

指导教师： 杨涛

提交时间： 2016/3/17

CVPR2015 Paper Translation

No: 01

姓名： 高俊龙

学号： 2013302571

班号： 10011305

深度轮廓：正数共享损失轮廓侦察的深度卷积特性

申伟¹, 王星光², 王艳³, 白项², 张志江¹

¹ 上海大学特种光纤与光接入网重点实验室

² 华中科技大学电子信息与通信学院

³ 南洋理工大学对象搜索实验

摘要

轮廓检测是用作各种计算机视觉任务的基础如图像分割和目标识别。而我们的主流工作致力于解决这个问题关注设计工程梯度特性, 在这项工作中, 我们表明轮廓检测精度可以改善但可以通过卷积神经网络(细胞神经网络)得到的。我们通过自定义分区轮廓(正)数据转换成子类和装配不同的模型参数子类来制定培训战略, 当然我们不是使用网络作为一个黑盒子特征提取。一个新的命名为共享正损的损失函数, 他的每个子类股共享整个大类的损失, 我们建议学习他的参数。相对于 SOFTMAX 损失函数, 我们所提出的函数引入了额外的正则化来强调的积极和消极的类损失, 这有利于探索更具有特征的损失。我们的实验结果证明, 学习深度特性可以实现伯克利分割数据集和水准(BSDS500)的最佳性能和获得竞争交叉数据泛化结果 NYUD 数据集。

1. 引言

在本文中, 我们研究了在计算机视觉自然图像轮廓检测中的经典和根

本的问题。许多任务是以正确地检测物体轮廓为基础的任务, 例如图像分割[2], 场景理解[3]和物体检测[17, 30, 48]。

轮廓检测是相当具有挑战性的, 比边缘检测更加困难。根据马丁的定义[34], 后者的目的是检测在亮度, 颜色和质地的特性的变化; 与此相反, 前者的目标是寻找从一个物体或表面到另一个像素所有权的改变。因此, 轮廓涉及对象的概念, 其投射一个障碍给我们: 如何辨别杂乱纹理引起的变化和那些对应于对象的边界的变化? 许多研究人员倾注了自己的努力解决这一问题, 并取得了的进展[34, 32, 38, 2, 39, 30, 11, 23, 12]。然而, 算法和人类注释给定轮廓检测观察到了明显的性能差距。用于轮廓检测的传统框架设计了多种用于每个图像像素梯度特征, 随后通过学习二元分类, 以确定一个图像的像素是否为轮廓。在过去的十年中, 虽然手设计的功能被广泛应用同时支持标准的轮廓检测指标排名靠前的算法[35, 2], 但我们不能忽视一个事实, 他们没有足够的辨别能力去区分语义的目标边界和低级影像中的突变。通过深度神经网络的广泛运用, 一些研究人员尝试学习深度特性来解决该轮廓与非轮廓分类问题[23, 19, 33]。对于许多计算机视觉任务的印象深刻的表现已经证明, 深层神经网络输出的深层功能是强大的

[1,37], 而且将取代传统的手工设计的功能, 例如像 SIFT[31]和 HOG[9]。因此, 引入深度学习技术去解决轮廓检测问题是合理可行的。然而, 学会如何辨别和表示对深为特征轮廓检测来说是不简单的。深度学习用于轮廓检测的当前使用情况是, 采取深网络作为黑箱模型了解轮廓[23]或本地等高线图[19]的概率为每个像素来得到数据。通过这样的方式, 深网络在对一个极其复杂的建筑研究时其灵活性可能是不足的。为什么深网络可以在如此多的识别任务中脱颖而出却被轮廓甄别拒绝呢? 原因是轮廓对于局部来说是巨大的, 因为他包含很多类型诸如直线, 平行线, T 连接和 Y 结的模式。尽管深网络功能十分强大但是把这么大的变化的数据视为一个类是不可取的。

我们的目标是学习著名的深卷积神经网络(细胞神经网络)[18, 42, 26], 了解了轮廓检测判别功能。但是, 我们不只是用 CNN 作为一个黑盒子来进行特征提取, 而是考虑轮廓的内在属性得到它。我们强调以下两点:(1) 分区块轮廓的紧凑集群根据其固有的结构是必要的有效的 CNN 模型。这样的集群过程中导致轮廓补丁中级形状表示。所形成的簇称为形状的和每个都通过一个形状标签分配。不同形状的拟合轮廓数据通过不同的模型参数是被分而治之策略拟合的[4], 从而缓解由于数据的多样性培训困难。(2) 如何界定 CNN 的损失函数是学习轮廓检测判别显著功能。上述聚类处理转换二元分类问题(即预测图像补丁是否属于轮廓或非轮廓)到多类问题(即预测图像补丁是否属于各形状类或负

类), 这似乎可以通过最小化损失 SOFTMAX 在标准 CNN 的使用损失函数得到很好的解决。然而, SOFTMAX 功能不利条件是相同的, 这是不适合学习的轮廓补丁和补丁背景之间的不同特性的。因为, 对于轮廓检测, 形状类之间的错误分类是可忽略的, 而一个轮廓补丁被分类为背景之一是相当大的误差, 或反之亦然。因此, 对于积极的和消极的损失应在训练被强调。基于这种观察, 我们定义一个新的结合轮廓与非轮廓与 SOFTMAX 损失的额外损耗目标函数。作为该功能的正损失其他所有类可以共享, 我们将其命名为积极损失。额外的损失带来更好的正规化的形式从而带来更好的轮廓特征的了解。随着深学特征, 轮廓检测可以通过将其送入任何分类后跟一个标准的非最大抑制方案进行[6]。

2. 相关工作

轮廓检测: 轮廓检测通常被认为是一个监督学习问题。在先驱工作 Konishi et al. [24], 轮廓检测是制定歧视任务指定的似然比检验的滤波器。Martin 等人[34]精心设计的功能学习分类的特点结合起来, 在亮度, 色彩, 以及与自然边界相关的纹理特征的变化分类。利用美元等[10]补丁丰富的功能提高概率树[46]检测轮廓。仁和博[39]发现稀疏编码梯度功能, 它是有效的轮廓检测。最近, 一个中等特征命名的草图令牌和随机基于结构分类, 分别在[30] [11]提出。除了监督学

习,[2],Arbelaez 等人结合当地多个信号变成一个全球化框架基于谱聚类的轮廓检测。

深度学习： 深度学习方法已经在计算机视觉的各种应用，如图像分类[25]，目标检测[20]，图像标记[15]，以及超高分辨率[14]上应用并获得巨大成功。深卷积神经网络（细胞神经网络）与 GPU 实现和纠正学习和监督分类和辍学[44]是流行的功能。受限玻尔兹曼机（RBM），自编码[47]及其变体是流行的无监督的深度学习。

轮廓检测的深度学习： 正如我们所知道的这里有两篇用于轮廓检测的论文。第一篇是 Kivinenet 等人[23]写的使用 RBM 技术和复杂分类器结合得到每层的值。第二篇是 Ganin 和 Lempitsky 发表的，特性的图像补丁是学会了使用传统的 CNN 然后特性映射到一个注释边缘地图使用 kd tree。这两个方法的不同，我们学会利用形状标签和 CNN 的一个新颖的损失来表示功能。

3. 数据准备

在本节中，我们将介绍如何准备训练验证数据，设置训练 CNN 模型。就拿 BSDS500 数据集作为一个例子，该 BSDS500 数据集包含 200 训练，100 验证和 200 个测试图像。每个图像都有手标记的地面真实轮廓。继[30]，我们得到来自代表手标记的地面真实轮廓二值图像中提取的集群补丁的形状类。只有补丁的中心是通过轮廓标记用于集群的。由于 CNN 的具有较强的

学习处理更多的信息的能力，在我们的实验中我们使用定制的为 45×45 尺寸的像素的(而不是 35×35 [30])补丁。NS 有较强的学习处理信息的能力。这个聚类过程导致图形 K 形类，它们的图案从直线到更复杂的结构，如在图 1 看到的。

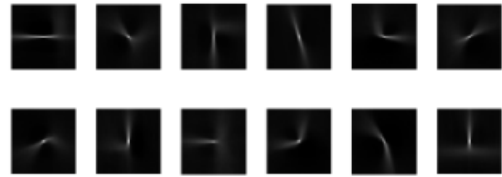


图 1: 某些选定的形状类的可视化

学会了 K 形类，我们可以给数据集中的 x 指定一个标签 y 。如果 x 是一个轮廓补丁（我们设定最大斜率等于允许的贴片中心之间缩略图距离轮廓 3 像素），它的类标签由形状聚类结果供给，即它的类标记 $y = k (k \in \{1, \dots, K\})$ 如果从手标记地面实况值图像中提取相应的补丁则属于第 k 个形状类。否则，如果 x 是一个背景贴片，它的标签由 $y=0$ 分配。为了标记简单，轮廓补丁和补丁程序的背景也被在本文其余部分称为的积极补丁和消极补丁。

我们从训练集中的 BSDS500 数据集，其中采样，积极消极的补丁的数量相等为 2,000,000 图像块，以形成用于学习我们 CNN 模型的训练数据。在测试 CNN 模型时，我们也使用了跟 BSDS500 数据集中相同的 2,000,000 做对比。

4. 学习 CNN 深度特征

在这部分，我们描述了如何通过我们的 CNN 模型学习轮廓检测。首先，我们介绍我们的 CNN 模型的体系结构。然后，我们讨论如何在我们的任务定义适当的损耗函数。

4.1 CNN 架构

我们培训 CNN 在一个多级分类任务，即图像补丁分类到形状类或消极类。图 2 描绘了 CNN 模型，含六层与要学习的参数的整体架构；前四个是卷积后两个是全接。只有卷积和完全连接层含有可学习的参数，而其他参数都是免费的。我们的 CNN 的输入尺寸是 45×45 ，其由第一卷积层（COV1）与尺寸 $5 \times 5 \times 3$ 的 32 粒的 2 个像素的填充过滤的 3 通道（RGB）图像补丁。所得 $45 \times 45 \times 32$ 的特征图，然后按顺序提供给本地响应正常化层（LRN1）和其上的 3×3 的空间的街区中的 2 个像素的步幅最大池层（MAXP1）。然后 MAXP1 的输出被传递到所述第二卷积层（COV2）。除了滤波器内核的数量，参数配置的四个卷积层是相同的。每个卷积层由整流线性单位（ReLU）组成，随后是一个 LRN 和 MAXP，除了最后一个，其他都施加的 MAP。卷积层是 CNN 的核心，提供各种特征图谱[13]。而 max-pooling 层是使得激活功能强劲的轻微变化的轮廓位置。第四最大值-池层（MAXP4）的输出被馈送到包括一个 ReLU 的第一完全连接层（FC1）。全层的每个输出单元连接到输入节点，它能够捕获特性之间的相

关性激活遥远地区的图像补丁[45]，这样像雄蕊的边界在图 2 中的示例图像补丁。为减少过度拟合的风险，我们在 FC1 使用差[25, 44]，它由设置以概率 0.5 到零每个神经元节点的输出。FC1 的在我们的 CNN 的输出将被用于轮廓检测，这是一个 128 维特征向量的深度特征。我们还可以使用更多的在这一层单位来形成高维的特征向量。然而，这将导致重计算负担。在一个标准的有线电视新闻网，最后完全连接的层的输出将被馈送到一个 SOFTMAX 分级以产生过类标记的分布。在我们的例子里假设我们有 $K = 50$ 形状类，然后是 CNN 的最后一层的单位数量应该是 51。根据每一层的参数配置，CNN 的架构可以简明地通过层符号与层大小描述：

$$\begin{aligned} & \text{COV1}(45 \times 45 \times 32) \rightarrow \text{LRN1} \rightarrow \text{MAXP1} \rightarrow \\ & \text{COV2}(22 \times 22 \times 48) \rightarrow \text{LRN2} \rightarrow \text{MAXP2} \rightarrow \\ & \text{COV3}(10 \times 10 \times 64) \rightarrow \text{LRN3} \rightarrow \text{MAXP3} \rightarrow \\ & \text{COV4}(4 \times 4 \times 128) \rightarrow \text{MAXP4} \rightarrow \text{FC1}(128) \rightarrow \text{FC2}(101). \end{aligned}$$

需要注意的是，在我们的 CNN 的体系结构中的层的数目小于用于图像的通用一网通 LSVRC[5]，如轮廓总是由一个本地图像补丁具有比一般对象尺寸较小所表示。从我们的经验看到卷积层足以捕捉轮廓和背景斑块之间的区别信息。

4.2 正面分享损失函数

训练一个标准的 CNN 的目标是概率最大化的正确分类，这是通过将 softmax 损失最小化来实现的。给出其中含有 m 图像补丁训练集：

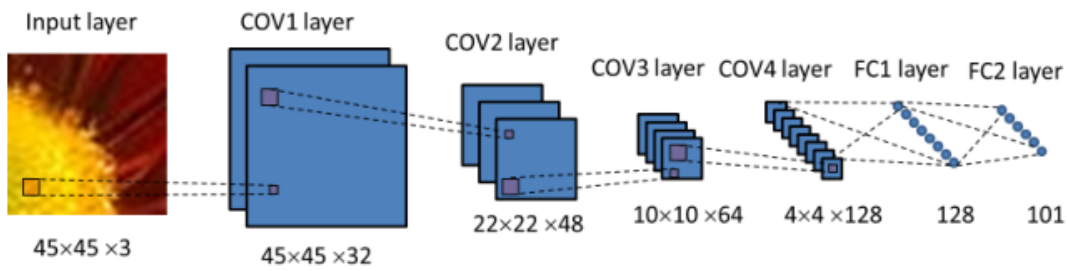


图 2. 我们的 CNN 的架构的图示, 其明确地显现每个网络层的尺寸。由于篇幅有限, 我们只显示与可学习有关的参数层。卷积层由蓝色方形表示, 而完全连接的是由蓝点标示。大和小的淡红色区块分别描述卷积内核和结果。

$\{x^{(i)}, y^{(i)}\}_{i=1}^m$, 其中, $x^{(i)}$ 是第 i 个图象贴片和 $y^{(i)} \in \{0, 1, \dots, K\}$ 是它的类的标签。如果 $y^{(i)} = 0$, 则 $x^{(i)}$ 是消极的补丁; 如果 $y^{(i)} = k > 0$, 则 $x^{(i)}$ 是一个正贴片和属于第 K 形状类。令 $(a_j^{(i)}; j = 0, 1, \dots, K)$ 为 FC2 中 j 对 $x^{(i)}$ 的输出。 $x^{(i)}$ 的概率由下式给出:

$$p_j^{(i)} = \frac{\exp(a_j^{(i)})}{\sum_{l=0}^K \exp(a_l^{(i)})} \quad (1)$$

在一个标准的 CNN 模型里, FC2 的输出是供给 $(K+1)$ SOFTMAX, 它旨在最小化以下损失函数:

$$J_0 = -\frac{1}{m} \left[\sum_{i=1}^m \sum_{j=1}^K \mathbf{1}(y^{(i)} = j) \log p_j^{(i)} \right] \quad (2)$$

其中 $\mathbf{1}(\cdot)$ 表示为指示函数。该 SOFTMAX 损失函数对每个类的分类错误等价的。然而, 在我们的情况下, 估计正贴剂的标签是一个错误的非零标签并不是一个重大的错误, 因为它仍

然是作为一个积极的预测。也就是说, 零标签和非零标签之间通过不正确的估计引起的损耗更应引起我们的关注。为此, 我们添加一个新的术语来规范损失:

$$J = J_0 - \frac{1}{m} \left[\sum_{i=1}^m \lambda \left(\mathbf{1}(y^{(i)} = 0) \log p_0^{(i)} + \sum_{j=1}^K \mathbf{1}(y^{(i)} = j) \log(1 - p_0^{(i)}) \right) \right], \quad (3)$$

其中 λ 是一个控制参数。当 λ 较小时, 式 3 往往是 SOFTMAX 损失函数; 而当 λ 越大, 形状分区的效果变弱, 式 3 趋于用于寻址二元分类问题, 轮廓为相对于非轮廓的目标函数。除非另有规定, 否则我们设置 $\lambda = 1$ 在我们所有实验的默认值。公式 3 告诉我们, 我们应该通过不同模型参数适应不同形状类的数据, 因为它们是在特征空间是不同的; 当计算分类损失的时候, 我们不能把他们当作“绝对”不同的类, 因为它们都属于积极类。我们称公式 3 共享正损失函数, 对于正类的损失每个形状类之间共享。应用标准[41,27]优化网络的参数, 计算偏导数的新损失

FC 的输出 w.r.t., $(a_j^{(i)}; j=0,1,\dots,K)$, 是必需的。式中的第一项, 公式 3 是标准 SOFTMAX 损失, 其衍生物在文献中已经提供。其余的是要获得的第二项 w.r.t. $a_0^{(i)}$ 和 $a_l^{(i)} (l=1,\dots,K)$ 的偏导数。我们可以展示:

$$\frac{\partial \log p_0^{(i)}}{\partial a_0^{(i)}} = 1 - p_0^{(i)}, \frac{\partial \log(1 - p_0^{(i)})}{\partial a_0^{(i)}} = -p_0^{(i)},$$

$$\frac{\partial \log p_l^{(i)}}{\partial a_l^{(i)}} = -p_l^{(i)}, \frac{\partial \log(1 - p_0^{(i)})}{\partial a_l^{(i)}} = \frac{p_l^{(i)} p_0^{(i)}}{1 - p_0^{(i)}}. \quad (4)$$

然后将新的损失求偏导数而得到

$$\frac{\partial J}{\partial a_0^{(i)}} = \frac{1}{m} \left[(\lambda + 1) \mathbf{1}(y^{(i)} = 0)(p_0^{(i)} - 1) + (\lambda + 1) \sum_{j=1}^K \mathbf{1}(y^{(i)} = j) p_0^{(i)} \right], \quad (5)$$

和

$$\frac{\partial J}{\partial a_l^{(i)}} = \frac{1}{m} [(\lambda \mathbf{1}(y^{(i)} = 0) + 1) p_l^{(i)} - \mathbf{1}(y^{(i)} = l) - \lambda \sum_{j=1}^K \mathbf{1}(y^{(i)} = j) \left(\frac{p_0^{(i)} p_l^{(i)}}{1 - p_0^{(i)}} \right)] \quad (6)$$

人们可以通过验证集每个补丁分类精度验证了 CNN 模型的有效性。然而, 对于检测问题, 性能更可能与阳性和阴性样品之间的对比来检测出来。因此, 我们定义一个对比得分作为量度。

$\{x_v^{(i)}, y_v^{(i)}\}_{i=1}^m$ 是验证集, $p_o^{(i)}$ 是 $x_v^{(i)}$ 的概率属于 CNN 的输出。对比得分通过下式得到:

$$\gamma = \frac{1}{m} \sum_{i=1}^m [(\mathbf{1}(y_v^{(i)} = 0) - \mathbf{1}(y_v^{(i)} > 0))(p_o^{(i)} - (1 - p_o^{(i)}))] \quad (7)$$

它的范围是-1 到 1, 测量正和负样本之间的学习模型的鉴别。我们经营的目标损失函数按梯度随机下降 (SGD),

通过设置设定为 0.001 学习速率。100000 次迭代后, 对比标准 CNN 大约是 0.56 分, 尽管我们大约是 0.59, 有 0.03 的改进。

5. 实验结果

我们用卷积分析了轮廓检测的性能。为了了解我们的 CNN 模型, 我们采取公开修改名为“咖啡” [22] 和修改 softmax 层的损失。由第一完全连接层 (FC1) 输出的 128 维特征向量是我们的特点。用于贴剂的所有像素的特征矢量被连接到一个被结构森林分类 [11,12] 来执行轮廓检测。我们通过可视化的了解该启动功能。接下来, 我们比较的是 BSDS500 数据集 [2] 在国家的最先进的轮廓检测结果。然后, 交叉验证数据集的泛化特性在 NYUD 数据集被确定 [43]。最后, 我们分析参数的影响。为结构化森林分类器, 我们使用在我们所有的实验中设置的默认参数。

5.1 深度可视化

CNN 是有效的特征提取器是一个常识, 看看我们观察学习数以百万计的图像补丁仍然是必要的, 它可以帮助我们了解学习。给定一个图像, 图像补丁密集采样输入到我们的 CNN 模型, 结果由 128 特性的 FC1 地图输出特性。我们随机选择一些具有可视化功能的地图和图在图 3 里, 令人鼓舞的是, 虽然其中一些来自致的纹理地区遭受

的噪音的影响，依然有很多是相当稀疏可以捕捉的对象，这将有利于轮廓检测。

5.2 BSDS500 数据集的结果

我们的大多数实验都在 BSDS500 数据集执行。该数据集的细节已经在第 3 章介绍。为了评价一个轮廓检测算法的性能，我们用到了 3 个标准量：在数据集固定比例的最佳 F 值(ODS)，在每个图像 (OIS) 的数据集的最佳比例的合计 F 值，平均精度 (AP) 的全面召回范围。我们把我们的轮廓检测的方法和其他的方法做对比，例如结构化的边缘 (SE)[11] 及其变种 (SE-Var)[12], 稀疏编码梯度 (SCG)[39] 和深层神经网络预测 (DeepNet)[23]。精度/召回曲线见图 4 和汇总统计见表 1。我们深轮廓的方法是优于国家的最先进的所有方法。我们比同类方法提高了 1 个点的竞争率/ OIS，同时可以实现和 AP 媲美。通过比较 SE 和 SE-Var，我们直接得出的证明是深度特点是比手设计的功能更详细识别的，如梯度信道特征和自相似的特性，在[11,12]. 的使用中。相较于其他处理轮廓检测的深度学习基础的方法，我们相当强于大盘 DEEPNET[23]，它的无监督学习的轮廓特征，因为学习的特点与监管提高了歧视。我们甚至胜过了 DeepNet 方法因为 DeepNet 是无监督的学习检测方法，而我们的是有监督的。引人注意的是，为了处理不同的形状类的轮廓之间的转换，深度网络

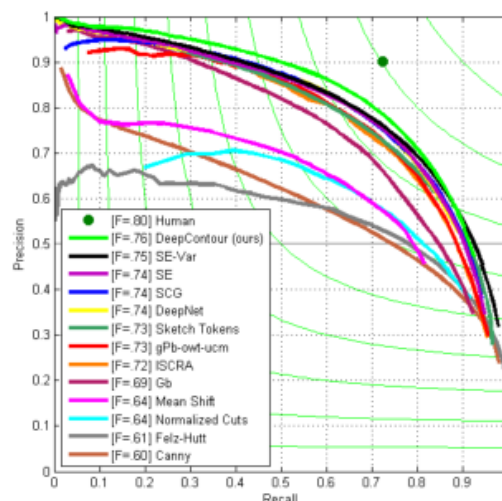


图 4. 轮廓检测器上的 BSDS500 数据集评价 [2]。轮廓检测方法是根据它最好的 F-尺寸 (ODS) 来归类的。我们的方法里深轮廓达到顶级结果，并同时显示改善在多精确召回制度的召回率和准确度。在表 1 有关于其他两个量和方法的引用的更多细节。

旋转版本，同时预测结果被平均化并在[23]上被增强体现出来，同时我们在训练过程也考虑了不同形状之间的多样性。不幸的是，深度网络的增强结果并没有体现出来。所以，我们没有办法表现出更多的有关不同形状效应的细节。同时，我们的方法也比其他深度学习方法 (N-Fields) [19]更成功的地方，尽管他们也提出了 CNNs 模型。CNN 模型是针对地区蜿蜒的局部地图的，它靠本身的模型隐式的将形状分区。。然而，它们 CNN 的特征空间使用最邻近搜索方法获得本地等高线图，但是 CNN 的特征图嘈杂响应表现不佳，如图 3 所示。我们在这个分类器应用随机 CNN 特性和嵌入式特征

选择机制提高了对噪声的鲁棒性。当我们根据草图令牌[30]的定义获得形状类时，他们之间的相当大的性能提升证明了我们深度特性方法的优越性。我们展示了几种获得的轮廓方法的检测结果在表 5 中定性比较。这些定性的例子表明，我们的方法触发了地面真识轮廓更加有力的回应，同时抑制了假阳性。

5.3 跨数据集泛化

某些人可能会涉及这样的失误，了解的一个数据集深度功能应用到其他数据集时可能会导致更高的泛化误差。为了探讨是否有这种情况，我们应用训练集中 BSDS500 数据集的数据集 NYUD[43]了解深度特性。该数据集 NYUD (V2) 包括 1449 双 480×640

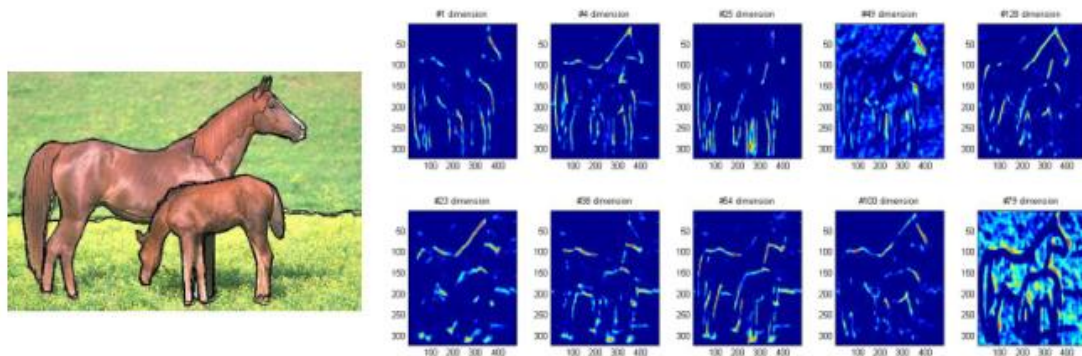


图 3.左：颜色测试图像，地面实况轮廓以黑色曲线显示。右：随机抽取深层特征图。

表 1. BSDS500 数据集轮廓检测结果[2]

	ODS	OIS	AP
Human	.80	.80	-
Canny [6]	.60	.63	.58
Felz-Hutt [16]	.61	.64	.56
Normalized Cuts [8]	.64	.68	.45
Mean Shift [7]	.64	.68	.56
Gb [29]	.69	.72	.72
ISCRA [40]	.72	.75	.46
gPb-owt-ucm [2]	.73	.76	.73
Sketch Tokens [30]	.73	.75	.78
DeepNet [23]	.74	.76	.76
SCG-[39]	.74	.76	.77
PMI+sPb [21]	.74	.77	.78
SE [11]	.74	.76	.78
SE-Var [12]	.75	.77	.80
N ⁴ -Fields [19]	.75	.77	.78
DeepContour (ours)	.76	.78	.80

分辨率的颜色和深度帧和地面实况语义区。此数据集是从各种室内场景组成的图像，而在 BSDS500 图像主要示出了室外场景。因此，在这两个数据集的对象完全不同。我们使用由[39]

表 2. 跨数据集泛化结果。训练/测试得出训练/测试所用的数据集。

	ODS	OIS	AP
gPb [2] (NYU/NYU)	.51	.52	.37
SCG [39] (NYU/NYU)	.55	.57	.46
SE [11] (BSDS/NYU)	.55	.57	.46
DeepContour (BSDS/NYU)	.55	.57	.49

中提出的相同的实验装置，它选择训练检测的图像为 60% 的图像和 40% 以缩小为 320×240 分辨率的图像。由于我们深度特性是从彩色图像 BSDS500 数据了解到，所以我们只将它们应用到 NYUD 数据集的彩色图像。为了比较其效果，我们在表 2 列出了我们方法和 SE[11]的跨数据集概括结果（从模型上的 BSDS500 数据集到 NYUD 数据集）。GOP 和 SCG 的结果被用于参考，而它们是由在 NYUD 数据集训练得到。虽然监督学习通常

减少一般性的博学深的特点实现了比 SE 相当或更好的跨数据集的泛化的结果, 即使是在 NYUD 数据集中培训也显著优于 GPB-OWT-UCM。选择的定性的例子在图 6 中可以看到并且从 BSDS500 数据集得到我们深度轮廓特征, 这证明我们深度特性是一般和便携式的轮廓表示。

5.4 参数探讨

我们在我们的测试集的 BSDS 数据集进行验证的方法介绍了参数的影响。我们有学习两个功能的重要参数: 形状类 k 的数量和我们的损失功能的控制参数 λ 。



图 5. 六个选定示例图像在 BSDS500 数据集轮廓检测结果的插图。前两行展示出了原始图像和地面实况。接下来的五行描绘 GPB-OWT-UCM[2], 素描令牌[30], SCG[39], SE-瓦尔[12]和深切的轮廓结果。请注意, 我们的方法触发地面上的真相轮廓强反应(如鲸在第五列中的轮廓), 并同时抑制了假阳性(如小鱼在第五列中的边缘)。这是更好地使用浏览器缩放功能去看细节。

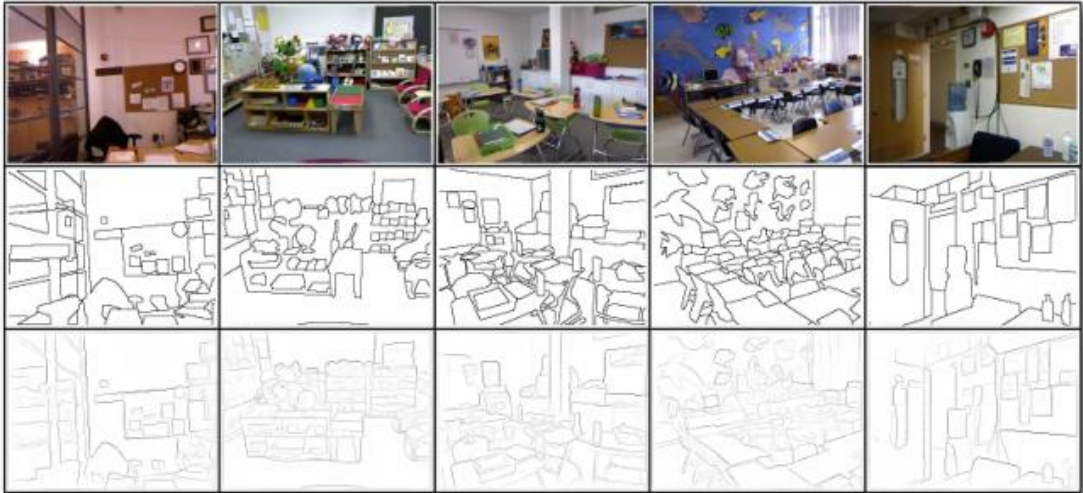


图 6. 在 NYUD 数据集提供了轮廓的检测结果的说明[43]为五个选定的示例图像（深度图像不使用）。在各实施例中，我们看到分别为原始图像和地面实况和我们的结果。我们从 BSD 系统数据了解到深度的轮廓特征，他们可以代表 NYUD 数据集中的物体轮廓。

在图 7 我们探讨的这两个参数的选择的影响。标准度量 OSD 被用于测量的精确度。为了节省时间，我们不再适用于结构化林[11]多尺度的战略，这将导致 OSD 减少（约 0.006）。需要注意的是，设置 $K=1$ 表示不进行形状类分区，则我们的模型等效于学习的二进制轮廓，而使用 SOFTMAX 损失非轮廓分类器，从而显著降低了性能。这证据表明形状类划分是必要的。最佳结果是另 $K=50$ 。如果 $\lambda=0$ ，那么，我们的损失函数减小到 SOFTMAX 损失，这也降低了性能。设置一个相当大的 λ ，我们的损失函数趋于只着眼于损失轮廓和非轮廓的类，这也导致性能降低。

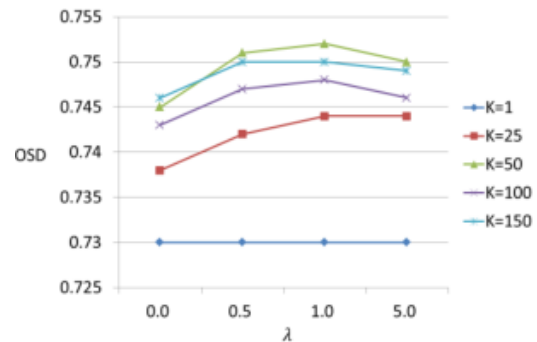


图 7. 有参数扫描得精度。K 是分区形状类的数量， λ 是在我们的损失函数引入的控制参数。

6. 总结与展望

在这项工作中，我们成功地展示了如何借鉴在自然图像轮廓检测深卷积神经网络的判别特征。我们强调两点：一个是分区的轮廓（正）数据的子类对培训有效的 CNN 模型是必要的，另一个是建议积极分享损失函数它强调了轮廓和非轮廓而不是针对每个损失的损失子类有利于探索比 SOFTMAX 损失函数更多判别功能。在 BSDS500 数据集[2]的实验表明该算法跑赢文献其他竞争方法。通过参

数扫描我们验证了积极的数据分区的必要性和拟议损失函数的有效性。我们也验证了对 NYUD 数据集[43]功能交叉的数据集通用性。

感谢 这项工作是由中国国家自然科学基金在格兰特 61303095 和 61222308, 中国高等教育博士学科研究基金在格兰特 20133108120017, 创新计划下上海教委 3989 格兰特 14YZ018, 中国中央电教馆项目-12-0217 的教育部 de 支持下进行的。同时我们感谢 NVIDIA 公司为我们的学术研究提供了 GPU 设备。

引用:

- [1] P. Agrawal, R. B. Girshick, and J. Malik. Analyzing the performance of multilayer neural networks for object recognition. In Proc. ECCV, pages 329–344, 2014.
- [2] P. Arbelaez, M. Maire, C. Fowlkes, and J. Malik. Contour detection and hierarchical image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 33(5):898–916, 2011.
- [3] J. T. Barron and J. Malik. Intrinsic scene properties from a single RGB-D image. In Proc. CVPR, pages 17–24, 2013.
- [4] J. L. Bentley. Multidimensional divide-and-conquer. *Commun. ACM*, 23(4):214–229, 1980.
- [5] A. Berg, J. Deng, and L. Fei-Fei. Imagenet large scale visual recognition challenge 2012. <http://www.image-net.org/challenges/LSVRC/2012/>, 2012.
- [6] J. Canny. A computational approach to edge detection. *IEEE Trans. Pattern Anal. Mach. Intell.*, 8(6):679–698, 1986.
- [7] D. Comaniciu and P. Meer. Mean shift: A robust approach toward feature space analysis. *IEEE Trans. Pattern Anal. Mach. Intell.*, 24(5):603–619, 2002.
- [8] T. Cour, F. Bédét, and J. Shi. Spectral segmentation with multiscale graph decomposition. In Proc. CVPR, pages 1124–1131, 2005.
- [9] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In Proc. CVPR, pages 886–893, 2005.
- [10] P. Dollár, Z. Tu, and S. Belongie. Supervised learning of edges and object boundaries. In Proc. CVPR, volume 2, pages 1964–1971, 2006.
- [11] P. Dollár and C. L. Zitnick. Structured forests for fast edge detection. In Proc. ICCV, pages 1841–1848, 2013.
- [12] P. Dollár and C. L. Zitnick. Fast edge detection using structured forests. arXiv preprint arXiv:1406.5549, 2014.
- [13] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell. Decaf: A deep convolutional activation feature for generic visual recognition. In Proc. ICML, pages 647–655, 2014.
- [14] C. Dong, C. C. Loy, K. He, and X. Tang. Learning a deep convolutional network for image super-resolution. In Proc. ECCV, pages 184–199. Springer, 2014.
- [15] C. Farabet, C. Couprie, L. Najman, and Y. LeCun. Learning hierarchical features for scene labeling. *IEEE Trans. Pattern Anal. Mach. Intell.*, 8(6):679–698, 1986.

- Mach. Intell., 35(8):1915–1929, 2013.
- [16] P. F. Felzenszwalb and D. P. Huttenlocher. Efficient graphbased image segmentation. *International Journal of Computer Vision*, 59(2):167–181, 2004.
- [17] V. Ferrari, L. Fevrier, F. Jurie, and C. Schmid. Groups of adjacent contour segments for object detection. *IEEE Trans. Pattern Anal. Mach. Intell.*, 30(1):36–51, 2008.
- [18] K. Fukushima. Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biological cybernetics*, 36(4):193–202, 1980.
- [19] Y. Ganin and V. S. Lempitsky. N⁴-fields: Neural network nearest neighbor fields for image transforms. In *Proc. ACCV*, 2014.
- [20] R. B. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate eobject detection and semantic segmentation. In *Proc. CVPR*, pages 580–587, 2014.
- [21] P. Isola, D. Zoran, D. Krishnan, and E. H. Adelson. Crisp boundary detection using pointwise mutual information. In *Proc. ECCV*, pages 799–814, 2014.
- [22] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell. Caffe: Convolutional architecture for fast feature embedding. *arXiv preprint arXiv:1408.5093*, 2014.
- [23] J. J. Kivinen, C. K. I. Williams, and N. Heess. Visual boundary prediction: A deep neural prediction network and quality dissection. In *Proc. AISTATS*, pages 512–521, 2014.
- [24] S. Konishi, A. L. Yuille, J. M. Coughlan, and S. C. Zhu. S-tatistical edge detection: Learning and evaluating edge cues. *IEEE Trans. Pattern Anal. Mach. Intell.*, 25(1):57–74, 2003.
- [25] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Proc. NIPS*, pages 1106–1114, 2012.
- [26] Y. LeCun, B. Boser, J. Denker, D. Henderson, R. Howard, W. Hubbard, and L. Jackel. Backpropagation applied to handwritten zip code recognition. *Neural computation*, 1(4), 1989.
- [27] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.
- [28] C.-Y. Lee, S. Xie, P. Gallagher, Z. Zhang, , and Z. Tu. Deeply-supervised nets. In *Proc. AISTATS*, 2015.
- [29] M. Leordeanu, R. Sukthankar, and C. Sminchisescu. Generalized boundaries from multiple image interpretations. *IEEE Trans. Pattern Anal. Mach. Intell.*, 36(7):1312–1324, 2014.
- [30] J. J. Lim, C. L. Zitnick, and P. Dollár. Sketch tokens: A learned mid-level representation for contour and object detection. In *Proc. CVPR*, pages 3158–3165, 2013.
- [31] D. G. Lowe. Distinctive image features

from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.

[32] M. Maire, P. Arbelaez, C. Fowlkes, and J. Malik. Using contours to detect and localize junctions in natural images. In *Proc. CVPR*, 2008.

[33] M. Maire, S. X. Yu, and P. Perona. Reconstructive sparse code transfer for contour detection and semantic labeling. In *Proc. ACCV*, 2014.

[34] D. R. Martin, C. Fowlkes, and J. Malik. Learning to detect natural image boundaries using local brightness, color, and texture cues. *IEEE Trans. Pattern Anal. Mach. Intell.*, 26(5):530–549, 2004.

[35] D. R. Martin, C. Fowlkes, D. Tal, and J. Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *Proc. ICCV*, pages 416–425, 2001.

[36] V. Nair and G. E. Hinton. Rectified linear units improve restricted boltzmann machines. In *Proc. ICML*, pages 807–814, 2010.

