Алгоритм построения классификатора пола.

На начальном этапе обозначим, что используются 3 вида особенностей:  
1) SIFT – используется вариант DSIFT, реализованный в библиотеке vl\_feat с параметрами Size = 4 – размер области, в которой вычисляется одна гистограмма и Step = 16 – шаг с которым вычисляются SIFT дескрипторы. Т.е. для каждого блока с шагом 16 по горизонтали и вертикали вычисляется дескриптор с окном 16х16. 16х16, так как в SIFT используются 4 на 4 сетка, а одна ячейка имеет размер 4 (Size).

2) HOG – используется, как в статье разбиение блока изображения на 4 части. В каждой части вычисляется гистограмма по 16 направлениям, после чего бины противоположных направлений складываются в один. Таким образом, получается 8 направлений в 4х частях блока или 32 коэффициента. Более детально: предварительно вычисляются свертки с фильтрами [1 0 -1] и   
[-1 0 1]’ – grad\_x и grad\_y . После чего вычисляются направления – atan2(grad\_x, grad\_y) и амплитуда – sqrt(grad\_x.^2+ grad\_y.^2). Гистограмма амплитуд, по указанным направлениям и вычисляется по интервалам –pi:pi шагом pi/8

3) GABOR – фильтры Габора вычисляются не для блоков в отдельности – как HOG и SIFT, а для всего изображения. Так как их вычисления для каждого блока – слишком затратное. Так как для реализации фильтров Габора требуется достаточно высокая апертура. Используется фильтр, с апертурой 19х19. Свертка выполняется через преобразование Фурье. 19x19. Изображение 60x60 дополняется до 64х64, так же как и фильтр 19х19. Используется 3 масштаба с шагом sqrt(2) и 6 направлений (с шагом pi/3). После получение коэффициентов фильтров Габора – происходит децимация. В зависимости от размеров блока в один отсчет превращается блок размером от 2х2 до 6х6. Так как минимальный размер 2х2, то при вычислении фильтров Габора сразу же осуществляется линейная децимация на 2 по обоим направлениям. Далее в зависимости от размера блока, происходит децимация по ближайшему соседу от 1х1 до 3х3.

Предварительный отбор блоков. Блоки – это часть изображения, в которой выбирается особенность. Эти части выбираются случайным образом, из которых уже отбираются пригодные к распознаванию с помощью алгоритма RealAdaBoost. Несмотря на то, что в статье описано, что сначала отбирается блок, а затем уже тип особенности – такой подход дал более низкий результат, чем выбор сразу пары особенность + блок. Блоки выбираются предварительно, чтобы результаты экспериментов можно было сравнивать при изменении параметров. Были отобраны 500 блоков площадью до 800 пикселей. Эти блоки использовались для вычисления особенностей типа HOG и GABOR. Также для особенностей типа SIFT были отобраны 300 блоков площадью до 1200 пикселей.

Для каждой пары блок + тип особенности – вычисляется вектор-строка признаков f. Для всей обучающей выборки вектор-строки f образуют матрицу F, где i-ая строка соответствует особенности вычисленной на выбранном блоке в i-м изображении обучающей выборки. Если за y обозначить вектор-столбец разметки, где 1- мужчины / -1 – женщины, то можно вычислить регрессию между матрицей F и вектор столбцом y. Для учета смещения к матрице F добавляется справа столбец из 1. F1 = [F **1**].

Теперь если обозначить за a вектор-столбец коэффициентов регрессии, то   
(F1’\*F1)\*a = F1’\*y.

Решение этой СЛАУ – есть вектор столбец коэффициентов регрессии. Теперь для каждого блока и особенности может быть посчитан вектор x, который приближает y и является: x = F1\*a;

Каждой паре особенность + блок на тестовом множестве будет соответствовать свой вектор-столбец, длиной в число отсчетов тестовой выборки. Совокупность таких вектор столбцов подается на вход RealAdaBoost. Алгоритм RealAdaBoost был реализован на базе совокупности описываемой статьи и статьи Robert E. Schapire, Yoram Singer Improved Boosting Algorithms Using Confidence-rated Predictions // Dec. 1999, Vol. 37, Is. 3, pp. 297-336  
http://adaptiveilp.googlecode.com/svn/trunk/thesis/bibliography/schapire99boosting%20-%20Confidence%20rated%20boosting.pdf

(стр 9 и далее)

Согласно этой статье необходимо выбрать разбиение области {-1,1} на интервалы. Далее каждому элементу обучающей выборки, в нашем случае – это вектор столбец x будет соответствовать определенный интервал. По базовой статье RealAdaBoost имеет дело с матрицей, обозначенной у нас, как F1 и требует задать преобразование из области особенностей (строк F1) на область вещественных чисел. Однако у нас такое преобразование уже выполнено, поэтому далее мы считаем, что i-ый тестовый пример принадлежит j-му интервалу, если x(i) – попадает в этот интервал. Число интервалов было выбрано 32. Было замечено, что при увеличении числа интервалов свыше 16 чуть повышается качество модели, но замедляется процесс обучения. Введем определение W+ или WP, W- или WM. WP(i) – сумма весов обучающих элементов, чьи значения попадают в i-ый интервал. Изначально все веса инициализируются константой, но в процессе обучения, веса распознанных примеров снижаются, а нераспознанных – растут.

Псевдокод алгоритма  
y – набор меток обучающей последовательности

|  |  |
| --- | --- |
| В статье | В реализации |
| Parameters  N number of training samples  M number of evaluated features each iteration  T maximum number of weak classifiers  **Input**: Training set  , i = 1,..,N;.  1. Initialize sample weight, classifier output, and false  positive rate  ; F() = 0; i = 1,..,N, fp0 = 1  2. Repeat for t = 1, 2,.., T  2.1 Update the sample weight wi using the hth weak  classifier output  (Формула неверна.) На самом деле , а не  2.2 For m = 1 to M  2.2.1 Generate a random region with a specific  feature extraction method (SIFT, HOG, or  Gabor)  2.2.2 Extract features and do least square to  2.2.3 Build the predict distribution function W+ and  W-  2.2.4 Select the best feature based on minimizing Z  in equation (6)  2.3 Update weak classifier using (3)  2.4 Update strong classifier  2.5 Calculate current false positive rate fpt  3. Output classifier | Параметры  X – матрица вектор-столбцов приближений y через регрессии. Матрица X в Matlab-коде представлена 3-мя матрицами SIFT\_, HOGi и GABOR N Число обучающих примеров  Nb = 32 Число бинов между -1 и 1  CA = 0.25 Complex Aware Factor  M = 800 Число итераций поиска блоков  T = 25 Число слабых классификаторов  eps = 0.003 Сглаживающий фактор  **Вход**: уже вычисленная матрица X.  1. Инициализируем falsePos = 0.5 ,  weights = 1/N  h = **0**  2. выполняем T итераций, с номером итерации – t  2.1. Обновляем веса примеров  weights = weights.\*exp(-(y.\*h));  weights = weights / sum(weights);  2.2. Для m от 1 до M (800) повторяем 2.2.1-2.2.4  2.2.1. Выбираем циклически SIFT, HOG, GABOR + индекс случайного блока  2.2.2 – Особенности для блока уже извлечены и МНК уже сделан на предварительном этапе. Поэтому просто возьмем нужный вектор-столбец **x** из матрицы Х.  2.2.3. Для вектора **х** вычислим гистограммы WP и WM. WP – это гистограмма для элементов вектора x, соответствующих мужчинам на интервале -1..1 с шагом 2/Nb. WM – аналогичная гистограмма для женщин.  2.2.4. Для каждого m сохраним соответствующее значение целевой функции:  ,  где С – это стоимость особенности в нашем случае SIFT – 10, HOG-2, GABOR –  (2 + число\_коэффициентов/150) (в статье 2-6 в зависимости от числа коэффициентов)  2.3. Выбираем особенность + блок, представленную вектор столбцом в матрице Х, соответствующую минимальному значению Обновляем классификатор по формуле из статьи. Для i-го примера x будет принадлежать j-му интервалу (1..Nb)  2.4. Обновляем сильный классификатор как в статье  Ft+1(xi) = Ft(xi) + ht(xi)  2.5. Рассчитываем аппроксимацию falsePos =  0.03 + 0.75\*(sum(abs(yEst(y==1)-y(y==1))) + sum(abs(yEst (y==0)-y(y==0))))/length(y)  3. Выходной классификатор аналогичен статье с той разницей, что необходимо вычислить особенности и умножить их на коэффициенты регрессии для получения |

Отобранные 25 особенностей, представленные парами тип + блок и 25 вектор столбцами подаются на вход алгоритма обучения линейного SVM.  
Для распознавания пола по входному изображению – необходимо извлечь 25 особенностей из изображения, скалярно умножить каждую на свой вектор регрессионных коэффициентов. Получить 25 чисел, по которым с помощью SVM модели получить распознанный пол.