**Конспект по книге**

«Программирование компьютерного зрения на языке Python»

Ян Эрик Солем

# Ссылка на книгу

<https://ru.pdfdrive.com/%D0%9F%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5-%D0%BA%D0%BE%D0%BC%D0%BF%D1%8C%D1%8E%D1%82%D0%B5%D1%80%D0%BD%D0%BE%D0%B3%D0%BE-%D0%B7%D1%80%D0%B5%D0%BD%D0%B8%D1%8F-%D0%BD%D0%B0-%D1%8F%D0%B7%D1%8B%D0%BA%D0%B5-python-e187857202.html>

# Информация

## Библиотека Pillow

Вместо библиотеки PIL, указанной в книге, следует использовать Pillow. Первая – для Python 2, вторая – для Python 3. Установка: *pip install Pillow*

В большинстве файлов (если не во всех) используется OpenCV вместо PIL.

## OpenCV из исходников

Для использования некоторых функций OpenCV (например, SIFT) нужно установить эту библиотеку из исходников. Cм. файл *OpenCV from soucres.md.*

## Точки для кусочно-аффинного преобразования

Точки на деформируемом нужно брать **в точно таком же порядке**, как и на конечном изображении. Если на деформируемом – слева направо сверху-вниз, то и на конечном – также слева направо сверху-вниз, **а не** сверху-вниз слева направо.

# Полезные страницы

* *Стр. 25*. MatPlotLib. Изолинии и гистограммы изображений
* *Стр. 26.* MatPlotLib. Интерактивное аннотирование
* *Стр. 33.* NumPy. Усреднение изображений
* *Стр. 70.* 2.3 Сопоставление изображений с геометками

# Алгоритмы машинного зрения

## Fast Non-Local Means Denosing

**Применение:** очистка изображения от шумов

Используется вместо модели Рудина-Ошера-Фатеми (ROF) по алгоритму Шамболя (Chambolle) – *стр. 47*

OpenCV: <https://opencv-python-tutroals.readthedocs.io/en/latest/py_tutorials/py_photo/py_non_local_means/py_non_local_means.html>

## Сингулярное разложение матрицы (SVD)

, где U – левые сингулярные вектора, V – правые синг. вектора, Σ – матрица с сингулярными числами

**Применение:** в PCA, в DLT, в МНК; вычисление ранга матрицы, обратной и псевдообратной матриц, решения систем уравнений и пр.

Википедия: <https://ru.wikipedia.org/wiki/Сингулярное_разложение>

machinelearning.ru: <http://www.machinelearning.ru/wiki/index.php?title=Сингулярное_разложение#.D0.93.D0.B5.D0.BE.D0.BC.D0.B5.D1.82.D1.80.D0.B8.D1.87.D0.B5.D1.81.D0.BA.D0.B8.D0.B9_.D1.81.D0.BC.D1.8B.D1.81.D0.BB_SVD>

## Метод главных компонент (PCA) для изображений

**Применение:** понижение размерности данных**;** получение мод среди изображений, нахождение «направления» данных, «центра» данных

*Страница 34*

OpenCV: <https://docs.opencv.org/3.4/d1/dee/tutorial_introduction_to_pca.html>

## Детектор углов Харриса

**Применение:** нахождение особых точек – углов

*Страница 53*

OpenCV: <https://docs.opencv.org/3.4/d4/d7d/tutorial_harris_detector.html>

## FAST

**Применение:** аналогично детектору углов Харриса

*Страница 78*

OpenCV: <https://docs.opencv.org/master/df/d0c/tutorial_py_fast.html>

В документации - описание алгоритма FAST и его использование с деревьями решений (раздел Machine Learning a Corner Detector)

## MSER

**Применение:** максимально-устойчивые области экстремума

*Страница 78*

OpenCV: <https://docs.opencv.org/3.4/d3/d28/classcv_1_1MSER.html>

## SIFT

Требуется собрать OpenCV из исходников

**Применение:** нахождение особых признаков на изображениях, независимых от поворота, мастшаба и пр.; сопоставление изображений, нахождение одинаковых деталей на разных изображениях

*Страница 62*

OpenCV: <https://docs.opencv.org/master/da/df5/tutorial_py_sift_intro.html>

Википедия: <https://ru.wikipedia.org/wiki/Масштабно-инвариантная_трансформация_признаков>

## SURF

Требуется собрать OpenCV из исходников

**Применение:** быстрый SIFT

См. также: ORB

## Метод наименьших квадратов (МНК)

**Применение:** решение матричных уравнений, переопределенных систем уравнений (когда количество уравнений превышает количество неизвестных), для поиска решения в случае обычных (не переопределенных) нелинейных систем уравнений, для аппроксимации точечных значений некоторой функции

Википедия: <https://ru.wikipedia.org/wiki/Метод_наименьших_квадратов>

## Прямое линейное преобразование (DLT)

## Аффинные преобразования

## Триангуляция по Делоне (Delaunay)

## Кусочно-аффинное деформирование