每周学习进展阶段汇报

---CS131 Stanford University 计算机视觉基础课程

汇报人 : 胡小婉

时间段 : 2018年2月5日至2018年8日&&2月14日至15日

[1. Visual Bag of Words 2](#_Toc1503620019)

[1.1 算法原理 2](#_Toc462767398)

[1.2 算法步骤 2](#_Toc915015364)

[2. Spatial Pyramid Matching(空间金字塔匹配) 4](#_Toc1118263347)

[2.1 Pyramids 4](#_Toc293224579)

[2.2 Bag of Words + Pyramids 4](#_Toc1724948002)

[2.3 Pyramid Match Kernels 5](#_Toc1866550019)

[3. Naive Bayes(朴素贝叶斯) 6](#_Toc68384369)

[3.1 基本原理 6](#_Toc1596759806)

[3.2 基本步骤 7](#_Toc592160159)

[3.2.1 Prior(先验) 7](#_Toc2001954450)

[3.2.2 Posterior(后验) 7](#_Toc1075102144)

[3.2.3 Classification(分类) 7](#_Toc2004510935)

[4. support vector machine (SVM) 8](#_Toc1617921769)

[4.1 基本原理 8](#_Toc1296772050)

[4.2 推导过程及应用 10](#_Toc2047409024)

[5. HOG 特征的提取--基于scikit-image 10](#_Toc815827713)

[5.1 基本思想 10](#_Toc1441139927)

[5.2 python实现 11](#_Toc583944024)

[Tips: 几种归一化方法（Normalization Method）python实现 12](#_Toc1194358413)

[1. (0,1)标准化： 12](#_Toc1892724099)

[2. Z-score标准化： 12](#_Toc1897554365)

[3. Sigmoid函数 12](#_Toc872995605)

[6. Evaluating Object Detection 13](#_Toc2135405948)

[7. A Simple Sliding Window Detector 14](#_Toc550688347)

[8. The Deformable Parts Model (DPM) 14](#_Toc1921453785)

[8.1 基本原理 15](#_Toc1558305677)

[8.2 算法实现 15](#_Toc1909593880)

[8.2.1 HOG 15](#_Toc1230562723)

[8.2.2 DPM模型 15](#_Toc2094752858)

[Tip:参考网址 17](#_Toc646346914)

[9. Hw7(作业题) 17](#_Toc586699094)

[9.1 Hog Representation 17](#_Toc410036609)

[9.2 Sliding Window 18](#_Toc1561362278)

[9.3 Image Pyramid 19](#_Toc1704962441)

[9.4 Pyramid Score 19](#_Toc703261188)

[9.5 Deformable Parts Detection 20](#_Toc1138826633)

[9.6 Human Parts Location 21](#_Toc1424028812)

[9.7 Gaussian Filter 24](#_Toc771645557)

[Tips:skimage.filters的Gaussian函数 26](#_Toc588102791)

**Lecture #14: Visual Bag of Words**

**Lecture 15: Detecting Objects by Parts**

**Lecture 16: Recognizing Objects by Parts**

# 1. Visual Bag of Words

## 1.1 算法原理

Bag of words模型最初被用在文本分类中，将文档表示成特征矢量。它的基本思想是假定对于一个文本，忽略其词序和语法、句法，仅仅将其看做是一些词汇的集合，而文本中的每个词汇都是独立的。简单说就是讲每篇文档都看成一个袋子（因为里面装的都是词汇，所以称为词袋，Bag of words即因此而来），然后看这个袋子里装的都是些什么词汇，将其分类。如果文档中猪、马、牛、羊、山谷、土地、拖拉机这样的词汇多些，而银行、大厦、汽车、公园这样的词汇少些，我们就倾向于判断它是一篇描绘乡村的文档，而不是描述城镇的

In Computer Vision, we can consider an image to be a collection of image features. By incorporating frequency counts of these features, we can apply the "Bag of Words" model towards images and use this for prediction tasks such as image classification and face detection.

There are two main steps for the "Bag of Words" method when applied to computer vision, and thesewill further be explored in the Outline section below.

1. Build a "dictionary" or "vocabulary" of features across many images - what kinds of common features exist in images? We can consider, for example, color scheme of the room, parts of faces such as eyes, and different types of objects.

2. Given new images, represent them as histograms of the features we had collected - frequencies of the visual "words" in the vocabulary we have built.

## 1.2 算法步骤

Bag of visual word类似于BoW模型，基本思想概括如下：

1）提取特征（Extract Features)

根据具体应用考虑，综合考虑特征的独特性、提取复杂性、效果好坏，处理是否方便等选择特征。

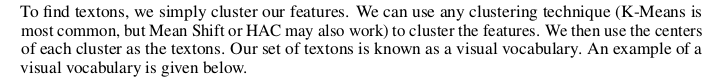
2）学习视觉词袋（Learn Visual Vocabulary）

统计图像数据库中出现的所有特征，去除冗余组成词袋。如果提取的图像特征过多，一般需要利用聚类算法先把相近的单词归为一类（类似于文档检索里的找词根），利用这些聚类结果来组成词袋。

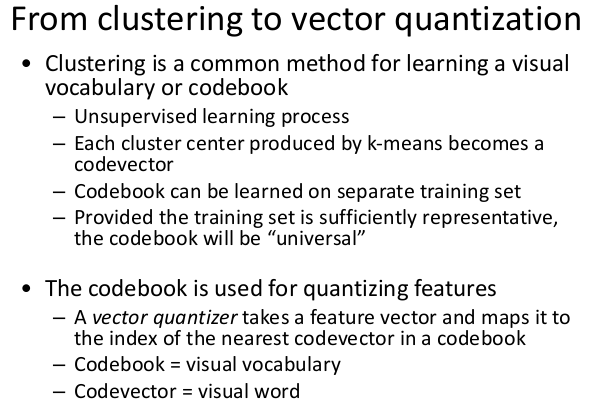
3）利用视觉词袋量化图像特征（Quantize features using visual vocabulary）

4）利用词频表示图像（Represent images by frequencies of visual words）

1. Extracting Interesting Features
2. Learning Visual Vocabulary

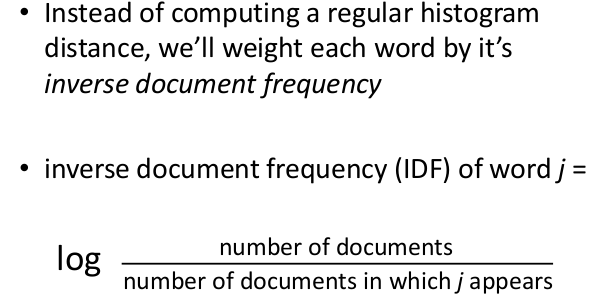


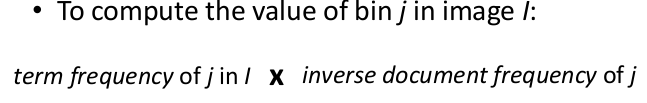
1. Quantize Features
2. Represent Images by Frequencies



1.3 Large-Scale Image Search

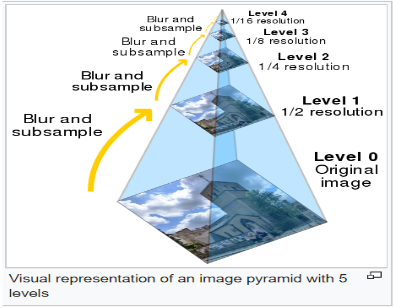
1.3.1 TF-IDF weighting



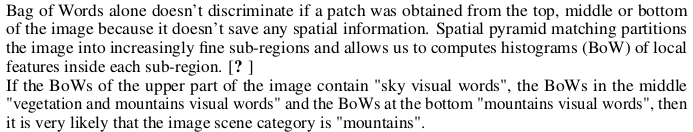


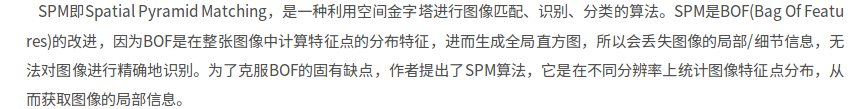
# 2. Spatial Pyramid Matching(空间金字塔匹配)

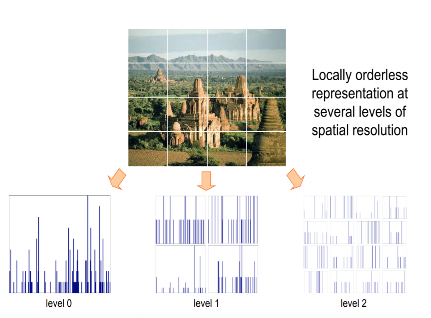
## 2.1 Pyramids



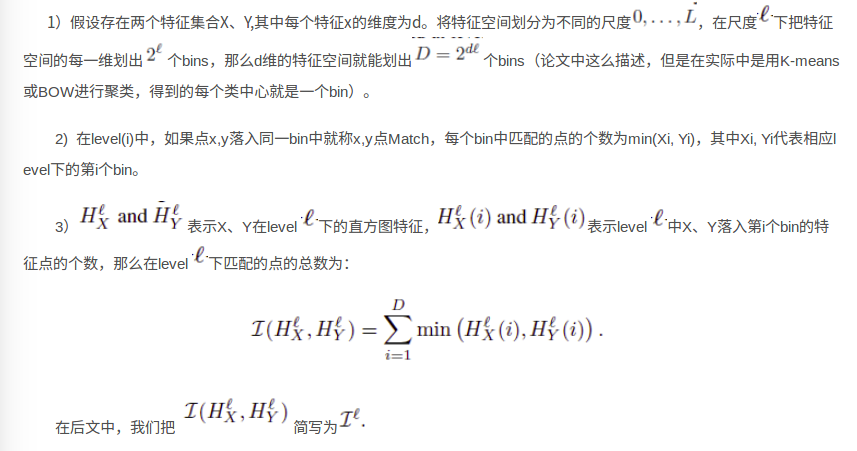
## 2.2 Bag of Words + Pyramids

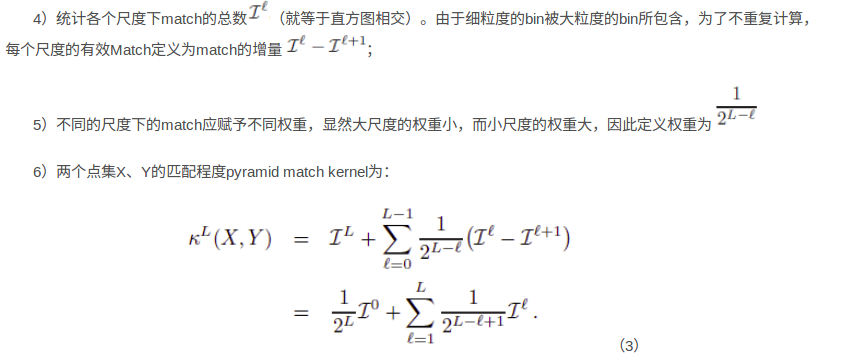


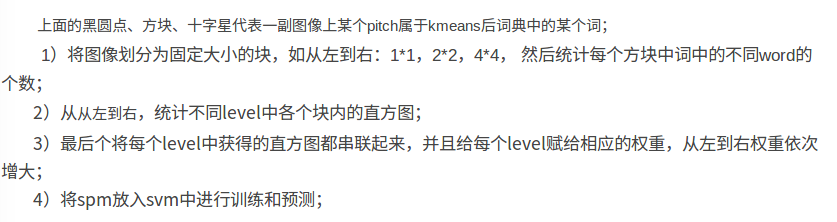
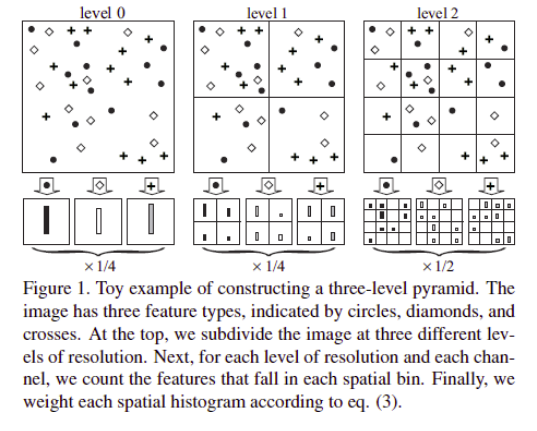




## 2.3 Pyramid Match Kernels







论文及参考网址:

Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories

**<http://blog.csdn.net/chlele0105/article/details/16972695>**

# 3. Naive Bayes(朴素贝叶斯)

## 3.1 基本原理

贝叶斯分类是一类分类算法的总称，这类算法以贝叶斯定理为基础，故统称为贝叶斯分类。贝叶斯定理解决了现实生活中经常遇到的问题：已知某条件概率，如何得到事件交换后的概率，即在已知P(A|B)的情况下求得P(B|A)。条件概率P(A|B)表示事件B已经发生的前提下，事件A发生的概率，叫做事件B条件下发生事件A的条件概率。其基本求解公式为：P(A|B)=P(AB)/P(B)。贝叶斯定理：



贝叶斯的主要思想可以概括为：先验概率+数据=后验概率。贝叶斯定理换个表达形式：



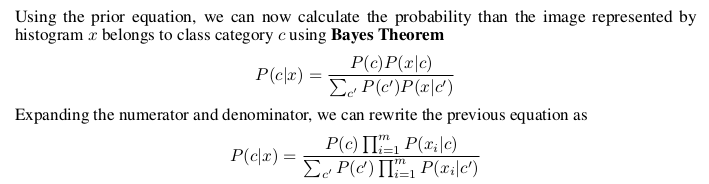
朴素贝叶斯的含义是：朴素——特征条件独立，贝叶斯——基于贝叶斯定理。朴素贝叶斯分类是一种十分简单的分类方法，这种分类的思想真的很朴素：对于给出的待分类项，求解此项条件下各个类别出现的概率，概率最大的类别就认为此项属于该类别

## 3.2 基本步骤

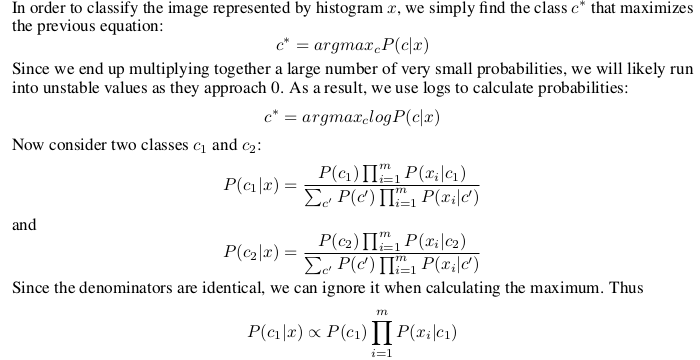
### 3.2.1 Prior(先验)

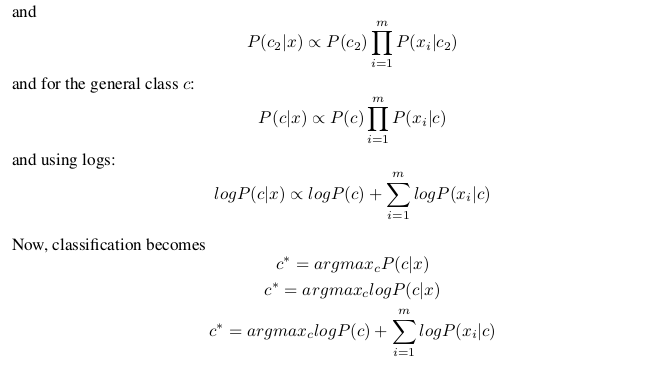


### 3.2.2 Posterior(后验)



### 3.2.3 Classification(分类)



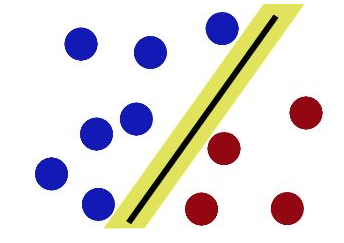
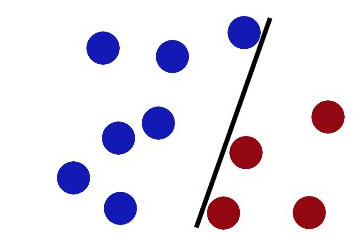


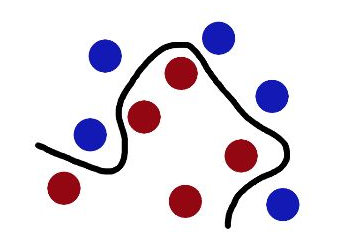
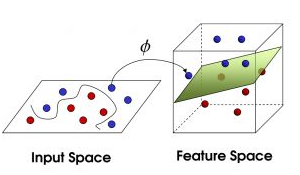
# 4. support vector machine (SVM)

## 4.1 基本原理

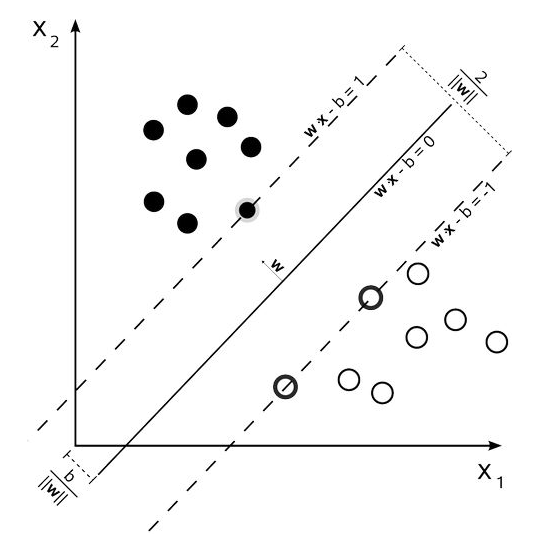
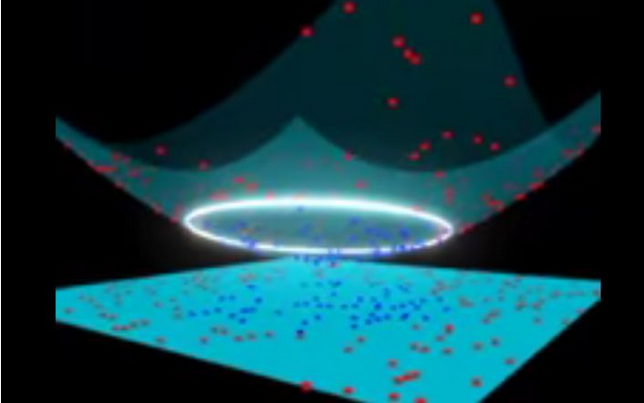
支持向量机（Support Vector Machine ,SVM）的主要思想是：建立一个最优决策超平面，使得该平面两侧距离该平面最近的两类样本之间的距离最大化，从而对分类问题提供良好的泛化能力。对于一个多维的样本集，系统随机产生一个超平面并不断移动，对样本进行分类，直到训练样本中属于不同类别的样本点正好位于该超平面的两侧，满足该条件的超平面可能有很多个，SVM正式在保证分类精度的同时，寻找到这样一个超平面，使得超平面两侧的空白区域最大化，从而实现对线性可分样本的最优分类。

支持向量机中的支持向量（Support Vector）是指训练样本集中的某些训练点，这些点最靠近分类决策面，是最难分类的数据点。SVM中最优分类标准就是这些点距离分类超平面的距离达到最大值；“机”（Machine）是机器学习领域对一些算法的统称，常把算法看做一个机器，或者学习函数。SVM是一种有监督的学习方法，主要针对小样本数据进行学习、分类和预测

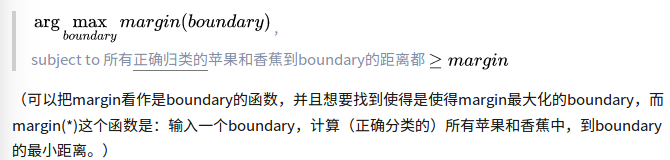


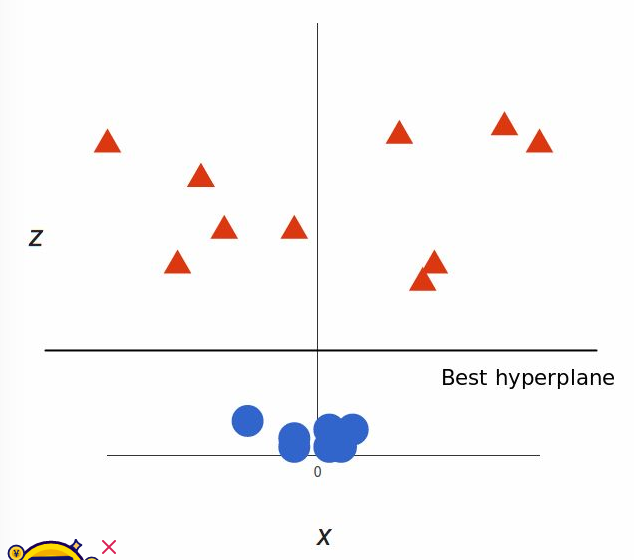
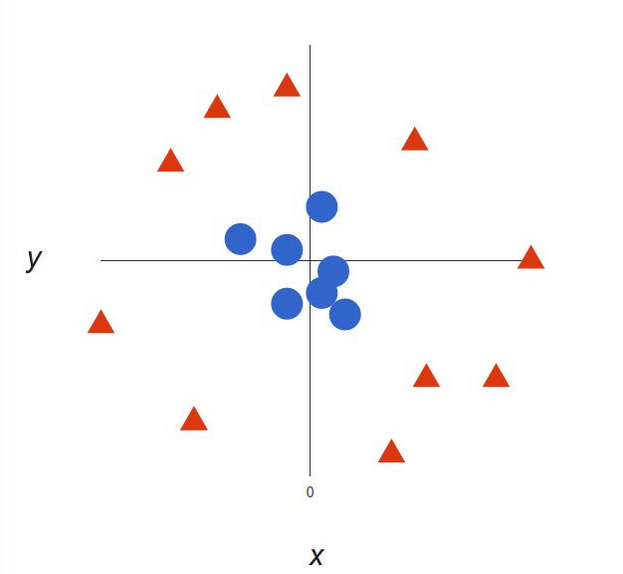


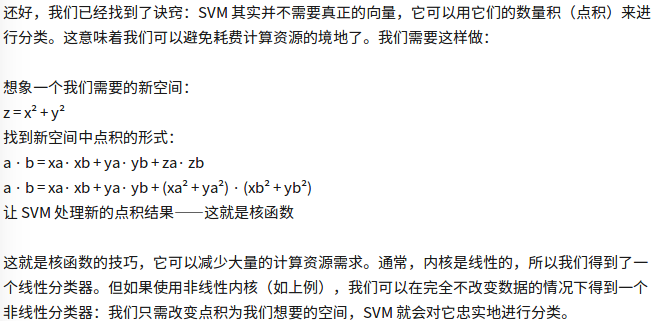
把这些球叫做 「data」，把棍子 叫做 「classifier」, 最大间隙trick 叫做「optimization」， 拍桌子叫做「kernelling」, 那张纸叫做「hyperplane」。



在SVM中，我们寻找一条最优的分界线使得它到两边的margin都最大,在这种情况下边缘加粗的几个数据点就叫做support vector，这也是这个分类算法名字的来源。







引入核函数，就是使用核函数对应高维空间内积的性质来使得线性分类器可以隐式地在高维空间建立分类面。因为在高维空间中的分类面更容易绕过一些低维空间中不可分的区域，这样可以达到更好的分类效果。

1. 它是针对线性可分情况进行分析，对于线性不可分的情况，通过使用非线性映射算法将低维输入空间线性不可分的样本转化为高维特征空间，使其线性可分，从而使得高维特征空间采用线性算法对样本的非线性特征进行线性分析成为可能；
2. 它基于结构风险最小化理论之上在特征空间中建构最优分割超平面，使得学习器得到全局最优化，并且在整个样本空间的期望风险以某个概率满足一定上界。

## 4.2 推导过程及应用

blog.csdn.net/american199062/article/details/51322852

知乎: https://www.zhihu.com/question/21094489

# 5. HOG 特征的提取--基于scikit-image

## 5.1 基本思想

HOG 特征, histogram of oriented gradient, 梯度方向直方图特征, 作为提取基于梯度的特征, HOG 采用了统计的方式(直方图)进行提取. 其基本思路是将图像局部的梯度统计特征拼接起来作为总特征. 局部特征在这里指的是将图像划分为多个Block, 每个Block内的特征进行联合以形成最终的特征. 具体来说:

将图像分块: 以Block 为单位, 每个Block以一定的步长在图像上滑动, 以此来产生新的Block.

Block作为基本的特征提取单位, 在其内部再次进行细分: 将Block 划分为(一般是均匀划分)NxN的小块, 每个小块叫做cell.

cell是最基本的统计单元, 在cell内部, 统计每个像素的梯度方向, 并将它们映射到预设的M个方向的bin里面形成直方图.

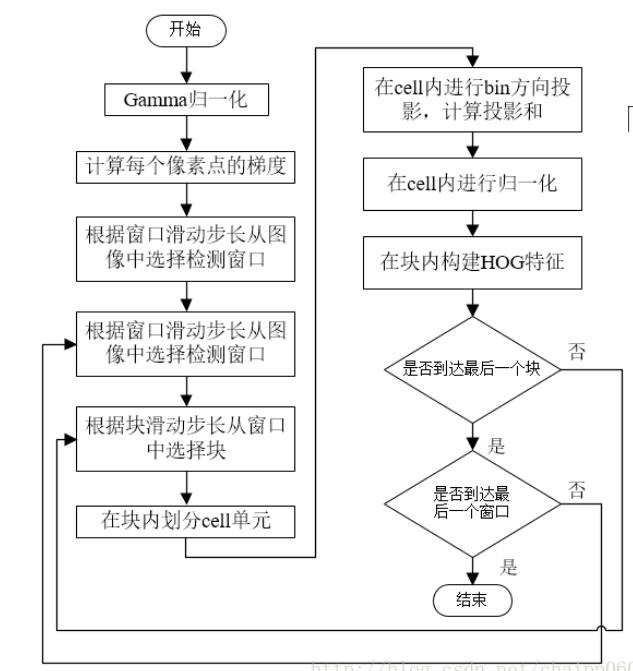
每个Block 内部的所有cell的梯度直方图联合起来并进行归一化处理(L1-norm, L2-Norm, L2-hys-norm, etc), 据说这样可以使特征具有光照不变性. 光照属于加性噪声, 归一化之后会抵消掉关照变化对特征的影响.

所有Block的特征联合起来, 就是最终的HOG特征

这里牵扯到一些技术细节, 比如将局部的梯度方向映射到预设的方向Bin里面需要双线性插值或三线性插值. 某点的梯度方向的映射是按照改点的梯度强度进行加权的.



## 5.2 python实现







## Tips: 几种归一化方法（Normalization Method）python实现

### 1. (0,1)标准化：

这是最简单也是最容易想到的方法，通过遍历feature vector里的每一个数据，将Max和Min的记录下来，并通过Max-Min作为基数（即Min=0，Max=1）进行数据的归一化处理



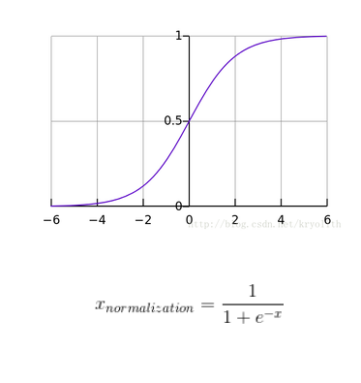
### 2. Z-score标准化：

这种方法给予原始数据的均值（mean）和标准差（standard deviation）进行数据的标准化。经过处理的数据符合标准正态分布，即均值为0，标准差为1，这里的关键在于复合标准正态分布，个人认为在一定程度上改变了特征的分布，关于使用经验上欢迎讨论，我对这种标准化不是非常地熟悉，转化函数为：



### 3. Sigmoid函数

Sigmoid函数是一个具有S形曲线的函数，是良好的阈值函数，在(0, 0.5)处中心对称，在(0, 0.5)附近有比较大的斜率，而当数据趋向于正无穷和负无穷的时候，映射出来的值就会无限趋向于1和0，是个人非常喜欢的“归一化方法”，之所以打引号是因为我觉得Sigmoid函数在阈值分割上也有很不错的表现，根据公式的改变，就可以改变分割阈值，这里作为归一化方法，我们只考虑(0, 0.5)作为分割阈值的点的情况：

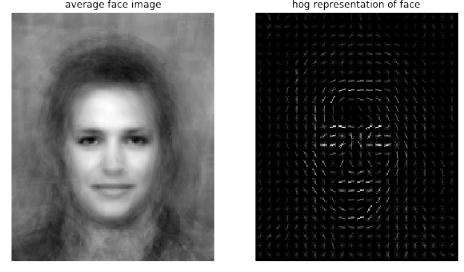
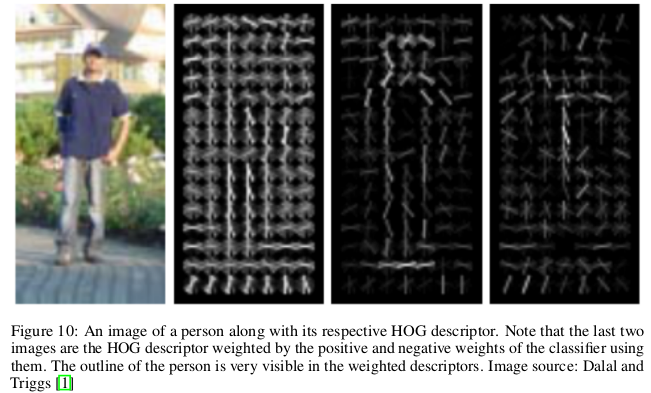


# 6. Evaluating Object Detection



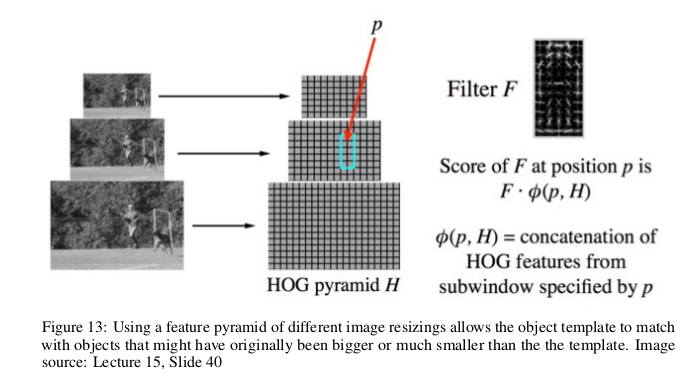
# 7. A Simple Sliding Window Detector



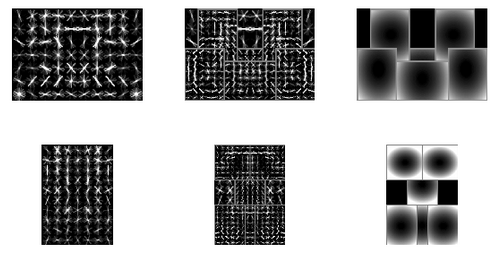
计算HOG分数

滑动窗口缩放

# 8. The Deformable Parts Model (DPM)



## 8.1 基本原理



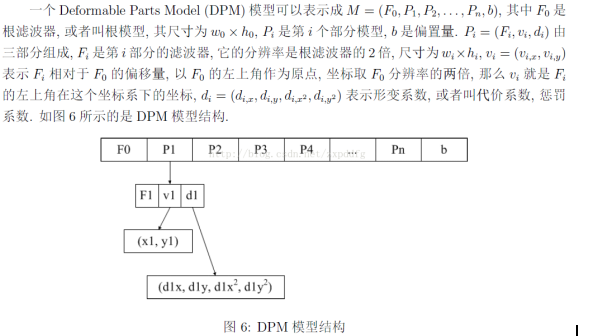
Deformable Parts Model (DPM) 是一种很成功的物体检测方法。可以说这种方法是从HOG继承而来的。传统的HOG特征只采用一个模板表示某种物体，而DPM把物体的模板划分成根模型和部分模型，其中的根模型等效于传统的HOG特征，部分模型则是物体某些部分的模板。在检测的时候，根模型用来对物体可能存在的位置进行定位，部分模型用来进行进一步的确认。付出了更多的运算量使得DPM的检测效果要优于传统的HOG。

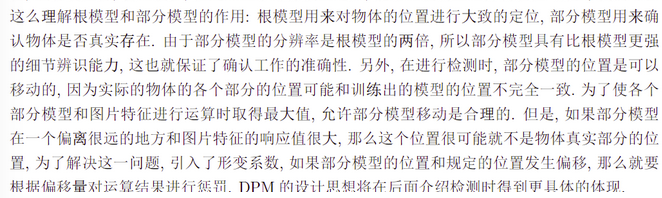
## 8.2 算法实现

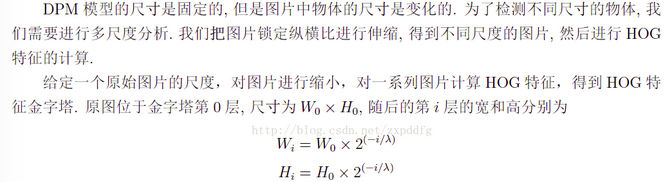
### 8.2.1 HOG

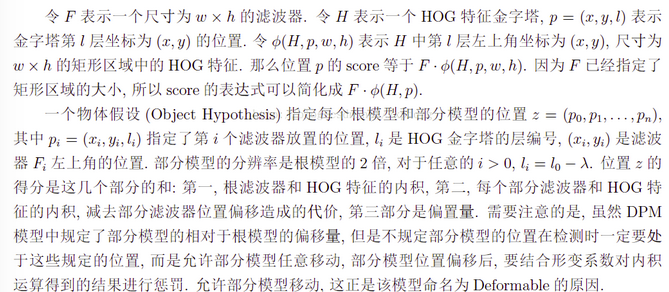
参考网址:<http://blog.csdn.net/zxpddfg/article/details/42263553>

### 8.2.2 DPM模型

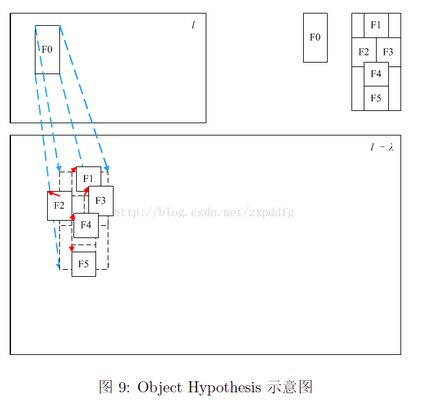


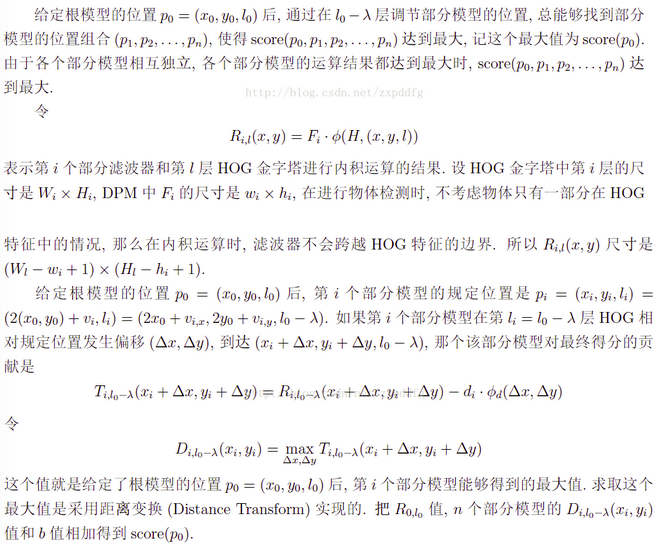












### Tip:参考网址

<http://blog.csdn.net/zxpddfg/article/details/42263553>

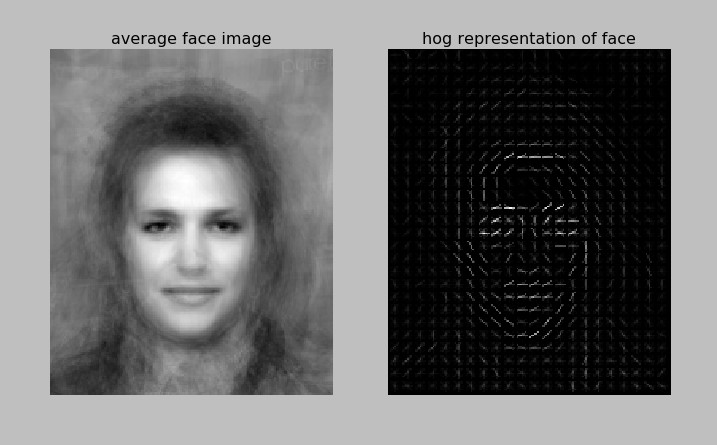
# 9. Hw7(作业题)

In this homework, we will implement a simplified version of object detection process.

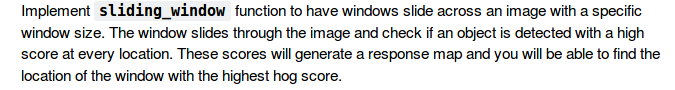
## 9.1 Hog Representation

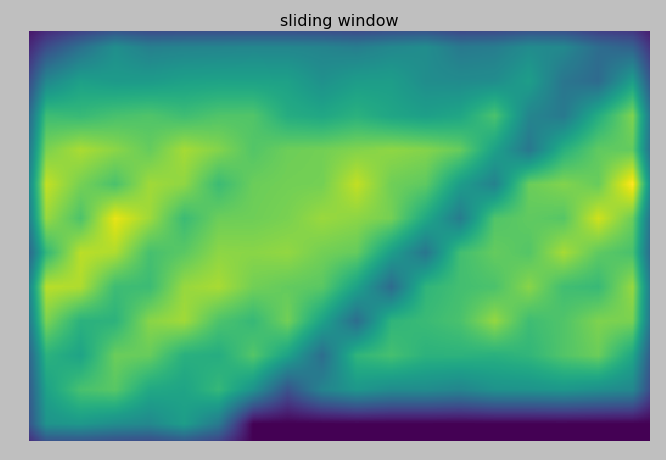
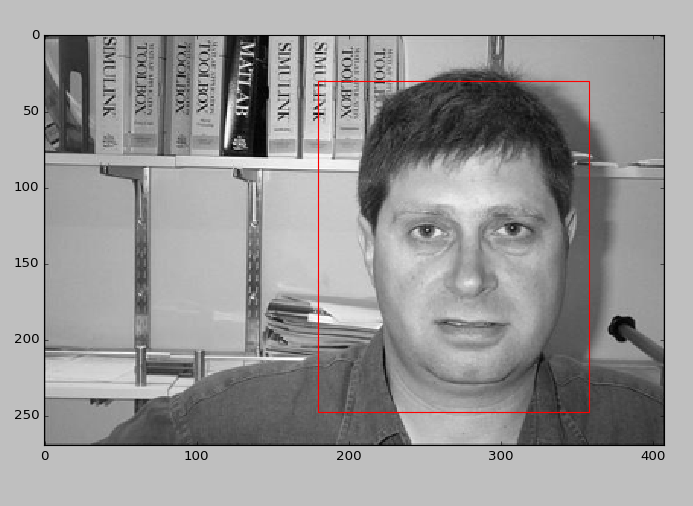
In this section, we will compute the average hog representation of human faces.

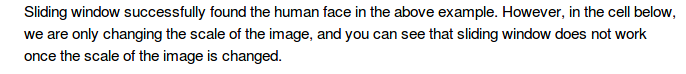
There are 31 aligned face images provided in the \face folder. They are all aligned and have the same size. We will get an average face from these images and compute a hog feature representation for the averaged face.

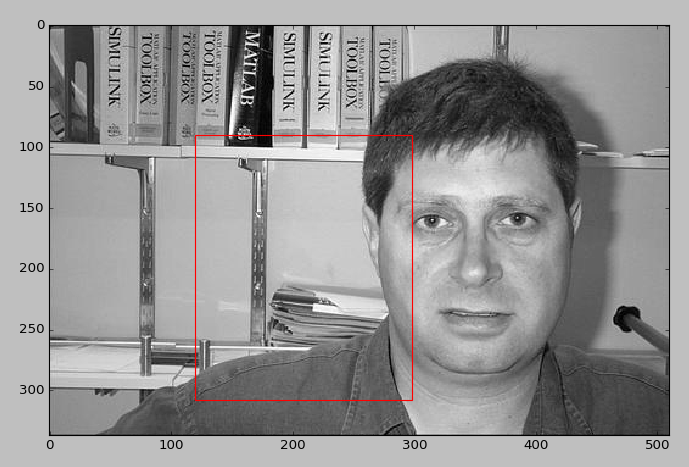


## 9.2 Sliding Window

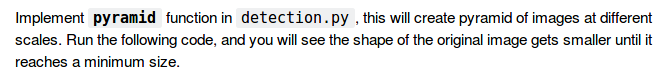






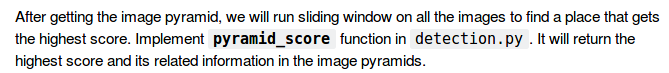


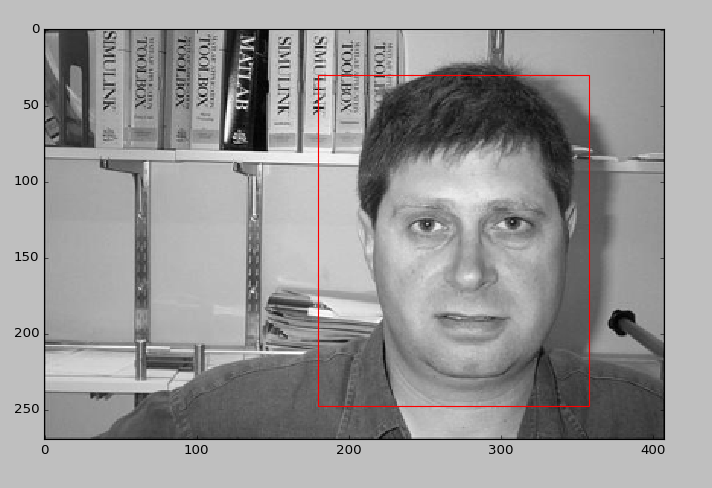
## 9.3 Image Pyramid

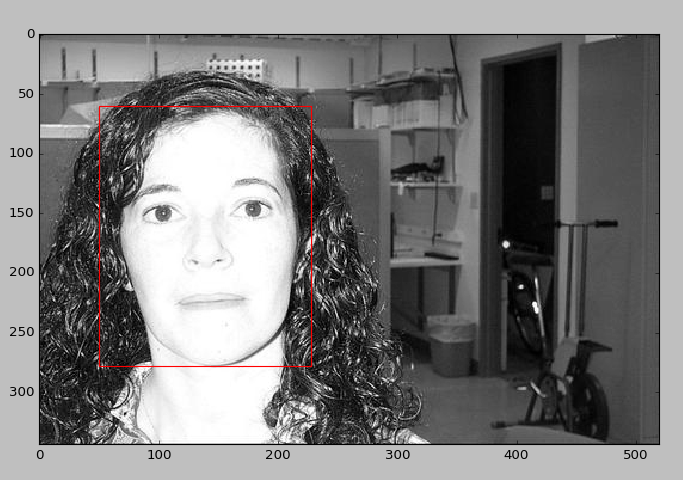


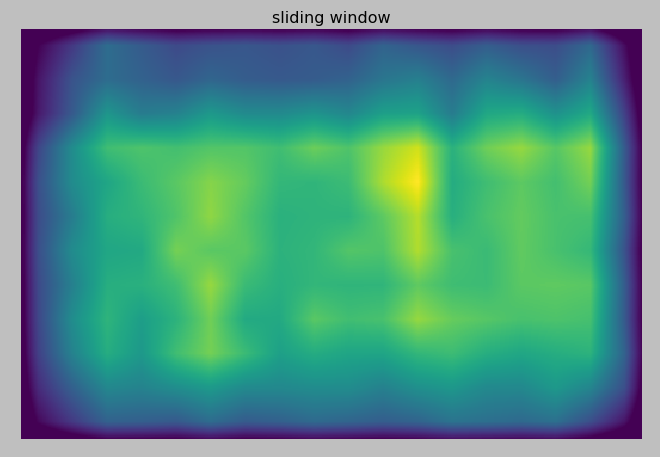


## 9.4 Pyramid Score

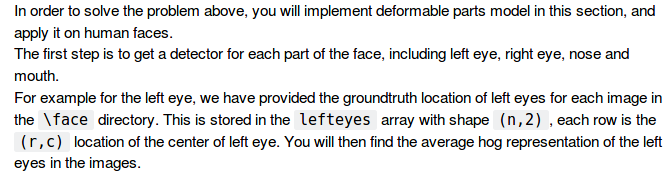


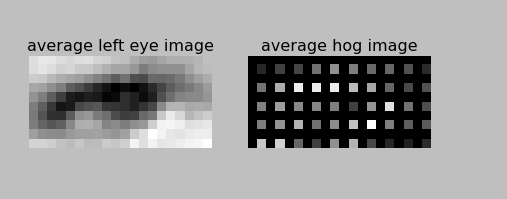


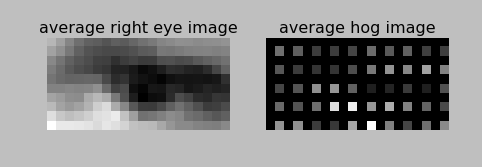


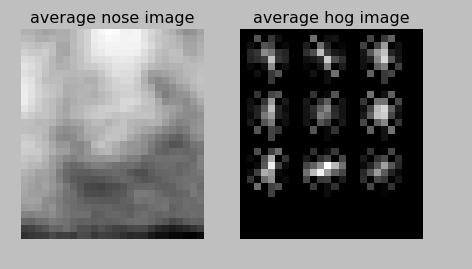


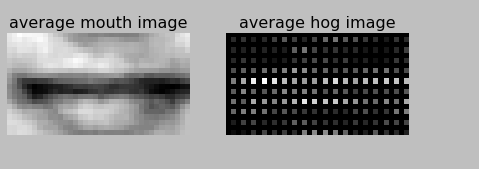
## 9.5 Deformable Parts Detection



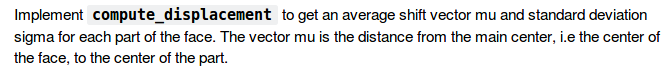


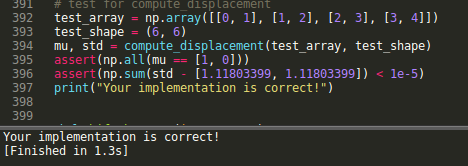


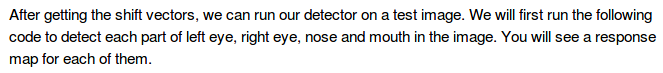




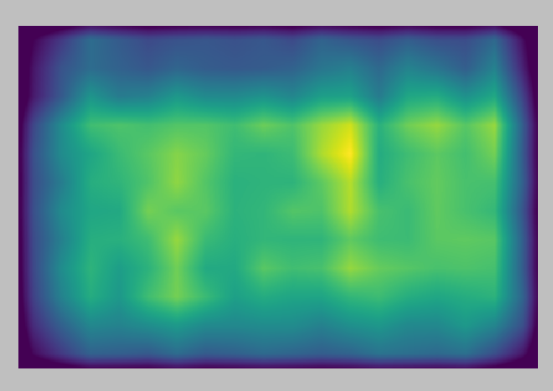
## 9.6 Human Parts Location

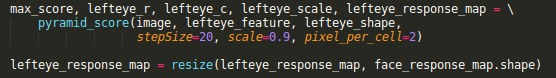


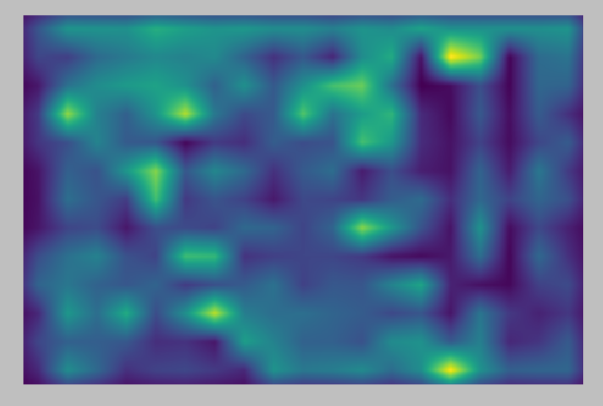


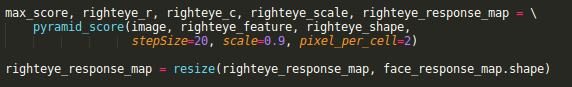


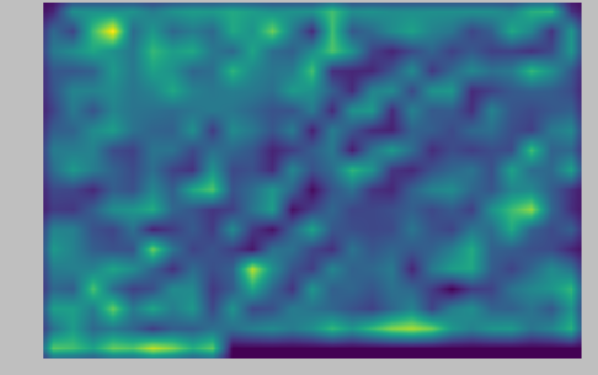


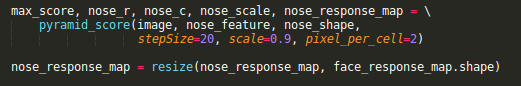


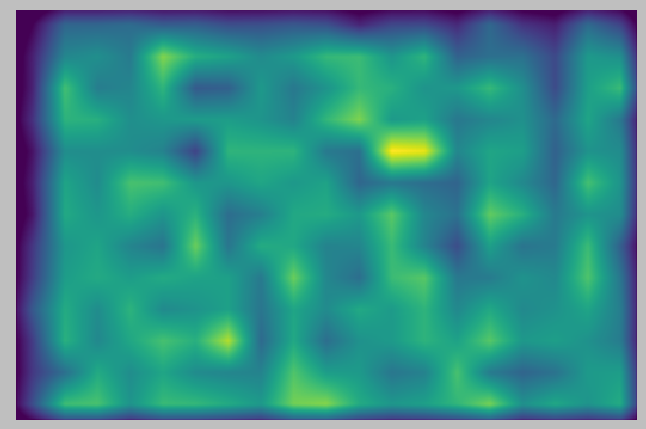


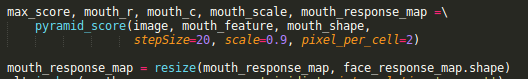




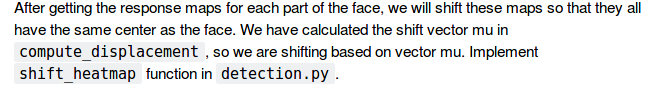


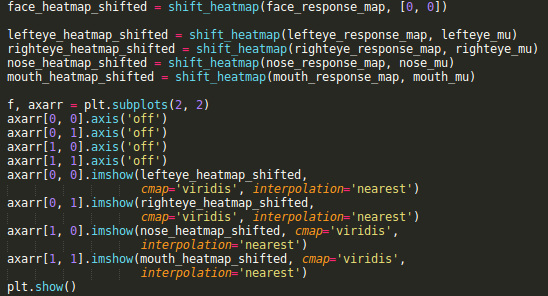


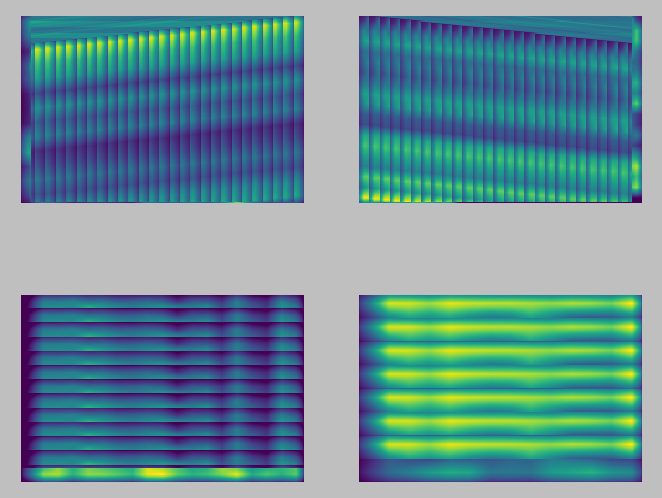




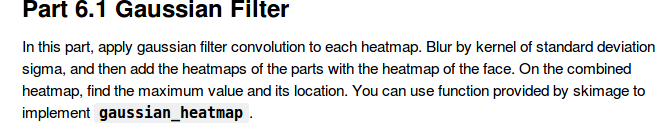




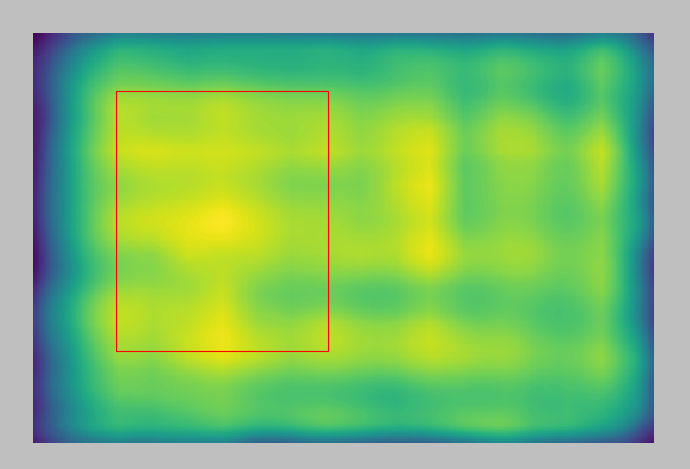
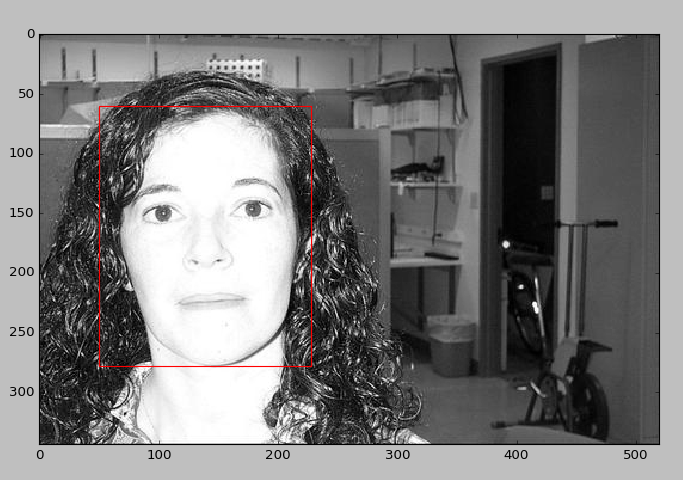




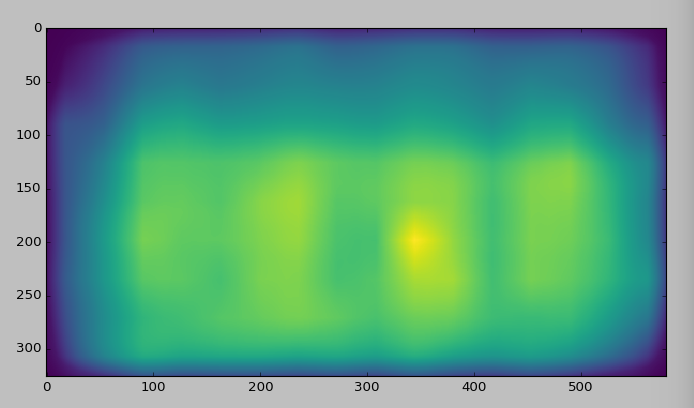
## 9.7 Gaussian Filter









### Tips:skimage.filters的Gaussian函数

