Практическое задание 2. EM алгоритм для детектива

Янаков Дмитрий Спартакович, 417 группа

17 ноября, 2022

1 Теория

1.1 Апостериорное распределение на координаты лица

$$q(\boldsymbol{d}) = p(\boldsymbol{d} \mid \boldsymbol{X}, \boldsymbol{\theta}, \boldsymbol{A}) = \prod_k p(\boldsymbol{d}_k \mid \boldsymbol{X}_k, \boldsymbol{\theta}, \boldsymbol{A})$$

$$p(\boldsymbol{d}_k \mid \boldsymbol{X}_k, \boldsymbol{\theta}, \boldsymbol{A}) = \frac{p(\boldsymbol{d}_k, \boldsymbol{X}_k, \boldsymbol{\theta}, \boldsymbol{A})}{p(\boldsymbol{X}_k, \boldsymbol{\theta}, \boldsymbol{A})} = \frac{p(\boldsymbol{X}_k \mid \boldsymbol{d}_k, \boldsymbol{\theta}) p(\boldsymbol{d}_k \mid \boldsymbol{A})}{p(\boldsymbol{X}_k, \boldsymbol{\theta}, \boldsymbol{A})} = \frac{p(\boldsymbol{X}_k \mid \boldsymbol{d}_k, \boldsymbol{\theta}) p(\boldsymbol{d}_k \mid \boldsymbol{A})}{\sum_{\hat{\boldsymbol{d}}_k} p(\boldsymbol{X}_k \mid \hat{\boldsymbol{d}}_k, \boldsymbol{\theta}) p(\hat{\boldsymbol{d}}_k \mid \boldsymbol{A})}$$

1.2 Точечные оценки на параметры

$$\mathbb{E}_{q(\boldsymbol{d})} \log p(\boldsymbol{X}, \boldsymbol{d} \mid \boldsymbol{\theta}, \boldsymbol{A}) = \sum_d q(\boldsymbol{d}) \log p(\boldsymbol{X}, \boldsymbol{d} \mid \boldsymbol{\theta}, \boldsymbol{A}) = \sum_d q(\boldsymbol{d}) \sum_k \log \left[p(\boldsymbol{X}_k \mid \boldsymbol{d}_k, \boldsymbol{\theta}) p(\boldsymbol{d}_k \mid \boldsymbol{A}) \right])$$

Через $\boldsymbol{q}(\boldsymbol{d}_k)$ далее будем обозначать $p(\boldsymbol{d}_k \mid \boldsymbol{X}_k, \boldsymbol{\theta}, \boldsymbol{A})$.

1.2.1 A

Составим Лагранжиан:

$$L = \sum_d \sum_k q(\boldsymbol{d}) \log p(\boldsymbol{X}_k \mid \boldsymbol{d}_k, \boldsymbol{\theta}) + \sum_d \sum_k q(\boldsymbol{d}) \log p(\boldsymbol{d}_k \mid \boldsymbol{A}) + \lambda \left(\sum_{ij} A(i, j) - 1 \right)$$

$$\frac{\partial L}{\partial \boldsymbol{A}(i, j)} = \lambda + \sum_d \sum_k \frac{1}{\boldsymbol{A}(i, j)} q(\boldsymbol{d}) \left[\boldsymbol{d}_k^h = i \right] \left[\boldsymbol{d}_k^w = i \right] = \lambda + \sum_k \frac{1}{\boldsymbol{A}(i, j)} \boldsymbol{q}(\boldsymbol{d}_k) = 0$$

$$\implies \boldsymbol{A}(i, j) = -\frac{1}{\lambda} \sum_k \boldsymbol{q}(\boldsymbol{d}_k)$$

Подставим полученное $\boldsymbol{A}(i, j)$ в уравнение связи:

$$-\sum_{ij} \frac{1}{\lambda} \sum_k \boldsymbol{q}(\boldsymbol{d}_k) = 1$$

$$\implies \lambda = -\sum_k \sum_{ij} \boldsymbol{q}(\boldsymbol{d}_k) = -\sum_k 1 = -K$$

$$\implies \boldsymbol{A}(i, j) = \frac{\sum_k \boldsymbol{q}(\boldsymbol{d}_k)}{K}$$

В случае MAP-EM алгоритма оценка имеет следующий вид:

$$\boldsymbol{A}(i, j) = \frac{\sum_k \boldsymbol{q}(\boldsymbol{d}_k) \left[\boldsymbol{d}_k = (i, j) \right]}{K}$$

1.2.2 F

$$L = \sum_{d}\sum_{k} q(\boldsymbol{d}) \log p(\boldsymbol{X}_k \mid \boldsymbol{d}_k, \boldsymbol{\theta}) + \sum_{d}\sum_{k} q(\boldsymbol{d}) \log p(\boldsymbol{d}_k \mid \boldsymbol{A})$$

$$= \sum_{d}\sum_{k} q(\boldsymbol{d}) \log \mathcal{N}(\boldsymbol{X}_k(i,j) \mid \boldsymbol{F}(i - d_k^h, j - d_k^w), s^2) + \sum_{d}\sum_{k} q(\boldsymbol{d}) \log p(\boldsymbol{d}_k \mid \boldsymbol{A})$$

Нормальное распределение имеет следующий вид:

$$\mathcal{N}(\boldsymbol{X}_k(i,j) \mid \boldsymbol{F}(i - d_k^h, j - d_k^w), s^2) = \frac{1}{s\sqrt{2\pi}} \exp\left(-\frac{\left(\boldsymbol{X_k}(i,j) - \boldsymbol{F}(i - d_k^h, j - d_k^w)\right)^2}{2s^2}\right)$$

Тогда:

$$\frac{\partial L}{\partial \boldsymbol{F}(i,j)} = \sum_{d_k}\sum_{k} \frac{\boldsymbol{X_k}(i + d_k^h, j + d_k^w) - \boldsymbol{F}(i,j)}{s^2} q(\boldsymbol{d_k}) + \frac{\partial}{\partial \boldsymbol{F}(i,j)} f(\boldsymbol{d}, s, A) = 0$$

$$\Longrightarrow \boldsymbol{F}(i,j) = \frac{\sum_{d_k}\sum_k q(\boldsymbol{d_k})\boldsymbol{X_k}(i + d_k^h, j + d_k^w)}{\sum_{d_k}\sum_k q(\boldsymbol{d_k})} = \frac{\sum_{d_k}\sum_k q(\boldsymbol{d_k})\boldsymbol{X_k}(i + d_k^h, j + d_k^w)}{K}$$

В случае MAP-EM алгоритма оценка имеет следующий вид:

$$\boldsymbol{F}(i,j) = \frac{\sum_k q(\boldsymbol{d_k})\boldsymbol{X_k}(i + i_k^*, j + j_k^*)}{K},$$

где (i_k^*, j_k^*) – точка, в которой $q(\boldsymbol{d}_k) = 1$.

1.2.3 B

Оценка на $\boldsymbol{B}$ выводится аналогично $\boldsymbol{F}$ за тем лишь исключением, что вместо $\mathcal{N}(\boldsymbol{X}_k(i,j) \mid \boldsymbol{F}(i - d_k^h, j - d_k^w), s^2)$ используется $\mathcal{N}(\boldsymbol{X}_k(i,j) \mid \boldsymbol{B}(i,j), s^2)$ и дополнительно навешивается условие того, что $(i,j) \notin faceArea(\boldsymbol{d}_k)$.

Получаем, что:

$$\boldsymbol{B}(i,j) = \frac{\sum_{d_k:(i,j)\notin faceArea(\boldsymbol{d}_k)} \sum_k q(\boldsymbol{d_k})\boldsymbol{X_k}(i,j)}{\sum_{d_k:(i,j)\notin faceArea(\boldsymbol{d}_k)} \sum_k q(\boldsymbol{d_k})}$$

MAP-EM оценка выписывается аналогично.

1.2.4 s^2

$$L = \sum_{d}\sum_{k} q(\boldsymbol{d}) \log p(\boldsymbol{X}_k \mid \boldsymbol{d}_k, \boldsymbol{\theta}) + \sum_{d}\sum_{k} q(\boldsymbol{d}) \log p(\boldsymbol{d}_k \mid \boldsymbol{A})$$

Вывод аналогичен выводу $\boldsymbol{F}$ и $\boldsymbol{B}$, только теперь в $p(\boldsymbol{X}_k \mid \boldsymbol{d}_k, \boldsymbol{\theta})$ учитываются оба нормальных распределения.

$$s^2 = \frac{1}{HWK}\sum_{k}\sum_{\boldsymbol{d}_k} q(\boldsymbol{d}_k)\Bigg(\sum_{(i,j)\in faceArea(\boldsymbol{d}_k)} \left(\boldsymbol{X}_k(i,j) - F(i - d_k^h, j - d_k^w)\right) +$$

$$+ \sum_{(i,j)\notin faceArea(\boldsymbol{d}_k)} (\boldsymbol{X}_k(i,j) - B(i,j))\Bigg)$$

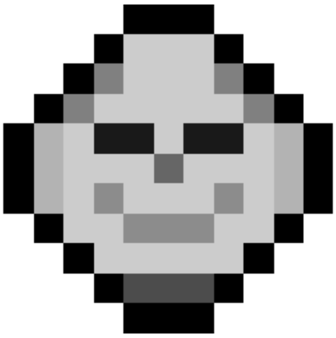
Множитель HWK получается при суммировании по всем (i,j) и умножении на $\sum_k \sum_{\boldsymbol{d}_k} q(\boldsymbol{d}_k) = \text{K}$.

MAP-EM оценка выписывается аналогично.

2 Анализ

2.1 Исходные данные

Данные, используемые далее для экспериментов, приведены ниже.



(a) Лицо. Размер: 11×13

(b) Фон. Размер: 20×30

Рис. 1: Исходные данные

2.2 Влияние начального приближения

Выясним, сильно ли влияет начальное приближение на параметры на результаты работы алгоритма. Для этого, зафиксируем количество изображений $K = 500$ и стандаратное отклонение шума $s = 125$ и посмотрим на выдаваемые результаты.

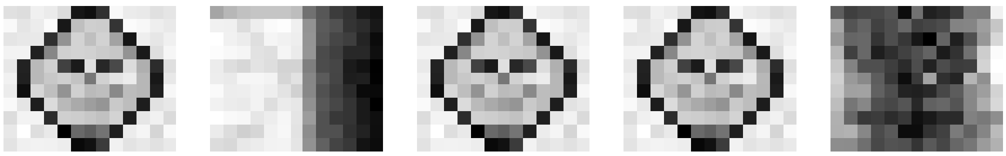


Рис. 2: Результаты работы алгоритма: лица

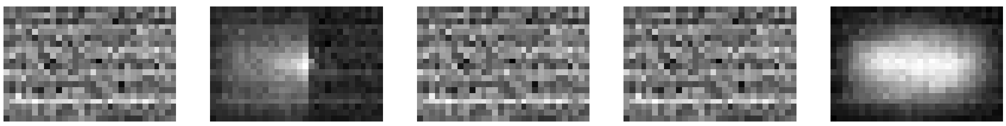


Рис. 3: Результаты работы алгоритма: фоны

Можно заметить, что вывод не всегда тот, который хотелось бы увидеть, поэтому для данной задачи стоит запускать EM-алгоритм из разных начальных приближений.

2.3 Влияние размера выборки и уровня шума

Выберем следующие шумы $s = 125, 250, 500$ и рассмотрим результаты (результатом в данном случае будем считать наилучший вывод при 5 запусках) при разных размерах выборок $K = 100, 250, 500, 750, 1000$.

Лица на рисунках ниже изображены сверху, фоны – снизу.

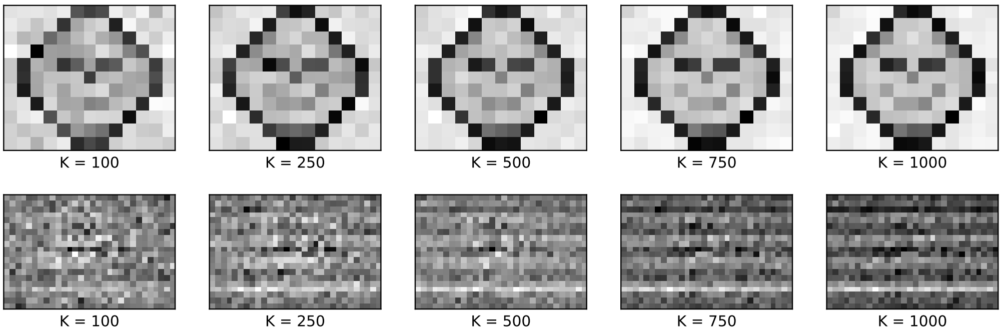


Рис. 4: $s = 125$

Для $s = 125$ алгоритм уже неплохо справляется при количестве изображений, равном 250. При 100 изображениях достаточно хорошо видны черты лица.

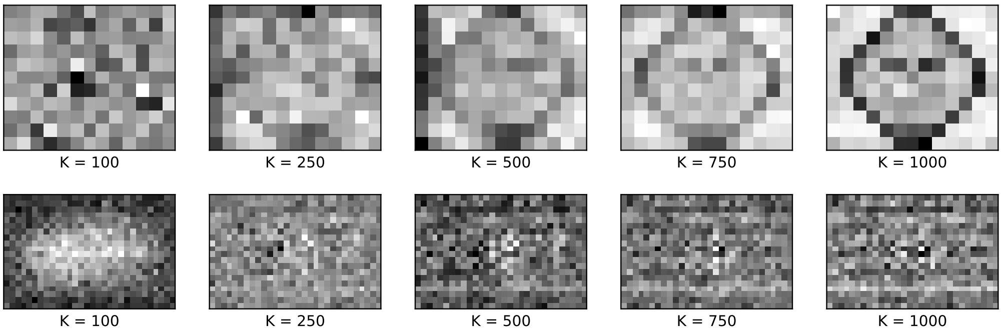


Рис. 5: $s = 250$

В случае с $s = 250$ для хорошего распознавания требуется выборка большей размерности, а именно: при 1000 – почти хорошо видно лицо, при 750 – слега начинают проявляться черты лица.

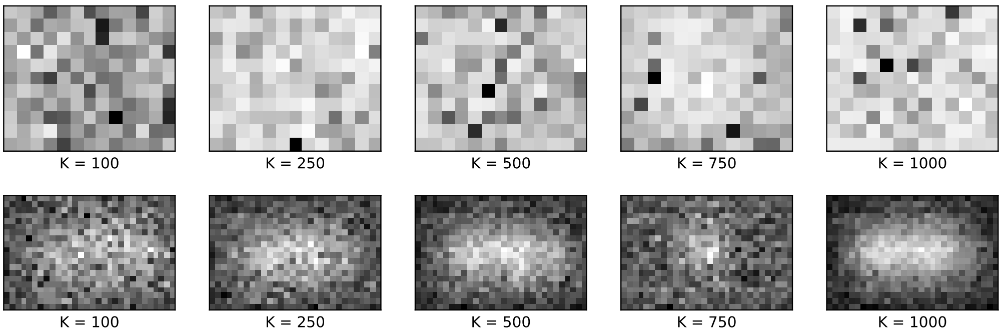


Рис. 6: $s = 500$

Однако, при $s = 500$ алгоритм уже не справляется с данной задачей даже при $K = 1000$. Видно, что получаемый результат невозможно интерпретировать.

Аналогичные утверждения справедливы и для фона.

Отсюда можно сделать вывод, что при ограниченном количестве изображений и достаточно большом уровне шума (в данном случае этот порог лежит в диапазоне $(250, 500]$) EM-алгоритм перестает выдавать вменяемые результаты.

Рассмотрим как изменения в обучающей выборке влияют на нижнюю оценку на логарифм неполного правдоподобия $\mathcal{L}(q, \boldsymbol{\theta}, \boldsymbol{A})$. Для этого рассмотрим больше значений K при фиксированном уровне шума $s = 250$:

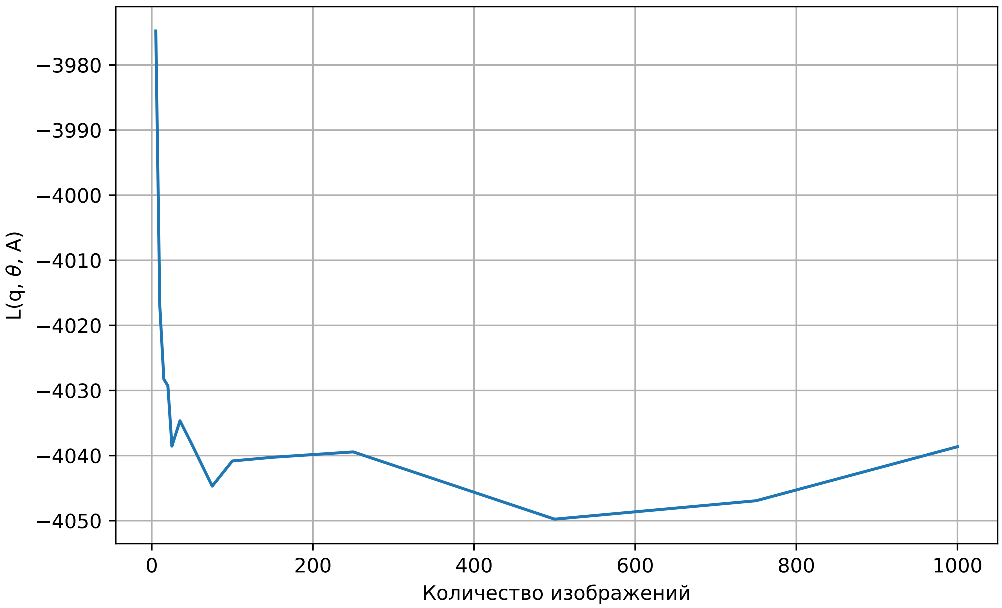


Рис. 7: Зависимость $\mathcal{L}(q, \boldsymbol{\theta}, \boldsymbol{A})$ от K при $s = 250$

Видно, что при достаточно малом количестве изображений $\mathcal{L}(q, \boldsymbol{\theta}, \boldsymbol{A})$ принимает высокие значения. Затем, с добавлением новых изображений, график выходит на плато, немного осциллируя.

2.4 EM и hard EM

Сравним качество и время работы EM и hard EM алгоритмов на сгенерированных данных при $K = 100, 250, 500, 750, 1000$ и $s = 100, 150$. Усреднение по времени производем по 10 запускам.

$s = 100$	$K = 100$	$K = 250$	$K = 500$	$K = 750$	$K = 1000$
EM	0.2322	0.6563	1.3086	1.8351	2.6202
hard EM	0.1070	0.3389	0.6183	0.8699	1.3096
$s = 150$	$K = 100$	$K = 250$	$K = 500$	$K = 750$	$K = 1000$
EM	0.2331	0.6265	1.2933	1.8785	2.6763
hard EM	0.1182	0.3129	0.7594	0.8701	1.2400

Таблица 1: Время работы EM и hard EM при разных s и K

Можно заметить, что во всех случаях hard EM работает быстрее, чем EM. Связано это с тем, что в hard EM участвует не все апостериорное распределение, а лишь одна точка – в которой достигается максимум вероятности, следовательно, уменьшается количество операций.

Рассмотрим результаты, выдаваемые алгоритмами. Сверху – выводы EM, снизу – hard EM.

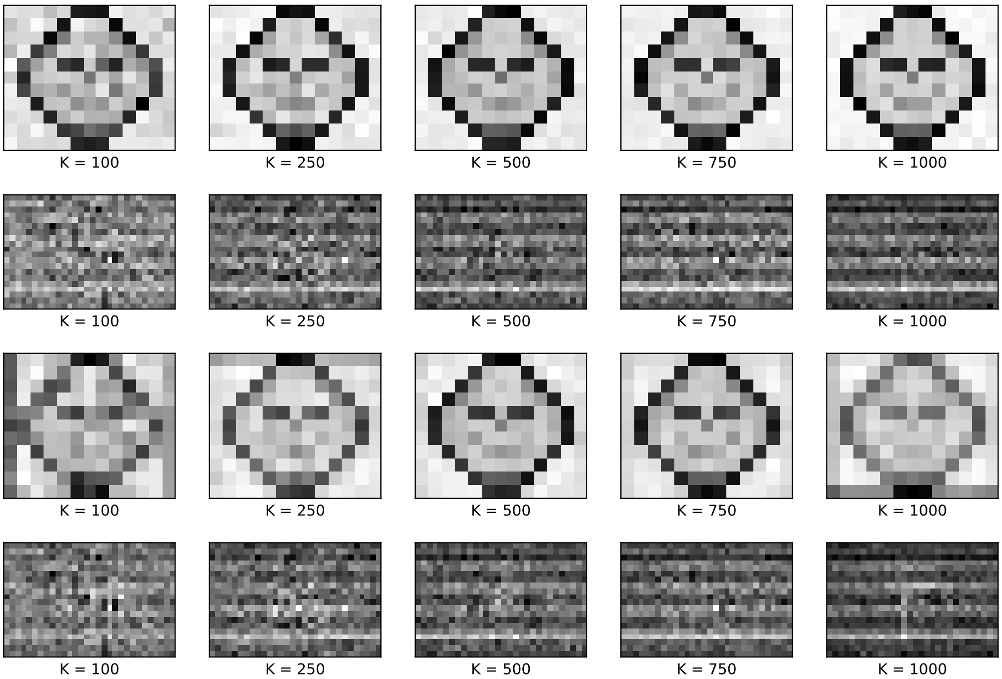


Рис. 8: $s = 100$

При $s = 100$ оба алгоритма достаточно неплохо справились со своей задачей, однако качество EM все-таки лучше.

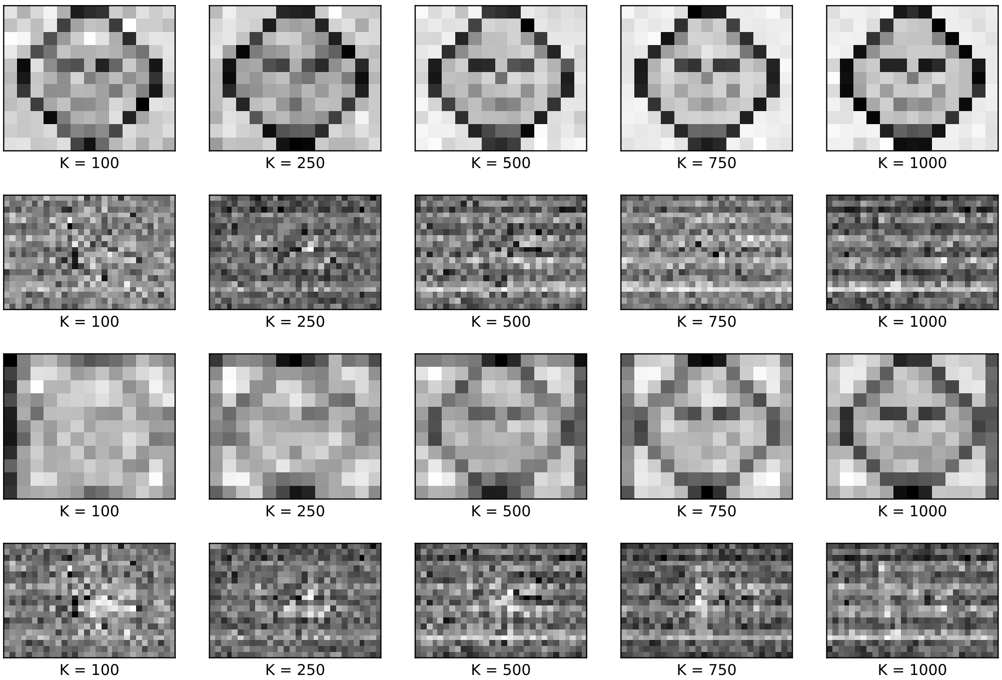


Рис. 9: $s = 100$

При $s = 150$ EM-алгоритм так же хорошо справляется, как и при $s = 100$, в то время, как качество hard EM понизилось.

Отсюда можно сделать вывод, что качество EM-алгоритма лучше, чем качество hard EM (особенно это видно при большом уровне шума), однако второй алгоритм работает быстрее первого.

2.5 Выявление преступника

Применим EM-алгоритм к данным с зашумленными снимками преступника при разных значениях размера выборки K:

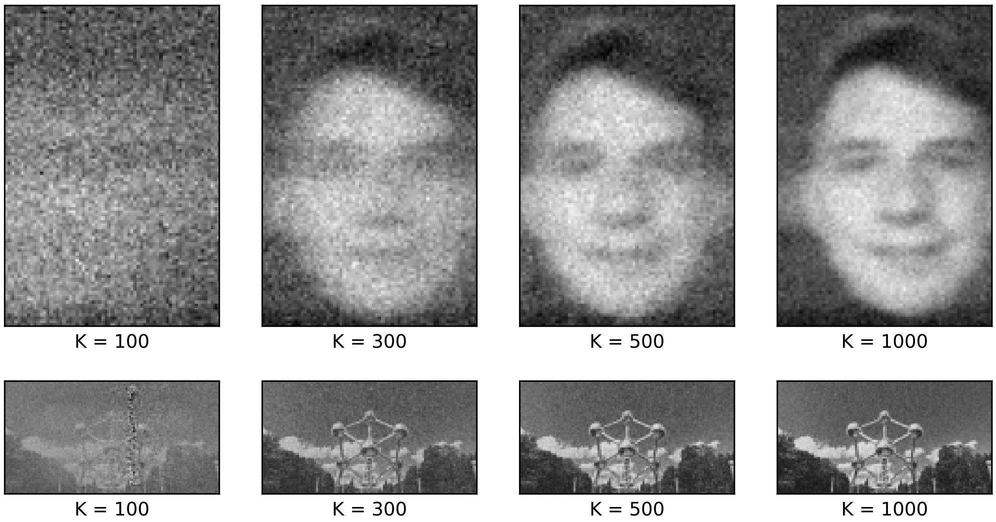


Рис. 10: Преступник и место происшествия

Вероятно, преступником является Сергей Трошин из Bayes Group.

2.6 Модификация алгоритма

2.6.1 Модификация 1: ускорение работы

В качестве модификации полученного EM-алгоритма можно предложить следующее: после E-шага преобразовать $q(\boldsymbol{d})$ так, что для каждого изображения $\boldsymbol{X}_k$ оценка $q(\boldsymbol{d}_k)$ принимает значение, не равное нулю, только в N точках - точках, в которых апостериорное распределение $p(\boldsymbol{d}_k \mid \boldsymbol{X}_k, \boldsymbol{\theta}, \boldsymbol{A})$ принимает наибольшие значения, а в $q(\boldsymbol{d})$ эти вероятности будут пропорциональны исходным вероятностям.

В таком случае, алгоритм должен работать быстрее EM, но медленнее hard EM, однако его качество будет лучше hard EM.

N является гиперпараметром и настраивается отдельно для каждой задачи.

2.6.2 Модификация 2: улучшение качества

Также, для улучшения работы алгоритма, можно воспользоваться специальными фильтрами, которые очищают изображение от шума: медианный фильтр, фильтры, управляющие величиной коррекции и т.д.

Воспользоваться этим можно следующим образом: во время работы алгоритма при получении очередного параметра F и B, применять к нему соответствующий фильтр и подавать на следующую итерацию. Однако, делать это надо после какой-то определенной итерации, чтобы не потерять основную информацию на первых этапах.

Возможно, данная модификация сможет повысить качество EM-алгоритма, хоть, и очевидно, работать она будет дольше.