebAR-прототипа с маркерным трекингом
	Статья	Тема 4	Статья по докладу "ARCore/ARKit: что скрыто в популярных AR-SDK"
Команда 10	Доклад	Тема 4	Визуально-инерциальный SLAM: ORB-SLAM3 vs VINS-Mono
	Практика	Тема 3	Автоматическое текстурирование модели с помощью UV-развертки
	Статья	Тема 5	Статья по докладу "Безмаркерный трекинг в WebAR на основе нейросетей"
Команда 11	Доклад	Тема 4	ARCore/ARKit: что скрыто в популярных AR-SDK
	Практика	Тема 5	Создание WebAR-сцены с использованием A-Frame и AR.js
	Статья	Тема 2	Статья по докладу "MDS: универсальная оценка глубины на одном изображении"
Команда 12	Доклад	Тема 5	WebXR Device API: работа с устройствами VR/AR в браузере
	Практика	Тема 2	Оценка глубины на видео с помощью Depth from Video моделей
	Статья	Тема 3	Статья по докладу "CloudCompare: продвинутые техники фильтрации облаков точек"
Команда 13	Доклад	Тема 5	Безмаркерный трекинг в WebAR на основе нейросетей
	Практика	Тема 4	Сравнение производительности SLAM-систем в динамической среде
	Статья	Тема 4	Статья по докладу "Проблема дрейфа в SLAM и методы борьбы с ней"

Гугл табличка -> “Темы”

Вопросы

План лекции

- 1) Введение и Фундамент
- 2) Классическая эра MVS
- 3) Революция глубокого обучения
- 4) Новая парадигма — Нейронные поля
- 5) Итоги и Заключение

Часть 1: Введение

Зачем нам третье измерение?

Цель: Воссоздание цифровых 3D-копий реального мира.

Применение: Навигация, кинематограф, игры, VR/AR, 3D-печать, научная визуализация.

Основная проблема: Как получить информацию о глубине из 2D-изображений?

Решение: Комбинированный пайплайн Structure from Motion (SfM) и Multi-View Stereo (MVS).

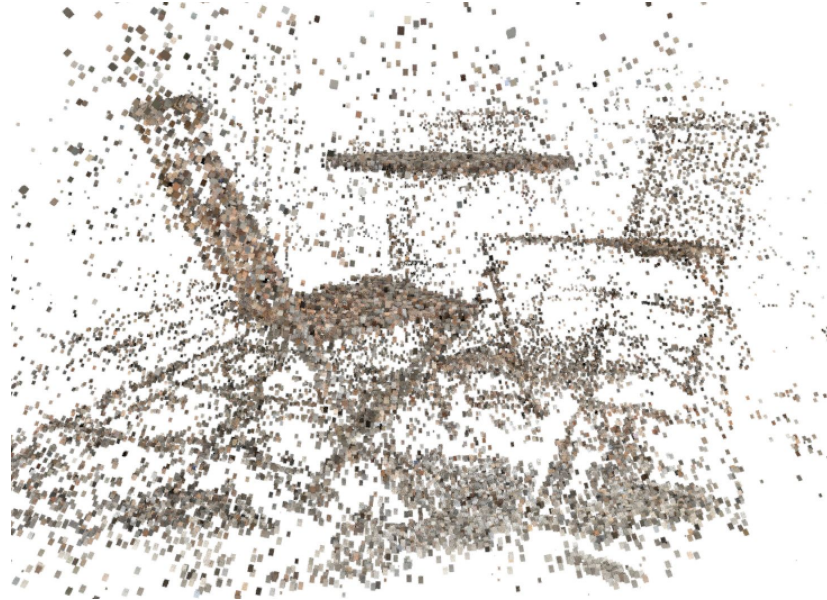


Шаг 1: Восстановление структуры по движению (SfM)

Задача: Определить положение камер в пространстве и восстановить общую геометрию сцены.

"Договор" камер: Камеры "договариваются" о 3D-сцене, находя общие особенности (features) на разных снимках.

Результат: Разреженное облако точек и точные позы камер.



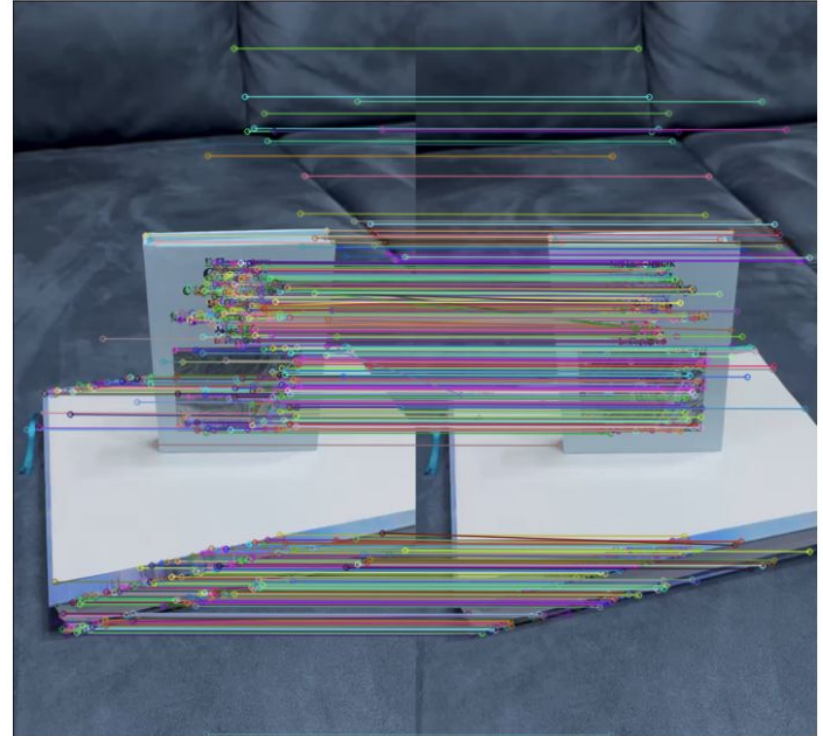
Что такое features и почему они так важны?

Особенность — уникальная, легко идентифицируемая точка или область на изображении (угол, край, пятно).

Роль: Служат опорными точками для сопоставления изображений.

Процесс:

- 1) Извлечение: Поиск особенностей на каждом изображении (SIFT, ORB).
- 2) Сопоставление: Поиск одинаковых особенностей на разных снимках.
- 3) Триангуляция: Восстановление 3D-координат этих точек.



Триангуляция

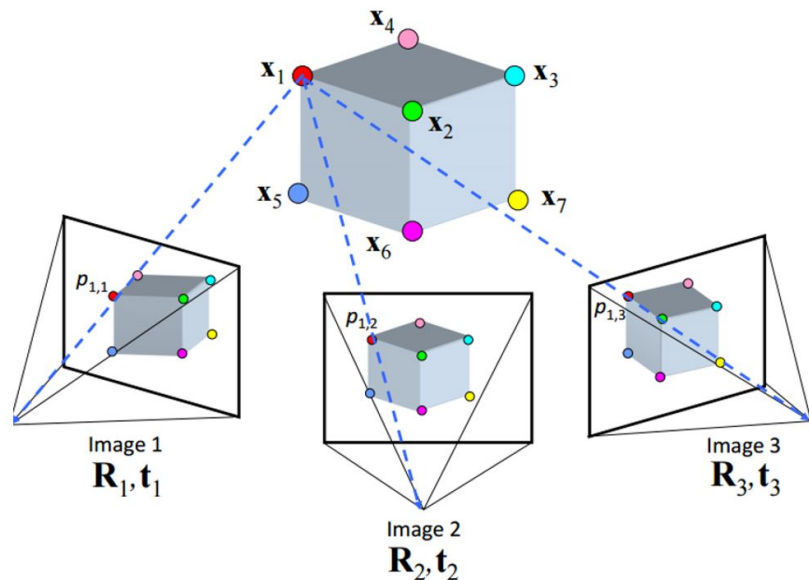
Цель: По двум (или более) 2D-измерениям одной и той же 3D-точки на разных изображениях восстановить её 3D-координаты.

Как работает:

- 1) Есть две камеры с известными матрицами проекции P_1 и P_2 .
- 2) Известны 2D-соответствия: точка x_1 на первом изображении и x_2 на втором.
- 3) 3D-точка X лежит на пересечении двух лучей (back-projection).

В SfM:

- 1) После сопоставления особенностей (features) между изображениями.
- 2) Для каждой пары (набора) соответствующих точек выполняется триангуляция, чтобы получить начальные оценки 3D-точек.
- 3) Эти точки составляют начальное разреженное облако.



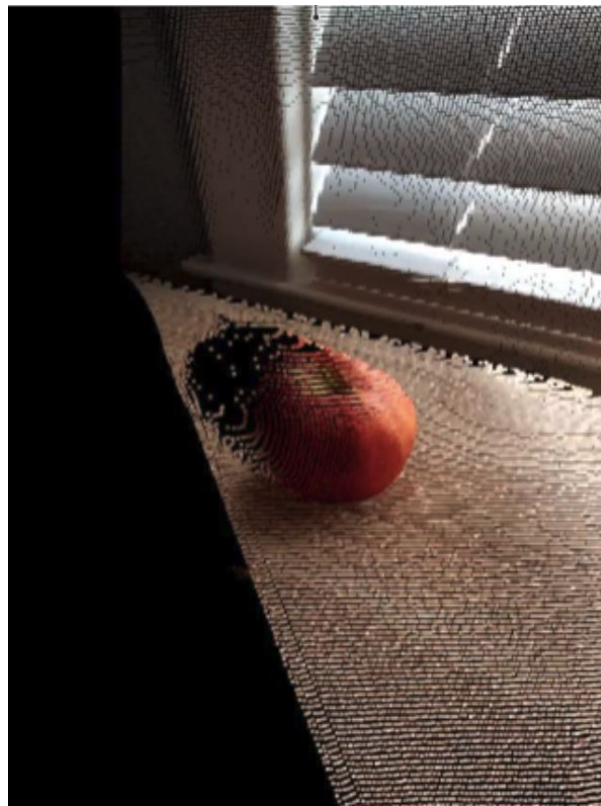
Source: http://cs.nyu.edu/~fergus/teaching/vision/11_12_multiview.pdf

Шаг 3(?): Multi-View Stereo (MVS)

Задача: Используя результаты SfM (позы камер), создать плотное облако точек.

Отличие от SfM: SfM создает "скелет" сцены (разреженные точки), MVS – наполняет его деталями.

Плотное облако точек: Набор 3D-точек, который детально описывает поверхность объекта.



Шаг 2(!): Bundle Adjustment

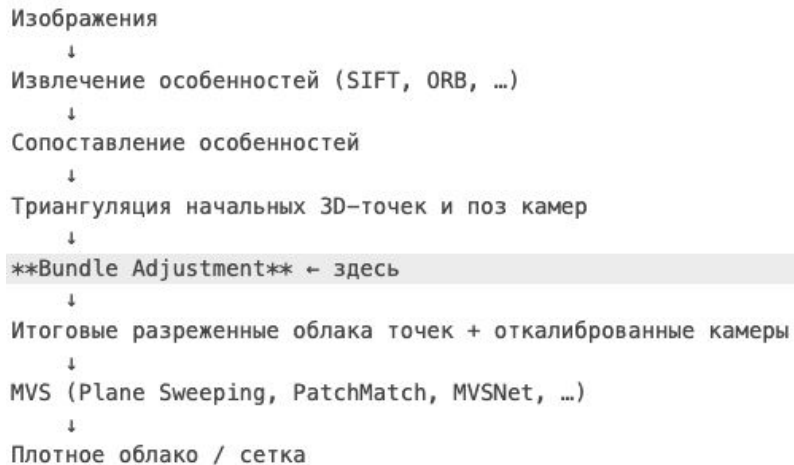
Цель: Совместно оптимизировать параметры камер (позы, внутренняя калибровка, дисторсия) и 3D-координаты точек сцены, чтобы минимизировать ошибку перепроецирования (reprojection error).

Вход:

- 1) Набор 2D-измерений (координаты особенностей на изображениях).
- 2) Начальные оценки поз камер и 3D-точек (от триангуляции).

Выход:

- 3) Оптимальные (в смысле МНК) позы камер и 3D-точки.



что если пропустить ВА?

ВА **уменьшает ошибки** в положениях и ориентациях камер и даёт **оптимальные** 3D-координаты опорных точек.

SfM без ВА:

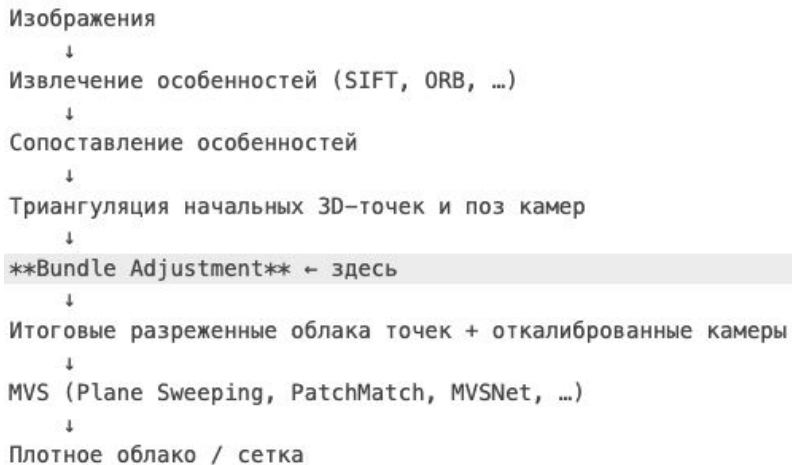
- 1) Позы камер имеют дрейф, 3D-точки "размазаны".

MVS на таких данных:

- 2) Plane Sweeping: warped-изображения будут размыты, cost volume — шумный.
- 3) PatchMatch: гипотезы будут сходиться медленно или к неверным значениям.

Результат:

- 4) Плотное облако точек — с артефактами, "двойными" поверхностями, пропущенными деталями.



Часть 2: Классические подходы

Multi-View Stereo (MVS)

MVS — это общий класс методов, которые по множеству изображений с известными параметрами камер (позами и внутренними калибровками) восстанавливают плотную геометрию сцены (карты глубины, облака точек, поверхности).

- 1) SfM даёт грубые оценки глубин для ключевых точек.
- 2) MVS-алгоритмы используют эти оценки, чтобы сузить интервал поиска глубин для каждого пикселя (минимальная и максимальная глубина во frustum-е камеры).

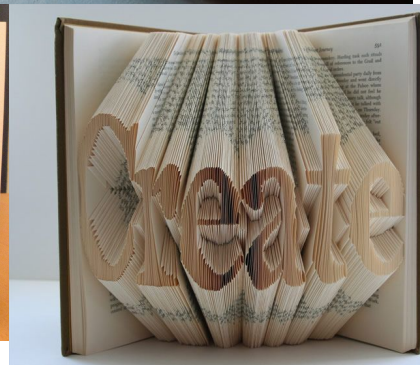
Plane Sweeping: Сканирование пространства плоскостями

Идея: "Пройти" виртуальными плоскостями через всю сцену на разной глубине.

Процесс:

- 1) Для каждой плоскости на определенной глубине "перенести" (warp) пиксели со всех видов на эту плоскость.
- 2) Оценить, насколько хорошо перенесенные изображения совпадают.
- 3) Там, где совпадение наилучшее, поверхность сцены и находится.

Преимущества: Концептуально прост, отлично параллелится на GPU.



Plane Sweeping

Проблема: В 1990-х годах классические стереоалгоритмы (block matching, graph cuts) хорошо работали для пары изображений, но плохо масштабировались на множество видов.

Идея "space sweep":

- Роберт Коллинз (RT Collins) в 1996 году предложил A Space-Sweep Approach to True Multi-Image Matching.
- Вместо перебора параллаксов для пары изображений он ввёл виртуальные плоскости, которые "прорежают" 3D-объём камеры (frustum) на разных глубинах.
- Для каждой плоскости изображения warp-ятся в общую систему координат, и считается фотометрическая ошибка между видами.

Почему это было важно?

- Параллелизм: Алгоритм легко ложился на GPU (появлялись первые эксперименты с аппаратным ускорением).
- Расширяемость: Метод работал с произвольным числом видов, не ограничиваясь стереопарой.
- Надёжность: Plane Sweeping устойчив к шуму калибровки, так как учитывает все виды сразу.

Дальнейшее развитие

- 2007: Дэвид Галлап (David Gallup) с соавторами представили Real-Time Plane-Sweeping Stereo на GPU.
- 2019: DPSNet (Deep Plane Sweep Network) показала, что идею Plane Sweeping можно встроить в нейросеть, обучив энд-ту-энд 3D-реконструкцию.

Plane Sweeping

Вход:

- 1) Набор изображений с известными позами камер.
- 2) Диапазон глубин $[d_{\min}, d_{\max}]$ (из SfM).

Определение семейства плоскостей:

- 1) Для опорной камеры выбрать N плоскостей Π_i , параллельных изображению, на глубинах d_i (равномерно или по облаку).

Warп-инг изображений:

- 1) Для каждой плоскости Π_i :
 - а) Для каждого вида j применить гомографию H_{ji} , которая переводит пиксели вида j в координаты опорного изображения, как будто они лежат на Π_i .

Построение cost-volume:

- 1) Для каждого пикселя p опорного изображения и для каждой плоскости Π_i :
 - а) Собрать warped-патчи со всех видов.
 - б) Посчитать фотометрическую стоимость $C(p, i)$ (например, SSD).

Выбор глубины:

- 1) Для каждого пикселя p выбрать глубину d_i , соответствующую минимальной стоимости $C(p, i)$.

Выход: Карта глубин для опорного изображения.

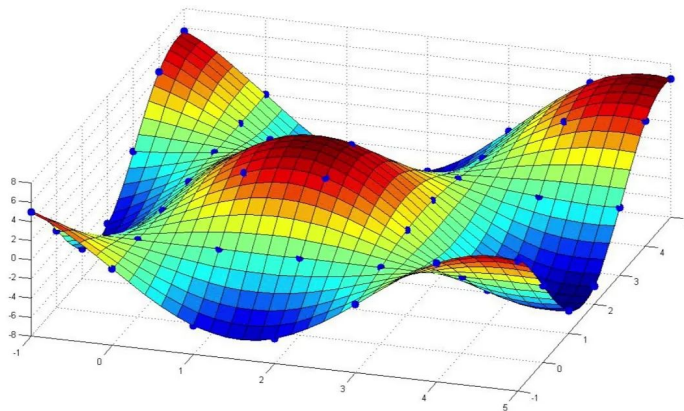
PatchMatch Stereo: Искусство распространения догадок

Идея: Для каждого пикселя поддерживать гипотезу о его глубине и ориентации поверхности.

Процесс:

- 1) Инициализация: Задать случайные гипотезы для всех пикселей.
- 2) Итерации: Последовательно улучшать гипотезы, "заимствуя" лучшие варианты у соседей (пространственное распространение).
- 3) Фотометрическая согласованность: Выбирать ту гипотезу, при которой фрагменты изображений (патчи) с разных видов выглядят наиболее похоже.

Преимущества: Очень быстрый, эффективный, хорошо параллелится.



PatchMatch Stereo

Появление (2009)

- Проблема: Для плотного восстановления стерео нужно было решать оптимизационную задачу для каждого пикселя, что было очень медленно.
- Авторы: Коннелли Барнс (Connelly Barnes) с соавторами (Princeton).
- Оригинальная задача: Быстрый поиск ближайших соседей для image-патчей (для задач редактирования изображений).

Ключевая идея:

- Вместо exhaustive search (полного перебора) использовать случайные итерации + пространственное распространение.
- На каждой итерации для каждого патча:
 - Propagation: "Заимствовать" гипотезу у соседа.
 - Random search: Случайно пробовать новые гипотезы.
- Алгоритм сходится очень быстро (20-100× ускорение).

Перенос в MVS (PatchMatch Stereo)

- 2011: Брайан Кларк (Brian Curless) и Ястжембски (Yastzembski) адаптировали PatchMatch для стерео, введя плоскостные гипотезы (глубина + нормаль).
- 2014: Коллаборация ETH Zurich и Microsoft Research (Michael Goesele, Steven M. Seitz) интерпретировали PatchMatch MVS в COLMAP, сделав его стандартом де-факто.

Почему это стало прорывом?

- Скорость: PatchMatch MVS работает в разы быстрее Plane Sweeping при сопоставимом или лучшем качестве.
- Память: Не требует хранения огромного cost-volume.
- Качество: Учитывает наклонные поверхности (slanted support windows), что важно для тонких деталей.

PatchMatch Stereo

Вход:

- 1) Набор изображений с известными позами камер (из SfM).
- 2) Для опорного изображения — диапазон глубин $[d_{\min}, d_{\max}]$ (из SfM).

Инициализация:

- 1) Для каждого пикселя p на опорном изображении:
- 2) Случайная глубина $d \in [d_{\min}, d_{\max}]$.
- 3) Случайная нормаль n (задаёт наклон локальной плоскости).

Итеративное распространение:

- 1) На каждой итерации для каждого пикселя:
 - a) Пространственное распространение: проверить гипотезы соседних пикселей (слева, сверху, справа, снизу).
 - b) Видовое распространение: проверить гипотезы из других видов (если доступно).
 - c) Случайный поиск: с некоторой вероятностью сгенерировать новую случайную гипотезу.

Оценка стоимости:

- 1) Для текущей гипотезы (d, n) построить локальный патч в 3D, спроецировать его на соседние виды.
- 2) Посчитать сумму квадратов разниц цветов (SSD) или нормализованную взаимную корреляцию (NCC) между патчами.

Выход:

- 1) Карта глубин для опорного изображения.

Часть 3: Развитие нейросетей

Эпоха глубокого обучения: MVSNet

Революция: Первый энд-ту-энд (end-to-end) нейросетевой подход для MVS.

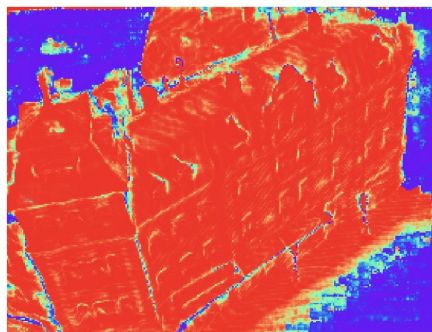
Архитектура:

- 1) Извлечение признаков: Сверточная сеть извлекает признаки из всех изображений.
- 2) Построение "объёма стоимости": Создание 3D-тензора, где для каждого пикселя и каждой гипотетической глубины хранится "стоимость" (ошибка соответствия).
- 3) Регуляризация: 3D-свертки обрабатывают этот объём для получения карты глубины.

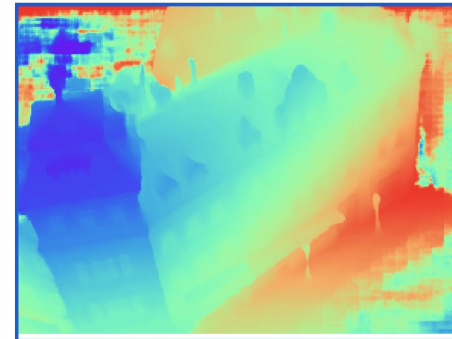
Результат: Значительное превосходство над классическими методами в точности и скорости.



reference image



probability map



depth map

Эпоха глубокого обучения: MVSNet



point cloud result



ground truth point cloud

R-MVSNet: Преодоление барьера памяти

Проблема MVSNet: 3D "объём стоимости" требует огромного количества памяти.

Решение R-MVSNet: Обработать объём последовательно, плоскость за плоскостью, используя рекуррентные нейронные сети (RNN).

Как работает: Сеть "просматривает" все гипотезы глубины для одного пикселя, запоминая информацию на каждом шаге, и только потом принимает решение.

Преимущество: Позволяет восстанавливать сцены с высоким разрешением.



R-MVSNet: Преодоление барьера памяти

Results Summary

Model	Accuracy ↓	Completeness ↓	F-Score ↑	Inference Time ↓
MVSNet	Baseline	Baseline	Baseline	Moderate
R-MVSNet	Improved	Improved	Higher	Significantly Faster

Часть 4: Концептуально новые нейросетевые подходы

Новая парадигма: Нейронные Поля Излучения (NeRF)

Идея: Представить сцену не как облако точек или сетку, а как непрерывную функцию, заданную небольшой нейронной сетью.

Как работает:

- 1) Вход сети: 3D-координаты точки + направление взгляда.
- 2) Выход сети: Плотность (насколько "твёрдая" точка) и цвет (излучение) в этой точке.
- 3) Обучение: Сеть учится рендерить известные виды, "пропуская" через себя лучи от камеры.

Результат: Невероятно фотореалистичный синтез новых видов, включая сложные эффекты освещения.



Instant-NGP: Нейронная графика в реальном времени

Проблема NeRF: Очень медленное обучение и рендеринг.

Решение Instant-NGP: Специальная многоразрешающая хэш-кодировка входных координат.

Суть: Вместо сложного позиционного кодирования используется быстрая хэш-таблица с обучаемыми признаками. Это позволяет использовать очень маленькую нейросеть.

Результат: Обучение за секунды, рендеринг в реальном времени.



Вопросы