

## Задание 2. Приложения ЕМ-алгоритма

Практикум 317 группы, 2018

Начало выполнения задания: 18 марта 2018 года.

Мягкий дедлайн: **31 марта 2018 года, 23:59.**

Жёсткий дедлайн: **14 апреля 2018 года, 23:59.**

Максимальный балл: 60 (плюс бонусные баллы).

Текст задания обновлён 17 марта 2018 г.

## Формулировка задания

Данное задание направлено на ознакомление с ЕМ-алгоритмом. Задание состоит из двух частей, каждая оценивается в 30 баллов:

1. В первой части необходимо написать на языке Python собственную реализацию ЕМ-алгоритма для восстановления смеси нормальных распределений. Во второй части необходимо написать на языке Python собственную реализацию модели PLSA с возможностью добавления регуляризации.

Прототипы функций должны строго соответствовать прототипам, описанным в спецификации, и проходить все выданные тесты. Задание, не проходящее все выданные тесты, приравнивается к невыполненному. При написании необходимо пользоваться стандартными средствами языка Python, библиотеками `numpy`, `scipy`, `scikit-learn` и `matplotlib`, но запрещается пользоваться уже готовыми реализациями выше упомянутых методов.

2. Провести описанные ниже эксперименты.
3. Написать отчёт о проделанной работе (формат PDF, подготовленный в системе  $\text{\LaTeX}$ , или формат html, конвертированный из jupyter-notebook).
4. Сдать в систему anytask .zip архив с отчётом и jupyter-notebook с экспериментами (если есть дополнительные модули с кодом, их тоже нужно поместить в архив). Архив должен иметь название `task2_фамилия_имя.zip`
5. Допускается сдача только одной из двух частей задания

## Часть 1: Вычитание фона

### Введение

Вычитание фона — важная прикладная задача. Часто оно является первой стадией в системах анализа видео с камер наблюдения. Необходимо классифицировать пиксели видеопоследовательности на принадлежащие фону и принадлежащие объектам переднего плана. Обычно предполагается, что камера статична. По обучающей выборке необходимо оценить модель фона, чтобы затем для каждого пикселя каждого кадра тестовой видеопоследовательности уметь предсказать вероятность того, что он является фоном. При предположении равномерного распределения принадлежности пикселей объектам переднего плана (обычно нет априорных сведений, чтобы использовать другое распределение) принятие решения для данного пикселя эквивалентно сравнению предсказанной вероятности фона с некоторым порогом.

Часто (хотя и не всегда) в качестве обучающей выборки удаётся взять один или несколько кадров, на которых отсутствуют объекты переднего плана. Но даже в этом случае задача оценки модели фона не является тривиальной из-за того, что фон никогда не бывает полностью статичен: всегда есть шум камеры, также может меняться освещение, тени, камера может дрожать, фон может быть динамическим (листва или вода в присутствии ветра), к фону могут добавляться дополнительные объекты. Вам предлагается протестировать несколько моделей разной сложности и сделать выводы о применимости моделей в разных ситуациях.

### Данные

В качестве тестовых последовательностей предлагается использовать стандартные данные с веб-сайта <http://wordpress-jodo.in.dmi.usherb.ca/dataset2012/>. Один набор данных является обязательным:

- Baseline/pedestrians (train — 1, ..., 299; test — 300, ..., 1099),

## Оценка алгоритмов

Для каждого кадра тестовой части последовательности можно оценить качество алгоритма, зная его верную разметку (директория `groundtruth`). Зная, какие части изображения на самом деле относятся к фону (класс `static`) и к переднему плану (класс `motion`), можно оценить количество верных положительных (TP) и отрицательных (TN) обнаружений пикселей, а также ложноположительных (FP) и ложноотрицательных (FN).

Для оценки качества вычитания фона следует использовать следующие три инструмента:

- Для каждого тестового кадра посчитайте количество ошибок 1 рода (FP) и 2 рода (FN). Постройте график зависимости ошибок от номера кадра. Опишите наблюдаемые эффекты и попытайтесь проанализировать их.
- Проведите визуальную оценку. Для этого на каждом из кадров выделите области, отнесённые к переднему плану. Сделайте вывод о характере ошибок.
- Проведите оценку величин TPR и FPR (доли верноположительных и ложноположительных обнаружений, соответственно) для разных значений порога по всей тестовой последовательности (не усредняя по кадрам) и постройте графики ROC характеристики [https://en.wikipedia.org/wiki/Receiver\\_operating\\_characteristic](https://en.wikipedia.org/wiki/Receiver_operating_characteristic). Площадь под графиком ROC кривой (AUC ROC) является одним из способов сравнить качество разных моделей.

Для удобства к заданию прилагается python код, который содержит функцию подсветки маски переднего плана, а так же функцию, которая позволяет проигрывать видео. Данный код может быть полезным для визуальной оценки качества работы методов. Для работы функции проигрывания видео в IPython Notebook необходимо установить JSAnimation из репозитория <https://github.com/jakevdp/JSAnimation> и импортировать следующий модуль – `from JSAnimation import IPython_display`. Подробнее информацию про установку можно посмотреть здесь: <https://gist.github.com/gforsyth/188c32b6efe834337d8a>

## Одномерная гауссиана для оценки фона

В самом простом варианте можно моделировать распределение цвета в каждом пикселе одномерным нормальным распределением. Для этого нужно перевести видеопоследовательность в полутоновую. Это можно сделать, например, с помощью формулы из стандарта NTSC:  $gs = 0.2126r + 0.7152g + 0.0722b$ . Для каждого пикселя обучающей части последовательности необходимо оценить параметры нормального распределения, используя яркости в каждой позиции пикселя (например, для **pedestrians** будет 299 точек для оценки плотности в каждом пикселе). Обратите внимание, что некоторые гауссианы получаются вырожденными. Для регуляризации разумно ограничивать возможное значение среднеквадратичного отклонения снизу. Можно использовать  $\sigma_{\min} = 5$ . Далее, при классификации пикселя тестовой выборки, его нужно относить к фону тогда и только тогда, когда его яркость отклоняется от  $\mu$  меньше, чем на  $k\sigma$ . Если установить  $k = 3$ , ожидается, что к переднему плану будут отнесены только 0.27% пикселей фона (“правило  $3\sigma$ ”). Изменяя этот порог, можно регулировать соотношение между количеством ложноположительных и ложноотрицательных обнаружений.

## Адаптивная гауссиана для оценки фона

Если в последовательности присутствует дрейф фона, желательно оценивать его модель по нескольким последним кадрам. Проблема заключается в том, что строить новую модель для каждого кадра может быть вычислительно затратно, а на предшествующих кадрах фон может быть загорожен объектами переднего плана. Чтобы этого избежать, можно оценивать модель в онлайн-режиме («на лету»). Итеративный алгоритм оценки параметров гауссовской модели фона был рассмотрен на 4-ой лекции курса.

## Усложнение модели фона

Предложенную модель фона можно усложнить, используя трёхмерное нормальное распределение в цветовом пространстве RGB. Решение о метке пикселя тестового кадра принимается сравнением плотности нормального распределения в данной точке с порогом. Для оценки качества модели используйте ROC кривую.

Ещё более сложная модель — смесь распределений — может быть полезна для моделирования фона, если он нестатичен, например, содержит воду или листву (то есть, когда распределение может иметь несколько мод).

## Требуемые эксперименты

1. Реализовать ЕМ-алгоритм для восстановления смеси многомерных нормальных распределений. Предусмотреть возможность нескольких запусков из случайных начальных приближений с выбором наилучшего по значению найденного правдоподобия. Требования по эффективности реализации: среднее время одной ЕМ-итерации для  $N = 10000$  объектов,  $D = 100$  признаков и  $K = 10$  компонент смеси не должно превышать одной секунды.

2. Провести тестирование реализованного ЕМ-алгоритма на двумерных модельных данных. Для этого сгенерировать данные из смеси распределений с заданными параметрами, а затем восстановить по этим данным параметры смеси с помощью ЕМ-алгоритма. Отобразить результат восстановления, где объекты выборки, соответствующие одинаковым компонентам смеси, показаны одинаковыми цветами. Убедиться в том, что значение правдоподобия в ЕМ-итерациях монотонно не убывает.
3. Реализовать оценку модели фона с помощью одномерной гауссианы и протестировать на последовательности **pedestrians**. Проанализировать качество вычитания фона с помощью трёх инструментов, описанных выше.
4. Реализовать оценку модели фона с помощью многомерной гауссианы в цветовом пространстве RGB и протестировать результат на последовательности **pedestrians**. Проанализировать ошибки метода и сравнить его с остальными.
5. Запустить алгоритм разделения смеси 3 трёхмерных гауссиан для вычитания фона в последовательности **traffic**. Проанализировать ошибки метода и сравнить результат с использованием одной гауссианы.

**Замечание.** Поскольку разделение смеси гауссиан на реальных данных вычислительно затратно, предлагается сначала отладить алгоритм на синтетических данных. Сгенерируйте выборку из смеси гауссиан и попробуйте восстановить её параметры с помощью ЕМ-алгоритма. Постройте график изменения логарифма правдоподобия, убедитесь в его монотонном росте.

6. Наилучшее из полученных решений для алгоритма вычитания фона должно быть добавлено в отчёт (или приложено отдельным файлом) в виде анимации. Формы анимации могут быть различными: виджет в IPython notebook **JSAnimation**, отдельный файл с видео, анимированные изображения кадров и т.д.

#### Бонусная часть:

1. (до +5 баллов) Реализовать оценку модели фона на основе адаптивной одномерной гауссианы. Протестировать метод на последовательности **pedestrians** и сравнить результаты с простой одномерной гауссианой.
2. (до +5 баллов) Реализовать оценку модели фона трёхмерной гауссианой и смесью гауссиан в цветовом пространстве HSV, в котором каналы коррелируют меньше. Для перевода изображения можно воспользоваться встроенной функцией `matplotlib.colors.rgb_to_hsv`.
3. (до +5 баллов) Реализовать ЕМ-алгоритм для восстановления параметров смеси из нормальных распределений с диагональными матрицами ковариаций. Оценить, как упрощение модели влияет на её качество и скорость работы.

## Часть 2: Тематическое моделирование

ToBeDone