

Задание 4. Методы восстановления плотности распределений в задаче вычитания фона

Практикум 317 группы, 2014

Начало выполнения задания: 28 ноября 2014 года.

Срок сдачи: **11 декабря 2014 года, 23:59.**

Максимальный балл: 5.0 (плюс бонусные баллы).

Содержание

1 Задание	1
2 Данные	1
3 Оценка алгоритмов	2
4 Реализация	2
4.1 Одномерная гауссиана	2
4.2 Оценка гауссианы на лету	2
4.3 Теперь в 3D!	3
4.4 Смесь гауссиан	3
5 Требования к реализации	3
6 Сдача задания	3
7 Бонусные баллы	4

1 Задание

Вычитания 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. Предполагая равномерное распределение на принадлежность 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.

2 Данные

Вам предлагается использовать стандартные тестовые последовательности с веб-сайта <http://wordpress-jodoin.dmi.usherb.ca/dataset2012/>. Два набора данных являются обязательными:

- Baseline/pedestrians (train — 1, . . . , 299; test — 300, . . . , 1099),
- Camera jitter/traffic (train — [581, . . . , 723, 752, . . . , 951]; test — 1000, . . . , 1570).

Вы можете использовать и другие последовательности для лучшего изучения свойств методов.

3 Оценка алгоритмов

Для каждого кадра тестовой части последовательности можно оценить качество алгоритма, зная его верную разметку (директория `groundtruth`). Обратите внимание, что только часть каждого из кадров является областью интереса. Например, на последовательности `traffic` область интереса ограничивается дорогой. 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). Пиксели с промежуточными метками (50, 85, 170) не входят в регион интереса и не участвуют в подсчёте точности. Другими словами, значения TP, FP, TN, FN не зависят от того, что вернул алгоритм на этих пикселях.

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 построьте графики ROC характеристики https://en.wikipedia.org/wiki/Receiver_operating_characteristic. Площадь под графиком ROC кривой (AUC ROC) является одним из способ 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>

4 Реализация

4.1 Одномерная гауссиана

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 оцените 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 μ меньше, чем на $k\sigma$. Если установить $\sigma = 3$, ожидается, что к переднему плану будут отнесены только 0.27% пикселей фона (“правило 3σ ”). Изменяя этот порог, можно регулировать соотношение между количеством ложноположительных и ложноотрицательных обнаружений.

Реализуйте такую оценку фона и запустите на последовательности `pedestrians`. Проанализируйте качество вычитания фона с помощью трёх инструментов, описанных в предыдущей секции.

4.2 Оценка гауссианы на лету

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 также на предшествующих кадрах фон может быть загорожен объектами переднего плана. Чтобы этого избежать, можно оценивать модель в онлайн-режиме («на лету»).

Реализуйте следующий итеративный алгоритм оценки параметров гауссовской модели фона на лету (оценка также производится по полутоновым изображениям независимо для разных пикселей):

Новый параметр ρ отвечает за величину памяти метода (размер окна): чем меньше ρ , тем большее влияние имеют старые кадры. Рекомендуемые значения: $\rho = 0.01$, μ_0 равен яркости на первом кадре, $\sigma_0^2 = 100$. Не забывайте про регуляризацию: устанавливайте минимальное значение дисперсии.

Протестируйте метод на последовательности `pedestrians` и повторите анализ из предыдущего пункта. Сделайте вывод о сравнении методов.

Инициализировать μ_1, σ_1^2 .

for $t = 1, \dots, \text{Number_of_frames}$ **do**

Принять решение о метке пикселя на кадре t : передний план тогда и только тогда, когда $\frac{|(I_t - \mu_t)|}{\sigma_t} > k$, где I_t — яркость пикселя на кадре t .

Если пиксель отнесён к фону, обновить $\mu_{t+1} = \rho I_t + (1 - \rho)\mu_t$, иначе $\mu_{t+1} = \mu_t$.

Если пиксель отнесён к фону, обновить $\sigma_{t+1}^2 = (I_t - \mu_{t+1})^2 \rho + (1 - \rho)\sigma_t^2$, иначе $\sigma_{t+1}^2 = \sigma_t^2$.

4.3 Теперь в 3D!

Предлагается изменить модель фона на более гибкую: трёхмерное нормальное распределение в цветовом пространстве RGB. Рассмотрите два варианта модели: с полной и с диагональной матрицей ковариаций. Оцените параметры на обучающей части (в офлайн-режиме), не забывая о регуляризации матрицы ковариаций (можно добавлять положительное число к диагонали, например, 5). Решение о метке пикселя тестового кадра принимается сравнением плотности нормального распределения в данной точке с порогом.

Постройте ROC кривую для последовательности **pedestrians**, выберите значение порога и проанализируйте ошибки метода. Сделайте вывод о сравнении метода с остальными.

Повторите эксперимент, используя цветовое пространство HSV, в котором каналы коррелируют меньше. Для перевода изображения можно пользоваться встроенной функцией `matplotlib.colors.rgb_to_hsv`. Обратите внимание, что она принимает и возвращает яркости на отрезке $[0,1]$. Сделайте вывод об уместности использования диагональной матрицы ковариаций для разных задач.

4.4 Смесь гауссиан

Смесь распределений может быть полезна для моделирования фона, если он нестатичен, например, содержит воду или листву, то есть, когда распределение может иметь несколько мод.

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 EM-алгоритма. При этом нельзя пользоваться сторонними средствами разделения смеси распределений. Постройте график изменения логарифма правдоподобия, убедитесь в его монотонном росте. Приведите пример данных, в котором существенна зависимость от начального приближения, а также результаты запусков из разных начальных приближений. Уделите внимание оптимизации кода, используйте векторизацию.

Запустите алгоритм разделения смеси 3 трёхмерных гауссиан для вычитания фона в последовательности **traffic**, выбрав модель с учётом результатов предыдущего пункта (важны точность и скорость работы). При этом желательно использовать несколько случайных начальных приближений и выбирать лучшее по значению правдоподобия. Проанализируйте ошибки и сравните результат с использованием одной гауссианы, сделайте выводы.

5 Требования к реализации

При работе разрешается использовать только два сторонних пакета — `numpy` и `matplotlib`. Подсчет матриц ковариации и разделение смесей гауссиан должно быть реализовано самостоятельно.

Постарайтесь уделить особое внимание оптимизации кода, используйте векторизацию. Оценка параметров гауссиан, вычитание фона и разделение смеси должны использовать по меньшей мере одну из двух векторизаций: по пикселям (одновременно обрабатываются все пиксели изображения), по времени (одновременно обрабатываются пиксели стоящие на фиксированной позиции на всех изображениях последовательности). Использование обоих приветствуется.

6 Сдача задания

Сдача задания осуществляется через репозиторий SVN. Задание должно быть выполнено в IPython Notebook, формат сдачи — `.ipynb` и `.ru` файлы, а также отчёт в формате pdf. Результаты всех экспериментов, за исключением видео, должны быть представлены в отчете. `.ipynb` файлы должны содержать код, с помощью которого можно повторить все эксперименты задания. За отсутствие анализа экспериментов и выводов балл будет снижаться!

`.ipynb` файл с видео имеет довольно большой размер, поэтому рекомендуется использовать видео для анализа экспериментов, при этом убирая само видео во время коммита в SVN. Удалить видео можно, нажимая `cell -> current output -> clear` для каждой клетки с видео. Не рекомендуется коммитить файлы больше 1МБ. При этом у проверяющего должна быть возможность собрать и посмотреть видео с результатами экспериментов с помощью вашего кода.

7 Бонусные баллы

- до +1.0 балла. Реализация и оценка работы модификаций предложенных методов. Размер бонуса зависит от качества работы и оригинальности модификации, а также является субъективным и безапелляционным.