

# Комбинирование ключевых точек и прямого подхода для стерео-SLAM

Елена Сергеевна Люлина

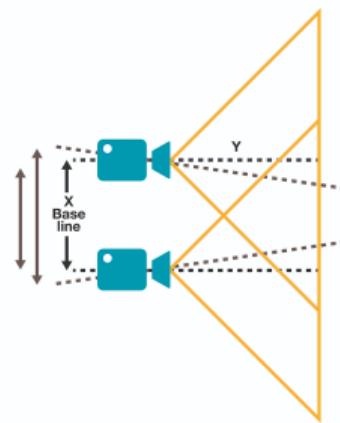
343 группа  
Лаборатория распознавания изображений  
СПбГУ

2019

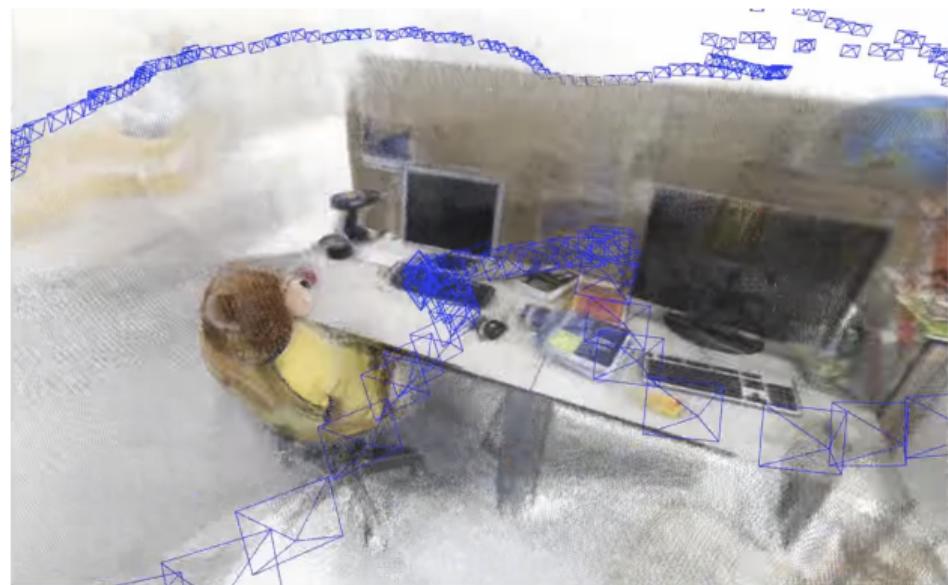
Руководитель: Пименов А. А.  
Консультант: Корчемкин Д. А.

# Стерео-SLAM

**SLAM** = **S**imultaneous **L**ocalization **A**nd **M**apping



Система из двух камер



Пример работы ORB-SLAM2

## Feature-based:

- получить новый кадр
- выделить особые (feature) точки:
  - углы
  - пятна
- с их помощью определить местоположение камеры и дополнить карту

## Direct:

- получить новый кадр
- оценить расположение нового кадра, используя оценки глубины значительного числа точек предыдущего кадра
- оптимизировать позу по совпадению интенсивностей точек
- обновить оценки глубины

**Но каждый из подходов имеет свои недостатки!**

# Общий смысл

## Feature-based:

### Достоинства:

- не требует инициализации
- лучшая точность
- определяет замыкание циклов

### Недостатки:

- требуются достаточно текстурированные поверхности
- строится неплотная карта окружающей обстановки

## Direct:

### Достоинства:

- строится плотная карта
- работает при однородных текстурах объектов

### Недостатки:

- оценки позы очередного кадра выполняются нелинейной оптимизацией и являются локально-оптимальными
- замыкание циклов “напрямую” требует вычислительных затрат

# Цели

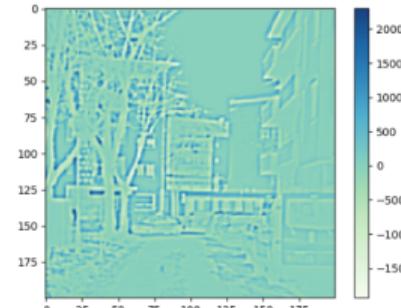
- 1 Ознакомиться с предметной областью, изучить основные подходы и особенности выбранных алгоритмов
- 2 Реализовать подзадачи, связанные с ключевыми точками:
  - детекция ключевых точек на кадрах
  - поиск соответствующих ключевых точек на разных кадрах
- 3 Оптимизировать время работы реализованных алгоритмов

# Детекция особых точек



Исходное изображение

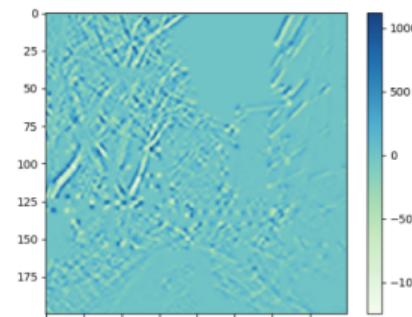
-1, -1, 0, 1, 1  
-1, -1, 0, 1, 1  
0, 0, 0, 0, 0  
1, 1, 0, -1, -1  
1, 1, 0, -1, -1



Локальные экстремумы

-1, -1, -1, -1, -1  
-1, 1, 1, 1, -1  
-1, 1, 8, 1, -1  
-1, 1, 1, 1, -1  
-1, -1, -1, -1, -1

Изображение после corner-свертки



Локальные экстремумы

Изображение после blob-свертки

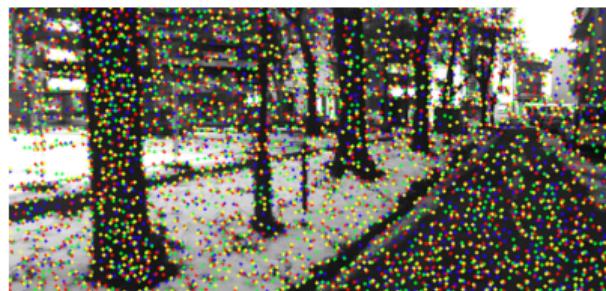
# Детекция особых точек



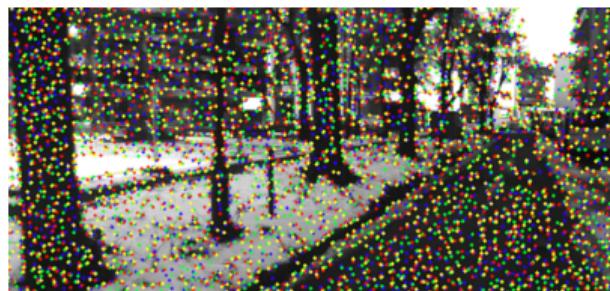
Левый кадр в момент времени  $k$



Правый кадр в момент времени  $k$



Левый кадр в момент времени  $k - 1$



Правый кадр в момент времени  $k - 1$

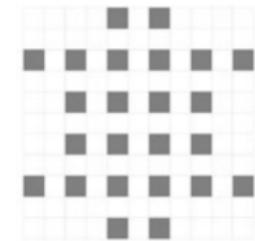
# Дескрипторы точек



Исходное изображение

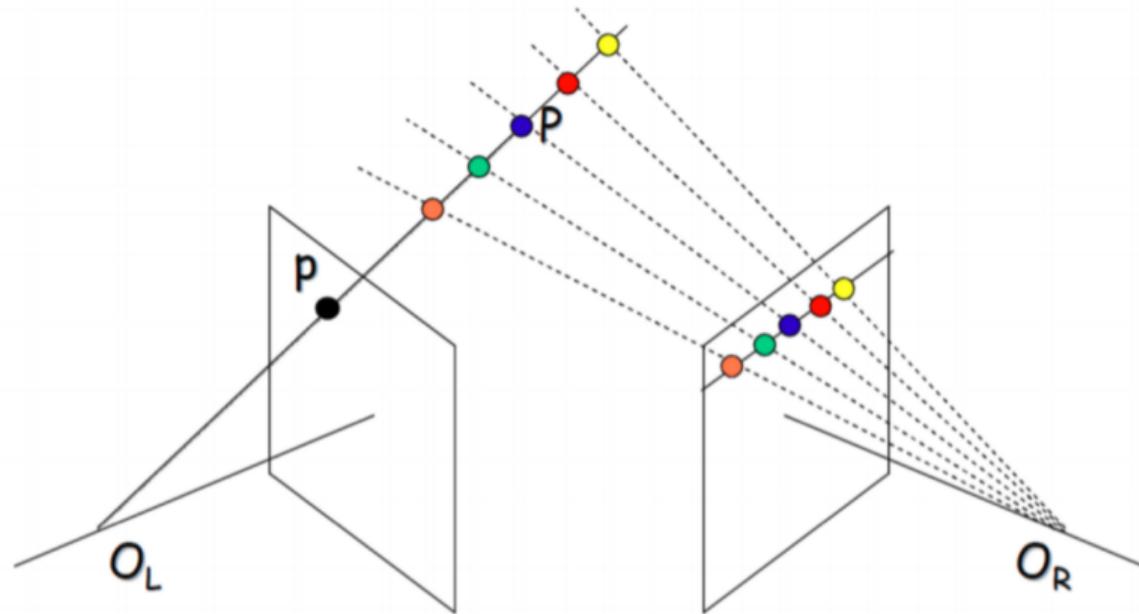


Градиент  
интенсивности  
изображения



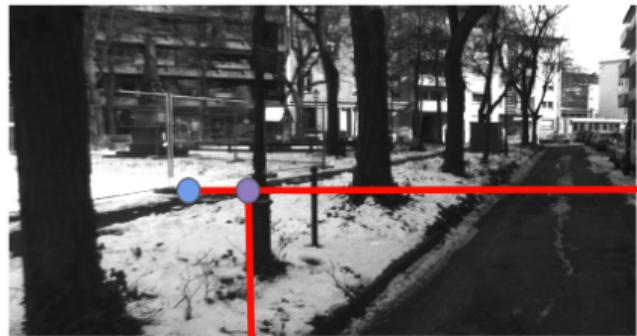
Пиксели, в которых  
берутся значения  
градиента

# Поиск соответствий на кадрах

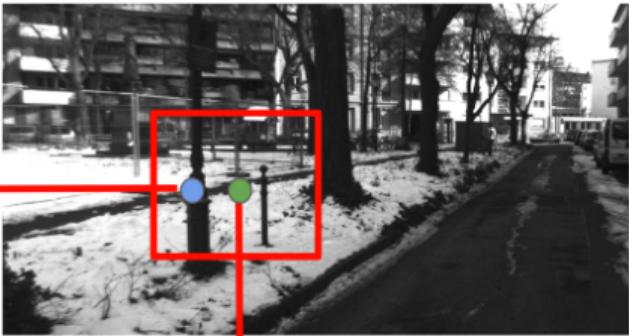


Эпиполярная линия

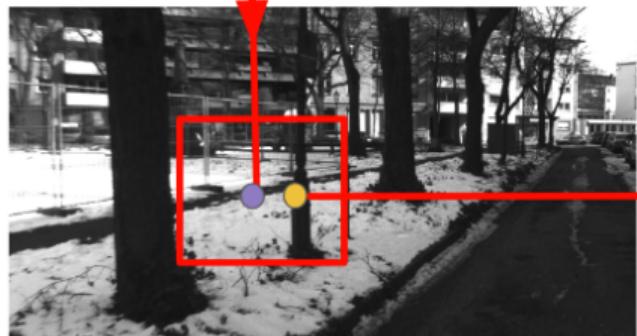
# Поиск соответствий на кадрах



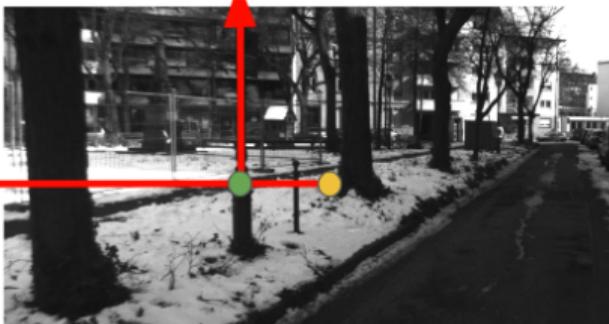
Левый кадр в момент времени  $k$



Правый кадр в момент времени  $k$



Левый кадр в момент времени  $k - 1$

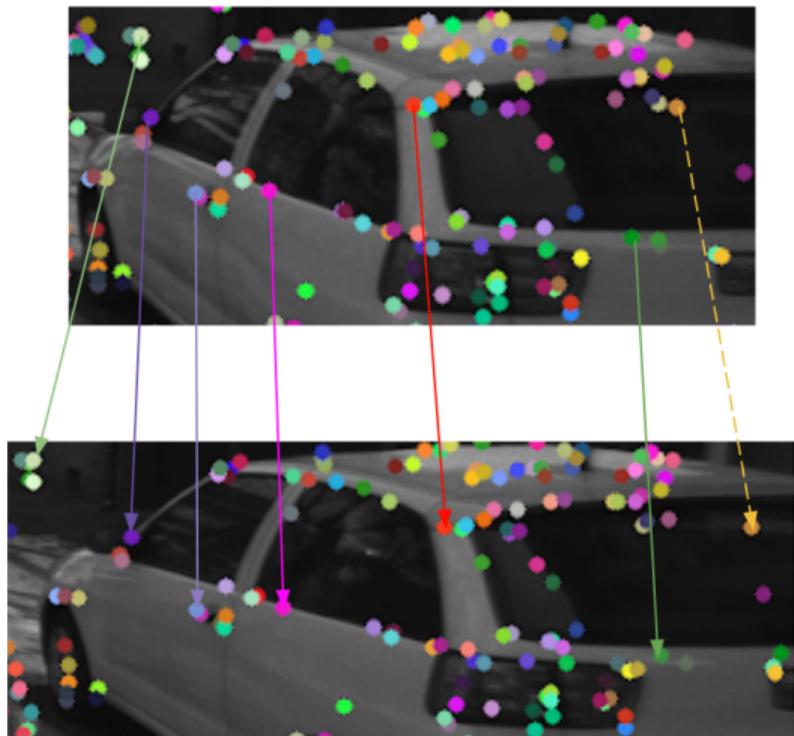


Правый кадр в момент времени  $k - 1$

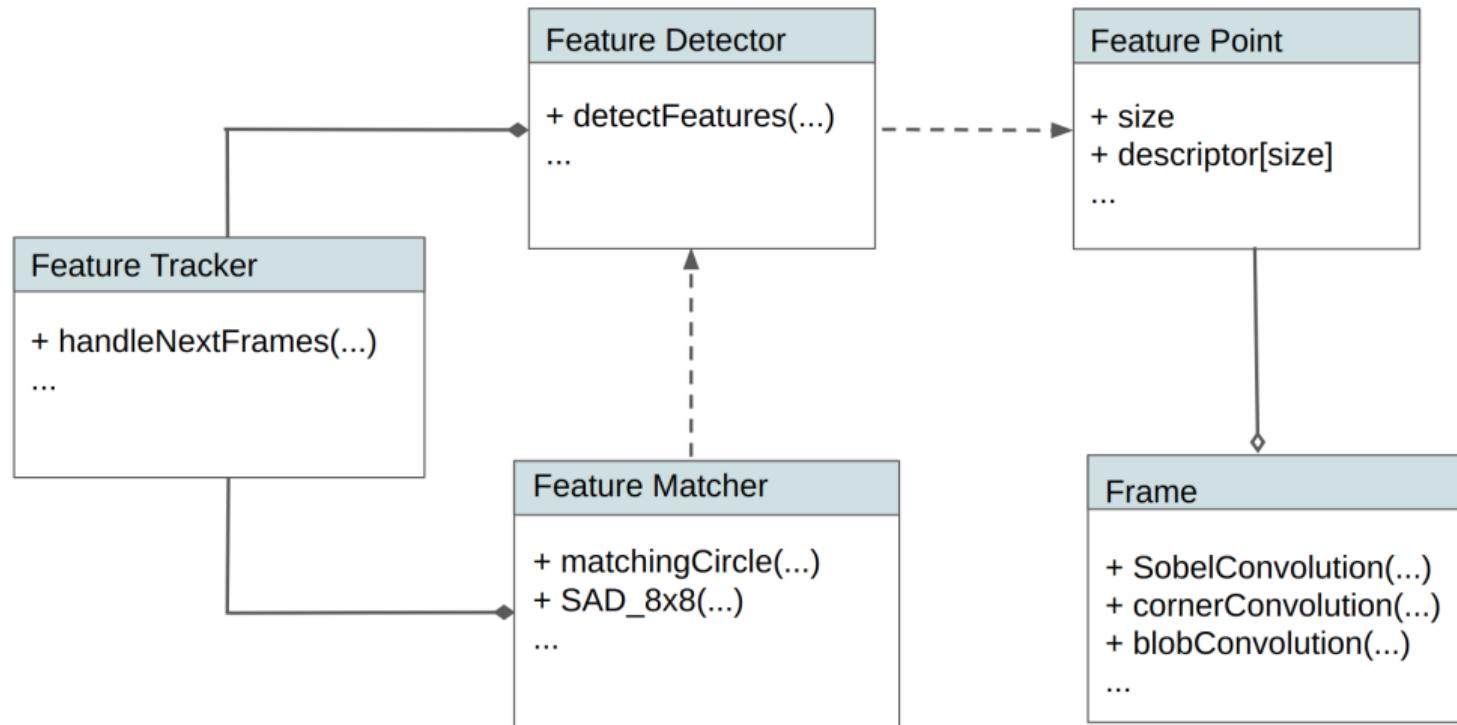
# Поиск соответствий на кадрах

Фрагмент правого  
кадра в момент  
времени  $k$

Фрагмент правого  
кадра в момент  
времени  $k - 1$

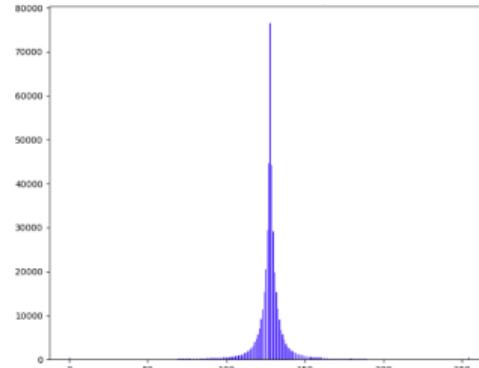
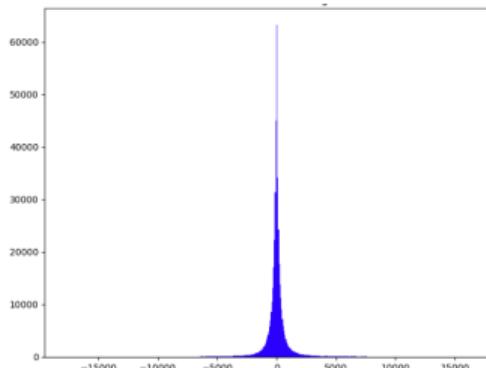


# Упрощенная диаграмма классов



# Оптимизация

- подсчет суммы абсолютных разностей (занимает 90% времени) - с помощью intel intrinsics
- дескриптор: 48 чисел по 32 бита
- для 8-битных чисел SAD работает в 4 раза быстрее
- из-за распределения градиента можем без особых потерь конвертировать в 8-битный формат



Распределение градиента до конвертации и после

# Результаты

Метод	До оптимизации	После оптимизации		Улучшение
Поиск экстремумов	0.045 s	0.0016 s		26 раз
Corner-свертка	0.007 s	0.004 s		2 раза
Blob-свертка	0.007 s	0.006 s		-
SAD	15 ns	32-битные	8-битные	6 раз
		10 ns	2.5 ns	
Фильтр Собеля	0.007 s	3 x 3	5 x 5	2 раза
		0.0013 s	0.003 s	

Данные получены при помощи Google Benchmark

# Результаты

Выполнены следующие задачи:

- ① Ознакомление с предметной областью, изучение основных подходов и особенностей выбранных алгоритмов
- ② Реализация подзадач, связанных с ключевыми точками:
  - детекция ключевых точек на кадрах
  - поиск соответствующих ключевых точек на разных кадрах
- ③ Оптимизация времени работы реализованных алгоритмов