

Компьютерное зрение '2014

Who? Александр Вахитов

When? March 28, 2014

План лекции

Модель,
параметры и
данные

Фильтрация
изображений

Морфология

Оптимизация

Задача оптимизации:

$$F(x) \rightarrow \min$$

Удобно думать и применять

Пример

Моя программа ...

- снижает ошибку измерения расстояния
- оценивает истинное положение линии квадратного маркера на изображении
- ищет наиболее вероятное значение пикселя в идеальном (без шума) изображении
- определяет наиболее вероятное положение человека по оптическому потоку
- считает количество проехавших мотоциклов в данном направлении

Модель

Дано: наблюдение y

Необходимо: найти (скрытые, ненаблюдаемые) параметры модели x , которые позволяют решить задачу, используя наблюдения y

$$F(x) = \text{distance}(y - M(x)) \rightarrow \min_x$$

Здесь $M(x)$ - модель, зависящая от параметров x

Пример

- измерение расстояния по изображению: x - положения точек на фотографиях, координаты точек и направлений съемки
- положение маркера на изображении: x - положение маркера, параметры модели смазывания (затенения) маркера
- фильтрация изображения: x - представление изображения в некотором базисе
- положение человека: x - размер, положение человека в 3D
- подсчет количества мотоциклов: x - положения каждого мотоцикла в каждый момент времени, в 3D или в 2D

Как решить задачу оптимизации

От класса функции зависит сложность метода решения

Виды задач:

- выпуклая

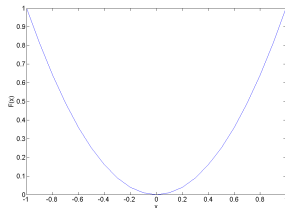
Нужно формулировать задачу в пригодном для решения виде

Как решить задачу оптимизации

От класса функции зависит сложность метода решения

Виды задач:

- выпуклая
- близкая к выпуклой



Нужно формулировать задачу в пригодном для решения виде

Как решить задачу оптимизации

От класса функции зависит сложность метода решения

Виды задач:

- выпуклая
- близкая к выпуклой

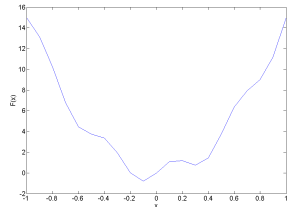
Нужно формулировать задачу в пригодном для решения виде

Как решить задачу оптимизации

От класса функции зависит сложность метода решения

Виды задач:

- выпуклая
- близкая к выпуклой
- сильно невыпуклая



Нужно формулировать задачу в пригодном для решения виде

Как решить задачу оптимизации

От класса функции зависит сложность метода решения

Виды задач:

- выпуклая
- близкая к выпуклой
- сильно невыпуклая

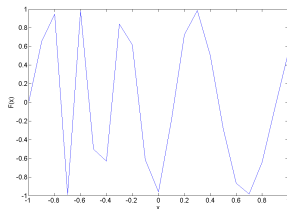
Нужно формулировать задачу в пригодном для решения виде

Как решить задачу оптимизации

От класса функции зависит сложность метода решения

Виды задач:

- выпуклая
- близкая к выпуклой
- сильно невыпуклая



Нужно формулировать задачу в пригодном для решения виде

Основные методы решения

- выпуклые: линейные (метод сопряженных градиентов) и нелинейные (градиентный и Ньютона)
- близкие к выпуклым: методы выпуклой оптимизации + сглаживание
- невыпуклые: отжиг, глобальный перебор...

“Проклятие” размерности: в общем случае, для того, чтобы гарантировать отличие оценки от истинного минимума на ϵ , нужно произвести перебор для $(D/\epsilon)^N$, где N - размерность, D - ширина интервала

Оптимизация в условиях помех

Пусть w, v - случайные величины (вектора). Тогда
Обычно наблюдение зависит от параметров и шумов:

$$y = g(x) + v$$

Иногда:

$$y = wg(x) + v$$

Общий случай:

$$y = g(x, w) + v$$

Оптимизация в условиях помех (2)

Наиболее частая постановка задачи:

$g(x)$ - наблюдение, предсказанное моделью

y - зашумленное наблюдение

$$\|g(x) - y\|^2 \rightarrow \min$$

y_* - истинное наблюдение

v - шум

$$y = y_* + v$$

Обычно, $Ev = 0$, $y \in \mathbb{R}^N$, $x \in \mathbb{R}^q$, $q \ll N$

Модель для компьютерного зрения

Модель формирования наблюдения

- формирование 3D (2D) сцены T
- проектирование на матрицу камеры P с параметрами p (положение, направление)
- оптическое искажение (дисторсия) D с параметрами d (радиальные, тангенциальные)
- размытие изображения B с параметрами b (степень размытия - дисперсия)
- дискретизация изображения Q

$$y = Q(B_b(D_d(P_p(T(x)))))) + v$$

$$x = ?$$

Улучшение качества изображений

- 1 Постановка задачи
- 2 Фильтр Винера
- 3 Гауссовский фильтр
- 4 Фильтр Ярославского (bilateral)
- 5 Анизотропный фильтр
- 6 NL-Means
- 7 BM3D

Основной источник: A non-local algorithm for image denoising (Buades, Coll, Morel, 2005)

Задача улучшения качества изображения

Определение

Одноканальное изображение $I(x, y)$, x, y - координаты пикселя,
 $(x, y) \in \mathbb{Z}^2 \cap [0, R]^2$, $I(x, y) \in \mathbb{Z} \cap [0, 255]$. Иногда будем обозначать $I(x)$, $x \in \mathbb{Z}^2$

$$I(x, y) = \bar{I}(x, y) + v(x, y),$$

- Наблюдаемое изображение $I(x, y)$
- Истинное изображение $\bar{I}(x, y)$
- Шум $v(x, y)$

Необходимо найти $\hat{I} \approx \bar{I}$

Задача восстановления изображения

$$I(x) = H(x) * \bar{I}(x) + v(x)$$

H - смазывание (размытие)

\bar{I} - исходное изображение

v - случайная помеха

I - результат

Необходимо найти $\hat{I} \approx \bar{I}$

Классификация задач

$$I(x) = H(x) * \bar{I}(x) + v(x)$$

- $H = Id$ - удаление шумов
- Deblurring (deconvolution): $H = B(\sigma)$ - оператор размытия, известен,
- Blind deconvolution: H неизвестен
- Super resolution: $H = DBW$, D - оператор понижения разрешения, B - размытие, W - геометрическое преобразование (сдвиг, поворот), известны, I - вектор из нескольких изображений

Модель появления изображения

Что такое \bar{I} - исходное изображение?

- 1 произвольный неизвестный сигнал
- 2 сигнал из какого-то известного пространства сигналов

Для случая 1), оптимален фильтр Винера

Для случая 2), зависит от конкретной формулировки

Фильтр Винера-Колмогорова в обработке изображений (1)

$$I(x, y) = H(x, y) * \bar{I}(x, y) + v(x, y)$$

Преобразование Фурье даст:

$$I_F(u, v) = H_F(u, v)\bar{I}_F(u, v) + v_F(u, v)$$

Если $v_F(u, v) = 0$, $\bar{I}_F = H^{-1}I_F$

Общий случай:

$$\hat{I}_F(u, v) = W_F(u, v)I_F(u, v),$$

$$W_F(u, v) = \frac{H_F^*(u, v)}{|H(u, v)|^2 + K(u, v)},$$

$$K(u, v) = \frac{S_I(u, v)}{S_v(u, v)} = \frac{|I_F(u, v)|^2}{E\{|N_F(u, v)|^2\}}$$

Обоснование фильтра Винера

$$\hat{I}_F = W_F H_F \bar{I}_F + W_F N_F$$

$$\bar{I}_F - \hat{I}_F = (1 - W_F H_F) \bar{I}_F - W_F N_F$$

$$\frac{\partial}{\partial W} E \|\bar{I}_F - \hat{I}_F\|^2 = 0$$

Для каждого (u, v) :

$$2(-(1 - W_F^* H_F^*) H_F |\bar{I}_F|^2 + W_F^* E\{|N_F|^2\}) = 0$$

$$W_F^* = \frac{H_F |\bar{I}_F|^2}{|H|^2 |\bar{I}_F|^2 + E\{|N_F|^2\}},$$

Известно $E\{N_F\}$, $\bar{I}_F(u, v) \approx I_F(u, v)$

$$W_F(u, v) = \frac{H_F^*(u, v)}{|H(u, v)|^2 + K(u, v)}.$$

Аппарат для сравнения фильтров

Определение

Шумом фильтра называется разность $\bar{I}(x) - F\bar{I}(x)$
где F обозначает применение фильтра.

В чем смысл определения?

Какая связь с концепцией “модель - данные”?

Гауссовский фильтр

$$\hat{I}(x) = \frac{1}{4\pi\sigma^2} \int e^{-\|y\|^2/(4\sigma^2)} I(x-y) dy$$

Theorem

(Габор, 1960) Шум гауссовского фильтра выражается как

$$\bar{I}(x) - F\bar{I}(x) = -\sigma^2 \Delta u + o(\sigma^2),$$

для достаточно малого σ .

$\Delta = \partial_x^2 + \partial_y^2$ - оператор Лапласа.

Фильтр Ярославского (bilateral)

Экспоненциальный вес:

$$c(t) = e^{-\|t\|^2}$$

Вес по расстоянию $c(u - x)$

Вес по интенсивности $c(I(x) - I(u))$

Комбинация весов:

$$\hat{I}(x) = \lambda \int_{-\infty}^{\infty} I(u) c(u - x) c(I(u) - I(x)) du$$

Анизотропный фильтр (Perona и Malik 1990)

$$\hat{I}(x) = \int G_{\sigma}(t) I(x + t \frac{\nabla I(x)^{\circ}}{|\nabla I(x)|}) dt,$$

при $\|\nabla I(x)\| \neq 0$, $(x, y)^{\circ} = (-y, x)$, G - гауссовская функция с дисперсией σ .

Theorem

Шум анизотропного фильтра выражается как:

$$I(x) - AFI(x) = -\frac{1}{2}\sigma^2 |\nabla I| \text{curv}(I)(x) + o(\sigma^2)$$

Total Variation фильтр (Rudin, Osher, Fatemi 1992)

$$\hat{I}(x) = \operatorname{argmin}_{\hat{I}} TV(\hat{I}) + \lambda \int |\hat{I}(x) - I(x)| dx,$$

при $TVI(x) = \sup \sum_{n=0}^N |I(x_n) - I(x_{n-1})| = \int |\nabla I(x)|$.

Theorem

Шум TV фильтра выражается как:

$$I(x) - TVFI(x) = -\frac{1}{2\lambda} \operatorname{curv}(\hat{I})(x)$$

Non-local means фильтр (Buades, Coll, Morel 2005)

$$\hat{I}(x) = \sum_y w(x, y) I(y),$$

$w(x, y) \in [0, 1]$, $\sum_y w(x, y) = 1$.

w определяется схожестью цвета в окрестности пикселей.

Заметим:

$$E\|x + v_1 - (y + v_2)\|^2 = \|x - y\|^2 + 2\sigma^2,$$

для независимых с.в. u, v с дисперсией σ и произвольных x, y .

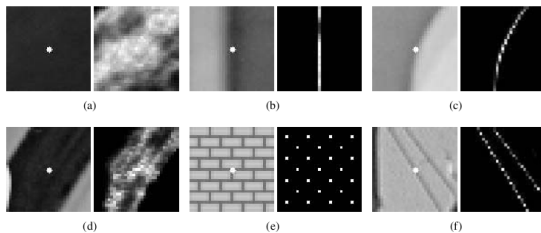
Non-local means фильтр (Buades, Coll, Morel 2005)

Определение весов:

$$w(x, y) = \frac{1}{Z(x)} e^{-\|I(N_x) - I(N_y)\|/\sigma^2},$$

где $I(N_x)$ - матрица пиксельных значений для квадратной окрестности N_x пикселя x , $Z(x)$ - нормализующая константа.

Примеры весов



(Buades, Coll, Morel 2005)

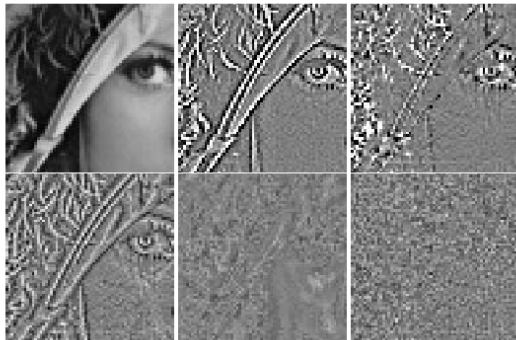
Non-local means фильтр (Buades, Coll, Morel 2005)

Theorem

Пусть $V = U + N$, где N - матрица независимых случайных величин с 0 средним. Пусть X_i - значения всех пикселей, кроме i -го. Тогда выполнено следующее:

- $E\{V(i)|X_i = x\} = E\{U(i)|X_i = x\}$ для всех i и x ;
- $E\{U(i)|V(N_i \setminus i)\}$ - функция $V(N_i \setminus i)$, минимизирующая (по g) среднеквадратичную ошибку $E\|U - g(V(N_i \setminus i))\|^2$

Non-local means фильтр: примеры



Исходное изображение, шум методов: гауссовский, анизотропный, TV, Ярославского (bilateral), NL-means. (Buades, Coll, Morel 2005)

Non-local means фильтр: примеры



Зашумленное изображение, результаты фильтров: гауссовский, анизотропный, TV, Ярославского (bilateral), NL-means. (Buades, Coll, Morel 2005)

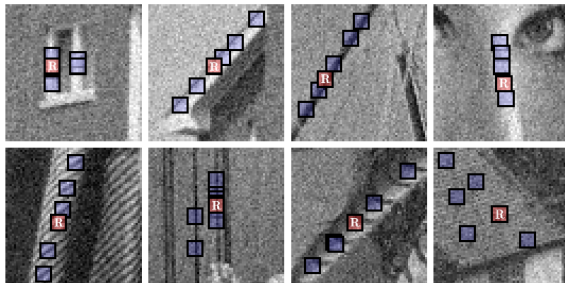
Block Matching 3D фильтр: примеры

Идеи

- поиск разреженного представления сигнала в некоем базисе
- агрегирование информации по всему изображению (аналогично NL-Means)
- совместная фильтрация похожих блоков (повышает разреженность представления)
(Dabov, Foi, Katkovnik, Egiazarian 2007)

Block Matching 3D: группировка

Идея 1: сопоставление (matching) блоков
реализация matching в BM3D:



Block Matching 3D: группировка

Идея 2: коллаборативная фильтрация

Блоки, близкие данному, собираются в стеки в порядке, соответствующем их близости первому блоку (R на рисунке выше).

Для каждого стека вычисляется 3D спектр вдоль 2х измерений изображения блока и 1 го измерения - направления стека.

Block Matching 3D: фильтр

Идея 3: Отсечение спектра

Информация сконцентрирована в небольшом числе коэффициентов => необходимо отсечь коэффициенты 3D спектра по абсолютному значению
Hard thresholding:

$$x_{ht} = x, |x| > \tau$$

$$x_{ht} = 0, |x| \leq \tau$$

Soft thresholding:

$$x_{st} = \mathit{sign}(x)(|x| - \tau)_+$$

$$a_+ = \mathit{id}(a > 0) \cdot a$$

Затем, выполняется обратное преобразование, и отфильтрованные блоки возвращаются на место. На каждом месте появляется набор блоков. Они усредняются с весами, пропорциональными близости блока R-блоку стека, в котором он фильтровался.

Block Matching 3D: фильтр

Идея 4: Винеровский фильтр

Есть модель изображения, есть зашумленное изображение

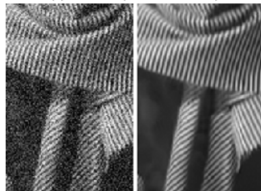
1. Оценить сигнал и уровень шума по изображениям
2. Применить винеровский фильтр к зашумленному изображению

Block Matching 3D: примеры

(a) *Lena* (PSNR 32.08 dB)



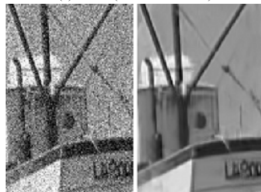
(b) *Barbara* (PSNR 30.73 dB)



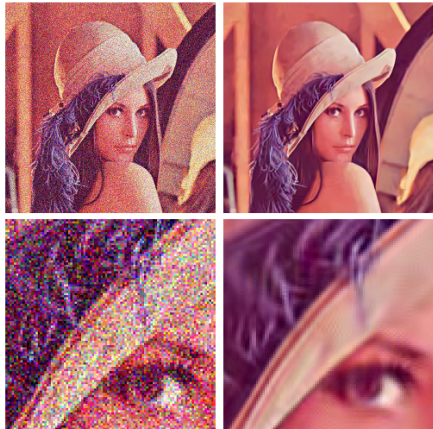
(d) *Man* (PSNR 29.62 dB)



(e) *Boats* (PSNR 29.91 dB)



Color Block Matching 3D: примеры



Color Block Matching 3D: наблюдения

YUV эффективнее RGB

корреляция по Y влечет корреляцию по U, V =>
делаем группировку по Y

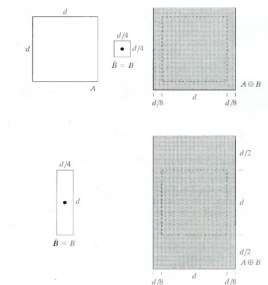
Морфология

Математическая морфология - представление и описание формы регионов изображения (скелетов, выпуклых оболочек,...)

Dilate

$$A \oplus B = \{z | (\hat{B})_z \cap A \neq \emptyset\}$$

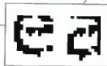
- $\hat{B} = \{w | w = -b, b \in B\}$
- $(B)_z = B + z$ - смещение на z
- B - структурирующий элемент



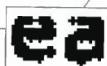
Операция Dilate

Пример dilate

Historically, certain computer programs were written using only two digits rather than four to define the applicable year. Accordingly, the company's software may recognize a date using "00" as 1900 rather than the year 2000.



Historically, certain computer programs were written using only two digits rather than four to define the applicable year. Accordingly, the company's software may recognize a date using "00" as 1900 rather than the year 2000.

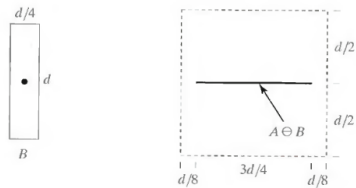
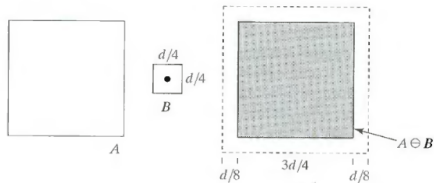


0	1	0
1	1	1
0	1	0

Операция Dilate: заполнение разрывов

Erode

$$A \ominus B = \{z | (\hat{B})_z \subseteq A\}$$



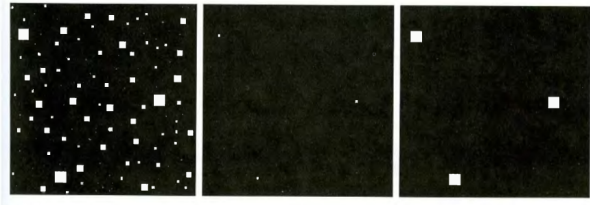
Erode & Dilate

Упражнение

$$(A \ominus B)^C = A^C \oplus \hat{B}$$

Erode & Dilate

Пример: фильтрация объектов по размеру



Исходное изображение, операция Erode, затем Dilate

Open & Close

Открытие (Open)

$$A \circ B = (A \ominus B) \oplus B (= \cup \{(B)_z \mid (B)_z \subseteq A\})$$

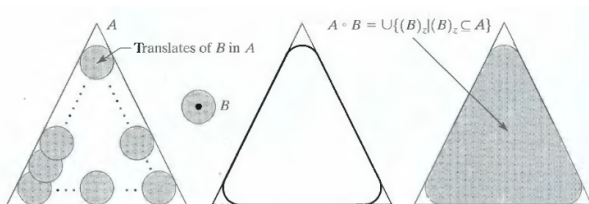
Закрытие (Close)

$$A \cdot B = (A \oplus B) \ominus B$$

Open

Открытие (Open)

$$A \circ B = (A \ominus B) \oplus B$$



Close

Закрытие (Close)

$$A \cdot B = (A \oplus B) \ominus B$$

