网络中的网络

原文来源：Lin M, Chen Q, Yan S. Network in network[J]. arXiv preprint arXiv:1312.4400, 2013.

1. 摘要

我们发布了一个新奇的深层网络结构“网络中的网络”（NIN）去增强模型对接受域中的块的识别能力。传统的卷积层运用线性的感知器后面加上非线性的激活函数去扫描输入。相反，我们建立了迷你型却具有更复杂的结构的神经网络去抽象接受域中的图数据。我们用一个多层的感知器（同时也是一个有效的函数模拟器）去初始化这个迷你网络，我们通过不断在与 CNN 相似方式得到的输入上滑动迷你网络去得到特征图。随后他们被送入到下一层。深层的 NIN 网络可以是多个上述结构的堆叠，通过借助迷你网络增强局部建模，我们能够利用在分类器层的对于特征图全局的平均池化来达到更简单的分类，而且比起传统的全连接层更不容易出现过拟合的情况。我们在 CIFAR-10 and CIFAR-100,上取得了领先的成绩，并在 SVHN 和 MNIST 上也取得了不错的成绩。

2 介绍

卷积神经网络由多个卷积层和池化层组成。卷积层拿取线性感知器的产出结果和在每个局部输入区域都有的非线性激活函数后面的潜在的接受域。然后输出被称为特征图的结果。

CNN 中的卷积感受器是一个针对于潜在数据块的广义的线性模型（GLM），我们可以指出 GLM 的抽象级别是很低的。这里的抽象可以理解为在相同的概念上变量的特征是不变的。将 GLM 模型替换成一个更强力的非线性函数模拟器可以增强对局部模型的抽象能力。当样本的特征处于线性分布时，GLM 可以达到一个较好的抽象水平。所以传统的卷积网络往往假设样本的潜在分布是线性的。不过样本的分布却往往是非线性的，所以抓取信息的模型往往是输入中的非线性函数。在 NIN 中，GLM 被带有非线性函数模拟器的迷你网络所代替。在这个工程中，我们选择多层感知器作为迷你网络的实例化。

这是一个普遍的函数模拟器和一个用反向传播训练的神经网络。

结果我们称之为一个 mlpconv 层结构。线性感知器和 mlpconv 都将接受域映射到一个输出特征向量。Mlpconv 用 MLP 将输入局部快映射到输出特征向量。MLP 在所有局部接受域中是共享的。特征图是将 MLP 在不断在与 CNN 相似方

式得到的输入上滑动迷你得到的。NIN 总体结构是多个 mlpconv 层的叠加。它被称为“网络的网络”(NIN)。

相比于采用传统的完全连接层分类的 CNN,我们直接从上一个 mlpconv 层输出特性的空间平均。当做通过全局平均池层的置信类别,然后生成的向量送入 softmax 层。在传统的 CNN 中，由于全连接层作为一个黑盒，因此很难解释如何将 softmax 的信息目标从本层传递回以前的卷积层。相比之下,全局平均池化更具有意义和可执行性，因为他加强了映射和类别的关联,这是可能由由于使用微网络建模导致的强大的局部模型。此外,完全连接层容易过度拟合和严重依赖 dropout,而全局平均池化本身就是一个结构调整,这将自然地防止所有结构的过拟合。

2.1 卷积神经网络

传统的卷积神经网络由很多的卷积层和池化层组成，这些卷积层依赖于线性感知器和后面的非线性激活函数产生特征图。用线性的 rectifier 函数做个例子，特征图计算如下：

$$f_{i,j,k} = \max(w_k^T x_{i,j}, 0). \tag{1-1}$$

公式中的 ij 是特征图中的像素索引，Xij 代表以 i.j 为中心的输入区域，K 被用来标记特征图中的通道。

这个线性卷积在输入特征是线性分布是非常好使，但是能最好表现数据特征的都是非线性的函数。在传统的卷积网络中，这个问题通常用利用一组过量的感知器涵盖所有级别的变量的方式解决。理论上，独立的线性感知器可以被训练到涵盖相同级别上的所有变量。尽管如此，需要考虑前一层输入的所有组合的过量的感知器给系统带来沉重的负担。在 CNN 中，相比于原输入，高层的感知器将映射出更大的区域。将之前低层特征组合起来会产生更高级别的特征。所以我们指出，将每个局部块在组合成更高层之前进行抽象是更好的。

在最近的 maxout 网络中，特征图的数量被用在仿射特征图上最大化池化所减少。(仿射特征图是指线性卷积后的未经过激活函数额直接结果)。线性函数上的最大化造成了可以胜任模拟任何卷积函数的分段线性函数。相比于行使线性分割的传统卷积层，maxout 网络是更加强力的，因为他可以分割在曲线集中的数据。这个提升赋予了 maxout 网络在数个标本集中的良好表现。

尽管，maxout 在处理输入曲线集中的隐含维度的优秀表现令人印象深刻。但是当输入数据的分布特性越来越复杂时，构造一个更加普遍的函数模拟器是更加必要的。为了解决这个问题，我们推出了 NIN 网络，在网络中的每一层都拥有

迷你网络去计算局部数据块中更抽象的特征。

在以前的网络中，滑动迷你网络就已经运用了。比如结构化复合层感知器（SMLP）运用了在不同局部块上的地共享的多层感知器。另一项，一个基于感知器的神经网络被用于人脸检测。虽然，他们都是为特定的问题而设计并都只包含一个滑动网络结构。NIN 从一个更加普遍的环境中被设计，为了追求对所有层次的特征的更好抽象，迷你网络被加入到 CNN 模型中。

3 卷积神经网络

我们首先聚焦于 NIN 网络中最关键的组成部分：在 3.1 和 3.2 节中介绍 LP 卷积层和全局平均池化层，在 3.3 中介绍 NIN 细节。

3.1 MLP 卷积层

我们不对数据的特征分布做任何的假设，那么我就会很需要运用一个普遍的函数模拟器对局部数据块的特征进行分析。Radial basis 网络和多层感知器是两个知名的普遍型的函数模拟器。我们选择多层的感知器有两个原因，第一：多层感知器与卷积神经网络是可以兼容的，其二：多层感知器也可在作为深度网络本身，他拥有特征重用的精神。在这篇文章中，在 mlpconv 中我们用 MLP 代替 GLM 去处理输入数据。Mlpconv 的计算公式如下：

$$f_{i,j,k_1}^{1} = \max({w_{k_1}^{1}}^{T} x_{i,j} + b_{k_1}, 0). \quad (3\text{-}1)$$

$$\vdots$$

$$f_{i,j,k_n}^{n} = \max({w_{k_n}^{n}}^{T} f_{i,j}^{n-1} + b_{k_n}, 0). \quad (3\text{-}2)$$

这里的 n 代表多层感知器的层数。在感知器中 Rectifird 线性单元被用作激活函数。

从交叉通道池化观点来看，等式 2 等价于相当于级联，在正常卷积层横通道参数池化。每一个池化层在输入特征图上行使权重线性解析，然后传递到激活函数。这个横通道池化特征图在下一层被一遍遍的横通道池化。这个级联横通道参数池化结构允许与横通道信息的复杂的和可学习的交互。

横通道参数池化层也等价于一个 1*1 的卷积核。这个交互是的他更直观的了解 NIN 的结构。

相比于 maxout 层：maxout net 中的 maxout 层对多重的隐含特征图进行了最大值池化的操作。Maxout 层的计算如下：

$$f_{i,j,k} = \max_m (w_{k_m}^T x_{i,j}). \quad (3\text{-}3)$$

在线性函数上的 maxout 形成了一个分片线性函数，他可以胜任模型化任何曲线函数。对于一个曲线函数，拥有特定阈值的样本形成一个曲线集。这样的话在局部块中模拟曲线函数，maxouot 就能分析出曲线集中样本的复杂维度的特征。Mlpconv 层与 maxout、
层在曲线函数模拟器被取代为普遍函数模拟器上，这样就能在模型化多样的特征分布上有更好的性能。

3.2 全局池化

传统卷积神经网络在网络的低层上进行卷积操作。比如分类，特征图由最后一个卷积层输入到全连接层中和后面的 softmax 层中。这是传统分类的结构，这是把卷积层当做特征提取器，结果特征也用传统方式进行分类。

尽管，全连接层容易过拟合，因此妨碍整个网络的工作能力。由 Hinton 发布的 Dropout 层会随机将一半的全连接层置为休眠。这样提升了整个网络的工作能力和防止过拟合的能力。

在这篇论文中，我们使用另一种策略叫做全局平均池化去代替传统的全连接层。这个想法是在最后一个 mlpcoonv 为每一个相关的分类任务产生一个特征图。代替了直接在特征图上直接添加全连接层，我们取每一个特征图的平均值，然后结果被直接送入 softmax 中。全连接层前的平均池化的一个优势在于增强特征图和输入对于卷积结构更加自然。因此特征图能被更容易的映射为输入置信图。另一个优势在于，全局平均池化
没有参数，所以这一层就没有过拟合的危险。而且，全局池化加上了空间信息，因此在输入的空间转换上更加鲁棒。

我们可以将全局平均池化看做一个专注于增强特征图转化为置信图的结构正则化。这也使得利用 mlpconv layers 模拟出比 GLMs 更好的函数模拟器。

3.3 网络中的网络结构

NIN 整体上是 mlpconbv 的堆叠，在这个上面有全局平均池化，和目标代价函数。Sub-sampling 层也可以添加在 mlpconv 层。在每一个 mlpconv 层中有一个三层的感知器。NIN 和迷你网络中的层数都是可以调整的。

4 实验

4.1 总览

我们在四个公开集上取得了实验效果：CIFAR-10 [12], CIFAR-100 [12], SVHN [13] 和 MNIST [1]. 使用的网络数据集都包含三叠 mlpconv 层, 在所有的实验中 mlpconv 层都是紧随一个空间最大值池化，降低了输入图像的取样。作为一个正则化,dropout 应用于所有的输出除了最后的 mlpconv 层。除非特别提到的,所有实验中使用的网络都使用全局平均池化而不是全连接层。另一个所使用的规范应用为权值衰弱。我们在由亚历克斯实现的 cuda-convnet 上实现我们的网络代码。数据的预处理,分割训练和验证集都遵循格 Goodfellow。

我们采用 Krizhevsky 等使用的培训过程。。也就是说,我们手动设置正确初始化权重和学习率。网络使用 mini-batches 规模为 128。训练过程从最初的重量和学习速率,仍继续直到训练集上的准确性停止变化,然后学习率降低 10 倍。这个过程会重复一次,这样最后的学习速率会是初始值的百分之一。

4.2 CIFAR-10

CIFAR-10 数据集是由 10 类 50000 张自然图像组成的训练集和 10000 张测试图像。每个图像 RGB 图像大小的 32×32。对于这个数据集,我们应用与被格拉汉姆·古德费勒相同的全局标准化和 ZCA 白化。在 maxout 网络中，我们用训练集的最后 10000 图像做验证数据。

每个 mlpconv 层的特征图的数量在这个实验中设置为与相应的 maxout 网络相同数量。两个 hyper-parameters 调谐使用验证集,即局部接受域大小和权值衰减。在 hyper-parameters 确定之后，我们利用训练集和验证集重新训练网络，由此产生的模型用于测试。我们在这个数据集获得的 10.41%的测试误差,比起之前的最佳效果提升了百分之一。比较如下表：

表 1. 在各种方式下 CIFAR-10 上的验证集错误率
Table1. Test set error rates for CIFAR-10 of various methods

Method	Test Error
Stochastic Pooling [11]	15.13%
CNN + Spearmint [14]	14.98%
Conv. maxout + Dropout [8]	11.68%
NIN + Dropout	**10.41%**
CNN + Spearmint + Data Augmentation [14]	9.50%
Conv. maxout + Dropout + Data Augmentation [8]	9.38%
DropConnect + 12 networks + Data Augmentation [15]	9.32%
NIN + Dropout + Data Augmentation	**8.81%**

这证明了我们在 mlpconv 间加 dropout 的实验通过增强模型的泛化能力而改善网络的表现。如图 3 所示，加入后减少的误差率超过了 20%。这个观测结果是在 Goodfellow 上获得的可靠结果。因此 dropout 被加在 mlpconv 层间。没有 dropout 的模型错误率在 CIFAR-10 上达到了 14.51%，这依然远远超过了之前许多名列前茅的结果（maxout 除外）。因为 maxout 没有 dopout，因此只有有 dropout 层的在这里比较

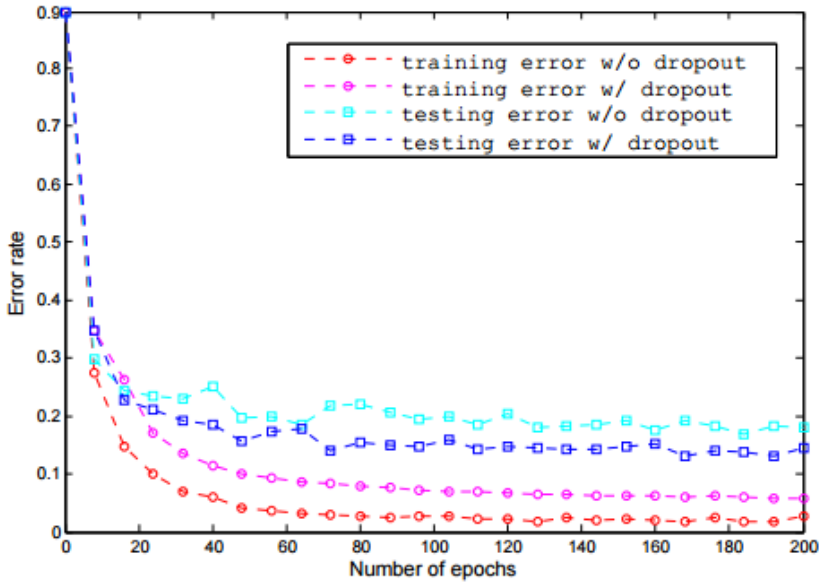

图 1.NIN 网络的测试训练错误率

Fig1. Trainingandtestingerror of NIN

4.3 CIFAR-100

CIFAR-100 数据集在格式和大小上和 CIFAR-10 相同，但他有 100 类，因此每个种类的图片只有 CIFAR-10 的十分之一。对于 CIFAR-100 我们不调整他的 hyper—parameters，用和 CIFAR-10 一样的设置。唯一的不同是最后一个 mlpconv 输出 100 维向量。错误率 35.68%，增加了 1%。详情见表

表 2. CIFAR-100 上的验证集错误率
Table2. Test set error rates for CIFAR-100 of various methods

Method	Test Error
Learned Pooling [16]	43.71%
Stochastic Pooling [11]	42.51%
Conv. maxout + Dropout [8]	38.57%
Tree based priors [17]	36.85%
NIN + Dropout	**35.68%**

4.4 Street View House Numbers

SVHN 由 630420 张 32*32 的图片，分为训练，测试和额外集。这个集合任务是识别中心的数字。训练测试步骤遵循 Goodfellow。约 400 张训练集中，额外集中 200 张样本用来作为验证集。剩下的图片用来训练。验证集只用来做参数选择，不用来训练。数据集的处理依照 Goddfellow。这里的结构和参数都和 CIFAR-10 相似，三个 mlpconv 和一个紧随的全局平均池化层。在这个集中我们获得了 2.35%的错误率，对比如下：

表 3. SVHN 上的验证集错误率
Table3. Test set error rates for SVHN of various methods

Method	Test Error
Stochastic Pooling [11]	2.80%
Rectifier + Dropout [18]	2.78%
Rectifier + Dropout + Synthetic Translation [18]	2.68%
Conv. maxout + Dropout [8]	2.47%
NIN + Dropout	2.35%
Multi-digit Number Recognition [19]	2.16%
DropConnect [15]	**1.94%**

4.5 MNIST

MNIST 由手写的 0-9 的 28*28 大小图片组成，6000 张训练图片，10000 张测试图片。我们用和 CIFAR-10 相同的结构。但每个 mlpconv 层的特征图数量减少。因为 MNIST 比起 CIFAR-10 要简单。不需要那么多参数，我们在上面获得的结果和以前的模型进行了比较：

表 4. MNIST 上的验证集错误率
Table4. Test set error rates for MNIST of various methods

Method	Test Error
2-Layer CNN + 2-Layer NN [11]	0.53%
Stochastic Pooling [11]	0.47%
NIN + Dropout	0.47%
Conv. maxout + Dropout [8]	**0.45%**

4.6 Global Average Pooling as a Regularizer

全局平均池化层类似于全连接层，因为他们都是对向量特征图的线性转换。但他们的不同点在于转换矩阵。对于全局平均池化层来说，转换矩阵事先设好且没有为零的项，在建块元素上他们参数共享。全连接层有密集的转换矩阵且值为反向传播而设置。为了学习其正则化的效果，我们把池化层换成了全连接层，保持其他部分一致。我们测算了这个没有 dropout 在全连接层前面的模型，这些都是在 CIFAR-10 数据集上测试，获得如下比较：

表 5. 全局平局池化与全连接层比较
Table5. Global average pooling compared to fully connected

Method	Testing Error
mlpconv + Fully Connected	11.59%
mlpconv + Fully Connected + Dropout	10.88%
mlpconv + Global Average Pooling	10.41%

如表所示，没有 dropout 的全连接层有最坏的表现（11.59%），这应该是全连接层过拟合造成的。加了 Dropout 后，错误率达到了 10.88%，池化达到最低的错误率 10.41%。

然后我们探究了池化层在传统卷积网络中是否有同样的效果，我们按照 Hinton 的方式实例化了一个网络。三个卷积层和一个局部连接层。产生一个 16 维特征图传递给带有 Dropout 的全连接层。为了公平的比较，我们减少了特征图的维数（16->10），每一个分类只有一个特征图。一个等价的带有池化层的网络随后被创建出来取代带有 dropout 的全连接层。然后在 CIFAR-10 上进行测试。

有全连接的 CNN 网络只达到了 17.56%，加了 dropout 后达到了 15.99%，池化层达到了 16.46%，比没有 dropout 的全连接层提高了 1 个百分点。这再次证明了池化层作为正则化层的效率性。虽然比带有 Dropout 的全连接稍弱，但是我们

觉得原因在于池化层也许对线性卷积层的要求太苛刻。他要求带有 rectified 激活函数的线性感知器构造出分类的置信向量。

4.7 Visualization of NIN

我们将在最后一层 mlpconv 的特征向量当做分类的置信向量，这只有当强健的局部接受域存在时候才可行。例如 NIN 中的 mlpconv，为了理解这个目标的完成度，我们抽取并直接可视化了最后一层 mlpconv 的特征图（CIFAR-10）

下图显示了一些样例图片和他们相关的特征图，预计能在特征图中看见被池化显示出来的最大的激活量，在特征图里我们能看到最大的激活量在原图中同样的位置也显现了出来。如在第二行的的车，但这只是用了类别信息训练，如果用了边缘信息，效果会更好。

可视化再一次证明了 NIN 的效果，他通过强烈的局部接受域体现，池化层随后保证了其对特征类别信息的学习，未来可对整体目标识别进行研究。检测结果可通过在同一级别的如 Farabel 的屏幕标签分类特征图实现。

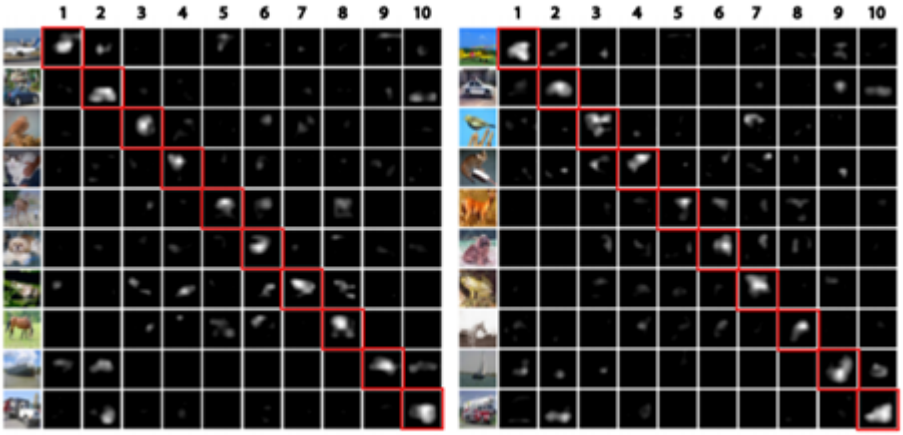

图 1.对最后一个卷积层特征图的可视化

Fig1. Visualization of the feature maps from the last mlpconv layer

5 结论

我们推出了一个叫做网络中的网络的新奇的深度网络来完成分类任务，这个新的结构由用多层感受器的 mlpconv 层去卷积输入和一个全局平均池化层作为全连接层的替代，Mlpconv 层能对局部数据块更好的建模作为一个正则化块也能更好的防止过拟合。这两个组成元素帮助我们在 CIFAR-10 CIFAR-100 SVHN 数据集上取得了领先的效果。通过特征图的可视化，我们将最后一个 mlpconv 层的特征图当做分类置信向量，这也增加了利用 NIN 做目标检测的可能性。

参考文献

[1] Yann LeCun, Leon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning ´
applied to document recognition. Proceedings of the IEEE, 86(11):2278–2324, 1998.
[2] Y Bengio, A Courville, and P Vincent. Representation learning: A review and new perspectives.
IEEE transactions on pattern analysis and machine intelligence, 35:1798–1828, 2013.
[3] Frank Rosenblatt. Principles of neurodynamics. perceptrons and the theory of brain mechanisms.
Technical report, DTIC Document, 1961.
[4] Alex Krizhevsky, Ilya Sutskever, and Geoff Hinton. Imagenet classification with deep convolutional
neural networks. In Advances in Neural Information Processing Systems 25, pages 1106–1114, 2012.
[5] Geoffrey E Hinton, Nitish Srivastava, Alex Krizhevsky, Ilya Sutskever, and Ruslan R Salakhutdinov.
Improving neural networks by preventing co-adaptation of feature detectors. arXiv preprint arXiv:1207.0580, 2012.
[6] Quoc V Le, Alexandre Karpenko, Jiquan Ngiam, and Andrew Ng. Ica with reconstruction cost
for efficient overcomplete feature learning. In Advances in Neural Information Processing
Systems, pages 1017–1025, 2011.
[7] Ian J Goodfellow. Piecewise linear multilayer perceptrons and dropout. arXiv preprint
arXiv:1301.5088, 2013.
[8] Ian J Goodfellow, David Warde-Farley, Mehdi Mirza, Aaron Courville, and Yoshua Bengio.
Maxout networks. arXiv preprint arXiv:1302.4389, 2013.
[9] C¸ aglar G ˘ ulc¸ehre and Yoshua Bengio. Knowledge matters: Importance of prior information for ¨
optimization. arXiv preprint arXiv:1301.4083, 2013.
[10] Henry A Rowley, Shumeet Baluja, Takeo Kanade, et al. Human face detection in

visual scenes.
School of Computer Science, Carnegie Mellon University Pittsburgh, PA, 1995.
[11] Matthew D Zeiler and Rob Fergus. Stochastic pooling for regularization of deep convolutional
neural networks. arXiv preprint arXiv:1301.3557, 2013.
[12] Alex Krizhevsky and Geoffrey Hinton. Learning multiple layers of features from tiny images.
Master's thesis, Department of Computer Science, University of Toronto, 2009.
[13] Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Y Ng.
Reading digits in natural images with unsupervised feature learning. In NIPS Workshop on
Deep Learning and Unsupervised Feature Learning, volume 2011, 2011.
[14] Jasper Snoek, Hugo Larochelle, and Ryan P Adams. Practical bayesian optimization of machine
learning algorithms. arXiv preprint arXiv:1206.2944, 2012.
[15] Li Wan, Matthew Zeiler, Sixin Zhang, Yann L Cun, and Rob Fergus. Regularization of neural
networks using dropconnect. In Proceedings of the 30th International Conference on Machine
Learning (ICML-13), pages 1058–1066, 2013.
[16] Mateusz Malinowski and Mario Fritz. Learnable pooling regions for image classification.
arXiv preprint arXiv:1301.3516, 2013.
[17] Nitish Srivastava and Ruslan Salakhutdinov. Discriminative transfer learning with tree-based
priors. In Advances in Neural Information Processing Systems, pages 2094–2102, 2013.
[18] Nitish Srivastava. Improving neural networks with dropout. PhD thesis, University of Toronto,
2013.
[19] Ian J Goodfellow, Yaroslav Bulatov, Julian Ibarz, Sacha Arnoud, and Vinay Shet. Multi-digit
number recognition from street view imagery using deep convolutional neural networks. arXiv

preprint arXiv:1312.6082, 2013.

[20] Clement Farabet, Camille Couprie, Laurent Najman, Yann Lecun, et al. Learning hierarchical ´

features for scene labeling. IEEE Transactions on Pattern Analysis and Machine Intelligence,

35:1915–1929, 2013.