Московский Государственный Университет

# EM алгоритм для детектива

Выполнил: Курцев Д.В.

Группа: 417

Факультет Вычислительной математики и кибернетики

Кафедра Математических методов прогнозирования

Ноябрь 2022

# Постановка задачи

Дана выборка $\boldsymbol{X} = \{\boldsymbol{X}_k\}_{k=1}^K$ сильно зашумленных черно-белых изображений размера $H \times W$ пикселей. Каждое из этих изображений содержит один и тот же неподвижный фон и лицо преступника в неизвестных координатах, при этом лицо попадает в любое изображение целиком. Будем считать, что изоражение лица имеет прямоугольную форму размера $h \times w$ пикселей.

Также пусть шум на изображении независимым для каждого пикселя и принадлежащим нормальному распределению $\mathcal{N}(0, s^2)$, где $s$ — стандартное отклонение. Таким образом для одного изображения имеем:

$$p(\boldsymbol{X}_k \mid \boldsymbol{d}_k, \boldsymbol{\theta}) = \prod_{ij} \begin{cases} \mathcal{N}(\boldsymbol{X}_k(i,j) \mid \boldsymbol{F}(i - d_k^h, j - d_k^w), s^2), & \text{если } (i,j) \in faceArea(\boldsymbol{d}_k) \\ \mathcal{N}(\boldsymbol{X}_k(i,j) \mid \boldsymbol{B}(i,j), s^2), & \text{иначе} \end{cases},$$

где $\boldsymbol{\theta} = \{\boldsymbol{B}, \boldsymbol{F}, s^2\}$, $faceArea(\boldsymbol{d}_k) = \{(i,j) \mid d_k^h \le i \le d_k^h + h - 1, d_k^w \le j \le d_k^w + w - 1\}$.

Распределение на неизвестные координаты лица на изображении зададим общим для всех изображений с помощью матрицы параметров $\boldsymbol{A} \in \mathbb{R}^{H-h+1, W-w+1}$ следующим образом:

$$p(\boldsymbol{d}_k \mid \boldsymbol{A}) = \boldsymbol{A}(d_k^h, d_k^w), \qquad \sum_{ij} A(i,j) = 1,$$

где $\boldsymbol{A}(i,j)$ — элемент матрицы $\boldsymbol{A}$.

В итоге имеем следующую совместную вероятностную модель:

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

Требуется решить задачу

$$p(\boldsymbol{X} \mid \boldsymbol{\theta}, \boldsymbol{A}) \to \max_{\boldsymbol{\theta}, \boldsymbol{A}}.$$

Для этого предлагается воспользоваться EM-алгоритмом, то есть перейти к следующей задаче оптимизации нижней оценки на логарифм неполного правдоподобия:

$$\mathcal{L}(q, \boldsymbol{\theta}, \boldsymbol{A}) = \mathbb{E}_{q(\boldsymbol{d})} \log p(\boldsymbol{X}, \boldsymbol{d} \mid \boldsymbol{\theta}, \boldsymbol{A}) - \mathbb{E}_{q(\boldsymbol{d})} \log q(\boldsymbol{d}) \to \max_{q, \boldsymbol{\theta}, \boldsymbol{A}}$$

$$\mathcal{L}(q, \boldsymbol{\theta}, \boldsymbol{A}) = \sum_{\boldsymbol{d}} q(\boldsymbol{d}) \log p(\boldsymbol{X}, \boldsymbol{d} \mid \boldsymbol{\theta}, \boldsymbol{A}) - \sum_{\boldsymbol{d}} q(\boldsymbol{d}) \log q(\boldsymbol{d})$$

# E-шаг

На E-шаге вычисляется оценка на апостериорное распределение на координаты лица на изображениях:

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

По формуле Байеса имеем:

$$p(\boldsymbol{d_k} \mid \boldsymbol{X_k}, \boldsymbol{\theta}, \boldsymbol{A}) = \frac{p(\boldsymbol{X_k}|\boldsymbol{d_k}, \boldsymbol{\theta}, \boldsymbol{A})\, p(\boldsymbol{d_k} \mid \boldsymbol{A})}{\sum\limits_{d_k^h} \sum\limits_{d_k^w} p(\boldsymbol{X_k}|\boldsymbol{d_k}, \boldsymbol{\theta}, \boldsymbol{A})\, p(\boldsymbol{d_k} \mid \boldsymbol{A})}$$

# M-шаг

На M-шаге вычисляется точечная оценка на параметры $\boldsymbol{\theta}, \boldsymbol{A}$:

$$\mathbb{E}_{q(\boldsymbol{d})} \log p(\boldsymbol{X}, \boldsymbol{d} \mid \boldsymbol{\theta}, \boldsymbol{A}) \to \max_{\boldsymbol{\theta}, \boldsymbol{A}}.$$

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

$$= \mathbb{E}_{q(\boldsymbol{d})} \sum_k (\log p(\boldsymbol{X}_k \mid \boldsymbol{d}_k, \boldsymbol{\theta}) + \log p(\boldsymbol{d}_k \mid \boldsymbol{A})) =$$

$$\sum_{\boldsymbol{d_k}} \sum_k q(\boldsymbol{d_k}) (\log p(\boldsymbol{X}_k \mid \boldsymbol{d}_k, \boldsymbol{\theta}) + \log p(\boldsymbol{d}_k \mid \boldsymbol{A})) \quad (*)$$

Выведем точечные оценки на параметры. Для этого продифференцируем полученное выражение и приравняем производные к нулю.

## A

Так как у нас имеется ограничение на $\boldsymbol{A}$: $\sum\limits_{i,j} A(i,j) = 1$, запишем лагранжиан. $p(\boldsymbol{X}_k \mid \boldsymbol{d}_k, \boldsymbol{\theta})$ не зависит от $\boldsymbol{A}$, поэтому не будем его писать, так как данное правдоподобие уйдёт при дифференцировании.

$$L = \sum_{\boldsymbol{d}} \sum_k q(\boldsymbol{d_k}) \log \boldsymbol{A}(\boldsymbol{d_k}) - \lambda (\sum_{\boldsymbol{d}} \boldsymbol{A}(\boldsymbol{d}) - 1)$$

$$\frac{\partial L}{\partial A(\boldsymbol{d})} = \frac{\sum\limits_k q(\boldsymbol{d_k})}{A(\boldsymbol{d})} - \lambda = 0 \quad \Rightarrow \quad A(\boldsymbol{d}) = \frac{\sum\limits_k q(\boldsymbol{d_k})}{\lambda}$$

$$\frac{\partial L}{\lambda} = \sum_{\boldsymbol{d}} \boldsymbol{A}(\boldsymbol{d}) - 1 = 0 \quad \Rightarrow \quad \sum_{\boldsymbol{d}} \boldsymbol{A}(\boldsymbol{d}) = 1$$

Тогда получим, что

$$1 = \sum_{\boldsymbol{d}} \boldsymbol{A}(\boldsymbol{d}) = \sum_{\boldsymbol{d}} \frac{\sum\limits_k q(\boldsymbol{d_k})}{\lambda} \quad \Rightarrow \quad \lambda = \frac{1}{\sum\limits_{\boldsymbol{d}} \sum\limits_k q(\boldsymbol{d_k})} = \frac{1}{\sum\limits_k \sum\limits_{\boldsymbol{d}} q(\boldsymbol{d_k})} = \frac{1}{\sum\limits_k 1} = \frac{1}{K}$$

Таким образом оценка на $\boldsymbol{A}$ равна:

$$\boldsymbol{A}(\boldsymbol{d}) = \frac{1}{K} \sum_k q(\boldsymbol{d_k})$$

## F

Аналогично $\boldsymbol{A}$, в (*) второе слагаемое не зависит от оставшихся параметров, поэтому для сокращения записи не будем его писать. Получим следующее выражение:

$$\sum_{\boldsymbol{d}} \sum_k q(\boldsymbol{d_k}) \log p(\boldsymbol{X}_k \mid \boldsymbol{d}_k, \boldsymbol{\theta}) = \sum_{\boldsymbol{d}} \sum_k q(\boldsymbol{d_k}) \sum_{i,j} \log \mathcal{N}(\boldsymbol{X}_k(i,j) \mid \mu_{ij}, s^2) =$$

$$= \sum_{\boldsymbol{d}} \sum_k q(\boldsymbol{d_k}) \sum_{i,j} [-\frac{1}{2} \log 2\pi s^2 - \frac{1}{2s^2} (\boldsymbol{X}_k(i,j) - \boldsymbol{B}(i,j))^2 \mathbb{1}\{i,j \notin face\} -$$

$$-\frac{1}{2s^2} (\boldsymbol{X}_k(i,j) - \boldsymbol{F}(i - d_k^h, j - d_k^w))^2 \mathbb{1}\{i,j \in face\}\,] \quad (**)$$

Для начала продифференцируем (**) по $\boldsymbol{F}$:

$$\frac{\partial (**)}{\partial \boldsymbol{F}(i,j)} = \sum_{\boldsymbol{d}} \sum_k q(\boldsymbol{d_k}) \frac{1}{2s^2}\, 2\, (\boldsymbol{X}_k(i + d_k^h, j + d_k^w) - \boldsymbol{F}(i,j)) \boldsymbol{F}(i,j) = 0 \quad \Rightarrow$$

$$\Rightarrow \quad \sum_{\boldsymbol{d}} \sum_k q(\boldsymbol{d_k}) \boldsymbol{F}(i,j) = \sum_{\boldsymbol{d}} \sum_k q(\boldsymbol{d_k}) \boldsymbol{X}_k(i + d_k^h, j + d_k^w)$$

Таким образом оценка на $\boldsymbol{F}$ равна:

$$\boldsymbol{F}(i,j) = \frac{1}{K} \sum_{\boldsymbol{d}} \sum_k q(\boldsymbol{d_k}) \boldsymbol{X}_k(i + d_k^h, j + d_k^w)$$

## B

Теперь найдём оценку на $\boldsymbol{B}$:

$$\frac{\partial(**)}{\partial \boldsymbol{B}(i,j)} = \sum_{\boldsymbol{d}} \sum_{k} q(\boldsymbol{d_k}) \frac{1}{2s^2} \, 2 \, (\boldsymbol{X}_k(i,j) - \boldsymbol{B}(i,j)) \boldsymbol{B}(i,j) \mathbb{1}\{i,j \notin face\} = 0 \quad \Rightarrow$$

$$\Rightarrow \quad \sum_{\boldsymbol{d}} \sum_{k} q(\boldsymbol{d_k}) \boldsymbol{B}(i,j) \mathbb{1}\{i,j \notin face\} = \sum_{\boldsymbol{d}} \sum_{k} q(\boldsymbol{d_k}) \boldsymbol{X}_k(i,j) \mathbb{1}\{i,j \notin face\}$$

Таким образом оценка на $\boldsymbol{B}$ равна:

$$\boldsymbol{B}(i,j) = \frac{\sum\limits_{\boldsymbol{d}} \sum\limits_{k} q(\boldsymbol{d_k}) \boldsymbol{X}_k(i,j) \mathbb{1}\{i,j \notin face\}}{\sum\limits_{\boldsymbol{d}} \sum\limits_{k} q(\boldsymbol{d_k}) \mathbb{1}\{i,j \notin face\}}$$

## S

Теперь найдём оценку на s:

$$\frac{\partial(**)}{\partial s} = \sum_{\boldsymbol{d}} \sum_{k} q(\boldsymbol{d_k}) \sum_{i,j} [-\frac{1}{s} + \frac{1}{s^3} (\boldsymbol{X}_k(i,j) - \boldsymbol{B}(i,j))^2 \mathbb{1}\{i,j \notin face\} -$$

$$+ \frac{1}{s^3} (\boldsymbol{X}_k(i,j) - \boldsymbol{F}(i - d_k^h, j - d_k^w))^2 \mathbb{1}\{i,j \in face\} \, ] =$$

$$= \frac{1}{s}(-HWK + \frac{1}{s^2}) \sum_{\boldsymbol{d}} \sum_{k} \sum_{i,j} q(\boldsymbol{d_k}) [(\boldsymbol{X}_k(i,j) - \boldsymbol{B}(i,j))^2 \mathbb{1}\{i,j \notin face\} +$$

$$(\boldsymbol{X}_k(i,j) - \boldsymbol{F}(i - d_k^h, j - d_k^w))^2 \mathbb{1}\{i,j \in face\}] = 0$$

Таким образом оценка на $s$ равна:

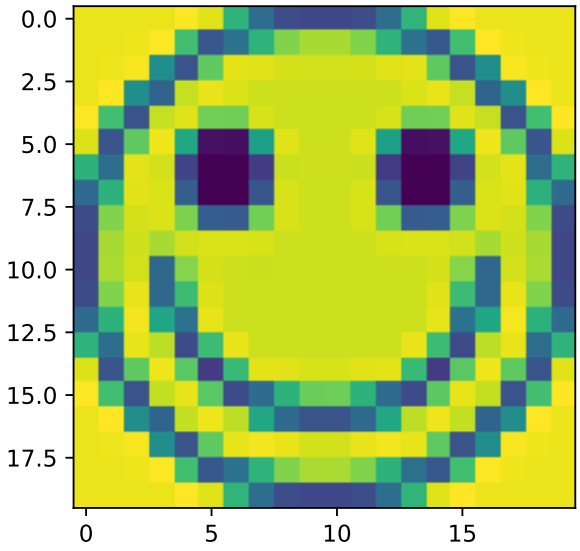
$$s^2 = \frac{1}{HWK} \sum_{\boldsymbol{d}} \sum_{k} \sum_{i,j} q(\boldsymbol{d_k}) [(\boldsymbol{X}_k(i,j) - \boldsymbol{B}(i,j))^2 \mathbb{1}\{i,j \notin face\} +$$

$$(\boldsymbol{X}_k(i,j) - \boldsymbol{F}(i - d_k^h, j - d_k^w))^2 \mathbb{1}\{i,j \in face\}]$$

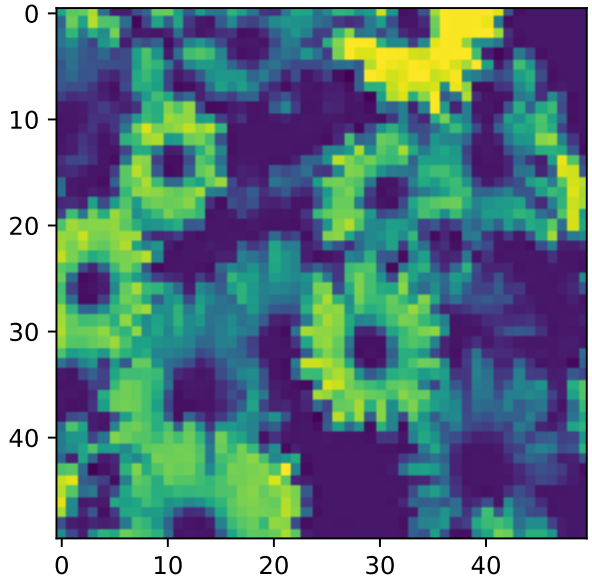
# Эксперименты

Посмотрим, как работает написанный алгоритм. Для этого возьмём произвольный объект - смайлик, который необходимо будет обнаружить на фоне цветочков.

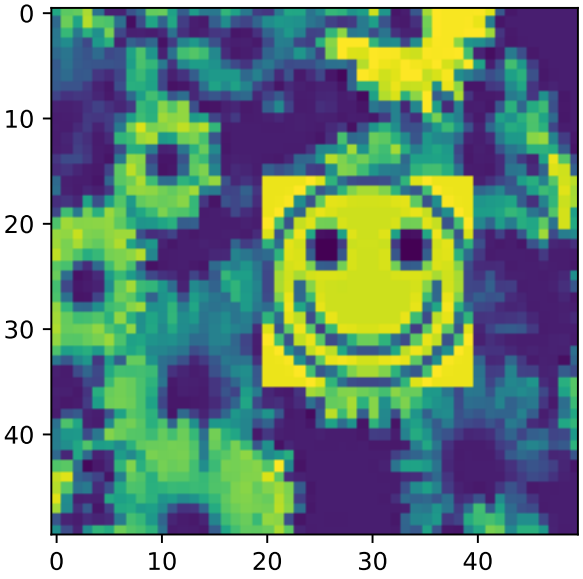
Создадим выборку из размещённых в разных местах на фоне цветочков смайликов. Пример одной картинки с шумом и без представлен ниже.



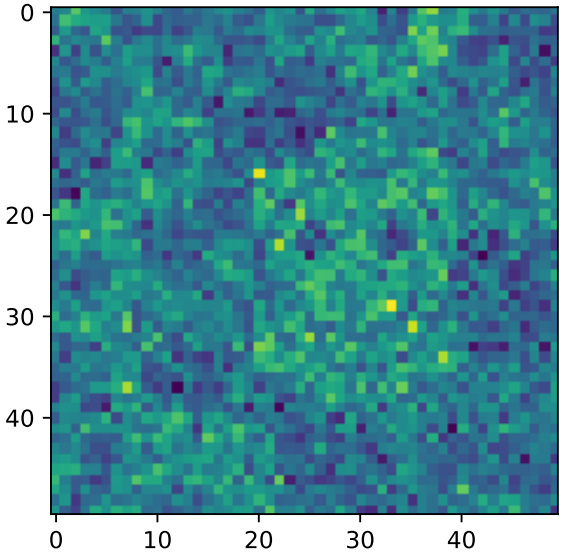

(a) Объект




(b) Фон




(a) Объект на фоне



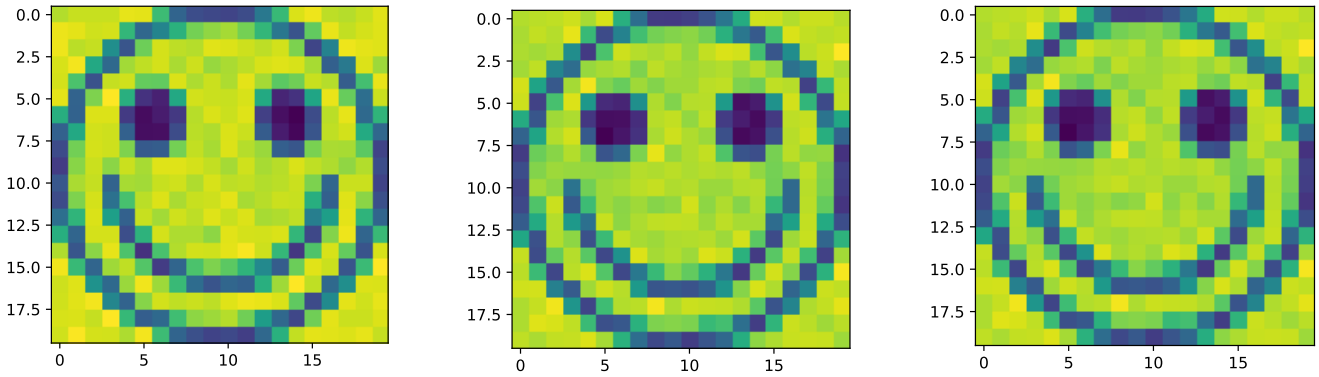

(b) Зашумлённый объект на фоне

Для начала посмотрим, как на результат работы влияет начальная инициализация параметров алгоритма. Будем использовать 3 стратегии: равномерное на отрезке от нуля до наибольшего значения из выборки - $\mathcal{U}[0, \max \boldsymbol{X}_k(i,j)]$, нормальное, где мат ожиданием является среднее значение из выборки с дисперсией 50 - $\mathcal{N}(mean\boldsymbol{X}_k(i,j), 50)$ и просто константым приближением - среднее. Матрицу $\boldsymbol{A}$ во всех случаях отнормируем на сумму.

Для обучения возьмём выборку из 100 изображений зашумлённых нормальных шумом с $s = 100$.

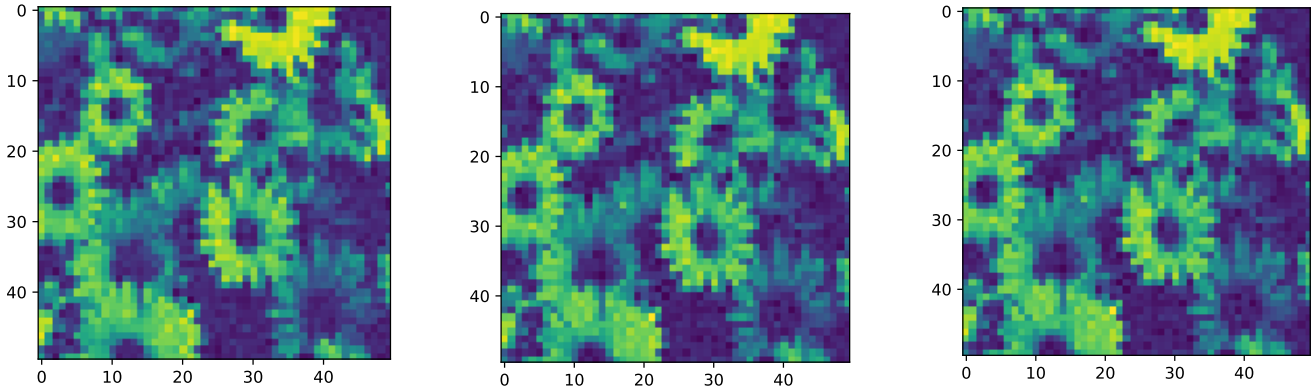
Результаты работы алгорима можно увидеть на Рис. 3 и 4

Как можно заметить, результат работы метода практически не зависит от



(a) Равномерное (b) Нормальное (c) Среднее

Рис. 3: Результат EM для объекта



(a) Равномерное (b) Нормальное (c) Среднее

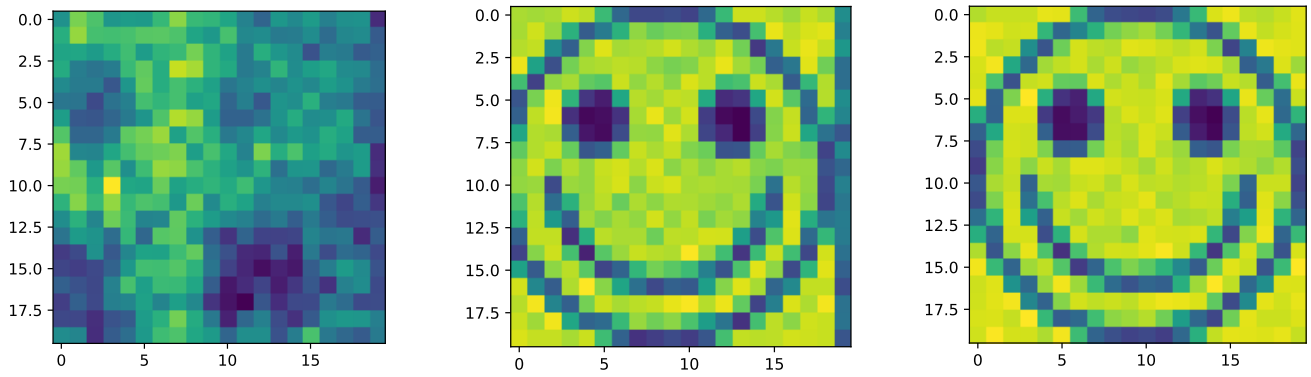
Рис. 4: Результат EM для фона

начальной инициализации. Неполное правдоподобие для равномерной инициализации чуть больше, чем для остальных двух (-1423467 против -1423942). Видно, что для первой стратегии результат получился чуть более точным, чем для оставшихся. Так же заметно, что последние 2 инициализации работают практически одинаково. Вероятно, это связано с тем, что для нормального распределения была выбрана не очень большая дисперсия, поэтому значения параметров не очень сильно отклонились от среднего значения.

Будем всюду далее использовать первую стратегию для задания начальных значений параметров.

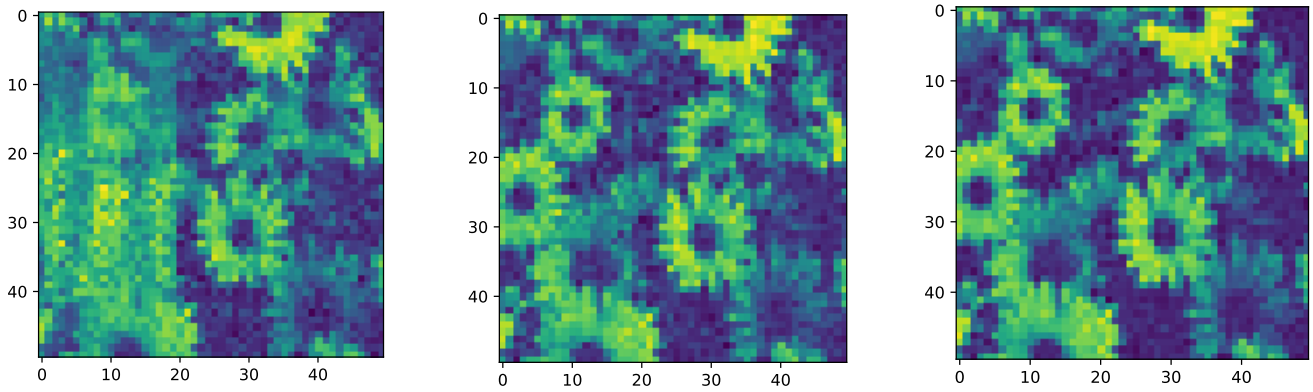
Теперь исследуем работу EM алгоритма в зависимости от размера обучающей выборки. Возьмём 30, 50 и 100 зашумлённых изображений.

Результаты работы метода приведены на Рис. 5 и 6. Можно сделать вполне логичный вывод, что чем больше обучающая выборка, тем точнее работает



(a) $K = 30$ (b) $K = 50$ (c) $K = 100$

Рис. 5: Результат EM для объекта



(a) $K = 30$ (b) $K = 50$ (c) $K = 100$

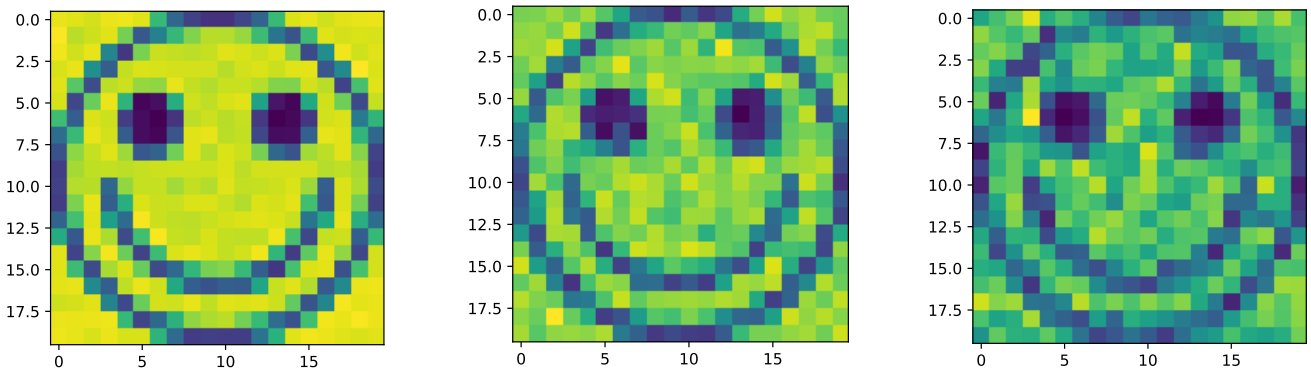
Рис. 6: Результат EM для фона

алгоритм. Видно, что на объёме из 30 изображений, вообще невозможно определить объект и часть фона. Возможно в обучении состояли картинки, на которых смайлик находился на левой стороне изображения. Для выборки объёма 50 EM уже вполне хорошо справляется. Однако полученные картинки чуть более размыты, чем для случая, когда в обучении использовалось 100 изображений.

Нормированное на объём выборки правдоподобие равняется $\mathcal{L}_{30} = -14403.3$, $\mathcal{L}_{50} = -14239.2, \mathcal{L}_{100} = -14234.6$. Видно, что чем лучше алгоритм справляется с задачей, тем выше правдободобие.

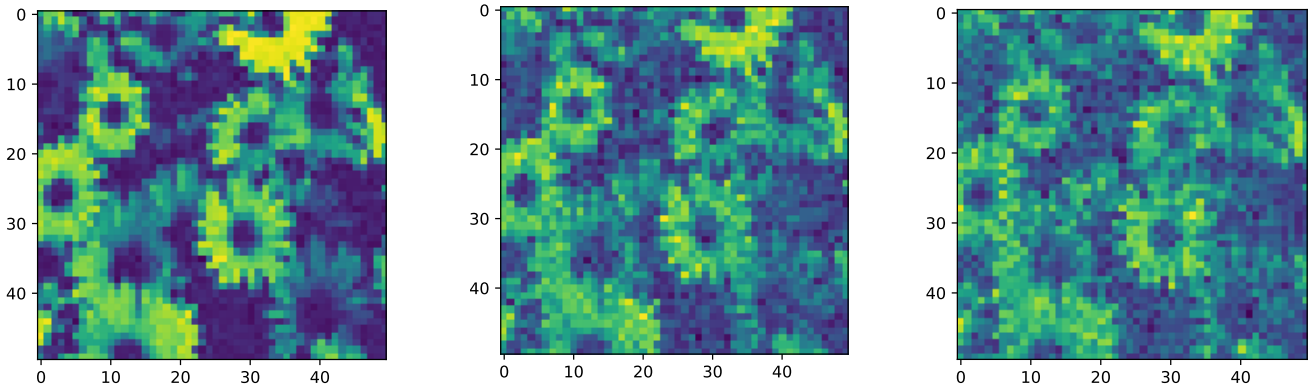
Далее посмотрим, как на качество влияет зашумлённость изображений. Раннее рассматривался шум с s=100. Теперь возьмём обучающую выборку размером из 50 картинок с уровнем зашумлённости $s \in \{50, 150, 200\}$

Результаты приведены на Рис. 7 и 8. Как можно заметить, чем больше шума в данных, тем сложнее алгоритму найти верное решение. Смайлик и фон с



(a) $s = 50$ (b) $s = 150$ (c) $s = 200$

Рис. 7: Результат EM для объекта



(a) $s = 50$ (b) $s = 150$ (c) $s = 200$

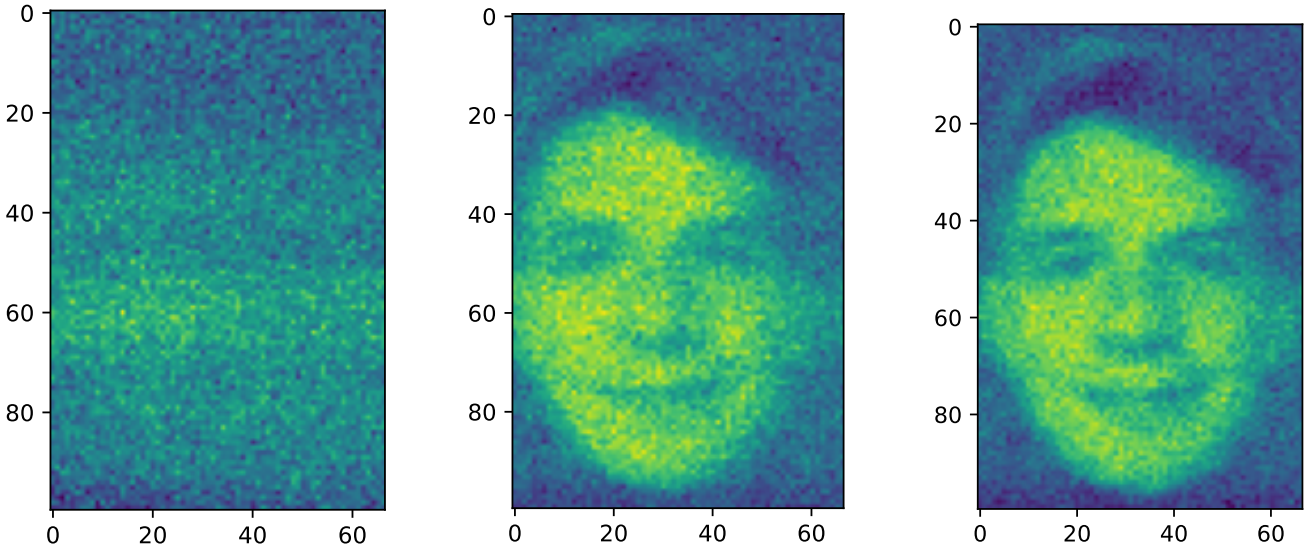
Рис. 8: Результат EM для фона

$s = 200$, получаются очень размытыми. Зато для $s = 50$ картинки получаются гораздо чётче, чем был раннее.

Таким образом можно заключить, что чем больше наши данные зашумлены, тем больше должен быть объём обучающей выборки, для корректной работы EM алгоритма

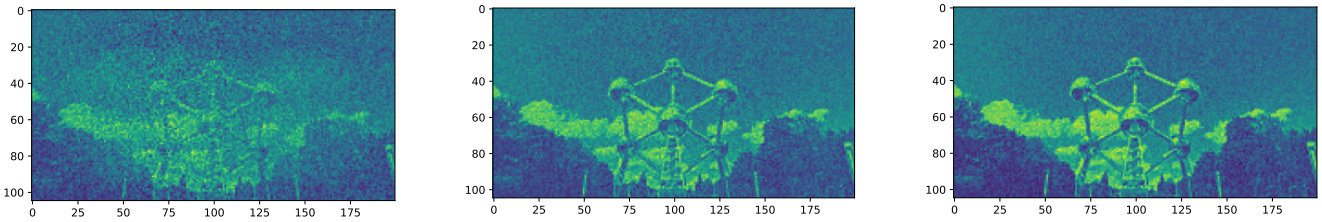
Теперь применим данный алгоритм для зашумлённых снимков преступника. Возьмём данные объёма 100, 300 и 500 изображений.

Результаты представлены на Рис. 9 и 10. Как и раннее здесь можно сделать аналогичный вывод. Чем больше изображений использовалось в обучении, тем точнее получается результат. Маленького объёма обучающей выборки не хватает для того, чтобы найти нужные закономерности в данных и оценить параметры.



(a) $K = 100$ (b) $K = 300$ (c) $K = 500$

Рис. 9: Результат EM для фото

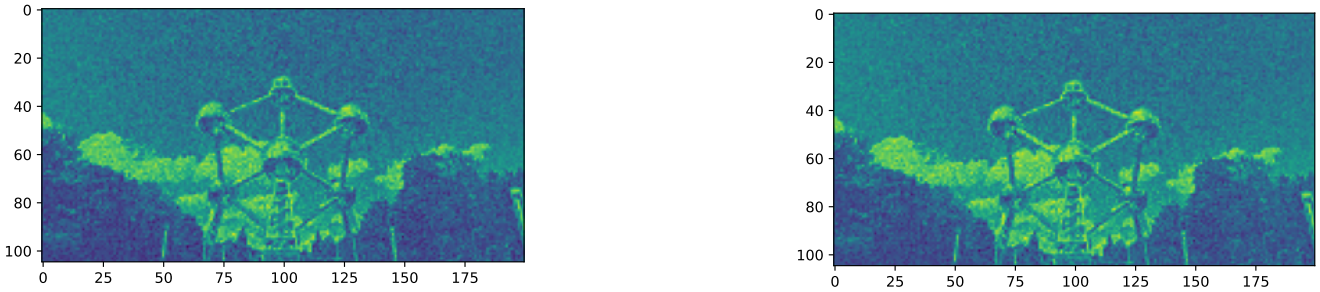


(a) $K = 100$ (b) $K = 300$ (c) $K = 500$

Рис. 10: Результат EM для фона

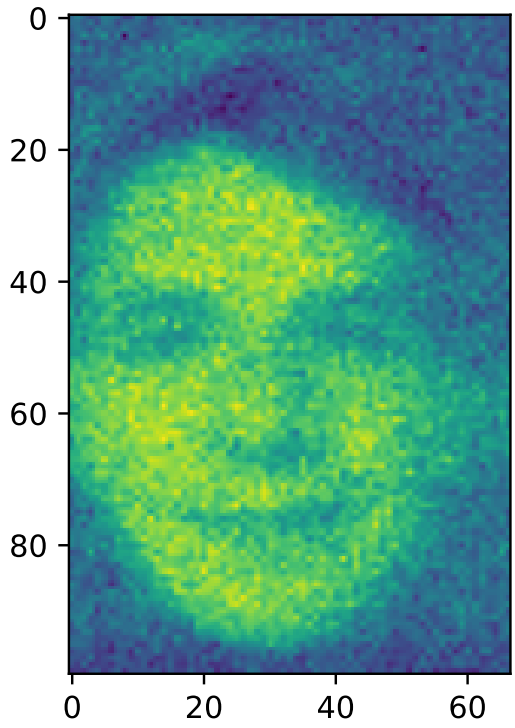
Нормированное на объём выборки правдоподобие равняется $\mathcal{L}_{100} = -128937.8$, $\mathcal{L}_{300} = -128874.7, \mathcal{L}_{500} = -128887.6$. Опять же видим, что чем больше объём выборки, тем больше правдоподобие и соответственно качество изображений.

Посмотрим на качество и время работы EM и hard EM этих данных. (На сгенерированных результаты получаются практически одинаковыми).

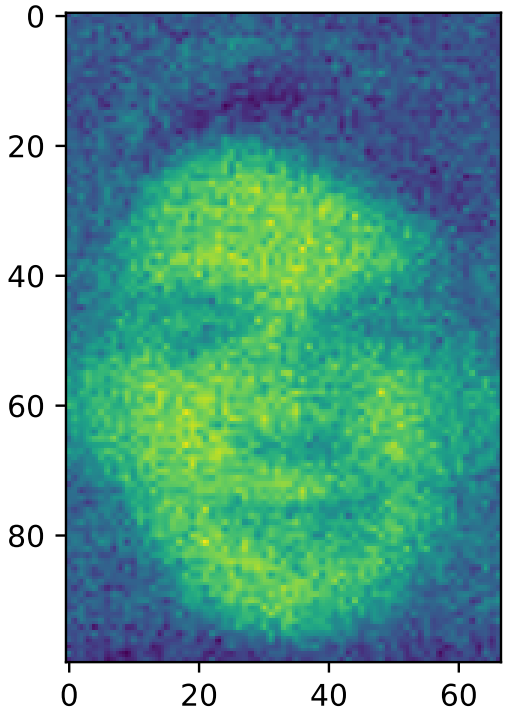


(a) Not MAP (b) MAP

Рис. 11: Результат EM для фона




(a) Not MAP




(b) MAP

Рис. 12: Результат EM для фото

Как можно заметить из Рис. 11 и 12 классическая реализация EM алгоритма работает чуть лучше. Его модернизация (hard EM) даёт более размытое изображение. Это можно объяснить тем, что мы делаем грубую аппроксимацию на латентные переменные. В связи с чем падает качество (правдободобие меньше): $\mathcal{L}_{EM} = -128874.7$, $\mathcal{L}_{hard} = -128877.2$. Однако hard EM при сопоставимом качестве работы сходится гораздо быстрее: $t_{EM} = 1min\,48sec$, $t_{hard} = 22sec$.

Из модификаций можно предложить выход не по $\mathcal{L}$, а по норме $F$. То есть в качестве критерия останова использовать следующее условие: $||F^{(t+1)} - F^{(t)}|| \leq tol$.

Так же усовершенствуя идею hard EM, можно выбирать не 1 точку, а k наиболее вероятных точек и преобразовать $q(\boldsymbol{d}_k)$ так, чтобы вся вероятностная масса была сосредоточена в них.

P.s. Ник очень невнимательный. Максимально невнимательный...