**Taking a Deeper Look at Pedestrians**

**摘要：**

本文我们研究了在行人检测任务中卷积神经网络的使用。尽管近来取得了不同程度的成功，相较于其他行人检测器，卷积神经网络在历史上表现的并不好。我们故意忽略把问题明确的建模成网络（部件或遮挡建模）并且证明我们可以达到不错的效果没有花哨的修饰.在一系列广泛的实验中，我们分析了小型与大型的卷积神经网络，他们结构上的选择，参数以及不同训练数据的影响，包括与代理任务中的预训练。

我们将在Caltech和KITTI数据集上呈现最好的卷积神经网络检测器。在Caltech我们的卷积神经网络达到了最好的效果不管是Caltech 1x和Caltech 10x训练设置。在训练时使用附加数据，我们最强的卷积神经网络模型甚至比在测试时使用附加数据（光流）的检测器更具竞争性。

**[本文思考]：**本文由Jan Hosang等人发表于2015年的CVPR，主要的贡献是将深度网络单纯的应用于行人检测问题上，并进行了一系列广泛的实验，分析了卷积神经网络中各种结构、参数、训练数据等各个方面的影响。该篇文章虽然并没有什么开创性的贡献，只是改变了前人的做法（将卷积神经网络用到某个特定问题）将卷积神经网络直接应用于整个行人检测问题上，但作者进行了十分广泛的实验，探究了该问题上的种种问题，工作量十分巨大。从另一方面来说，研究就是这样，不一定要做出多么超前，开创性的工作，也许细致的分析问题的方方面面，从本质出发也是一条正确的科研道路。

1. **简介**

近年来计算机视觉领域见证了大量涉及卷积神经网络的成功案例。该结构目前为通用目标分类，通用目标检测，特征匹配，立体匹配、场景识别，姿态估计，动作识别以及许多其他任务提供了最好的结果。行人检测是目标检测的典型案例，其应用包括车辆安全，监控以及机器人学。该问题的许多想法都被研究过了并且已建立的基准数据集是可用的。我们想知道卷积神经网络的成功是否能转移到行人检测任务中。

之前行人检测上神经网络的工作都依赖于特殊目的的设计，例如手工制作的特征，部件和遮挡建模。尽管被提出的这些方法表现的很合理，目前最好的方法都是基于决策树通过Adaboost学习的。在本研究中，我们重新研究该问题，并证明小型和大型vanilla卷积神经网络都可以在具有挑战性的Caltech数据集上达到最好的表现。我们提供了关于训练的细节，网络的参数和不同proposal的方法的大量的实验。

**[问题的提出]：**卷积神经网络已经大量应用于计算机视觉领域，并取得了成功。但似乎并没有相关研究将卷积神经网络直接应用于行人检测问题上。这种成功能否转移到行人检测任务上，是本文所探究的主要内容。

**[本文的工作]：**本文主要研究了卷积神经网络在行人检测问题上使用的各方面细节问题。

1. 卷积神经网络设计空间
2. 卷积神经网络实现选择

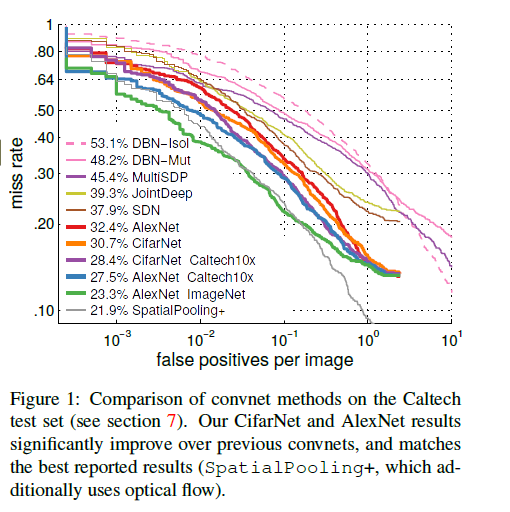
作者主要考虑了两种结构：CifarNet、AlexNet

**[CifarNet]:**

对于CifarNet网络来说，作者主要过滤器的数量和大小、各层的数量和阈值以及输入通道等方面的影响。这些对于网络的构建来说是最基本也最重要的方面。对于这些参数的选择并没有任何定论，因此作者通过一系列的实验来得到哪些设置对于行人检测来说是较为合理的。每个参数的改变，就要进行相应的实验，这也体现了本研究的工作量之大。

**[AlexNet]:**

不同于CifarNet网络，AlexNet是大型网络，实现的选择与方式不同于小型网络。本文中作者主要探究了代理任务、预训练、训练数据集、网络层数等方面的影响。



* 1. **相关工作**

尽管行人检测任务十分流行，只有少量的研究对该任务应用了深度神经网络：我们所知只有6个。

第一篇在行人检测上使用卷积神经网络的论文关注于如何解决有限的训练数据（他们使用INRIA数据集，该数据集提供了614个正例和1218个反例用于训练）。首先，使用卷积稀疏编码的方式对每一层进行初始化，然后整个网络对检测任务进行调参。他们提出了一种使用检测最后一层和倒数第二层特征的结构。该方法被名命名为ConvNet。

另一类型研究扩展了一个可变形部件模型，其中有一堆受限波尔兹曼机器，该机器被训练用来推出部件和遮挡。该模型被扩展到来解释人与人之间的关系（DBN-Mut）并最终共同优化了所有这些方面：JointDeep 共同优化了特征，部件变形，遮挡和人与人关系。

MultiSDP网络使用在候选行人检测附近使用不同尺度计算的上下文信息来喂每一层。最后SDN，目前行人检测中最好的卷积神经网络，使用了附加的“可交换层”（RBM变体）来自动的学习低级特征和高级部件（如“头”，“腿”等）。

注意到目前没有论文依赖于与最初的LeNet（卷积的各层，非线性的，池化的，内积以及一个顶层softmax层）相似的“简单的”卷积网络。我们将在本文重新研究该决策。

当准备本原稿时，附加相关的论文出现在[46,54]。

**目标检测** 除了行人检测外，相关卷积神经网络也被用来对ImageNet以及Pascal VOC类别进行检测。最成功的通用目标检测器是基于R-CNN框架的变体。给定一个输入图像，便会产生一个简化的检测提案集合，这些提案会通过卷积神经网络进行评估。这本质上是一个两级级联滑窗法。最近的一些提案方法见[22]。

**检测提案** 通用目标检测最流行的提案方法是SelectiveSearch。最近的评论[22]也指出EdgeBoxes是一个快速有效的方法。对于行人检测，DBN-Isol和DBN-Mut使用DPM来进行提案。JointDeep, MultiSDP和SDN使用HOG+CSS+线性SVM检测器（与[49]相似）来提案。只有ConvNet应用卷积神经网络以一种滑动的方式。

**决策森林** 大部分提出的行人检测方法不使用卷积神经网络。抛开那些使用光流的方法，目前最好的方法（在Caltech 和 KITTI数据集）是SquaresChnFtrs, InformedHaar, SpatialPooling, LDCF以及Regionlets。所有的这些都是提升的决策森林并可以看作是积分通道特征结构的变体。Regionlets和SpatialPooling使用一个大型特征的集合，包括HOG，LBP和CSS，而SquaresChnFtrs, InformedHaar, 和LDCF建立在HOG+LUV之上。在Caltech基准上，最好的卷积神经网络（SDN）比前面提到的所有方法都要好。

**卷积神经网络的输入** 值得强调的是ConvNet从YUV输入像素来学习预测，然而所有其他方法都是使用附加的手工制作的特征。DBN-Isol和DBN-Mut使用HOG特征作为输入。MultiSDP使用HOG+CSS特征作为输入。JointDeep和SDN使用YUV+Gradients作为输入（HOG+CSS来作为检测提案）。我们的实验表明只使用RGB信息也可以达到最好的结果，但我们也同样证明了更加复杂的输入系统的提高检测质量。我们的数据表明对立的“手工制作特征VS卷积神经网络”是错觉。

**[相关工作的思路]：**

1. 首先指出很少研究把神经网络运用在行人检测上。
2. 接着介绍了一系列目前在行人检测上使用卷积网络的例子，包括：

* ConvNet, 第一个在行人检测上使用的卷积网络，用于解决训练数据有限的问题。
* JointDeep，拓展了DPM来推出部件和遮挡，最后联合优化了特征、部件变形、遮挡以及人与人关系。
* MultiSDP,每层使用候选行人检测附近的上下文特征作为输入。
* SDN，目前行人检测上最好的卷积网络方法，使用附加“可交换层”，自动学习低层特征和高层部件。

并指出所有之前的这些工作都不是依赖于“直接的”卷积网络。

1. 之后介绍了卷积网络在物体检测上的应用。
2. 又介绍了相关检测提案的内容。包括：

* 一般物体检测提案方法。SelectiveSearch、EdgeBoxes。
* 在行人检测上卷积网络所使用的提案方法。DBN-Iso1和DBN-Mut使用DPM，JointDeep和MultiSDP以及SDN使用HOG+CSS+linearSVM。ConvNet使用一种滑动的方式。

1. 介绍了行人检测目前最好的方法都是决策森林方法。包括：

* SquaresChnFtrs
* InformedHaar
* SpatialPoolig
* LDCF
* Regionlets

在Caltech上，SDN比这些方法都好。

1. 最后介绍了其他卷积网络的输入，并表明自己的实验仅用RGB输入就可以达到很好的效果，使用复杂输入则效果更好。
   1. **贡献**

本文中我们通过仔细的研究设计空间（层数，过滤器大小等）以及重要的实现选择（训练数据预处理，检测提案的影响等）来使用卷积神经网络重访行人检测问题。我们的研究表明小型（10^5个参数）和大型（6·10^7个参数）网络从零开始训练都可以达到最好的表现（即使与之前方法使用相同的数据）。我们也表明了使用扩展和外部数据的好处，这使在Caltech上出现了最强的单帧检测器。我们报告了在富有挑战的Caltech数据集上卷积神经网络最好的表现（提高了10个百分点还要多），以及在KITTI数据集上第一个卷积神经网络结果。

1. **训练数据**

总所周知的是训练数据的容量对卷积神经网络达到较好性能来说十分重要。以下是我们本文考虑的数据集。

**Caltech** Caltech数据集以及其相关联的基准，是最流行的行人检测数据集之一。它包括从一辆行驶在良好天气环境下的美国街道的汽车所捕捉到的视频画面。标准的训练集在“Reasonable”设置下包含4250帧，大约2·10^3个标注行人，在测试集中包含4024个帧大约1·10^3个行人。

**Caltech 验证集** 在我们的实验中我们还使用了Caltech训练数据来做验证。在这些实验中，我们使用了一个建议的验证集分割：前五个训练视频用来验证训练第六个训练视频用来验证测试。

**Caltech10x** 因为Caltech数据集视频全部被标注了，所以可以通过重采样视频来增加训练数据量。受到[30]的启发，我们通过每三帧取一帧的采样增加了十倍的训练数据（而不是标准设置中每三十帧取一帧）。这样从42787帧中产生了大约2·10^4个标注行人用来训练。

**KITTI** KITTI数据集包含了从一辆行驶过德国街道（也是在良好天气环境下）所捕获到的视频数据。虽然在外表上和Caltech相似，但它表现出不同的统计数据（见[5,补充材料]）。它的训练集在7481帧上包含了4445个行人（4024个比40像素要高），它的测试集包含7518帧。

**ImageNet, Places** 在第五节中我们将考虑使用可以为代理任务利用预训练的大型卷积神经网络。我们考虑了两种该类型任务（以及他们相关联的数据集），ImageNet2012千种物体类别分类以及205种场景类别分类。数据集分别提供了1.2·10^6和2.5·10^6标注的图像用于训练。

1. **从决策森林到神经网络**

在进行实验之前，值得注意的是我们所使用的提案方法，SquaresChnFtrs（见4.1节），可以被转换为一个卷积神经网络。整个系统变为两个神经网络的级联。

SquaresChnFtrs[4,5] 是一个由2048个两层决策树组成的决策森林，应用在十个手工制作的特征通道上（HOG+LUV）。这些通道在矩形区域上被联合池化（sumpooled）并被输入到树上分离的结点中。这种结构可以简单的映射为一个卷积神经网络。

如1.1节中提到的，使用非RGB输入是带有卷积神经网络的行人检测标准做法（更详细的见4.4节），因此我们专注于对池化和决策森林的转换。联合池化阶段会直接映射到内积层。每个决策树映射成两个隐层的一小列，伴随符号函数非线性（硬非线性）。最后所有决策树的输出通过线性加权进行组合。

从SquaresChnFtrs到深度神经网络的映射是准确的：对同样的输入进行评估，它将返回完全相同的输出。生成网络的特殊之处是它不是通过反向传播训练，而是通过Adaboost训练。该网络已经比Caltech上最好的卷积神经网络SDN表现的要好。

可惜的是，减轻非线性和使用反向传播来调参的实验并没有表现出重大的改进。我们猜想通过Adaboost建立的参数是局部最小值，并不能通过随机梯度下降法来逃出（escape）。

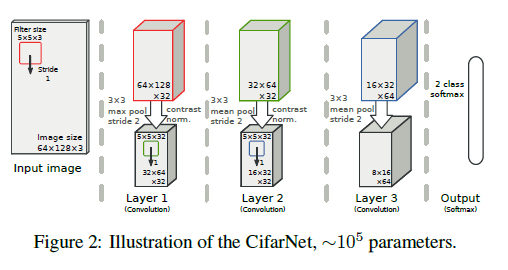
1. **Vanilla卷积神经网络**

根据我们的经验，许多卷积神经网络结构和训练超参数并不能使多样和挑战性的任务有效的学习。因此从已知能够很好工作的并对现有任务逐渐适应的结构和参数来开始研究是最佳的实践。这是接下来几节的策略。

本节中我们首先考虑CifarNet，一个用来解决CIFAR-10分类问题（10个物体类别，（5+1）·10^5个32x32像素的彩色图像）的小型网络。在第五节中我们考虑AlexNet，一个参数比CifarNet多600倍的网络，用来解决ILSVRC2012分类问题（1000个物体类别，（1.2+0.15）·10^6个VGA分辨率彩色图像）。这些网络都在[27]中有介绍，并在开源项目Caffe中进行了重实现。

尽管行人检测与CIFAR-10相比大不相同，我们决定从在CIFAR-10上表现权威的CifarNet开始我们的研究。图2描绘了该网络的结构，除非另外说明，我们都使用原始RGB输入。

我们首先讨论如何使用CifarNet 网络（4.1节）。这种简单的方法已经在最好的卷积神经网络上得到了提升（4.2节）。4.3节和4.4节探讨了CifarNet的设计空间并进一步推进了检测质量。本节中所有模型都是仅在Caltech上训练的（见第二节）。

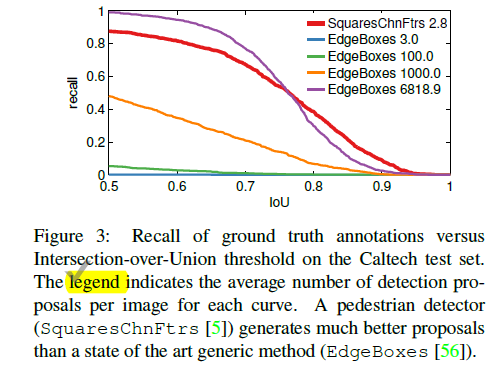


* 1. **如何使用CifarNet?**

给定一个初始化的网络规格，依然还有一些设计选择来影响最后的检测质量。在下面几段中我们将讨论它们其中的一部分。

**检测提案** 除非另外说明，我们使用SquaresChnFtrs[4,5]检测器来生成提案，因为在写作本文时，这是有可用源码的最好的行人检测器（在Caltech数据集上）。图3中我们比较了SquaresChnFtrs和一种先进的类别无关提案方法——EdgeBoxes。使用特定类别的提案可以将提案的数量降低三个数量级。

除了ConvNet（不使用提案），所有其他竞争的卷积神经网络都是用一个行人检测器来进行提案（见4.2节）。

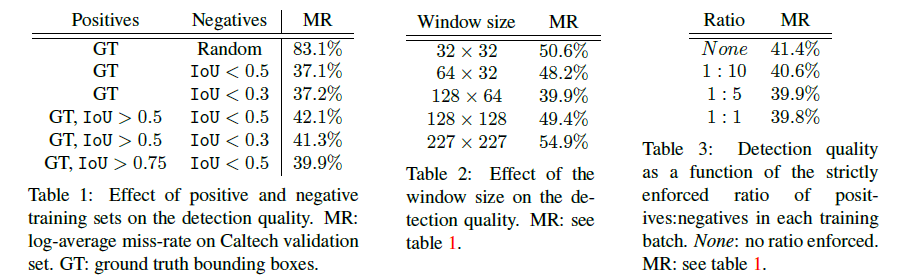


**正样本和负样本的阈值** 给定训练提案和ground truth 标注，我们现在考虑为每个提案分配哪个训练标注。某个提案被认为是正样本如果它超过了最少一个GT标注的一个特定IoU阈值。某提案被认为是负样本如果它不超过任何GT标注的第二IoU阈值，否则就被忽略。我们发现使用GT标注作为正样本是十分有益的（例如不施加显著的抖动）。

**模型窗口大小** 行人检测器典型的选择是一个128x64像素的模型窗口，在该窗口中行人占据了96x48像素的区域。这是否是卷积神经网络的理想输入还是不清楚的。尽管CifarNet被设计用来操作32x32像素的图像，表2表明一个大小为128x64像素的模型确实工作的最好。我们对其他变体进行了实验（拉伸vs剪裁，更大的上下文边界），并没有明显的改善。

**批训练** 在检测设置中，训练样本通常与背景类非常不平衡。尽管在我们验证设置中这种不平衡被限制了，我们发现对每批随机梯度下降优化实施一个严格的正负样本比例对整个实验都是有益的。只要保持某个比例，最终的表现对该参数就是不敏感的。我们使用1：5的比例。

在补充材料中我们详细讨论了所有其他训练参数。

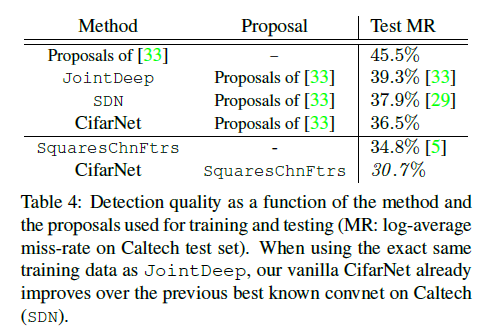


* 1. **使用CifarNet可以达到什么程度？**

给定上一节中验证集上的参数选择，CifarNet如何与在Caltech测试集上之前的卷积网络的结果进行比较？表4和图1表示了我们naïve 网络立刻提高了最好的卷积网络（30.7%MR vs SDN37.9%MR）。

为了对我们强大的SquaresChnFtrs提案在CifarNet表现上的贡献进行解耦，我们也使用了JointDeep提供的提案来训练CifarNet。当在训练和测试时使用相同的检测提案，vanilla CifarNet已经提升了定制 的JointDeep和SDN。

我们CifarNet结果与在Caltech1x训练的最好的行人检测器出乎预料的接近（30.7%Mr vs SpatialPooling29.2%MR）。



* 1. **探索不同的结构**

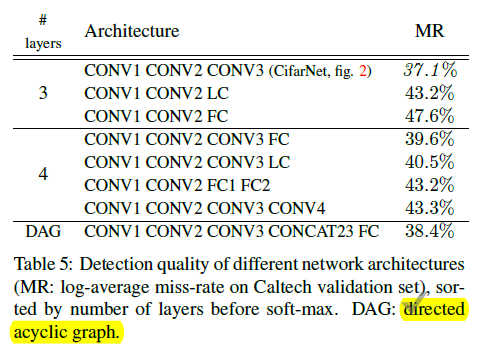
受到我们初始结果的鼓舞，我们继续探索CifarNet结构的不同参数。

* + 1. **卷积过滤器的数量和大小**

使用Caltech 验证集，我们在每层执行卷积过滤器大小（3x3,5x5,或7x7）和数量（16，32，或64个过滤器）。我们把整个表格放在补充材料中，我们观察到使用大型过滤器尺寸会影响检测质量，而改变过滤器的数量展现出很少的影响。虽然我们观察到了一些丢失率的起伏，整体上还没有明显的趋势表明某个配置明显的比其他的更好。因此，为了简单，我们在后续的实验中继续使用CifarNet（32-32-64个5x5像素过滤器）.

* + 1. **每层的数量和类型**

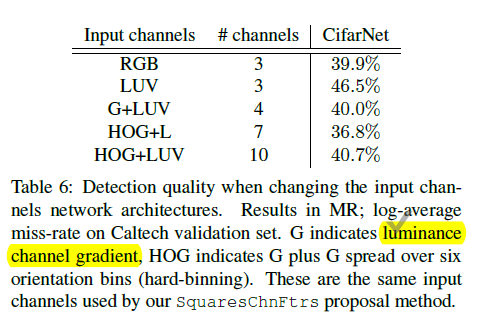
表5中我们评估了在其他CifarNet参数固定时，改变层数量和类型的影响。除了卷积层（CONV）和全连接层（FC），我们也考虑了局部连接层（LC）和跨层连接特征（在ConvNet中使用的）。这些考虑的结构改变对原始的CifarNet都没有提升。



* 1. **输入通道**

如在1.1节中讨论的，之前行人检测器大部分使用梯度和颜色特征来作为输入，而不是原始RGB。在表6中我们评估了CifarNet上不同输入特征的影响。似乎HOG+L通道比RGB要好一点。

为了直接对大型网络进行直接的比较，在下一节中我们将在我们CifarNet实验中使用原始RGB输入。我们将在第六节报告CifarNet测试集结果。



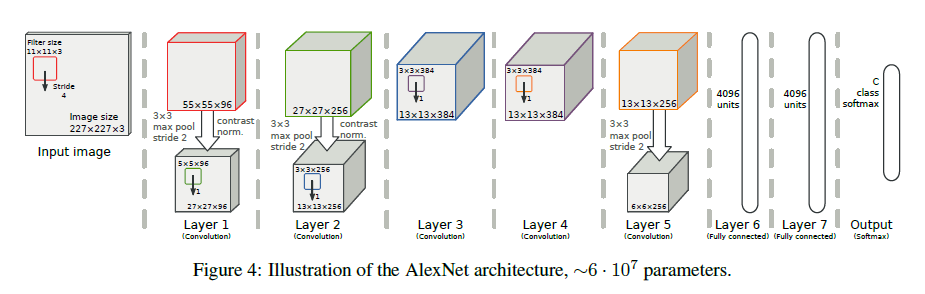
1. **大型卷积神经网络**

卷积神经网络一个吸引人的特征就是它能够扩展训练数据的规模。在本节中我们探索使用更多数据训练的大型网络。

我们基于R-CNN方法开展我们的实验，该方法是目前在Pascal VOC检测任务中表现最好的方法之一。因此我们对其在行人检测上的表现充满好奇。

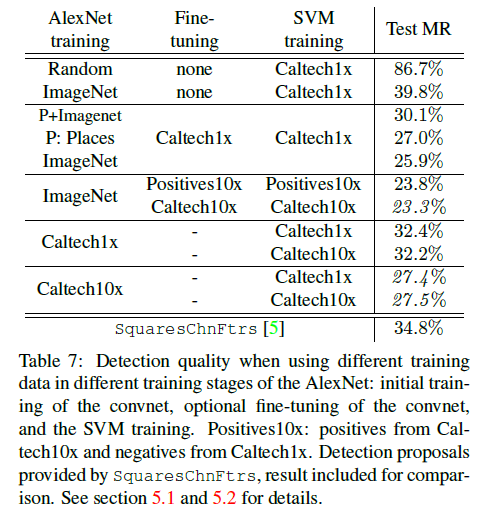
* 1. **改进检测的代理任务**

R-CNN方法（“具有CNN特征的区域”）包装了之前在ImageNet分类任务上训练的大型网络，我们称之为AlexNet（见图4）。在上下文区别明确时，我们使用“AlexNet”来作为“具有AlexNet的R-CNN”的简写。在R-CNN训练时，AlexNet会为检测任务被调参，在第二步中，softmax的输出被一个线性SVM所替代。除非另有说明，我们使用开源的，基于Caffe的R-CNN实现的默认参数。如在前面几节中，我们使用SquaresChnFtrs来做检测提案。为了与文献中其他AlexNet实验一致，我们使用默认RGB以及227x227输入大小（CifarNet上最佳的参数可能不适用于大型AlexNet）。



**预训练** 如果我们只训练顶层SVM，不对底层AlexNet进行调优，我们在Caltech测试集上得到了39.8%MR。这已经出乎意料的接近于该任务上最好的卷积网络（SDN 37.9%MR）。当在Caltech上对所有层进行调优时，测试集上表现急剧提高，达到了25.9%MR。这证明了同源R-CNN检测方法（在ImageNet上训练AlexNet，为感兴趣的任务进行调优）的有效性。

表7中我们通过考虑在场景识别（“Places”,见第二节）和Places以及ImageNet（“Hybird”）上训练过的AlexNets来研究预训练任务的影响。“Places”提供了接近ImageNet的结果，表明确切的预训练任务并不是关键以及ImageNet并没有特殊之处。



**Caltech10x** 出于AlexNet有着大量的参数，我们考虑使用Caltech10x提供供附加的训练数据来对网络进行调优（见第二节）。尽管训练样本之间有强相关性，我们确实看到了进一步提升（见表7）。有趣的是，大部分的提升是由于更多的行人（Positives10x，使用了Caltech10x中的正样本以及Caltech1x的负样本）。我们最好的结果，23.3%MR，是我们的AlexNet设置成为在Caltech上最好的单帧检测器。

* 1. **只使用Caltech训练**

为了和CifarNet比较，并验证预训练是否是必要的，我们只使用Caltech训练数据来训练AlexNet“从头开始”。我们收集的结果在表7中。

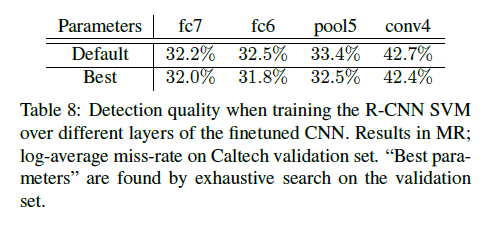
单独的在Caltech上训练AlexNet，产生32.4%MR，相比于提案（SquaresChnFtrs 34.8%MR）和之前最好的Caltech上卷积网络（SDN39.8%）有所提升。使用Caltech10x会更进一步提升性能，达到27.5%MR。

尽管这些数字比从ImageNet预训练中获取的要差（23.3%MR，见表7），我们可以只使用行人数据而不管AlexNet模型的10^7自由参数获得出乎意料的具有竞争力的结果。使用Caltech10x的AlexNet是在Caltech上第二好的单帧行人检测器（最好的是LDCF 24.8%MR，也使用Caltech10x）。

* 1. **附加实验**

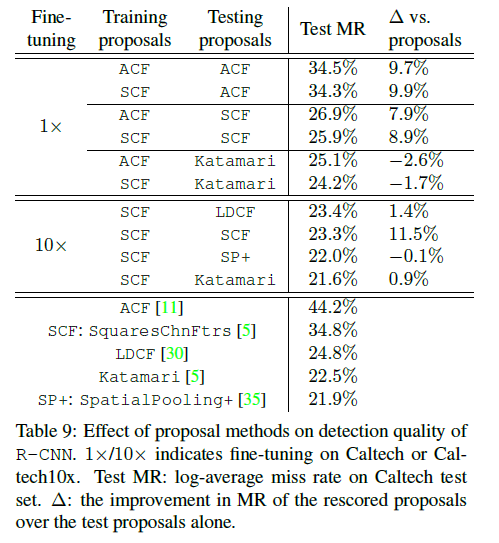
**多少层？** 目前为止所有实验都使用了R-CNN默认参数。之前的工作报告了，根据任务的不同，使用AlexNet低层特征可以提供更好的结果。表8报告了当在上层4~7层训练SVM输出层时Caltech验证集的结果。我们报告了使用默认参数和使用网格搜索优化参数的结果（详细的网格搜索在补充材料中给出）。

我们观察到默认的和优化过的参数只有微不足道的差异（最多1个百分点）。默认参数的结果表现出在高层有更好结果的轻微趋势。这些验证集结果表明，整体上R-CNN默认参数对于行人检测来说是很好的选择。



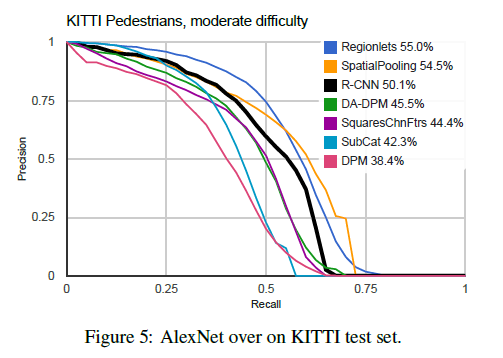
**提案方法的影响** 当对在Caltech1x上调优的AlexNet与提案方法进行性能比较时，我们发现了在丢失率上9个百分点的提升。在表9中我们研究了使用弱提案还是强提案的影响。ACF和SquaresChnFtrs都提供了源代码，允许我们生成训练提案。Katamari和SpatialPooling+是目前Caltech数据集上最好的方法，两者都在测试时都使用了光流等附加信息。ACF，SquaresChnFtrs与Katamari/SpatialPooling之间有10个百分点差距，允许我们覆盖不同的工作点。

表9的结果表明，尽管有10个百分点的差距，使用ACF训练的与用Squares-ChnFtrs训练的AlexNet模型之间并没有明显的差异。看起来只要提案不是随机的（见表1的第一行），获得的结果质量就很稳定。该结果也表明AlexNet饱和附近的质量提升大约22%MR。使用强提案并不会进一步获得提升。这意味着我们训练的AlexNet的辨别能力与Caltech数据集上最好的方法是相当的，但不能赶超他们。



**KITTI 测试集** 图5中我们展示了在KITTI行人检测基准上下文环境中AlexNet的表现。该网络在ImageNet上预训练并使用KITTI训练数据调优。SquaresChnFtrs达到了44.4%AP（平均精度），AlexNet可以提高到50.1%AP。这是KITTI行人检测数据集上第一批公布的卷积网络结果。

虽然SpatialPooling的排名改变了，但应该注意到的是a)两种数据集使用不同的评估度量，b）两个数据集比表面上看起来大不相同，c）总体上AlexNet在KITTI上保持了令人满意的结果；使用具有高召回率的提案可能进一步提高结果。



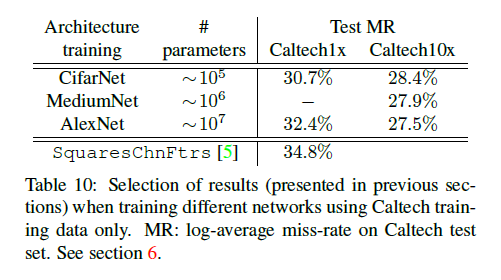
* 1. **误差分析**

之前一节的结果令人鼓舞，但不如在对Pascal VOC上的提升所预期的要好。所以到底是什么限制了性能？提案方法？卷积网络的定位质量？

观察最高得分的虚警描绘了一张提案方法，R-CNN甚至GT的定位误差图。为了量化该影响，我们重新运行了Caltech评估但是去除了接触到标注的所有虚警。该实验为解决定位问题以及进行完美非极大值抑制时提供了一个性能上界。我们意外的发现所有的方法一致的提高了2%MR（见补充材料）。这意味着我们从虚警中收集的直觉是错误的，实际上大部分使MR降低的错误都是背景窗口误认为是行人。从该结果中可以明显的看出­这不是我们在检测提案的R-CNN实验中才会出现的，同样会出现在被训练为滑动窗口检测器的方法上。

1. **小型还是大型卷积神经网络？**

由于我们已经分别分析过CifarNet和AlexNet，在本节中我们将共同的比较它们的性能。表10展现了只在Caltech1x和Caltech10x上训练的模型在Caltech测试集上的表现。具有较少训练数据的CifarNet达到了30.7%MR，比AlexNet好2个百分点。在Caltech10x上，我们发现CifarNet表现提高到了28.4%，而AlexNet提高到了27.1%MR。这种趋势证明了当增加训练数据时，具有较低容量的模型比具有较高容量的模型更容易饱和。我们也可以得出当在Caltech1x上训练时，AlexNet会得到更好的正则化的结论。



**时间** 检测运行时间大约3ms每个提案窗。这对于滑窗来说十分慢，但使用具有高召回率，每张图少于100个窗口的快速提案方法，评分任务则大约300ms每张图。在我们的实验中，SquaresChnFtrs每张图运行2s，所以提案检测任务占据了大部分检测时间。

1. **额外说明（Takeaways）**

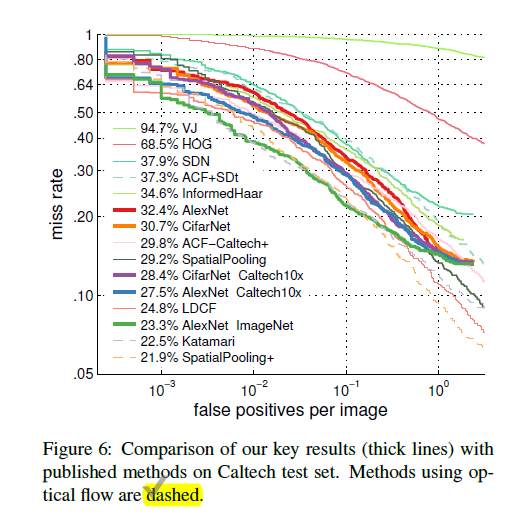
之前的工作表明尽管卷积神经网络涉及结构，但在行人检测上表现不好（见[5]关于行人检测的调查）。本文表明不一定是这种情况。我们使用两个现成的模型进行了一系列广泛的实验，达到了具有竞争力的表现：小型CifarNet和大型AlexNet。

我们提出了两个只在Caltech上训练的网络，都胜过之前所有Caltech上公布的卷积网络。CifarNet比相关工作展现出更好的表现，即使使用与相应方法相同的训练数据（4.2节）。尽管规模较大，AlexNet也提高了所有卷积网络，即使它只在Caltech上训练（5.2节）。

我们推动了已在Caltech1x和Caltech10x上训练的最好的行人检测器。CifarNet是Caltech1x上训练出的最好的单帧行人检测器，而AlexNet是在Caltech10x上训练处的最好的单帧行人检测器。

在图6中，我们将包括所有在Caltech上公布的方法纳入比较，也增加了在测试时使用附加信息的方法。在ImageNet上预训练的AlexNet相比公布的最好方法达到了具有竞争力的结果，但在测试时并没有使用附加信息（5.1节）。

我们公布了KITTI行人检测基准上第一个卷积网络结果。AlexNet单独提高了提案方法，提供了令人鼓舞的结果以进一步推进使用卷积网络在KITTI上的表现。



1. **结论**

我们提出了广泛和系统的实验证据，表明了卷积网络对于行人检测的有效性。相比于之前应用于行人检测的卷积网络，我们的方法避免了用户定制。当与之前方法使用同样的提案与训练数据时，我们的“vanilla”网络比之前的方法表现要好。

我们已经证明在代理任务上预训练，卷积网络可以在本任务上达到最好的表现。有趣的是我们证明了即使没有预训练，也能达到具有竞争力的结果并且该结果对于模型大小（10^5到10^7个参数）并不不敏感。我们的实验也详细的证明了哪些参数对于达到最好的效果是最重要的。我们在Clatech和KITTI数据集上都报告了卷积网络最好的方法。

我们使用卷积网络的经验表明它们在行人检测上表现出良好的希望并且报告过的最佳实践也可转移到该任务。话虽如此，在这个更加成熟的领域，我们并没有观察到在类似Pascal VOC和ImageNet的数据集上巨大的提升。

**[文末总结]：**纵览全文，大部分内容是对卷积神经网络中各种实现细节的讨论，探究。虽然本文发表于2015年，但那时深度学习已经普及起来，许多视觉任务都应用了深度学习并获得了不错的效果。“将卷积神经网络应用于行人检测任务上”这一想法一定被许多研究者尝试过，但所做的相关工作都不如这篇论文中的详细。论实验的广度与深度，这篇论文都做到了尽可能详细，这也是它能以这类型的工作在CVPR上脱颖而出的原因。这同时也是对我们的一种提醒，某个领域的发展不仅仅需要先进算法的提出，更需要基本问题的探索与发现。

[其他]：因为这周在准备期末考试，所以看论文的时间比较少，对论文中的实验理解不算太深刻，之后会再重新回顾文章中所提到的一些参数选择，或者动手做做实验。