Object Detection

Студент 5 курса А. М. Грачев

ВМК МГУ

26 ноября 2013 г.

Input


0
50
100
150
200
250
300
350
50
100
150
200
250
300
350
400
450
500

Рис. 1.

Desired output

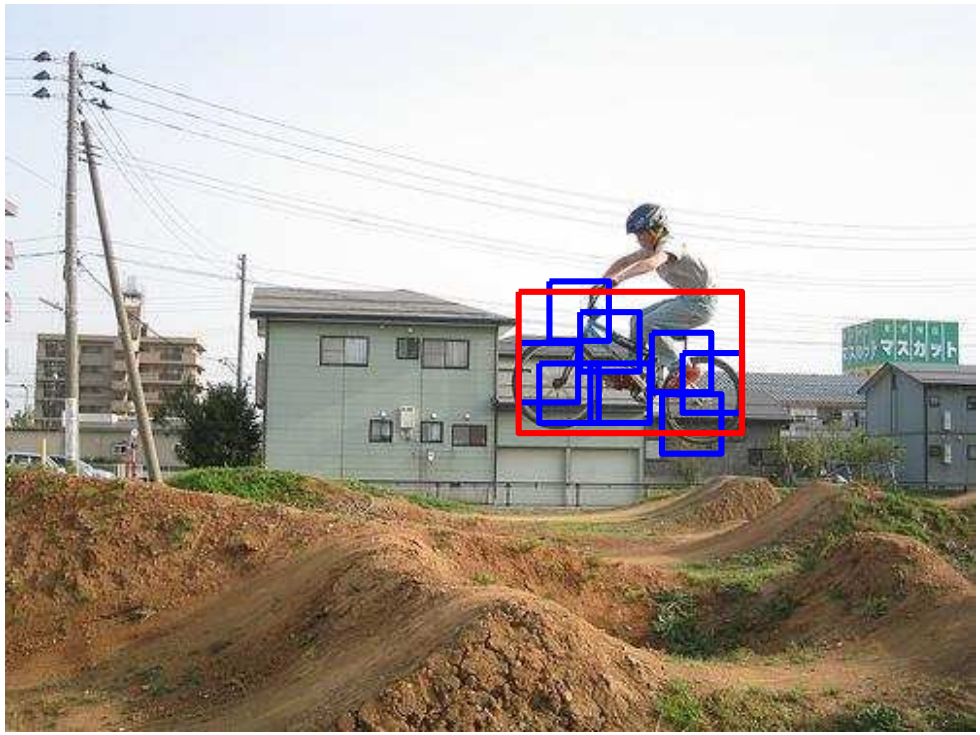

Рис. 2.

Another desired output

Рис. 3.

Application

- Semantic image and video search
- Human-computer interaction (e. g., Kinect)
- Automotive safety
- Camera focus-by-detection
- Surveillance
- Semantic image and video editing
- Assistive technologies
- Medical imaging
- ...

High-level structure

1. Сильные низкоуровневые признаки на основе гистограммы направленных градиентов
2. Эффективный алгоритм объединяющий part-based модель (pictorial structures)
3. Обучение с использованием latent SVM

Градиент изображения

Определение

Градиент изображения — направление максимального изменения яркости изображения

$$\nabla f = \left[\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}\right]$$

$$r(x,y) = ||\nabla f|| = \sqrt{\left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial y}\right)^2}, \quad \theta(x,y) = \tan^{-1}\left(\frac{\partial f}{\partial x} \Big/ \frac{\partial f}{\partial y}\right)$$

$r(x,y)$ — сила, $\theta(x,y)$ — направление градиента

Гистограмма направленных градиентов

- Вычислим направление градиента в каждом пикселе
- Разобьем направления градиентов на сколько-то групп (основных направлений)
- Для блоков фиксированных размеров посчитаем гистограмму распределения направлений градиентов

$$score = \sum_{x',y'} F\left[x', y'\right] \cdot G\left[x + x', y + y'\right] \quad (1)$$

Пусть F — это фильтр размера $w \times h$, H — пирамида признаков Через $\phi(H, p, w, h)$ будет обозначим вектор признаков, где $p = (x, y, l)$ определяет позицию (x, y) на l-том уровне пирамиды. Тогда

$$score = F' \cdot \phi(H, p, w, h) \quad (2)$$

$$score = \sum_{x',y'} F\left[x', y'\right] \cdot G\left[x + x', y + y'\right] \quad (1)$$

Пусть F — это фильтр размера $w \times h$, H — пирамида признаков Через $\phi(H, p, w, h)$ будет обозначим вектор признаков, где $p = (x, y, l)$ определяет позицию (x, y) на l-том уровне пирамиды. Тогда

$$score = F' \cdot \phi(H, p, w, h) \quad (2)$$

Feature pyramid

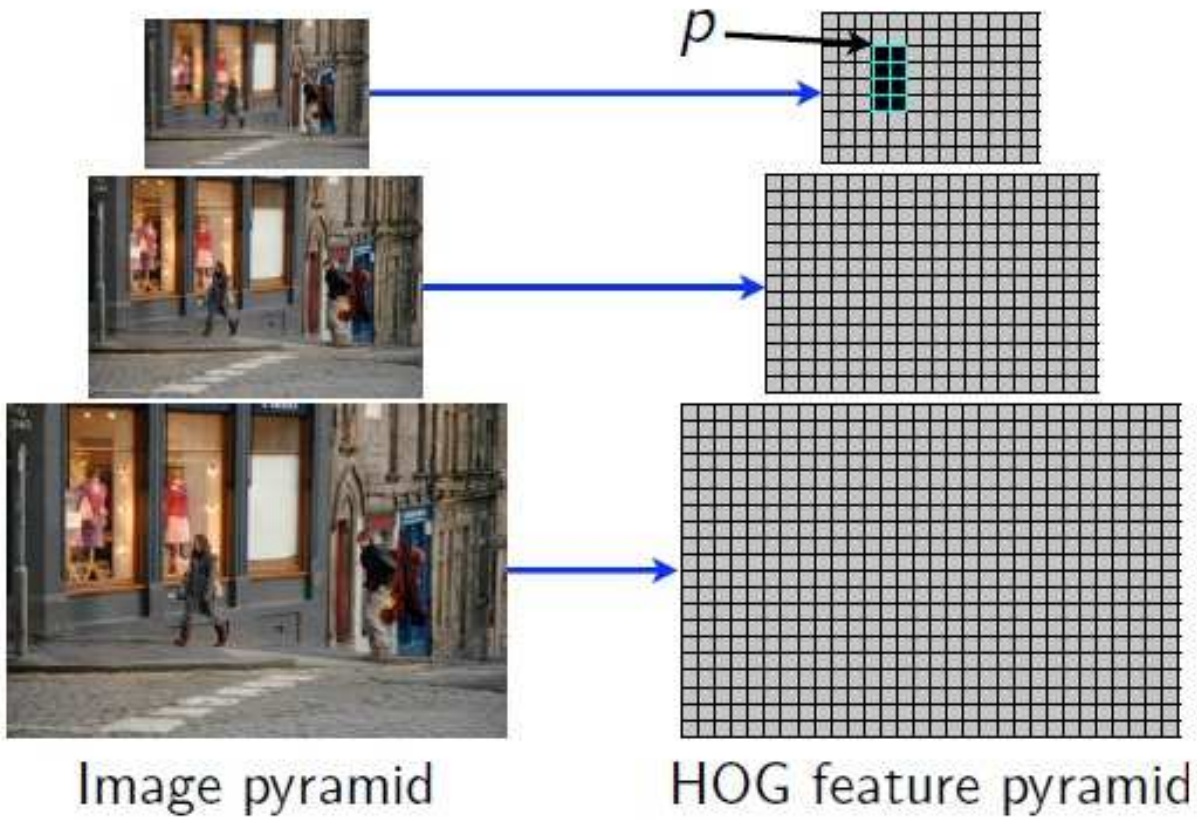

Рис. 4.

$$\mathrm{M} = (F_0, P_1, \ldots, P_n, b)$$

$$P_i = (F_i, v_i, d_i)$$

Здесь v_i — координаты точки прикрепления части к основному фильтру, d_i — вектор весов, который отвечает за расстояние от части до точки крепления

$$z = (p_0, \ldots, p_n)$$

$$score(p_0, \dots p_n) = \sum_{i=0}^{n} F_i' \cdot \phi(H, p_i) - \sum_{i=1}^{n} d_i \cdot \phi_d(dx_i, dy_i) + b \quad (3)$$

$$\phi_d(dx, dy) = (dx, dy, dx^2, dy^2) \quad (4)$$

$$dx_i = x_i - (2x_0 + v_i)$$

$$dy_i = y_i - (2y_0 + v_i)$$

$$score = \beta \cdot \psi(H, z)$$

$$\beta = (F_0', \dots, F_n', d_1, \dots, d_n, b)$$

$$\psi(H, z) = (\phi(H, p_0), \dots, \phi(H, p_n), -\phi_d(dx_1, dy_1) \dots, -\phi_d(dx_n, dy_n), 1)$$

$$score(p_0, \dots p_n) = \sum_{i=0}^{n} F_i' \cdot \phi(H, p_i) - \sum_{i=1}^{n} d_i \cdot \phi_d(dx_i, dy_i) + b \quad (3)$$

$$\phi_d(dx, dy) = (dx, dy, dx^2, dy^2) \quad (4)$$

$$dx_i = x_i - (2x_0 + v_i)$$

$$dy_i = y_i - (2y_0 + v_i)$$

$$score = \beta \cdot \psi(H, z)$$

$$\beta = \left(F_0', \dots, F_n', d_1, \dots, d_n, b\right)$$

$$\psi(H, z) = (\phi(H, p_0), \dots, \phi(H, p_n), -\phi_d(dx_1, dy_1) \dots, -\phi_d(dx_n, dy_n), 1)$$

$$score(p_0) = \max_{p_1,\dots p_n} score(p_0, \dots, p_n)$$

- Используем динамическое программирование.
- Используем обобщенное преобразование расстояния (generalized distance transform).
- Результирующий метод дает $O(nk)$, где n — количество частей модели, k — количество локаций в пирамиде признаков

$$R_{i,l}(x,y) = F_i' \cdot \phi\left(H,(x,y,l)\right)$$

$$D_{i,l}(x,y) = \max_{dx,dy}\left(R_{i,l}(x+dx,y+dy) - d_i \cdot \phi\left(dx,dy\right)\right)$$

$$score(x_0,y_0,l_0) = R_{0,l_0}(x_0,y_0) + \sum_{i=1}^{n} D_{i,l_0-\lambda}\left(2(x_0,y_0)+v_i\right) + b$$

$$P_{i,l}(x,y) = \arg\max_{dx,dy}\left(R_{i,l}(x+dx,y+dy) - d_i \cdot \phi_d\left(dx,dy\right)\right)$$

The matching process

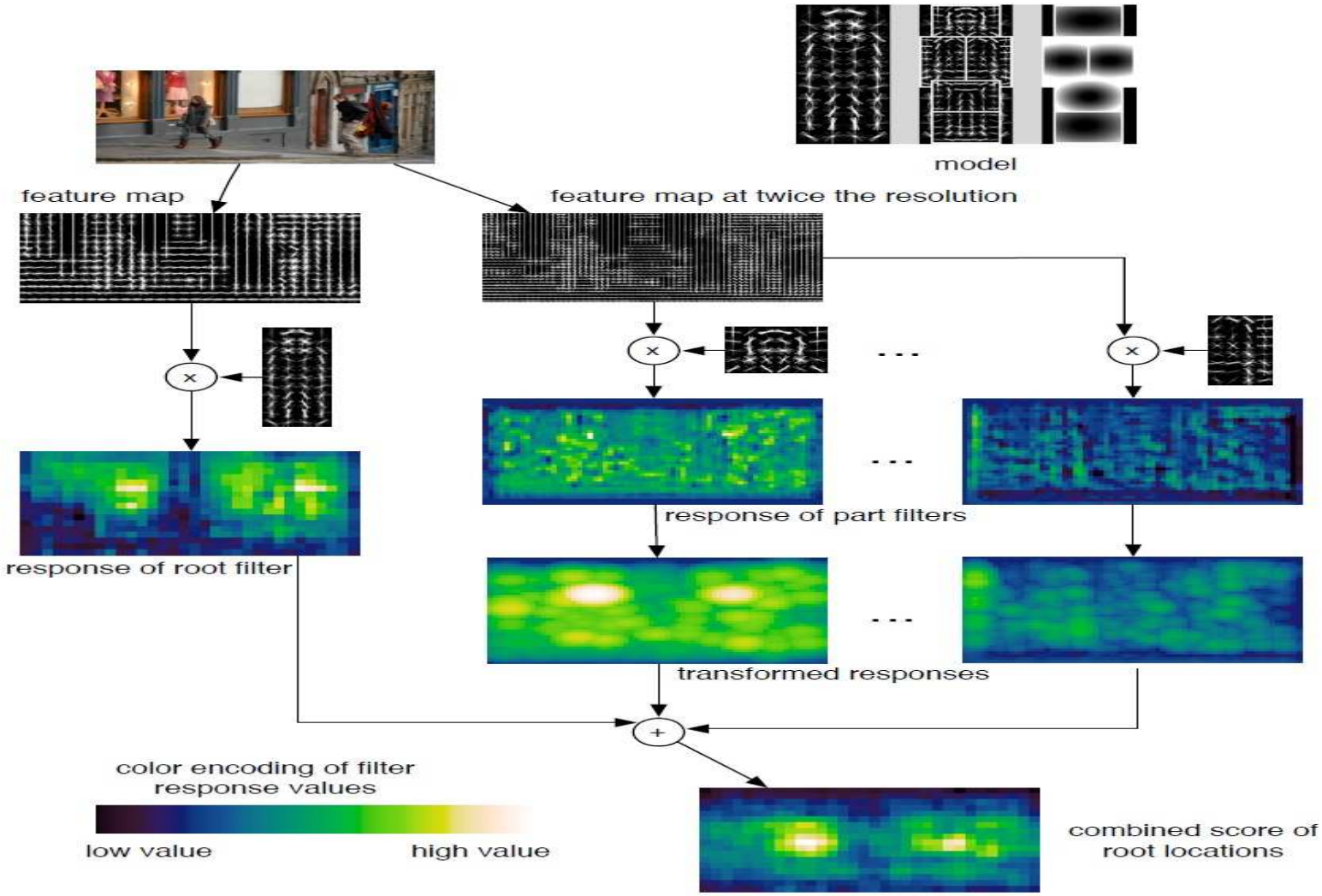

Рис. 5.

$$M = (M_1, \dots M_n) \tag{5}$$

Здесь M_c — это модель для c-той компоненты.
Теперь $z = (c, p_0, \dots, p_{n_c})$ и обозначим через $z' = (p_0, \dots, p_{n_c})$.

$$\beta = (\beta_1, \dots \beta_n)$$

$$\psi(H, z) = \left(0, \dots, 0, \psi\left(H, z'\right), 0, \dots, 0\right)$$

$$\beta \cdot \psi(H, z) = \beta_c \cdot \psi\left(H, z'\right)$$

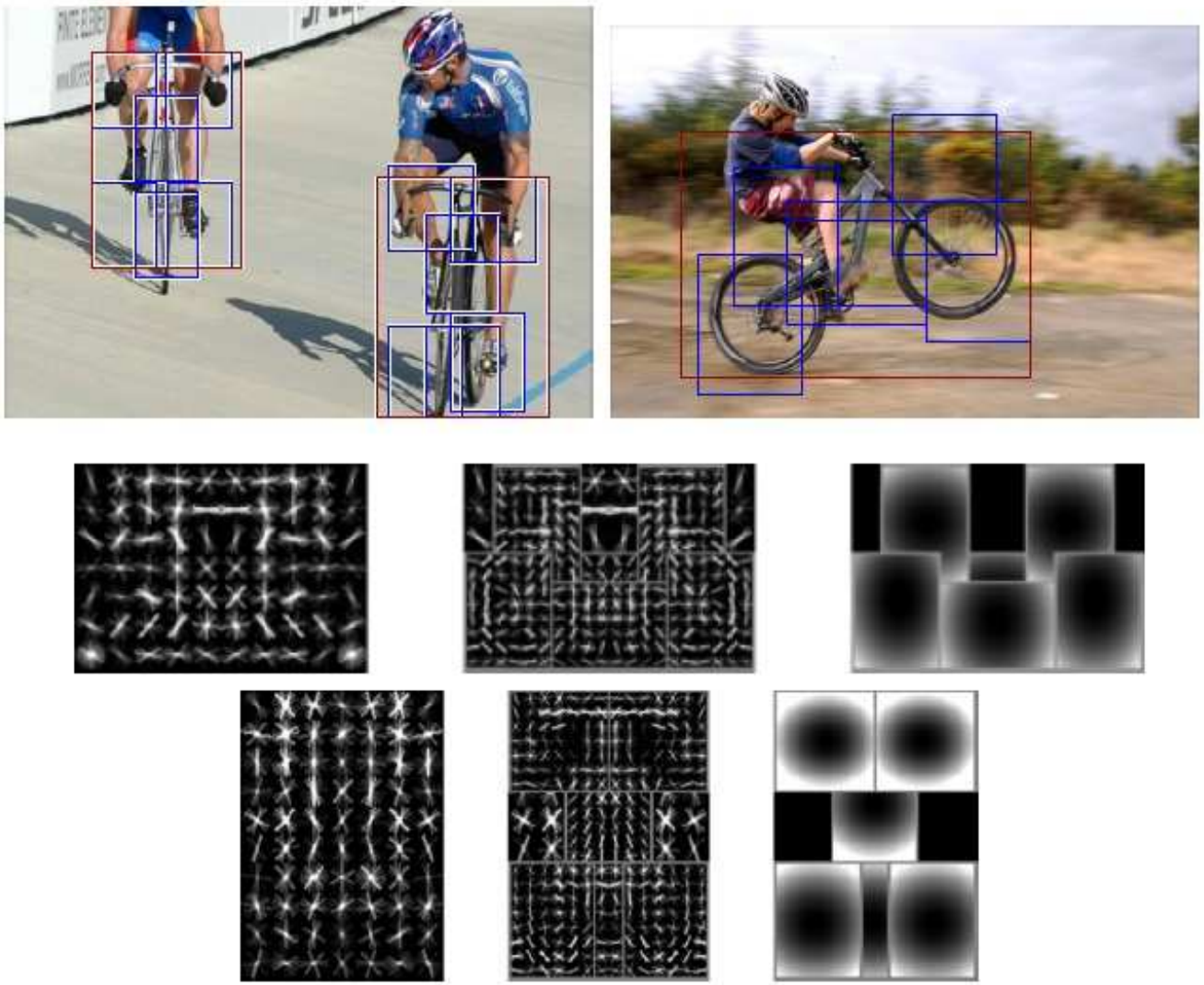

Рис. 6.

$$f_\beta = \max_{z \in Z(x)} \beta \cdot \Phi(x, z) \tag{6}$$

$$D = ((x_1, y_1), \ldots, (x_n, y_n))$$

$$y_i = \{-1, 1\}$$

$$L_D(\beta) = \frac{1}{2}||\beta||^2 + C \sum_{i=1}^{n} \max(0, 1 - y_i f_\beta(x_i))$$

Полувыпуклость

$$L_D(\beta) = \frac{1}{2}||\beta||^2 + C\sum_{i=1}^{n} \max(0, 1 - y_i f_\beta(x_i))$$

Warning!

Latent SVM приводит к невыпуклой оптимизационной задаче.

1. Если $y_i = -1$, то функция $\max(0, 1 - y_i f_\beta(x_i))$ выпукла по β.
2. Рассмотрим latent SVM, в котором возможно только одно скрытое значение для каждого положительного объекта. В таком случае $f_\beta(x_i)$ становится линейной для положительного объекта.

Полувыпуклость

$$L_D(\beta) = \frac{1}{2}||\beta||^2 + C\sum_{i=1}^{n} \max(0, 1 - y_i f_\beta(x_i))$$

Warning!

Latent SVM приводит к невыпуклой оптимизационной задаче.

1. Если $y_i = -1$, то функция $\max(0, 1 - y_i f_\beta(x_i))$ выпукла по β.
2. Рассмотрим latent SVM, в котором возможно только одно скрытое значение для каждого положительного объекта. В таком случае $f_\beta(x_i)$ становится линейной для положительного объекта.

1. Оптимизируем $L_D(\beta, Z_p)$ на множестве Z_p выбирая, скрытые значения так чтобы они давали максимальный отклик на каждом положительном примере $z_i = \arg\max_{z \in Z(x_i)} \beta \cdot \Phi(x_i, z)$.
2. Оптимизируем $L_D(\beta, Z_p)$ по переменной β, решая выпуклую задачу определенную на $L_{D(Zp)}(\beta)$.

Пусть $z_i(\beta) = \arg\max_{z \in Z(x_i)} \beta \cdot \Phi(x_i, z)$, тогда $f_\beta(x_i) = \beta \cdot \Phi(x_i, z_i(\beta))$ и

$$\nabla L_D(\beta) = \beta + C \sum_{i=1}^{n} h(\beta, x_i, y_i)$$

$$h(\beta, x_i, y_i) = \begin{cases} 0 & \text{if } y_i f_\beta(x_i) \geq 1 \\ -y_i \Phi(x_i, z_i(\beta)) & \text{otherwise} \end{cases}$$

1. Пусть α_t — это скорость обучения на t-ом шаге.
2. Пусть i — случайный объект
3. Пусть $z_i(\beta) = \arg\max_{z \in Z(x_i)} \beta \cdot \Phi(x_i, z)$
4. Если $y_i f_\beta(x_i) = y_i \left(\beta_i \cdot \Phi(x_i, z_i)\right) \geq 1$ set $\beta := \beta - \alpha_t \beta$
5. Иначе $\beta = \beta - \alpha_t \left(\beta - Cny_i \Phi(x_i, z_i)\right)$

Этап 1. Инициализация основного фильтра

Input.
$P = (I, B)$ — положительные объекты
N — фоновые объекты

1. Делим все объекты на m групп приблизительно одинакового размера $P_1 \dots P_m$
2. Обучаем m различных основных фильтров $F_1 \dots F_m$ используя обычный SVM

Этап 2. Объединение компонент

1. Объединяем основные фильтры в смешанную модель. Пока что без частей.
2. Обучаемся на полной выборке
3. Скрытая переменная здесь — это метка класса

Этап 3. Инициализация фильтров частей

1. Инициализируем части каждой компоненты.
2. Фиксируем количество частей и размер для каждой компоненты
3. Располагаем их, покрывая регионы с высокой энергией
4. Полагаем $d_i = (0, 0, 1, 1)$

Что использовать в качестве финальной границы изображения?

Вариант 1

Самый простой способ — это использование в качестве финальной границы объекта корневой фильтр.

Вариант 2

Пусть у нас есть функция $g(z)$ — границы всех частей и корневого фильтра.Также есть настоящие границы. Обучим четыре линейных функции для предсказания координат x_1, y_1, x_2, y_2 границ объекта.


P. Felzenszwalb, D. McAllester, D. Ramanan
"Object Detection with Discriminatively Trained Part Based Models."
— IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2008.


P. Felzenszwalb, D. Huttenlocher "Distance transforms of sampled functions"
— Cornell University CIS, Tech. Rep. 2004-1963, 2004.


P. Felzenszwalb, D. Huttenlocher "Pictorial structures for object recognition"
International Journal of Computer Vision, vol. 61, no. 1, 2005.


N. Dalal and B. Triggs
"Histograms of oriented gradients for human detection"
— IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2005.


R. Girshick
"From Rigid Templates to Grammars: Object Detection with Structured Models"
— Ph.D. dissertation, The University of Chicago, Apr. 2012