

МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение
высшего образования

«Костромской государственный университет» (КГУ)

Институт автоматизированных систем и технологий

Кафедра информационных систем и технологий

Направление подготовки/Специальность* 09.03.02

Информационные системы и технологии

Профиль Разработка программного обеспечения информационных систем

Дисциплина Технологии компьютерного зрения

ОТЧЁТ

Лабораторная работа №5: Локальные особенности.

Выполнили студенты Смирнов Кирилл Андреевич

Шкунов Владимир Викторович

Группа 22-ИСбо-2а

Проверил Орлов Александр Валерьевич

Оценка _____

Подпись преподавателя _____

Кострома 2024

Задание

Задание 1 Создайте приложение на Python, которое выполняет следующие операции:

1. Загружает изображение из заданного файла
2. Использует детектор Ши-Томаси для обнаружения удачных точек (функция `cv2.goodFeaturesToTrack`)
3. Отображает загруженное изображение и найденные точки в окне.
4. По нажатию клавиш "вверх" и "вниз" должен изменяться параметр функции GFTT `qualityLevel` (диапазон 0.01 - 0.9).
5. По нажатию клавиш "влево" и "вправо" должен изменяться параметр функции GFTT `minDistance` (диапазон 5 - 100).
6. По нажатию Esc окно должно быть закрыто.

Задание 2: Загрузите изображение-сцену `bookz.jpg`, и один из образцов `bookN.jpg` по выбору пользователя.

Используя детектор **`cv2.SIFT`** и сопоставитель **`cv2.BFMatcher`** с параметрами по умолчанию, найдите список точек и их дескрипторов для обоих изображений методом **`.detectAndCompute()`**.

Примените критерий Лёве для отсева неоднозначных пар точек. Покажите сопоставления на экране, используя функцию **`cv2.drawMatches()`**.

Нажатие стрелок "вверх" и "вниз" должно менять порог отсева для критерия Лёве (диапазон 0.1...1.0).

Задание 3: Модифицируйте программу, составленную в задании 2, следующим образом.

Вместо использования функции `cv2.drawMatches()`, постройте по найденным соответствиям гомографическое преобразование методом RANSAC.

Используйте `cv2.perspectiveTransform()` для нахождения точек сцены, соответствующих углам объекта, и обведите его рамкой, вызвав функцию `cv2.polylines()`.

Задание 4: Модифицируйте программу, составленную в задании 3, следующим образом.

Вместо статического файла сцены `bookz.jpg` загрузите видео `bookz.mp4`, и обведите искомый объект на каждом кадре.

Дополнительное задание: Используя функцию `time()` из модуля `time`, отдельно оцените время, требуемое для выполнения операций:

- поиска точек в кадре;
- проведения сопоставления точек;
- отсева точек и поиска гомографии.

Оценку усредните по всему клипу `bookz.mp4`. Попробуйте произвести оценку для алгоритмов извлечения особенностей SIFT, ORB и KAZE. Для сопоставления точек используйте `BFMatcher`. Для SIFT и KAZE используйте метрику L1 (`cv2.NORM_L1`), для ORB - расстояние Хэмминга (`cv2.NORM_HAMMING`).

Составьте таблицу следующего вида:

Алгоритм	Поиск, мс	Сопоставление, мс	Гомография, мс
SIFT
ORB
KAZE

На основании табличных данных составьте гистограмму, и добавьте её и таблицу в отчёт.

Вопросы

1. Что такое дескриптор локальной особенности?
2. Что такое инвариантность алгоритма поиска по отношению к чему-либо?
3. Какие виды локальных особенностей вы знаете? Какие алгоритмы применяются для их поиска?

Вопрос №1

Дескриптор локальной особенности — это набор числовых значений, которые описывают ключевые характеристики фрагмента изображения вокруг локальной особенности (угол или точку). Значения должны быть инвариантными относительно различных трансформаций, чтобы обеспечить корректное сопоставление и распознавание особенностей на изображениях.

Вопрос №2

Инвариантность алгоритма означает, что он сохраняет свои свойства и характеристики при изменении входных данных или при различных преобразованиях.

Инвариантность к повороту: выдает верные результаты при перевернутом изображении.

Инвариантность к освещению: при изменении яркости или контрастности изображения выдает верные результаты.

Инвариантность к масштабу: алгоритм независимо от размера изображения способен находить объекты.

Вопрос №3

Метод Харриса – это алгоритм обнаружения углов на изображении, где анализируется изменение интенсивности пикселей в окрестности каждого пикселя и выявление областей, в которых такие изменения являются наиболее значимыми.

Дескриптор SIFT – алгоритм для обнаружения и описания локальных особенностей, устойчивых к небольшим сдвигам, изменениям освещения и поворотам.

Алгоритм IBR – это метод обнаружения областей на изображении, основанный на итеративном применении бинаризации с последующей переклассификацией.

Алгоритм MSER – это метод обнаружения экстремально стабильных областей на изображении. Он позволяет выделять области с однородной текстурой или яркостью, которые устойчивы к изменениям масштаба и освещения.

Скриншоты работы

Задание 1

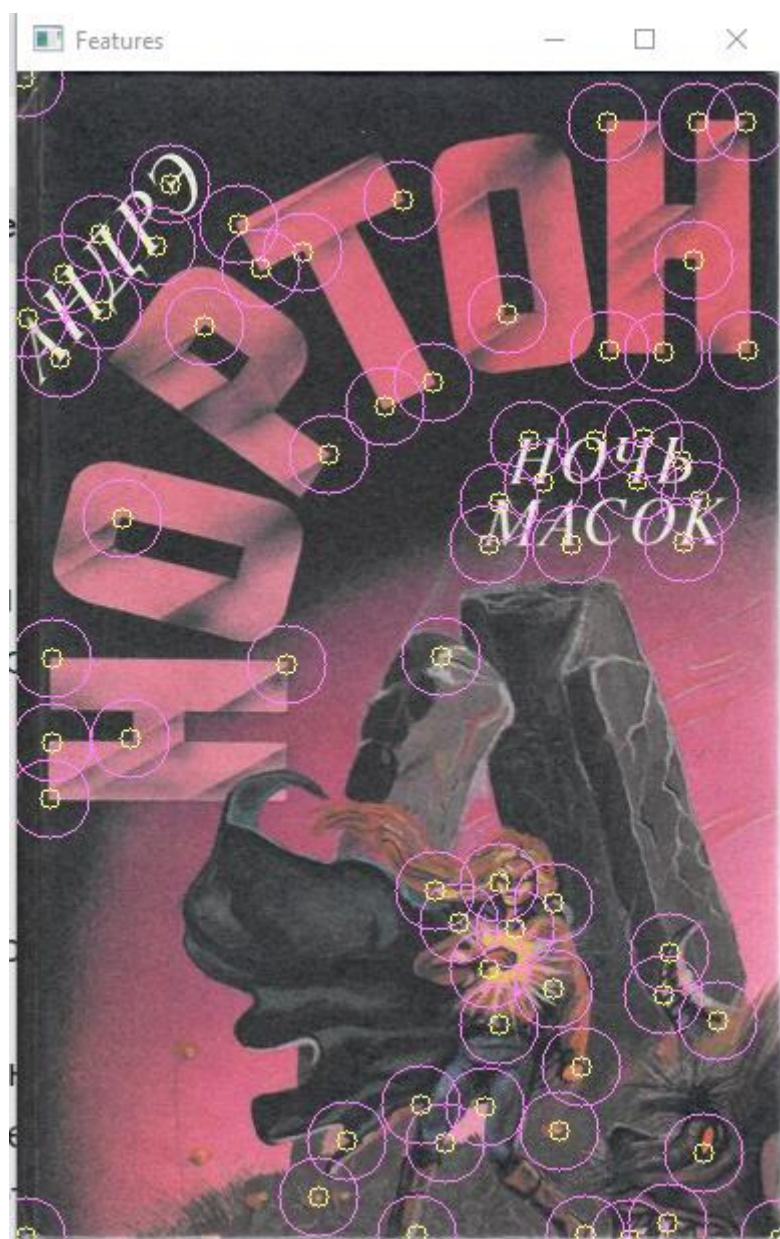


Рис. 1. Вывод первого задания.

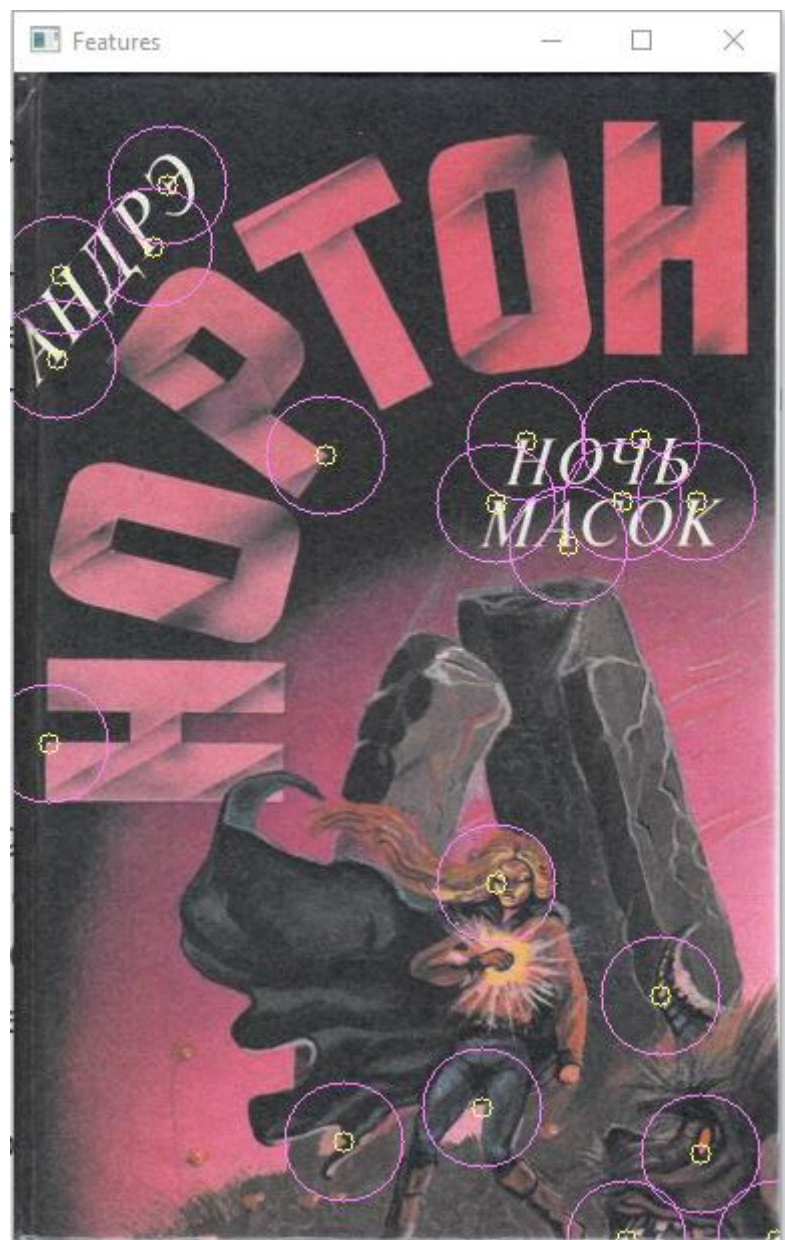


Рис. 2. После изменения **qualityLevel** и **minDistance**

Задание 2



Рис. 3. Вывод второго задания (максимальный порог).



Рис. 4. Вывод второго задания (почти минимальный порог).

Задание 3



Рис. 5. Вывод третьего задания.

Задание 4



Рис. 5. Вывод четвертого задания.

Дополнительно задание

```
Среднее время поиска точек: 0.06992 сек.  
Среднее время сопоставления точек: 0.03659 сек.  
Среднее время поиска гомографии: 0.00114 сек.
```

Рис. 6. Вывод средних значений SIFT.

```
Среднее время поиска точек: 0.00908 сек.  
Среднее время сопоставления точек: 0.00124 сек.  
Среднее время поиска гомографии: 0.00295 сек.
```

Рис. 7. Вывод средних значений ORB.

```
Среднее время поиска точек: 0.24148 сек.  
Среднее время сопоставления точек: 0.01161 сек.  
Среднее время поиска гомографии: 0.00122 сек.
```

Рис. 8. Вывод средних значений KAZE.

Алгоритм	Поиск в мс	Сопоставление в мс	Гомография в мс
SIFT	69.92	36.59	1.14
ORB	9.08	1.24	2.95
KAZE	241.48	11.61	1.22

