**使用判别训练的部件模型进行目标检测**

Pedro F. Felzenszwalb, Ross B.Girshick, David McAllester and Deva Ramanan

# [使用判别训练的部件模型进行目标检测 Object Detection with Discriminatively Trained Part Based Models](http://blog.csdn.net/masibuaa/article/details/17924671)

# ****摘要****

本文介绍了一个基于**混合多尺度可变形部件模型**(**mixtures of multiscale deformablepart model**) 的目标检测系统。此系统可以表示各种多变的目标并且在PASCAL目标检测挑战赛上达到了目前最优结果(state-of-the-art)。虽然可变形 部件模型现在很流行，但它的价值并没有在类似PASCAL这种较难的测试集上进行展示。此系统依赖于使用未完全标注(partially labeled)的样本进行**判别训练**的新方法。我们提出了一种间隔敏感(margin-sensitive)的难例挖掘方法(data-mining hard negativeexample)，称为**隐藏变量SVM**(latent SVM, **LSVM**)，是MI-SVM加入隐藏变量后的重新表示。LSVM的训练问题是一个**半凸规划**(semi-convex)问题，但如果将正样本的隐藏变量的值指定后，LSVM的训练问题变为**凸规划**问题。最终可以使用一个迭代训练方法来解决，此迭代算法不断交替地固定正样本的隐藏变量和最优化目标函数。

# ****关键词****

目标识别(ObjectRecognition)，**可变形模型**(**Deformable Models**)，**图结构模型**(**Pictorial Structures**)，**判别训练**(**Discriminative Training**)，**隐藏变量SVM**(**Latent SVM**)

# ****1 引言****

目标检测是计算机视觉领域内一项基础性的工作。本论文研究在静态图片中检测并定位某一类目标(例如人或车)的问题。由于这些类别中的目标外表可能千差万 别，使得此项工作变得有些复杂。而且，变化不仅来自亮度和视角，还有由于目标不是刚体而引起的形变，以及同一类目标的形状和其他视觉上的变化。例如，人可 能穿不同的衣服，做不同的姿势，车可能有不同的形状和颜色。

本文介绍了一个基于混合多尺度可变形部件模型的目标检测系统，它可以表示各种多变的目标。此模型使用判别程序进行训练，训练过程只需要用到图片集中目标的**矩形框**(**包围盒**)(**意思是说只需要整个目标的标注信息，不需要各个部件的标注信息**)。训练好的系统既高效又精确，能够在PASCAL VOC测试集**[11-13]**和INRIA人体测试集[[10]](http://blog.csdn.net/masibuaa/article/details/14056807" \t "http://blog.csdn.net/masibuaa/article/details/_blank)上达到目前最佳结果。

我们的方法基于**图结构**(**Pictorial Structures**)框架**[15][20]**。图结构使用一系列部件以及部件间的位置关系来表示目标。每个部件描述目标的一个局部属性，通过部件间的**弹簧连接**(**Spring-like Connection**)表示模型的可变形配置。

可变形部件模型(例如图结构)是目标检测中的优秀方法，但是很难在实际中建立价值。在一些难度大的数据集上，可变形部件模型经常被一些简单的模型——例如**固定**(刚体rigid)**模版[10]**或**特征袋**(**bag-of-features**)**[44]**所超越。本论文的目标之一就是解决这一问题。

虽然可变形模型可适应很多外表变化，但单个可变形模型还不足以表示一个变化丰富的目标类别。例如对图像中的自行车外表建模的问题。自行车有**各种类型**(例如，山地车，双人自行车，以及19世纪的有一个大轮一个小轮的自行车)，并且**观察视角也会有不同**(例如从前面看和从侧面看)，所以本文中的系统**使用混合模型来适应这些变化**。

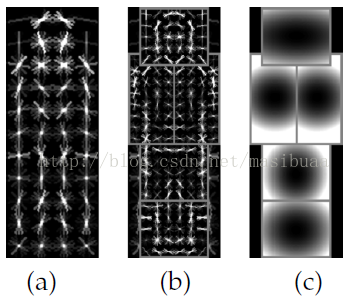
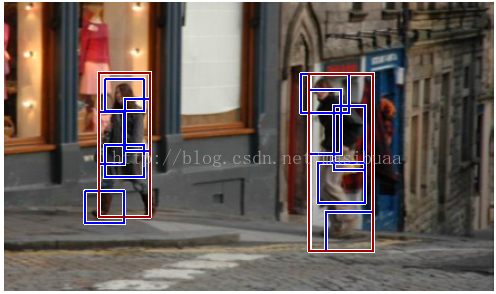
我们最终感兴趣的是使用**视觉语法**(**Visual Grammars**)对目标进行建模。基于语法的模型(例如**[16][24][45]**)使用可变层次结构来表示目标，是可变形部件模型的扩展和一般化。**基于语法的模型**(grammar based model)中的每个部件都可以被直接定义，或者根据其他部件进行定义。此外，基于语法的模型考虑到结构的变化。这些模型还提供在不同目标类别间共享信息和计算的框架，例如，**不同的模型共享可重用的部件**。

基于语法的模型是我们的终极目标，现在我们采取了一种研究方法，使用此方法使得我们可以在保证高性能的前提下逐步改进模型，使它变得更丰富。通过丰富模型来改进性能是非常困难的。在计算机视觉、语音识别、机器翻译和信息检索领域中，**一直以来简单模型都要比复杂模型表现更为出众**。例如，直到最近基于n元语言模型(n-gram language model)的语音识别和机器翻译系统才在性能上超过基于语法和词组结构的系统。根据我们的经验，只有逐步地丰富模型才能保证性能不降低。

之所以简单模型在实际中表现比复杂模型好的原因之一就在于**复杂模型很难训练**。对于目标检测，固定模型和特征袋模型很容易使用判别方法(例如SVM)进行训练。复杂模型很难训练，因为**复杂模型经常使用隐藏信息**。

例如从只标注了整个目标的包围盒的图片中训练一个基于部件的模型的难题。**因为部件的位置没有进行标注，所以这些信息在训练时只能被当做隐藏(latent或hidden)变量**。如果使用更完全的标注信息可能会训练出更好的模型，但也可能由于未能准确标注出各个部分的位置而导致更差的结果。通过自动发现有效部件来进行自动部件标注有可能达到更佳的性能。精细制作标注信息是费时而昂贵的。  
 Dalal和Triggs的检测器[[10]](http://blog.csdn.net/masibuaa/article/details/14056807" \t "http://blog.csdn.net/masibuaa/article/details/_blank)(在PASCAL 2006目标检测挑战赛上表现最好)使用基于HOG特征的单独**滤波器**(**模版**) 来表示目标。它使用滑动窗口方法，将滤波器应用到图像的所有可能位置和尺度。可以将此检测器看做一个分类器，它将一张图片以及图片中的一个位置和尺度作为 输入，然后判断在指定位置和尺度是否有目标类别的实例。考虑到此检测器只有一个滤波器，我们可以使用β·Φ(x)表示滤波器在某位置的得分，其中β是滤波 器参数，x是指定位置和尺度的图片，Φ(x)是x的HOG特征向量。Dalal-Triggs检测器的主要创新点是提出了一个非常有效的特征。

此论文的第一个创新点是丰富了Dalal-Triggs的模型，我们使用**星型结构**的部件模型，此模型由一个**根滤波器**(**root filter**，与Dalal-Triggs的滤波器相似)和一系列**部件滤波器**(**part filter**)以及相应的可变形模型构成。**星型模型在图像特定位置和尺度的得分**等于根滤波器在给定位置的得分加上各个部件的得分的总和，每个**部件的得分**等于此部件在所有空间位置的得分的最大值，**部件在某位置的得分**等于部件滤波器在此位置的得分减去此位置的变形花费，**位置的变形花费**衡量了部件**偏离**其理想位置(这里说的位置是指与根滤波器的相对位置)的程度。根滤波器和部件滤波器的得分都是由滤波器参数与特征金字塔中一个窗口的特征向量的**点积**(dot product)定义的。**图1**显示了人体的星型模型。



**图1**，**单组件**人体模型的检测结果。此模型由一个**粗糙的根滤波器**(a所示)，和**几个高分辨率的部件滤波器**(b所示)以及每个部件相对于根的**空间位置模型**(图c所示)组成。滤波器指定了HOG特征的权重。图中滤波器的可视化模型显示的是不同方向的正权重。空间位置模型的可视化图显示的将部件的中心放置到相对根的不同位置的变形花费(越白花费越高，表示部件偏离其理想位置越大)。

在我们的模型中，部件滤波器表示的图像特征是根滤波器所表示的图像特征所在的空间分辨率的两倍，也就是说我们是在多尺度对目标的外表建模。

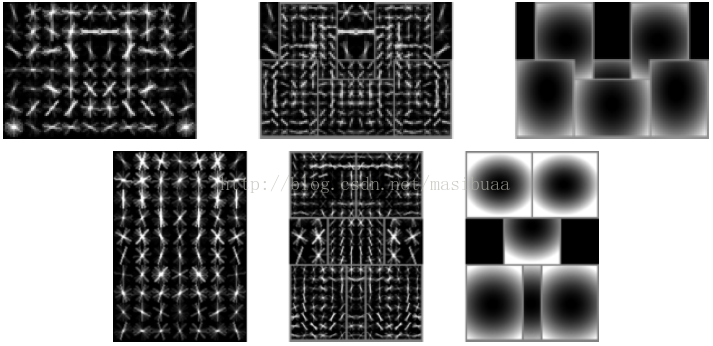
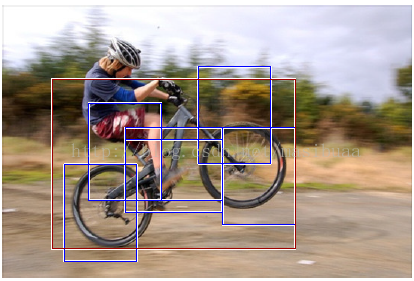
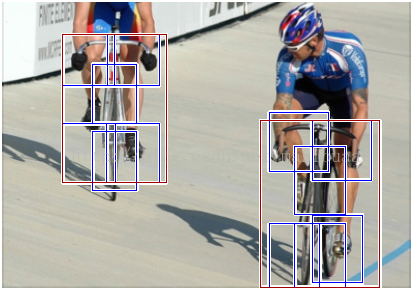
为了使用未完全标注(partially labeled，意思应该是目标的部件没有进行标注，只标注了整个目标)的数据进行模型训练，我们使用了论文**[3]**中的**多实例SVM**(**MI-SVM**)中的一个隐藏变量公式，我们称作**隐藏变量SVM**(**Latent SVM**,**LSVM**)。在LSVM中，每个样本x使用下面形式的公式进行评分：



其中β是模型参数向量，z是隐藏变量，Φ(x, z)是特征向量。模型参数β是根滤波器、部件滤波器、变形花费权重串联起来构成的参数向量，z是目标配置参数，Φ(x, z)是特征金字塔中的一个窗口对应的HOG特征和部件变形特征串联起来构成的特征向量。

我们意识到**公式(1)**可以处理更通用形式的隐藏信息，例如，z可以用来指定富视觉语法(rich visual grammar)中的派生词。

本文的第二种模型使用**混合星型模型**来表示目标类别。**混合模型**在指定位置和尺度的得分是各个**组件**(component)模型在给定位置得分的最大值。在这种情况下，隐藏变量z表示组件类别及组件配置。**图2**展示了自行车的混合模型。



**图2**，含有两个**组件**模型的**混合自行车模型**的检测结果。这几个例子表明了混合模型的重要性。第一个组件捕捉自行车的侧视特征，第二个组件捕捉自行车的正视或接近正视特征。侧视组件模型可以**变形来匹配自行车前轮抬起的姿势**。

为了通过判别训练来获得好的结果，往往需要使用大量训练样本。在目标检测中，训练问题是非常不平衡的，因为相比于特定目标来说有更大量的未知的背景。这就需要我们通过搜索背景数据来找到一个相对少量的潜在的**误报**(**虚警**)**的负样本集**，或者叫做**难例**(**负样本难例**，**HardNegative Example**)。

Dalal和Triggs在论文[[10]](http://blog.csdn.net/masibuaa/article/details/14056807" \t "http://blog.csdn.net/masibuaa/article/details/_blank)中采用了一种**对难例进行数据挖掘**的方法，但倒回到了1995年前后中用到的**自举法**(**bootstrap**)**[35][38]**。我们分析了SVM和LSVM训练中的数据挖掘算法，证明使用数据挖掘方法可以收敛到整个训练集上的最优模型。

本文中的目标模型是由滤波器定义的，滤波器可以对特征金字塔中的子窗口进行评分的。我们调查了与论文**[10]**中的HOG特征类似的特征，并找到了与原特征性能表现相同的维数更低的特征。通过对HOG特征做**主成分分析**(**Principal Component Analysis**)，可以大大减少特征向量的维数，同时不产生显著的信息丢失。此外，通过分析主特征向量，我们找到了一个容易解释并且可高效计算的低维特征。

本文还分析了PASCAL目标检测挑战赛和其他相似数据集中的一些特定问题。我们展示了**通过目标中部件的位置来估计目标的包围盒**的方法，这是通过用最小二乘回归训练的特定模型的预测器来实现的。我们还展示了一个用来**聚合几个目标检测器的输出结果**的 简单方法。此方法的基本思想是同一张图片中的某一类目标可以为其他类别的目标提供正例支持或反例驳斥。我们实现了这一思想，首先训练一个特定类别的分类 器，用此分类器对此类别所有检测出的目标进行重新评分，重新评分的依据是该目标的原始得分和用其他类别的分类器对此目标进行评分两者中的最高值。

# ****2 相关研究工作****

关于目标检测中各种类型的可变形模型已经有了大量研究工作，包括几种可变形模版模型(例如**[7][8][21][43]**)，和各种基于部件的模型(例如**[2][6][9][15][18][20][28][42]**)。

在**[18][42]**的**星座模型**中，部件被限制在由兴趣点所确定的稀疏(小的)位置集合中，并且他们的几何分布由高斯分布进行描述。相反，**图结构模型**(**Pictorial Structure Model**)**[15][20]**的匹配问题中，位于稠密位置集中的部件有各自独立的匹配花费，几何分布由部件两两之间的弹簧连接进行描述。**[2]**中的**部件拼接模型**(**patchwork of parts** model)也类似，但它清楚地考虑了重叠部件之间如何进行相互作用。

本文中所用的模型很大程度上基于论文**[15][20]**中的**图结构**(**Pictorial Structure**)框架。我们的模型中使用位置和尺度的稠密集，并定义了将部件滤波器放置在每个位置的得分。部件滤波器之间的几何配置由将每个部件滤波器与根滤波器连接的**变形花费**(**弹簧**)来描述，形成了一个星型的图结构模型。注意我们**并不对重叠部件之间的相互作用进行建模**，虽然如果对这种相互作用进行建模可能会有好处，但这并不是使用判别程序训练的模型中存在的问题，并且不对相互作用建模的话会大大简化模型与图像进行匹配的问题。

本文中提出的新的局部和半局部特征在改进目标检测方法的性能上起了重要作用，这些特征对光照变化和小的形变具有不变性。很多最近的目标检测方法使用**类小波特征**(Wavelet-like)**[30][41]**或局部归一化的梯度直方图特征**[10][29]**。其他的方法，例如**[5]**，从训练图片中学习局部框架词典。在我们的方法中，以论文[[10]](http://blog.csdn.net/masibuaa/article/details/14056807" \t "http://blog.csdn.net/masibuaa/article/details/_blank)中的HOG特征为基础，在不引起性能损失的前提下降低特征维数。与论文**[26]**中的方法类似，我们也使用**PCA**方法对特征进行降维，但我们还注意到获得的特征向量有清晰的结构，并最终找到一个解析特征集。这就省去了计算稠密特征映射时代价很大的投影步骤。

二维可变形模型不能很好地适应由于视角改变而引起的较大的目标形状和外形变化。方位图(Aspect Graph)**[31]**是一种经典的描述由于视角改变而引起的变化的方法。**混合模型**提供了一种简单的替代方法。例如，使用多个模型来编码人脸和汽车的正视图和侧视图**[36]**。混合模型也被用来描述其他的外表变化，例如目标类别有多个子类别时**[5]**。

将可变形模型与图片进行匹配是一个复杂的优化问题。局部搜索方法需要在正确的位置附近进行初始化**[2][7][43]**。为了保证全局最优匹配，需要更激进的搜索方法。其中一个流行的部件模型的搜索方法是将部件的位置约束在一个小的可能位置集合中，此位置集合是根据兴趣点检测得到的**[1][18][42]**。星型(树型)图结构模型**[9][15][19]**允许使用**动态规划**和**广义距离变换**方法来高效搜索所有可能的目标配置，不约束每个部件的可能位置，我们使用此方法来**匹配模型和图像**。

基于部件的可变形模型的参数由每个部件的外表以及描述部件间空间关系的几何模型来确定。学习模型时可以使用最大似然估计。在全监督条件下，训练图片中标注了每个部件的位置，所以可以使用简单的方法来学习模型**[9][15]**。在**弱监督**条件下，训练图片中并没有标注每个部件的位置，这时可以使用**EM算法[2][18][42]同时估计部件位置**和**学习模型参数**。

**判别训练**(**Discriminative Training**)方法通过最小化检测算法在训练集上的失败率来选择模型参数。这种方法可以直接优化正负样本间的决策边界。这也就是为什么使用判别训练方法训练的简单模型(例如Viola-Jones**[41]**和Dalal-Triggs[[10]](http://blog.csdn.net/masibuaa/article/details/14056807" \t "http://blog.csdn.net/masibuaa/article/details/_blank)的检测器)可以取得成功的原因。用判别方法来训练基于部件的模型更加困难，虽然对应的策略已经存在**[4][23][32][34]**。

**隐藏变量SVM**(**LSVM**)与**隐条件随机场**(**Hidden CRF**)**[32]**有关。然而，在LSVM中我们最大化隐藏的部件位置变量而不是忽视它们，在训练时使用**铰链损失**(**hinge-loss**)而不是**log-loss**。最终变为一个用来训练的**坐标下降算法**(**coordinate descent**)，以及一个可以用超大数据集进行学习的数据挖掘算法。LSVM可以看做**基于能量的模型[27]**的一种。

LSVM与**多实例学习**(Multiple InstanceLearning, **MIL**)**[3]**中的**MI-SVM**的公式是等价的，但我们发现LSVM的公式更适合我们的问题(我们在论文**[17]**中定义了LSVM，当时还未意识到与MI-SVM的关系)。此外，之前还有个不同的MIL框架被用来在弱标注的数据集上训练目标检测器**[40]**。

本文中使用的在训练时**对难例进行数据挖掘**(Data-mining HardExamples)的方法与SVM中的**工作集方法**(Working Set Method)(例如**[25]**)有关联。本文中的方法相对来说更少地遍历整个训练数据集，并且非常适合只有部分能读进内存的超大规模数据集的训练。

目标检测和识别中**图像上下文**(context)的使用近年来受到越来越多的关注。一些方法(例如**[39]**)使用低级别的整个图像的特征来定义可能的目标假设。**[22]**中的方法使用粗糙但语义丰富的场景表达，包括3D几何。本文中首先在图像上应用各种目标检测器，用检测结果来定义图像上下文。这种思想与论文**[33]**中使用CRF描述目标同时出现的方法相似，但我们使用不同的方法来描述此信息。

我们在论文**[17]**中介绍了系统的一个初步版本，本文中介绍的系统与**[17]**中有以下几点不同：**混合模型**的引入；使用**随机梯度下降算法**(**Stochastic Gradient Descent**)优化了真正的LSVM目标函数，论文**[17]**中是用SVM工具包优化目标函数的启发式近似；使用了**新的特征**，做到既低维又高信息量；通过包围盒预测和上下文再次评分，对检测结果进行后处理。

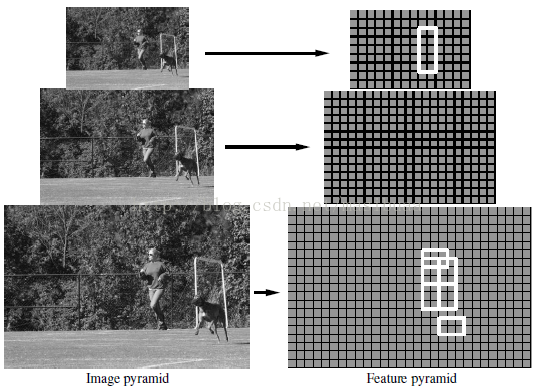
# ****3 模型****

我们的所有模型都涉及将线性滤波器应用到稠密**特征映射**中。**特征映射**(**feature map**)是一个数组，其元素由从一张图像中计算出来的所有d维特征向量组成，每个特征向量描述一块图像区域。实际中我们用的是论文[[10]](http://blog.csdn.net/masibuaa/article/details/14056807" \t "http://blog.csdn.net/masibuaa/article/details/_blank)中的HOG特征的变体，但这里讨论的框架是与特征选择无关的。

**滤波器**(filter)是由d维权重向量定义的一个矩形模版。滤波器F在特征映射G中位置(x,y)处的**响应值**(response)或**得分**是滤波器向量与以(x, y)为左上角点的子窗口的特征向量的**点积**(DotProduct)：



我们想要定义在图像不同位置和尺度的得分，这是通过使用**特征金字塔**来实现的，特征金字塔表示了一定范围内有限几个尺度构成的特征映射。首先通过不断的平滑和子采样计算一个标准的图像金字塔，然后计算金字塔中每层图像的所有特征。**图3**展示了特征金字塔。



**图3**，特征金字塔和其中的一个人体模型实例。**部件滤波器位于根滤波器两倍空间分辨率的金字塔层**。

根据参数λ在特征金字塔中进行尺度采样，λ定义了组中层的个数。也就是说，**λ是我们为了获得某一层的两倍分辨率而需要在金字塔中向下走的层数**。实际中在训练时λ=5，在测试时λ=10。**尺度空间的精细采样**对于我们的方法获得较高的性能表现非常重要。

[[10]](http://blog.csdn.net/masibuaa/article/details/14056807" \t "http://blog.csdn.net/masibuaa/article/details/_blank)中的系统使用单个滤波器来定义整个目标模型，它计算滤波器在HOG特征金字塔中所有位置和层的得分，通过对得分阈值化来检测目标。

假设F是一个w \* h 大小的滤波器，H是特征金字塔，p = (x, y, l)指定金字塔中l层的位置(x, y)，**φ(H,p, w, h)**表示金字塔H中以p为左上角点的w\*h大小的子窗口中的所有特征向量按行优先顺序串接起来得到的向量。滤波器F在位置p处的得分为F’·φ(H, p, w, h)，F’表示将F中的权重向量按行优先顺序串接起来得到的向量。此后我们使用F’·**φ(H,p)**代替F’·φ(H, p, w, h)，因为子窗口的大小已隐含包括在滤波器F中。

## ****3.1 可变形部件模型****

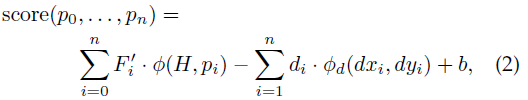
我们的星型模型由一个大体上**覆盖整个目标的粗糙的根滤波器**和**覆盖目标中较小部件的高分辨率的部件滤波器**构成。**图3**显示了特征金字塔中这种模型的一个实例。根滤波器定义了检测窗口(滤波器所覆盖的特征空间部分的像素)。部件滤波器被放置在根所在层的λ层之下，该层特征的分辨率是根所在层的特征的两倍。

我们发现**用高分辨率特征来定义部件滤波器对获得高识别性能至关重要**。用这种方法部件滤波器可以捕捉相对于根滤波器更精确定位的特征。例如建立人脸的模型，根滤波器捕捉的是人脸边界这些粗糙边缘信息，部件滤波器可以捕捉眼镜、鼻子、嘴这些细节信息。

含有n个部件的目标模型可以形式上定义为一个(n+2)元组：**(F0,P1,..., Pn, b)**，F0是根滤波器，Pi是第i个部件的模型，b是表示**偏差**的实数值。每个部件模型用一个三元组定义：**(Fi,vi, di)**，Fi是第i个部件的滤波器；vi是一个二维向量，指定第i个滤波器的**锚点位置**(**anchor position**，即未发生形变时的标准位置) 相对于根的坐标；di是一个四维向量，指定了一个二次函数的参数，此二次函数表示部件的每个可能位置相对于锚点位置的**变形花费**(**deformation cost**)。

每个**目标假设**都指定了模型中每个滤波器在特征金字塔中的位置：z = (p0,..., pn)，其中pi = (xi,yi, li)表示第i个滤波器所在的层和位置坐标。我们这里需要**每个部件所在层的特征分辨率都是根滤波器所在层的特征分辨率的两倍**，即li层特征是l0层特征的分辨率的两倍，并且li = l0 – λ(i > 0)。

**目标假设的得分**等于**每个滤波器在各自位置的得分**(从数据来看)减去**此位置相对于根位置的变形花费**(从空间来看)再加上**偏差值**：



其中，



给出了第i个部件相对于其锚点位置的位移：(x0,y0)是根滤波器在其所在层的坐标，为了统一到部件滤波器所在层需乘以2。vi是部件i的锚点相对于根的坐标偏移，所以2(x0, y0)+vi表示未发生形变时部件i的绝对坐标(**锚点的绝对坐标**)。

而

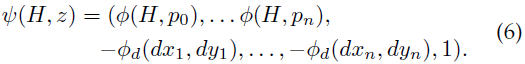


是**变形特征**(水平、垂直位移及其平方)。

如果di= (0, 0, 1, 1)，则第i个部件的变形花费就是它实际位置与锚点位置距离的平方。通常情况下，变形花费是位移的任意可拆分二次函数。

引入**偏差值**是为了在将多个模型组成**混合模型**时，使多个模型的得分具有可比性。

目标假设z的得分可以表示成点积的形式：β·ψ(H, z)，β是模型参数向量，ψ(H, z)是特征向量，如下：



这就将模型和线性分类器联系起来了，我们使用**隐藏变量SVM**(**LSVM**)来学习模型参数。

## ****3.2 匹配****

在图像中检测目标时，根据各个部件的最佳位置计算每个根位置的**综合得分**(**overallscore**)，如下：



高得分的根位置定义了一次检测，产生高得分根位置的部件位置定义了一个完整的目标假设。

通过定义每个根位置的**综合得分**(overall score)，我们可以检测目标的多个实例(假设每个根位置上最多一个实例)。这种方法与滑动窗口检测器有关联，因为可以认为score(p0) 是检测窗口在指定根位置的得分。

我们使用**动态规划**和**广义距离变换**(min-convolution)**[14][15]**来计算部件的最优位置(是根位置的函数)。此方法非常高效，花费O(nk)时间计算一个滤波器的响应值，n是模型中部件的个数，k是特征金字塔中位置的总数。我们在此简单介绍一下此方法，细节请参考论文**[14][15]**。

设是存放**部件i在特征金字塔第l层的响应值**的数组。匹配算法首先会计算这些响应值。注意Ri, l是滤波器Fi和特征金字塔第l层的交叉相关。



计算完这些滤波器响应值后，对其进行转换来允许具有空间不确定性：

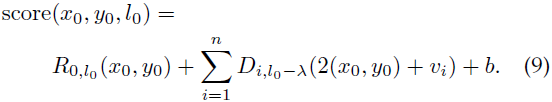
(8)



这种变换会将滤波器高分扩展到邻近位置，同时也将变形花费考虑在内。Di, l(x, y) 值表示将第i个部件的锚点放在l层的位置(x,y)时它对根位置得分的最大贡献值。

转换后的数组Di,l可根据论文**[14]**中的**广义距离变换**算法在线性时间内从数组Ri, l计算得到。

每一层根位置的**综合得分**可以表示为该层根滤波器响应值加上经过变换和子采样的部件响应值，

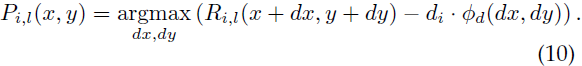


前面说过，λ是我们为了获得某一层的两倍分辨率而需要在金字塔中向下走的层数。

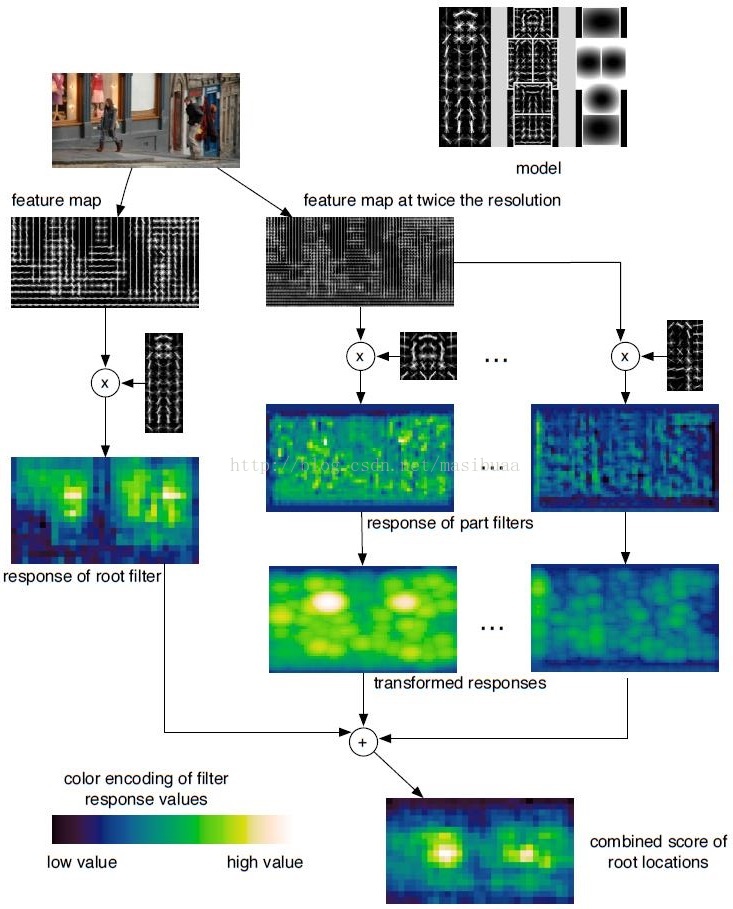
**图4**阐述了匹配过程。

为了理解**公式(9)**，需要知道对于一个固定的根位置，我们可以独立的选择每个部件的最佳位置，因为在目标假设的得分计算中没有考虑部件之间的相互作用。**变换后的数组Di,l表示第i个部件对根综合得分的贡献值**，它是部件锚点位置的函数。所以，通过**将根滤波器响应值和各个部件的贡献值相加，就得到了l层一个根位置的综合得分**，其中每个部件的贡献值已预先计算好存储在数组Di, l-λ中。

此外，在计算Di,l的过程中**[14]**中的算法还能计算出部件的最优位置(是锚点坐标的函数)，



找到一个高分的根位置(x0, y0,l0)后，可以在中查找对应的部件最佳位置。



**图4**，在某一尺度上的匹配过程。根滤波器和部件滤波器的响应值在特征金字塔的不同分辨率中进行计算。**变换后的响应值**(Di, l)和根滤波器的响应值一起产生出每个根位置的最终得分。图中显示了“头部”和“右肩”的响应值及变换后响应值。注意到“头部”滤波器区分度更大。最终根位置的综合得分清楚地表明在本尺度有两个目标假设。

## ****3.3 混合模型****

含有m个**组件**(component)的混合模型可由一个m元组定义：M = (M1, ..., Mm)，其中Mc是第c个组件的模型。

一个混合模型的**目标假设**指定了一个组件Mc(1<=c<=m)以及Mc中每个滤波器的位置：，这里nc表示模型Mc中的部件个数。此混合模型目标假设的得分等于第c个组件模型的目标假设的得分。



如果是单组件混合模型，混合模型的目标假设得分可以用模型参数向量β和特征向量ψ(H, z)的点积来表示。如果是多组件混合模型，向量β是每个组件的模型参数向量的串接。特征向量ψ(H, z)是一个稀疏向量，其中的非零元素为ψ(H, z’)，与β中的βc位置相对应，



所以有



用混合模型检测目标时，使用上节介绍的匹配算法，找到对所有部件产生高分假设的根位置。

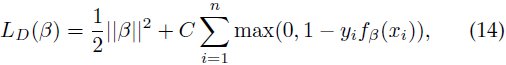
# ****4 隐藏变量SVM****

考虑对样本x用如下公式进行评分的分类器，



这里β是模型参数向量，z是隐藏变量。集合Z(x)定义了样本x所有可能的隐藏变量值。通过对此得分值进行阈值化，可以获得样本x的二分类类标。

类比经典SVM算法，我们使用带类标的样本集D = (<x1, y1>, ..., <xn,yn>)，yi∈{-1, 1}来训练参数β，通过最小化下面的目标函数：



其中是标准**铰链损失函数**(**hinge loss**)，常数C控制正则项的相对权重。



如果每个样本的隐藏变量有唯一可能值(|Z(xi)|=1)，则是β的线性函数，此时变为线性SVM问题，这是LSVM的一个特例。



## ****4.1 半凸规划****

LSVM最终是一个**非凸规划**(**non-convex**)问题。然而，在下面所讨论的情况下LSVM是**半凸规划**(**semi-convexity**)问题，一旦将隐藏信息指定给正样本则训练问题变为**凸规划**问题。

几个凸函数的最大值问题是凸规划问题。在线性SVM中，有是β的线性函数，此时**铰链损失函数**对于每个样本都是凸的，因为它是两个凸函数的最大值。



注意到**公式(13)**中定义的是一系列函数的最大值，而这些函数都是β的线性函数，因此是β的凸函数。所以当yi = -1时，两个函数f(x) = 0和都是β的凸函数，所以铰链损失函数是β的凸函数。也就是说，只有当样本为负样本时，损失函数是β的凸函数。我们将损失函数的这一性质叫做**半凸**(**semi-convexity**)。



对于正样本(yi= 1)来说，LSVM的铰链损失函数不是凸函数，因为它是一个凸函数f(x)= 0 和一个凹函数 的最大值。



但是，当LSVM的正样本的隐藏变量具有唯一可能的取值时，是β的线性函数，因此损失函数是β的凸函数，再加上半凸性质，**公式(14)**变为凸规划问题。



## ****4.2 最优化****

设Zp指定训练集D中正样本的隐藏变量值。定义一个辅助目标函数：



其中D(Zp)表示将D中正样本的隐藏变量取值约束在Zp中而得到的一个子集。也就是，对于正样本，设置Z(xi) = {zi}，根据前面的定义，集合Z(x)定义了样本x所有可能的隐藏变量值，zi是由Zp指定的xi的隐藏变量值。注意到



特别的，。辅助目标函数给出了LSVM目标函数的上界，这证明了通过最小化LD(β, Zp)来训练LSVM是合理的。



在实际中我们使用**坐标下降算法**(**Coordinate Descent**)来最小化LD(β, Zp)：

1)**估计部件位置**(重新标注正样本)：通过选择每个正样本的最高得分的隐藏值来最优化Zp，



2)**学习模型参数**(最优化β)：通过解决定义的凸优化问题来得到最优β值。



以上两个步骤或者优化LD(β, Zp)的值，或者维持其不变。在算法达到收敛之后就会得到一个相对强壮的局部最优值，**步骤1在指数空间内搜索正样本的隐藏变量值**，同时**步骤2搜索所有可能的模型和指数空间内负样本的隐藏变量**。

需要注意的是，**模型参数β必须合理进行初始化**，否则在**步骤1**中可能选择出非法的隐藏变量值，这会导致产生错误模型。

LSVM的半凸性质非常重要，正是由于具有半凸性质所以**步骤2**中的问题变为容易解决的凸优化问题(即使负样本的隐藏变量值是不确定的)。还有一种相似的处理过程，在每一轮迭代时固定所有样本的隐藏变量值，但这可能无法产生最优结果。假设Z指定训练集D中**所有样本**的隐藏变量值。由于LD(β)遇到负样本时可能最大化，所以LD(β)可能比LD(β,Z)还大，因此不能指望通过最小化LD(β,Z)来获得一个好的模型。

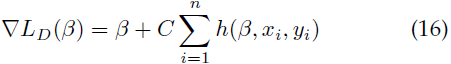
## ****4.3 随机梯度下降****

坐标下降算法中的**步骤2**(最优化β)可以通过**二次规划**(**Quadratic Programming**)**[3]**来解决，也可以通过**随机梯度下降算法**(**Stochastic Gradient Descent**)来解决。这里我们介绍一种在任意训练集D上用梯度下降算法来优化β的方法。在实际中我们使用这种方法的一个带有D(Zp)的特征向量缓存的版本(见**4.5节**)。

设，即zi(β)是使正样本xi得分最高的隐藏变量值(位置配置)。所以有。



计算LSVM目标函数(**公式14**)的梯度，如下



在随机梯度下降算法中，用样本的子集来近似▽LD，然后向梯度下降的方向走一步。用来近似表示。算法按如下步骤迭代更新β值：



1) 设αt是第t次迭代的学习率(步长)。

2) 随机选取样本xi。

3) 设



4) 如果，则β∶＝β－αtβ



5) 否则，



就像线性SVM中用的坐标下降方法，我们这里的方法也和**感知机算法**(**Perceptron**)很相似。如果当前对随机样本xi分类正确(大于函数间隔，如**步骤4**所述)，则收缩β；否则(如**步骤5**所述)，收缩β之后再加上Φ(xi, zi)乘以一个标量。



对于线性SVM来说，**学习率**(**步长**)αt = 1/t 时表现不错**[37]**。然而，收敛时间依赖于训练样本个数，对我们的问题来说，样本数非常巨大。特别的，如果存在很多“简单”样本(即信息量比较小的样本)，**步骤2**会经常选中这些简单样本，这样迭代过程会进展比较缓慢。

## ****4.4 难例挖掘，SVM版****

在训练目标检测的模型时，经常需要用到非常大量的负样本(对扫描窗口分类器来说，一张图片就可以产生大约105个负样本)，这使得同时处理所有负样本变得不可能。相反，用正样本和**负样本难例**(**Hard Negative**)来构成训练数据集变得更为常用。

**自举法**(**Bootstrap**)首先使用初始负样本集来训练一个模型，然后收集被这个初始模型错误分类的负样本来形成一个负样本难例集。用此负样本难例集训练新的模型，此过程可以重复多次。

这里我们介绍一种受自举法启发而提出的适用于经典(非隐藏变量)SVM的数据挖掘算法。此方法使用一个相对较小的难例集来解决一系列训练问题，并且可以收敛到由更大数据集定义的训练问题的精确解。此方法需要用到难例的间隔敏感定义。

训练集D关于分类器β的困难和简单实例的定义如下：



也就是说，**H(β,D)是D中被分类器β错误分类或位于分类器间隔内的样本**；**E(β,D)是D中被分类器β正确分类并位于分类间隔外的样本**。在分类器边界上的样本既不是困难样本也不是简单样本。

设，即**β\*(D)表示样本集D上的最优β值**(或者说**根据样本集D训练得到的模型参数β**)。



由于LD是严格凸的(**为什么？不是半凸吗？**)，所以β\*(D)是唯一的。

给定一个大训练集D，我们想要找到一个较小的样本集，使得β\*(C) = β\*(D)。



我们的方法需要用到一个**样本缓存**(cache)，从初始样本缓存开始，交替训练模型和更新样本缓存。在每次迭代时，从样本缓存中移除简单样本，然后加入新的难例。所有正样本一直在样本缓存中不动，在负样本上进行难例挖掘。

设是初始样本缓存，算法按如下步骤迭代的**训练模型**和**更新缓存**：



1) 令βt := β\*(Ct)，即用Ct训练一个模型βt。

2) 如果，即βt在样本集D上获得的难例都已包括在Ct中，停止迭代，返回模型βt。



3) 对任意，令，即收缩缓存。



4) 对任意并且，令 ，即扩大缓存。



在**步骤3**中，通过删除Ct中βt对应的简单样本来收缩缓存。在**步骤4**中，通过向缓存中加入难例来扩大缓存，这些难例是用βt在D上获得的并至少有一个不在原Ct中，这样的样本肯定是存在的，否则在**步骤2**中就已经返回了。

下面的定理说明当停止迭代时可以找到β\*(D)。

**定理1 设 并且β=β\*(C)，如果，则β=β\*(D)。**



**证明**：因为，所以LD(β\*(D))≥ LC(β\*(C)) = LC(β)(**不明白为什么是大于等于？**)。因为，所以集合D\C中的所有样本的损失函数(铰链损失)值都是0，所以LC(β) = LD(β)(这是因为函数LD(β)由两部分构成，前面是β的平方，后面是所有样本的损失函数值累加和)。所以可得LD(β\*(D)) ≥ LD(β)，又因为LD有唯一的最小值，所以β=β\*(D)。



下一个定理说明算法在有限次迭代后会停止。直观上看是由于 会在每次迭代时变大，但最大不会超过LD(β\*(D))。



**定理2 本数据挖掘算法可收敛。**

**证明**：收缩后的样本缓存包含Ct中所有相对于βt 损失函数值不为零的样本。这表明对于βt 来说，和是等价的。既然βt 是的最小值，所以也必定是的最小值，所以。



在扩大缓存时，中至少包含一个对于βt损失函数不为0的样本<x, y>。因为，所以对于所有β有 。如果 ，则(**不明白为什么是大于等于？**)，因为有唯一最小值。如果，由于存在至少一个样本<x, y>的损失函数不为0，所以有 。



所以我们得出结论。因为缓存数量有限，所以缓存中损失的增长次数也有限。



## ****4.5 难例挖掘，LSVM版****

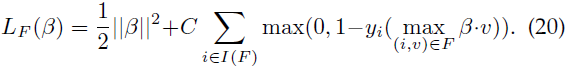
现在我们介绍一种适用于LSVM的难例挖掘算法，此算法要求LSVM正样本的隐藏变量值是固定的。也就是说，我们最优化 而不是LD(β)。基于前面的讨论，我们知道这个约束可以保证最优化问题是凸规划。



对于LSVM来说，用样本和隐藏变量二元组(x, z)的缓存来代替样本缓存。这样可以避免在优化算法(例如梯度下降)的内循环中对所有Z(x)做判断。此外，在实际中我们使用特征向量Φ(x, z)的缓存来代替二元组(x, z)的缓存，这样表示更简单紧凑。

**特征向量缓存F**由一系列二元组(i, v)构成，其中1≤i≤n是样本索引，v = Φ(xi , z)，z∈Z(xi)。注意对于每个样本xi，可能有多个二元组(i, v) ∈F。

设**I(F)**是缓存F对应的样本集合。F中的特征向量定义了β的目标函数(**公式20**)，在此函数中我们只考虑由I(F)指定的样本，对每个样本只考虑其在缓存F中的特征向量。



对**4.3节**中的梯度下降算法进行修改后可用来最优化LF。设**V(i)**是F中的特征向量的集合。**梯度下降算法**的迭代过程如下：

1) 设αt是第t次迭代的学习率(步长)。

2) 从I(F)中随机选择样本i

3) 令



4) 如果yi(β·vi) ≥1，令β＝β﹣αt β

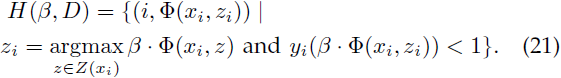
5) 否则，令β＝β﹣αt(β－Cnyivi)

注意，I(F)的大小控制着收敛所必需的迭代次数，V(i)控制执行**步骤3**所需要的时间。在**步骤5**中，n = |I(F)|。

设，即β\*(F)是根据特征向量缓存F训练得到的模型参数。我们想要找到D(Zp)一个较小的缓存F，使得β\*(F) = β\*( D(Zp) )。



定义训练集D上关于模型β的**难特征向量**(HardFeature Vector)为：



也就是说，难特征向量集H(β, D)由一系列二元组(i, v)组成，其中v是位于分类器β的分类间隔内(即分错类或无法分类的)的样本xi的最高得分特征向量。

定义缓存F中的**简单特征向量**为：



即缓存F中位于分类器β的分类间隔外(即分类正确的)的特征向量。

对于样本i，如果存在一个特征向量使得yi(β·v) ≤1，则即使样本i有其他更高得分的特征向量，也不再将(i, v)看做简单样本。

下面介绍计算β\*(D(Zp) )的数据挖掘算法。

此算法需要用到训练集D(Zp)的一个**特征向量缓存**，算法交替**训练模型**和**更新缓存**。

设F1是初始特征向量缓存，算法迭代过程如下：

1) 令βt := β\*(Ft)，即用Ft训练一个模型βt。

2) 如果，即βt在D(Zp)上找到的所有难特征向量都已包括在Ft中，则停止迭代，返回模型βt。



3) 对任意，令 ，即收缩缓存。



4) 对任意，令 ，即扩大缓存。



在**步骤3**中，通过删除简单特征向量来收缩缓存。在**步骤4**中，通过向缓存中加入新的特征向量来扩大缓存，这些特征向量是用βt在D(Zp)上获得的并至少有一个不在原Ft中。随着时间的推移，缓存中会累积起同个负样本的多个特征向量。

算法最终会收敛并返回β\*(D(Zp) )，与**4.4节**中类似。

# ****5 训练模型****

现在考虑从标注有目标包围盒的图片数据集中训练模型的问题。这种类型的数据可以从PASCAL数据集中获得。数据集中包含几千张图片，每张图片对应一个标注文件，标注文件指明了图片中出现的每个目标的类别以及目标的包围盒。需要注意的是，这是一种**弱标注**(weakly labeled)的数据集，因为其中没有指明部件的类别和位置。

我们介绍一种初始化混合模型结构并学习所有参数的方法。参数是通过训练LSVM来学习到的。我们用**4.2节**中描述的**坐标下降算法**，以及**4.5节**中的带有特征向量缓存的**数据挖掘**和**梯度下降算法**来训练LSVM。由于坐标下降算法容易受到局部极小值影响，所以必须合理进行模型的初始化。

## ****5.1 学习参数****

设c是一个目标类别。假设c的训练样本由正样本集P和背景图像集(负样本集)N给定。P是二元组(I, B)的集合，其中I是图片，B是I中c类目标的包围盒。

设M是一个有固定结构的(混合)模型。如前所述，一个模型的参数可由向量β定义。为了学习β，我们定义了一个带有隐式训练集D的LSVM训练问题，正样本来自集合P，负样本来自集合N。

每个样本<x,y>∈D都有一个相关的图片和特征金字塔H(x)。隐藏变量z∈Z(x)指明了模型M在特征金字塔H(x)中的一个实例。

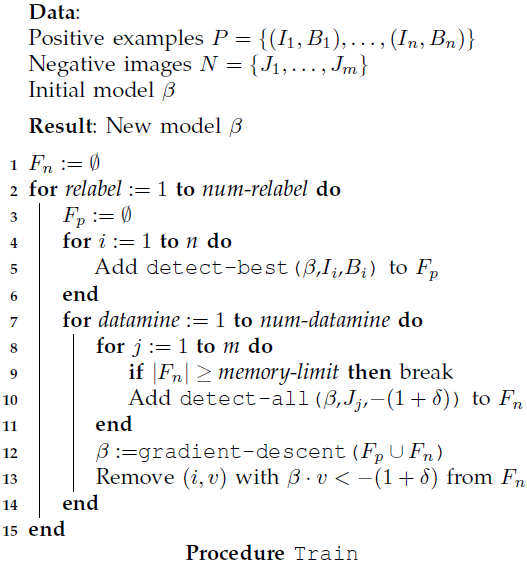
定义**Φ(x,z) = ψ(H(x), z)**，所以β·Φ(x, z)是模型M的假设z在H(x)上的得分。

(I, B)∈P指明了目标检测器应该在包围盒B定义的位置处检测出目标(“开火”)，也就是说由B定义的根滤波器位置的总体得分(**公式7**)应该很高。

对于每个(I,B)∈P，定义一个用来训练LSVM的正样本x。定义隐藏变量集合Z(x)，使得假设z∈Z(x)指定的根滤波器窗口与B重叠至少50%。我们发现将根滤波器的位置看做隐藏变量有助于补偿正样本标注中的噪声造成的损失。相似的思想在论文**[40]**中也有使用。

对于背景图(**负样本原图**)I∈N，我们不希望检测器在I的特征金字塔的任何位置检测出目标。这意味着每个根滤波器的总体得分(**公式7**) 都应该很低。设G是特征金字塔中位置的稠密集。对于G中的每个位置(i,j,l)，定义一个负样本x。定义集合Z(x)，使得z∈Z(x)指定的根滤波器 的层级是l，其检测窗口的中心坐标是(i, j)。从每张图片中可以获得大量的负样本，这与扫描窗口检测器需要有较低的误报率是相一致的。

下面给出了训练过程**Train**。最外层循环限定了**坐标下降算法**在LD(β, Zp)上的迭代次数。3-6行实现了坐标下降算法中(**步骤1**)**重新标注正样本**(**估计部件位置**)的过程，执行完后每个正样本对应一个特征向量，保存在正特征向量集合Fp中。7-14行实现了坐标下降算法中(**步骤2**)**最优化β**的过程。由于由N定义的负样本个数非常大，我们使用LSVM数据挖掘算法进行训练。我们并非运行算法直到收敛，而是进行固定次数的迭代。在每次迭代中，首先将难例收集到Fn中，用**梯度下降算法**训练一个新的模型，然后通过移除简单特征向量来收缩Fn。在数据挖掘过程中，通过在N中的图像上顺序迭代来增大缓存，直到达到内存限制。



函数detect-best(β,I, B)在图像I中用与矩形框B重叠超过50%的根滤波器寻找具有最高得分的目标假设。函数detect-all(β, I, t)计算每个根位置的最佳目标假设并选择其中得分大于t的。这两个函数都可以用**3.2节**中的匹配过程实现。

函数gradient-descent(F)利用缓存F中的特征向量来训练模型β(如**4.5节**所述)。在实际中会对算法进行修改，将可变形模型中的二次项参数限定在大于0.01，这保证了变形花费是凸函数，但又不至于太平坦。同时还**将模型限定为沿垂直轴对称**，位于垂直轴中心的滤波器必须是自对称的，不在垂直轴上的部件滤波器在模型的另一侧一定有一个对称部件。这些约束可以将要学习的参数个数降低一半。

## ****5.2 初始化****

LSVM中使用的坐标下降算法容易受到局部极小值的影响，对初始化很敏感。这也是其他使用隐藏信息的算法所共有的缺点。我们分三个阶段来初始化和训练混合模型。

**Phase1 初始化根滤波器**：为了训练含m个**组件**(component)的混合模型，我们首先将正样本集P中的包围盒按其**长宽比**(aspectratio)排序并分类为P1, ..., Pm组，每组内的包围盒具有相似的长宽比。这里，长宽比被当做区分同类目标的外观变化的指示器。共训练m个不同的根滤波器F1, ..., Fm，每组包围盒对应一个。

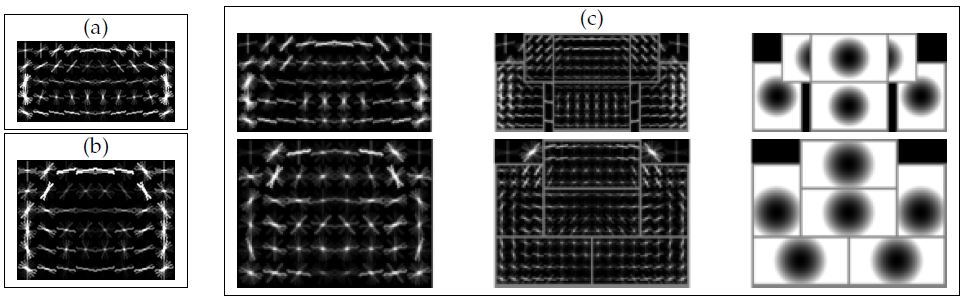
为了确定滤波器Fi的维数，选择Pi中包围盒的平均长宽比，并保证其面积不大于Pi中80%的包围盒。这就保证了对于Pi中的多数二元组(I, B)，可以将Fi放在I的特征金字塔中使Fi与B显著重叠。

我们用标准SVM(不含隐藏信息)来训练Fi，类似论文[[10]](http://blog.csdn.net/masibuaa/article/details/14056807" \t "http://blog.csdn.net/masibuaa/article/details/_blank)中的方法。对于(I, B) ∈Pi，缩放包围盒B中的图像区域使得其特征映射与滤波器Fi的维数相同，这样就产生了一个正样本。从负样本集N中的图片上随机选择合适维数的子窗口来产生负样本。**图5(a)**和**5(b)**显示了训练含两个组件的汽车模型的Phase1的结果。

**Phase2 合并组件**：将m个初始根滤波器联合起来形成不含部件的混合模型，在完整数据集P和N上(不分组不缩放)用**5.1节**中的**Train**过程重新训练联合模型的参数。这种情况下**组件标识**和**根位置**成为每个样本的隐藏变量。此时坐标下降训练算法可以看做一个判别聚类方法，不断迭代地给每个正样本分配聚类标识和估计聚类均值(根滤波器)。

**Phase3 初始化部件滤波器**：我们使用一个简单的**启发式算法**来初始化每个组件模型中的部件。每个组件模型的部件个数固定为6个，**贪心**地放置部件在根滤波器的**高能量区域**(能量是子窗口内正权重的范数)。部件或者放在根滤波器的中心垂直轴上，或者**关于中轴两两对称**。一个部件的位置确定后，它所覆盖区域的能量值设为0，然后继续寻找下一个最高能量区域，直到6个部件的位置都确定。

将根滤波器的值插值到两倍空间分辨率，用来初始化部件滤波器。每个部件的变形参数初始化为di = (0, 0, .1, .1)，这使得部件的位置与其锚点位置很近。**图5(c)**展示了训练含两个组件的汽车模型的Phase3的结果。结果作为最后一轮参数学习的初始模型(**？**)，最终的汽车模型如**图9**所示。



**图5**，(a)和(b)是汽车模型的初始根滤波器(即初始化过程中Phase1的结果)。(c)是汽车模型的初始部件滤波器(即初始化过程中Phase3的结果)

# ****6 特征****

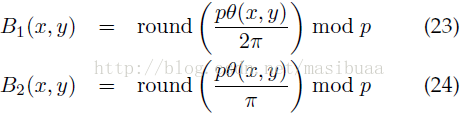
下面介绍来自论文[[10]](http://blog.csdn.net/masibuaa/article/details/14056807" \t "http://blog.csdn.net/masibuaa/article/details/_blank)的36维HOG特征(这里所用的HOG特征与论文**[10]**中稍有不同，但并不影响性能)，以及可选的包含相同信息13维特征集。我们发现通过包含对比度敏感和对比度不敏感的特征来扩大此低维特征集到31维特征向量，可以改善在大多数PASCAL目标类别上的表现性能。

## ****6.1 HOG特征****

### ****6.1.1 像素级特征映射****

令θ(x,y)和γ(x,y)分别表示(x,y)处像素点的**梯度方向**和**梯度幅值**。如论文**[10]**中所述，使用有限差分滤波器[-1,0,+1]及其转置来计算梯度。对于彩色图像，使用具有最大梯度幅值的颜色通道来定义θ和γ。

每个像素点的梯度方向离散到p个值中的一个，可以使用**对比度敏感**定义B1(方向范围为0-360)或**对比度不敏感**定义B2(方向范围为0-180)，



下面我们用B表示B1或B2。

我们定义一个像素级的特征映射，它指定了每个像素梯度幅值的稀疏直方图。设b取值范围为{0, ..., p-1}，则(x,y) 处的特征向量为



可以认为F是有p个方向通道的有向边映射，对于每个像素，通过离散化其梯度方向来选择一个方向通道，梯度幅值看做是有向边的强度。

### ****6.1.2 空间聚合****

设F是一个w\*h大小图像的像素级特征映射。k>0是一个方形图像区域的边长。定义一个矩形cell的稠密网格，将像素级特征聚合来获得基于cell的特征映射C，特征向量为C(i,j) 。空间聚合使特征具有对微小形变的不变性，并可以减少特征映射尺寸。



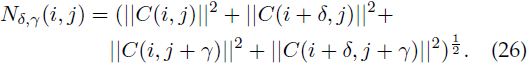
最简单的特征聚合方法是将像素(x,y)映射到一个cell 中，定义cell的特征向量是此cell内所有像素的特征向量的和(或均值)。



我们这里没有将每个像素映射到唯一的cell，而是采用论文**[10]**中的方法，用“软合并”使每个像素通过双线性插值对周围4个cell的特征向量产生贡献。

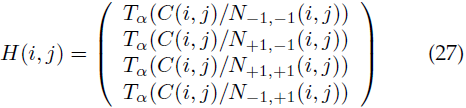
### ****6.1.3 归一化和截断(限幅)****

梯度对偏置改变(changesin bias)具有不变性，这种不变性可以通过归一化获得。Dalal和Triggs[[10]](http://blog.csdn.net/masibuaa/article/details/14056807" \t "http://blog.csdn.net/masibuaa/article/details/_blank)对特征向量C(i,j)用了4种不同的归一化因子。我们可以将这些因子写成 的形式：



每个因子都是对包含(i,j)在内的4个cell组成的块(block)的梯度能量的度量。

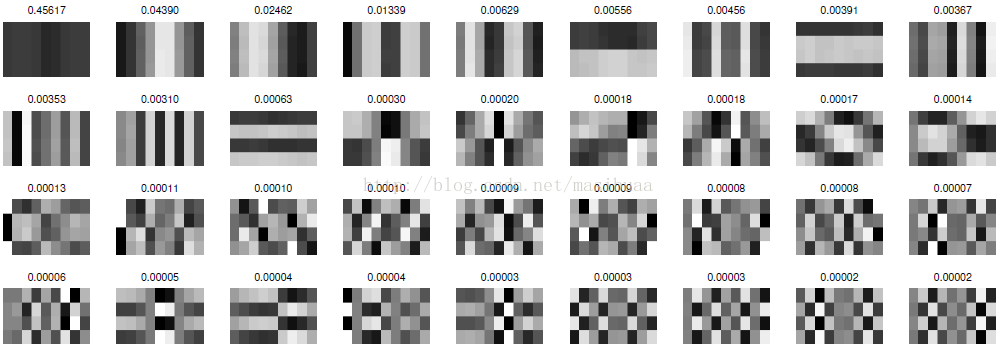
表示向量v被α**截断**(**限幅**)后形成的向量(的第i个元素是v中第i个元素和α两者中的最小值)。对基于cell的特征映射C进行归一化并截断(限幅)，然后串接起来就得到了HOG特征映射，如下：



通常使用的HOG特征中，p=9，用对比度不敏感定义进行梯度方向的离散化(B2公式，0-180度)，cell尺寸k=8，限幅阈值α=0.2。这会产生一个36维的特征向量，我们接下来的分析都使用以上参数设置。

## ****6.2 PCA和解析降维****

我们从大量各种分辨率的图片中收集了很多36维的HOG特征，并在这些特征向量上进行PCA分析。**图6**展示了分析出的主成分，从中我们发现很多有趣的现象。

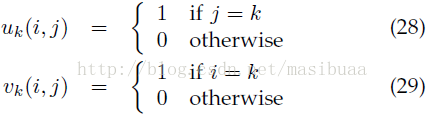


**图6**，HOG特征的PCA分析。每个特征向量(**eigenvector**)都显示为4\*9的矩阵，所以每一行对应一个归一化因子，每一列对应一个方向bin。特征值在特征向量上方。由前11个主特征向量定义的线性子空间基本上包含了HOG特征的所有信息。注意到，所有主特征向量沿其矩阵表达的行或列是定值。

**由前11个主特征向量(eigenvector)定义的线性子空间基本上间包含了HOG特征的所有信息**。 事实上，我们用原始36维特征和向主特征向量投影得到的11维特征在PASCAL 2007数据集的所有目标类别上都获得了同样的结果。使用低维特征可以产生参数较少的模型，并加速学习和检测算法。然而由于在计算特征金字塔时需要相对耗 时的投影步骤，使获得的加速优势有所减少。

36维HOG特征向量来自4个不同的归一化9维方向直方图，所以36维HOG特征向量可以自然地看做一个4\*9的矩阵。**图6**中的主特征向量有一个非常特殊的结构：他们沿其矩阵表达的行或列(近似)是定值。所以主特征向量所依赖的线性子空间可由沿其矩阵表达的某一行或列为定值的稀疏向量定义。

设V={u1,..., u9} ∪{v1, ..., v4}，其中ui和vi都是36维向量，其4\*9的矩阵表达形式满足下列条件：



例如，



定义一个**13维特征**，其中的元素是36维HOG特征与每个uk和vk的点积。HOG特征向每个uk的投影通过计算对应方向的4个归一化值的和(即矩阵表达的某列的和)来获得，HOG特征向每个vk的投影通过计算对应归一化方法的9个方向值的和(即矩阵表达的某行的和)来获得。(**注**：13维特征并不是36维特征向V的线性投影，因为uk和vk不是正交的。事实上，由V定义的线性子空间的维数是12)

使用11维PCA特征和使用36维HOG特征或由V定义的13维特征可以获得同样的性能表现。然而，由于uk和vk是**稀疏向量**，计算由V定义的13维特征比计算向PCA主特征向量的投影要简单的多。此外，13维特征有一个简单的解释：9个方向特征和反应cell周围区域梯度能量的4个特征。

我们也可以定义对比度敏感的低维特征。我们发现有些目标类别适合使用对比度敏感特征，有些目标类别又适合用对比度不敏感特征。所以在实际中我们既使用对比度敏感特征又使用对比度不敏感特征。

设C是聚合有9个对比度不敏感方向的像素级特征映射而获得的基于cell的特征映射，D是聚合有18个对比度敏感方向的像素级特征而获得的基于cell的特征映射。用**公式(26)**来 定义C和D的4种归一化因子。通过对C(i,j)和D(i,j)进行归一化和截断(限幅)，可以获得一个4\*(9+18)=108维的特征向量 F(i,j)。实际中我们使用此108维向量的一个解析投影，此投影由下面几个统计量定义：27个在不同归一化因子上的累加和(即列的和)，F中的每个方 向通道对应一个；以及4个在不同方向(9维对比度不敏感方向)上的累加和(即行的和)，每个归一化因子对应一个。cell尺寸k=8，截断(限幅)阈值 α=0.2。**最终的特征映射是31维向量**G(i,j)，其中27维对应不同的方向通道(9个对比度不敏感方向和18个对比度敏感方向)，其中4维捕获(i,j)周围4个cell组成的block的梯度能量。

最后，我们注意到**图6**中的主特征向量(eigenvector)可以粗略地解释为二维可分离的傅里叶基。每个特征向量可以粗略地看做是一个变量的正弦或余弦函数。如此一来，可以使用有限个傅里叶基函数代替有限个离散方向来定义特征。

**图6**中**傅里叶基**(Fourier basis)的出现是一个有趣的实验结果。当d\*d的协方差矩阵Σ是循环矩阵时，其特征向量可以形成一个傅里叶基，例如。**循环协方差矩阵**(Circulant Covariance Matrix)来自对坐标旋转具有不变性的向量的概率分布。**图6**中二维傅里叶基的出现证明图像上**HOG特征向量的分布(大致)具有二维旋转不变性**。所以我们可以旋转方向bin，以及分别旋转4个归一化块。



# ****7 后处理****

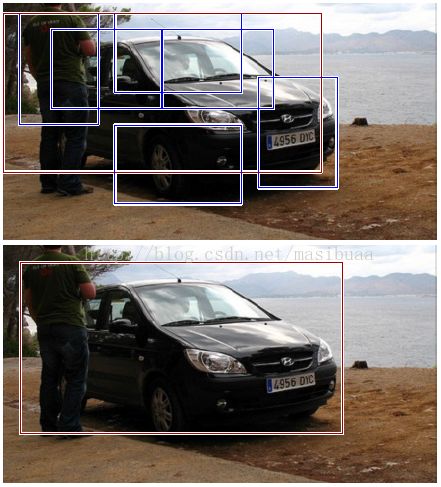
## ****7.1 包围盒预测****

目标检测系统想要得到的输出并不是完全统一的。PASCAL挑战赛的目的是预测目标的**包围盒**(BoundingBox)。我们之前的论文**[17]**是根据根滤波器的位置产生包围盒。但我们的模型还能定位每个部件滤波器的位置。此外，部件滤波器的定位精度要大于根滤波器。很明显我们最初的方法忽略了使用多尺度可变形部件模型所获得的潜在有价值信息。

在当前的系统中，我们使用**目标假设的完全配置**，z = (p0,..., pn)，来预测目标的包围盒。这是通过一个将特征向量g(z)映射为包围盒左上角点(x1,y1)和右下角点(x2,y2)的函数来实现的。对于一个含n个部件的模型，g(z)是一个2n+3维的向量，包含以像素为单位的根滤波器宽度(指出尺度信息)和每个滤波器在图像左上角点的位置坐标。

PASCAL训练集中的每个目标都用包围盒进行了标注。训练完模型后，用检测器在每个实例上的输出根据g(z)训练4个分别预测x1,y1,x2,y2的线性函数，这是通过**线性最小二乘回归**来训练的，对于混合模型中的每个组件是独立的。

**图7**是汽车检测中包围盒预测的一个例子。这种简单方法在PASCAL数据集的某些类别上可以产生较小但值得注意的性能提升(见**第8节**)。



**图7**，汽车检测中根据目标假设进行包围盒预测

## ****7.2 非极大值抑制****

使用**3.2节**中的匹配过程经常会得到每个目标实例的多个重叠检测。我们用一个贪心的**非极大值抑制**(**Non-maximum Suppression**)程序来消除重复检测。

应用完上节介绍的包围盒预测算法后，会得到图像中的某类目标的一个检测结果集合D。D中每个检测结果对应一个包围盒和一个得分。按得分对D中的检测结果排序，贪心地选择具有最高得分的检测结果并跳过被之前选择的检测结果的包围盒覆盖超过50%的结果。

## ****7.3 上下文信息****

我们实现了一个用**上下文信息**(**Contextual Information**)对检测结果进行**重新评分**(**二次评分**)的简单程序。

设(D1,..., Dk)是用k个不同目标类别的模型在图像I中获得的检测结果。每个检测结果(B, s)∈Di由一个包围盒B=(x1, y1, x2,y2)和一个得分s定义。我们用一个k维向量 c(I) = (σ(s1), ..., σ(sk) )来定义图像I的上下文，其中si是Di中最高得分检测结果的分数，σ(x) = 1/(1 + exp(-2x))是一个对分数进行重归一化的逻辑函数。

为了对图像I中的检测结果(B, s)进行重新评分，我们用原始检测得分、包围盒的左上角和右下角坐标、以及图像上下文构建了一个25维(c(I)是20维)特征向量：



坐标x1, y1,x2, y2∈[0,1]用图像的宽和高进行归一化。我们用一个特别分类器对这个新的特征向量进行评分，这样就获得了检测结果的新得分。此分类器结合g定义的上下文信息将正确检测从误报结果中区分出来。

为了获得**重评分分类器**(rescoring classifier)的训练数据，在标注好包围盒的图片(例如PASCAL数据集)上运行我们的分类器。每个返回的检测结果都会产生一个标注为正确检测 (true positive)或误报(false positive)的样本g，标注结果取决于检测结果是否与原图的标注包围盒有显著重叠。

此重评分程序在PASCAL数据集中某些类别的平均检测精度上可以产生性能提升(见**第8节**)。我们的实验中用同样的数据集来训练检测模型和训练重评分分类器，在重评分分类器的训练中使用二次核函数SVM。

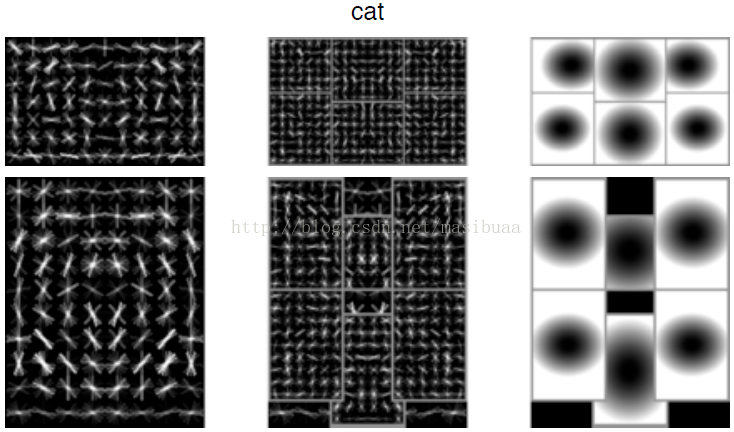
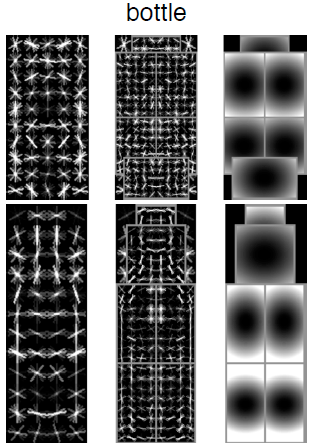
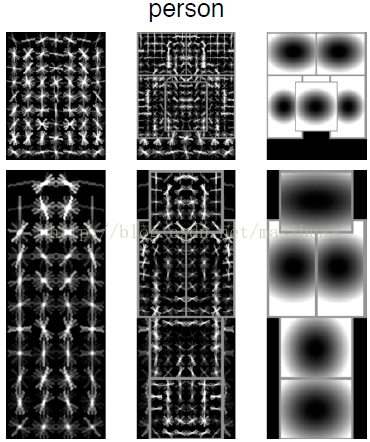
# ****8 实验结果****

我们用PASCALVOC 2006、2007和2008 comp3挑战赛的数据集和协议对系统进行评价。论文**[11]-[13]**中有详细介绍，需要强调的是，这些测评集是公认的目标检测中难度很大的测试集。

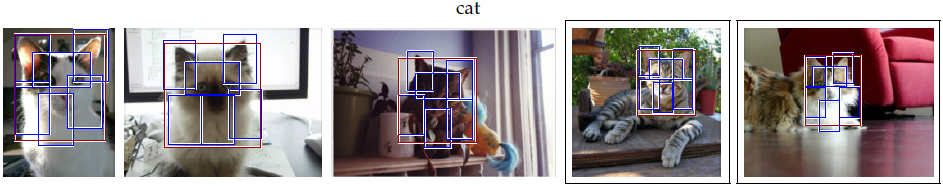
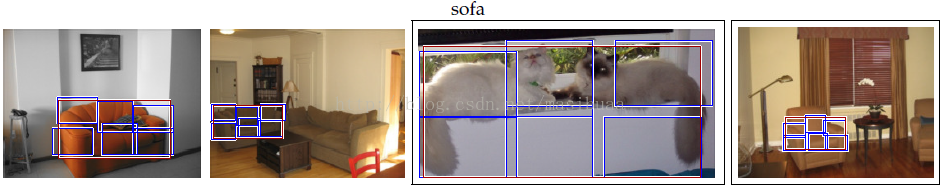
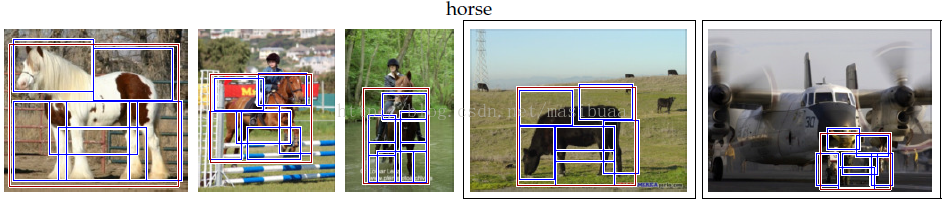
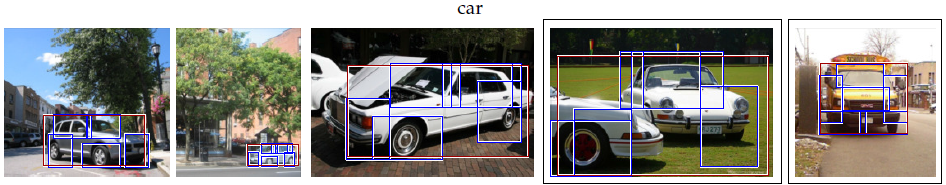
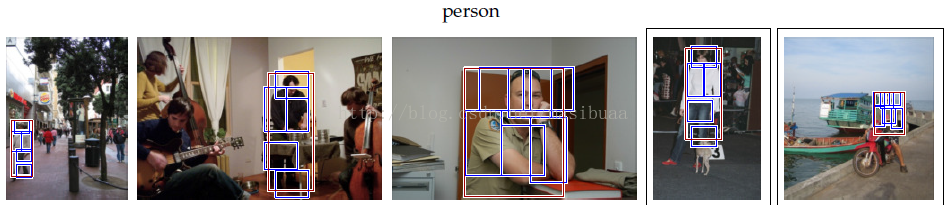
每个数据集都包含数千张真实世界场景的图片。数据集给出了几个目标类别的包围盒真值(**ground-truth**)。测试时，目标是预测图像中给定类别的所有目标的包围盒(如果有的话)。在实际中，系统会输出一系列带评分的包围盒，我们可以在不同点对这些分数进行阈值化来获得一个测试集中所有图片的**精度-召回率曲线**(**precision-recall curve**)。**精度(precision)**表示检测正确的目标占检测的总目标的比例。**召回率(recallrate)**表示检测到的目标占ground-truth中标注的所有目标的比例。

如果预测的包围盒与ground-truth包围盒重叠超过50%则认为此预测包围盒是正确的检测，否则预测包围盒被认为是**误报**(**虚警**，**False Positive**)检测。**重复检测**会被惩罚，如果系统预测的多个包围盒都与一个ground-truth包围盒重叠，则只有其中一个检测被认为是正确的，其他所有都被看做误报。通过系统在测试集上的精度-召回率曲线的**平均精度**(**AP**,Average Precision)对系统进行打分。

对于数据集中的每个目标类别，我们都训练了一个含两个组件的混合模型。**图9**是在2007数据集上学习得到的几个模型。**图10**是使用这些模型的检测结果，图中显示了高得分的正确检测和高得分的误报检测。



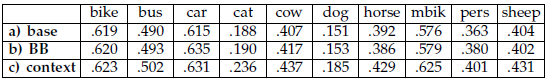
**图9**，在PASCAL 2007数据集上学习得到的一些模型



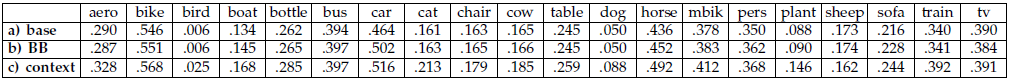
**图10**，在PASCAL 2007数据集上的高得分检测示例，选自每个类别中的前20个高得分检测。每行最后两张加框的图片是每类中的误报检测(false positive)，注意其中很多误报都是由于包围盒标准过于严格(例如人体和猫类别中的误报)。

在一些类别上发生错误检测是由于类别间容易发生混淆，比如马和牛或小汽车和巴士。还有些错误检测是由于包围盒标准过于严格。人体类别最后两个误报就是由于 预测的包围盒与ground-truth包围盒重叠不够充分。在猫的检测中，经常只检测出猫脸，所以预测的包围盒太小无法将猫身体的剩下部分包含进来。事 实上，猫类别中得分最高的20个误报的包围盒都是只有猫脸。这是一个较极端的例子但也解释了为什么我们在猫类上的AP得分较低。猫类别训练数据的正样本中 有很多都是只有猫脸，所以我们的猫模型的其中一个组件模型就是猫脸模型，见**图9**。

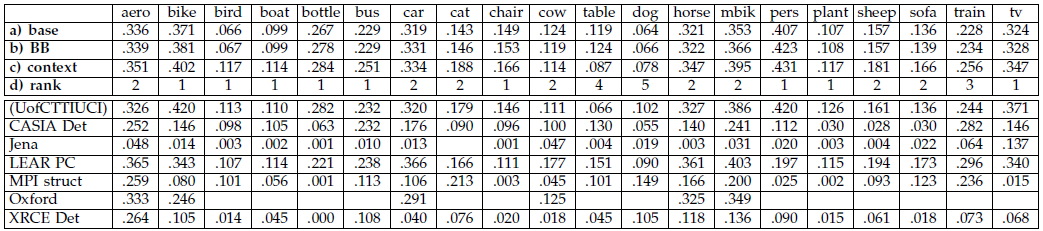
**表1**和**表2**总结了我们的系统在PASCAL 2006和2007挑战赛数据集上的结果。**表3**总 结了在PASCAL 2008数据集上的结果，以及进入2008官方挑战赛的其他系统的结果。空方格表示该方法在对应的目标类别上没有进行测试。其中”UofCTTIUCI” 是我们系统的一个初始版本。我们的系统在20个类别中的9个上得到最高AP，在其中8个上获得第二名。此外，在某些类别上例如人体中我们系统的得分比第二 名要高出很多。



**表1**，PASCAL VOC 2006 结果。(a) 基础系统的平均精度(AP)得分，(b) 加上**包围盒预测**后的结果，(c) 加上**包围盒预测**和**上下文重新评分**后的结果。



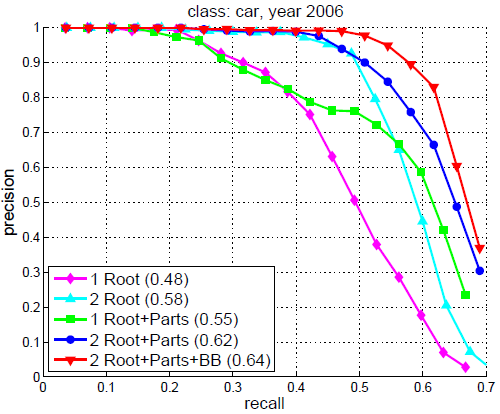
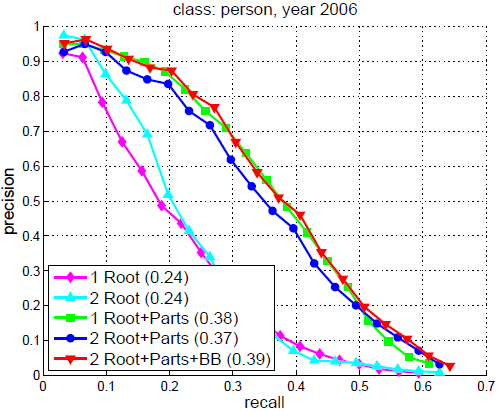
**表2**，PASCAL VOC 2007 结果。(a) 基础系统的平均精度(AP)得分，(b) 加上**包围盒预测**后的结果，(c) 加上**包围盒预测**和**上下文重新评分**后的结果。



**表3**，PASCAL VOC 2008 结果。第一个表：(a) 基础系统的平均精度(AP)得分，(b) 加上**包围盒预测**后的结果，(c) 加上**包围盒预测**和**上下文重新评分**后的结果，(d) 最终得分在2008挑战赛中的排名。第二个表：参加挑战赛的其他系统的结果(UofCTTIUCI是我们系统的初始版本)

我们所做的所有试验中都是用PASCAL trainval数据集中没有标注为“困难”的目标实例来训练模型(我们使用了标注为“被截断”的实例)。我们的系统非常高效。用桌面电脑在PASCAL 2007 trainval数据集上需要4个小时来训练一个模型，需要3小时在test数据集上评价模型。在test数据集中有4952张图片，所以每张图片的平均 运行时间是大约2秒。所有试验都是在配置有2.8Ghz 8核Intel Xeon CPU和Mac OS X 10.5 系统的Mac Pro电脑上进行的。系统在计算滤波器响应值时使用多核架构并行计算，其余的计算都是在单线程上完成的。

我们在耗费更长时间建立的PASCAL2006数据集上测试了系统的不同方面。**图8**总结了不同配 置的模型在人体和汽车检测上的结果。其中包括含1个或2个组件的带有部件或没有部件的模型，以及带有部件和包围盒预测的2组件模型的结果。我们发现部件 (以及包围盒预测)的使用可以显著地提升检测精度。带有多个组件模型的混合模型在汽车的检测中很有必要，但在人体的检测中并没有那么重要。



**图8**，在PASCAL 2006数据集上训练的人体和汽车模型的**精度-查全率曲线**。图中展示了含1个或2个组件的带有部件或没有部件的模型，以及带有部件和包围盒预测的2组件模型的结果。圆括号内是每个模型的**平均精度**(AP)得分。

我们还在INRIA人体数据集[[10]](http://blog.csdn.net/masibuaa/article/details/14056807" \t "http://blog.csdn.net/masibuaa/article/details/_blank)上训练和测试了一个单组件模型。我们用PASCAL评价方法(用PASCAL开发包)在完全测试集上对此模型进行了评分，用带有包围盒预测的基础系统获得了高达.869的AP得分。

# ****9 讨论****

本文介绍了一个基于混合多尺度可变形部件模型的目标检测系统。我们的系统依赖于利用隐藏信息进行判别训练的新方法，以及将部件模型和图片进行匹配的高效方法。系统高效而精确，在较难的数据集上达到了目前最高水平。

我们的模型已经能够表示高度可变的目标类别，但我们还想往更丰富的模型发展，还能够探索其他的隐藏变量框架。例如**更深的部件层次**(**带部件的部件**)或**带有更多组件的混合模型**。将来我们还要建立**基于语法的模型**(Grammar Based Models)，它利用可变层次结构来表示目标，允许部件级别的混合模型，以及**部件的重用**(reusability)，包括在一个目标的不同组件中的重用和不同目标模型中的重用。