

Занятие 4: Основы обработки изображений

Попов Артём, Бугаевский Владимир

Машинное обучение 1, программа OzonMasters

Пример: устранение шума

noisy



non-local means



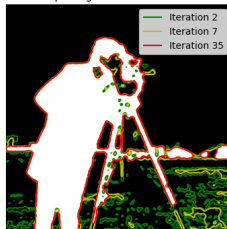
http://scikit-image.org/docs/dev/auto_examples/filters/...

Пример: выделение краёв

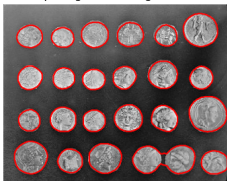
Morphological ACWE segmentation



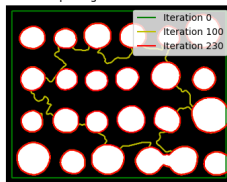
Morphological ACWE evolution



Morphological GAC segmentation



Morphological GAC evolution



http://scikit-image.org/docs/dev/auto_examples/segmentation/...

Пример: изменение баланса цвета



<http://studioeszkozok.hu/uploads/images/...>

Зачем нужна обработка изображений?

1. Улучшение изображения для восприятия человеком (изображение должно стать «лучше» с субъективной точки зрения человека)
2. **Улучшение изображения для восприятия компьютером**
(улучшение качества работы алгоритмов)
3. Технические нужды (например, уменьшение размера изображений для пересылки по почте)
4. Построение спецэффектов

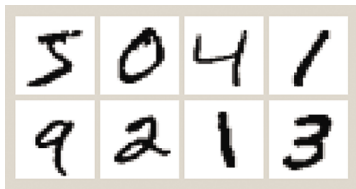
Обо всём этом подробнее в курсе «Компьютерная графика»

Улучшение качества работы алгоритма на изображениях

- ▶ Предобработка входных изображений
 - ▶ Удаление шума, преобразование цветов
- ▶ Выделение дополнительных признаков
 - ▶ Выделение важных объектов, областей
- ▶ Генерация дополнительных изображений для обучения
 - ▶ Добавление изображений, полученных из исходных с помощью некоторых преобразований
- ▶ Аугментация объектов
 - ▶ Преобразование объектов в ходе обучения/применения модели

Пример: классификация цифр

Первое практическое задание: классификация датасета MNIST



Класс изображения не меняется при:

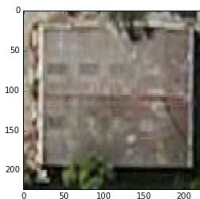
- ▶ сдвигах на 1-10 пикселей
- ▶ поворотах на 10-15 градусов в каждую из сторон
- ▶ размытии, удалении шумов

Добавив преобразованные объекты в исходную выборку, можно существенно повысить качество

Пример: классификация типов крыш

Задача: определение типа крыши, один из четырёх классов:

1. North-South orientation
2. East-West orientation
3. Flat roof
4. Other



1. При повороте на 90° объект 1 и 2 класса меняет класс.
2. При повороте на 180° объект 1 и 2 класса не меняет класс.
3. При повороте на 90° объект 3 класса не меняет класс.

Аугментация объектов улучшила точность с 80% до 84%

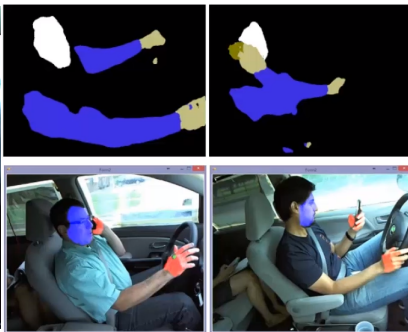
Пример: детектирование действий водителя

Детектировать запрещённые действия водителей по видео:

Исходное изображение:



Выделение областей:



Алгоритм можно настраивать отдельно на выделенные области

Библиотека scipy

- ▶ Scipy — библиотека для научных вычислений
<https://scipy.org/>
- ▶ В том числе, есть несколько модулей для работы с изображениями (misc, ndimage)
- ▶ Только самые базовые алгоритмы

Библиотека scikit-image

- ▶ Scikit-image — библиотека для работы с изображениями
<http://scikit-image.org/>
- ▶ Большое количество реализованных алгоритмов для работы с изображениями
- ▶ Много выложенных примеров использования библиотеки, но плохая документация

Библиотека OpenCV

- ▶ OpenCV — продвинутая библиотека для компьютерного зрения, есть хороший интерфейс для Python 2.7
<https://opencv-python-tutroals.readthedocs.io/en/latest/index.html>
- ▶ Огромное количество реализованных алгоритмов для работы с изображениями, видео, 3D моделями
- ▶ Хорошая документация

Заключение

- ▶ Используя методы обработки изображений, можно повысить качество работы алгоритма
- ▶ Используя классические подходы, можно получить бейзлайн решения даже для сложных задач
- ▶ В Python есть несколько библиотек с уже реализованными алгоритмами обработки
- ▶ У многих библиотек есть свои встроенные средства обработки