

图像处理方法、装置、计算机设备和计算机存储介质

相关申请的交叉引用

本申请基于申请号为 201810500185.0、申请日为 2018 年 5 月 23 日的中国专利申请提出，并要求该中国专利申请的优先权，该中国专利申请的全部内容在此以全文引入的方式引入本申请。

技术领域

本申请实施例涉及深度学习领域，涉及但不限于图像识别方法及装置、计算机设备和存储介质。

背景技术

卷积神经网络（Convolutional Neural Networks, CNN）已经成为计算机视觉领域的主流方法。对于图像理解任务例如图像分类，物体检测和语义分割，现有的主流卷积神经网络例如是牛津大学计算机视觉组（Visual Geometry Group, VGG），残差网络（Residual Network, ResNet），密集连接卷积网络（Dense Convolutional Network, DenseNet）等均采用批归一化（Batch Normalization, BN）来加速训练。然而，这些卷积神经网络对于图像外观的变化有较差的鲁棒性。例如，当图像的颜色、对比度、风格、场景等发生变化时，这些卷积神经网络的性能会显著下降。

此外，在图像外观转换领域，实例归一化（Instance Normalization, IN）被用于卷积神经网络中来提升其对不同外观图像的适应能力。然而，实例归一化并未被成功用于图像理解任务中，而且在现有技术中无论是在 CNN 中采用 BN 还是 IN 都没有很好的提升卷积神经网络的性能。

发明内容

本申请实施例提供一种图像识别方法及装置、计算机设备和存储介质。

本申请实施例的技术方案是这样实现的：

本申请实施例提供一种图像识别方法，所述方法包括：获取待识别图像；将所述待识别图像输入，经过训练得到的神经网络模型，得到所述待识别图像的识别结果，其中，所述神经网络模型是对所述神经网络中卷积层输出的特征图进行 IN 和 BN 处理后得到的；输出所述待识别图像的识别结果。

本申请实施例提供一种图像识别装置，所述装置包括：第一获取模块、第一处理模块和第一输出模块，其中：第一获取模块，配置为获取待识别图像；第一处理模块，配置为将所述待识别图像输入，经过训练得到的神经网络模型，得到所述待识别图像的识别结果，其中，所述神经网络模型是对神经网络进行 IN 和 BN 处理得到的。第一输出模块，配置为输出所述待识别图像的识别结果。

本申请实施例提供一种计算机程序产品，所述计算机程序产品包括计算机可执行指

令, 该计算机可执行指令被执行后, 能够实现本申请实施例提供的图像识别方法中的步骤。

本申请实施例提供一种计算机存储介质, 存储有计算机可执行指令, 该计算机可执行指令被执行后, 能够实现本申请实施例提供的图像识别方法中的步骤。

- 5 本申请实施例提供一种计算机设备, 包括存储器和处理器, 所述存储器上存储有计算机可执行指令, 所述处理器运行所述存储器上的计算机可执行指令时可实现本申请实施例提供的图像识别方法中的步骤。

10 本申请实施例一种计算程序产品, 其中, 所述计算程序产品包括计算机可执行指令, 该计算机可执行指令被执行后, 能够实现本申请实施例提供的图像识别方法中的步骤。

本申请实施例中, 通过将 IN 和 BN 结合后应用在神经网络中, 有效的提高了图像识别的正确率。

附图说明

- 图 1A 为本申请实施例网络架构的组成结构示意图
15 图 1B 为本申请实施例图像识别方法的实现流程示意图;
图 1C 为本申请实施例实现图像识别方法的网络架构图;
图 1D 为本申请实施例又一实现图像识别方法的网络架构图;
图 2 为本申请实施例图像识别方法的又一实现流程示意图;
图 3 为本申请实施例基于残差网络的组成结构图;
20 图 4 为本申请实施例又一基于残差网络的组成结构图;
图 5 为本申请实施例图像识别装置的组成结构示意图;
图 6 为本申请实施例计算机设备的组成结构示意图。

具体实施方式

- 本实施例先提供一种网络架构, 图 1A 为本申请实施例网络架构的组成结构示意图,
25 如图 1A 所示, 该网络架构包括两个或多个计算机设备 11 至 1N 和服务器 31, 其中计算机设备 11 至 1N 与服务器 31 之间通过网络 21 进行交互。计算机设备在实现的过程中可以为各种类型的具有信息处理能力的计算设备, 例如所述计算机设备可以包括手机、平板电脑、台式机、个人数字助理、导航仪、数字电话、电视机等。本实施例提出一种图像识别方法, 能够有效解决输出图像的结构信息与输入图像相比发生变化的问题, 该方法
30 应用于计算机设备, 该方法所实现的功能可以通过计算机设备中的处理器调用程序代码来实现, 当然程序代码可以保存在计算机存储介质中, 可见, 该计算机设备至少包括处理器和存储介质。

- 为了更好的理解本实施例, 这里对神经网络相关的专业术语加以解释, 通道 (Channel), 该词语有两种不同的含义, 第一种是对于样本图像 (图像作为训练样本),
35 通道是指颜色通道, 下面将用颜色通道来表示样本图像的通道; 第二种是输出空间的维

数，例如卷积操作中输出通道的个数，或者说每个卷积层中卷积核的数量。

颜色通道，把图像分解成一个或多个颜色成分或颜色分量。单颜色通道，一个像素点只需一个数值表示灰度，0 为黑色。三颜色通道，如果采用红绿蓝（Red Green Blue, RGB）色彩模式，把图像分为红绿蓝三个颜色通道，可以表示彩色，全 0 表示黑色。四颜色通道，在 RGB 色彩模式的基础上加上 alpha 通道，表示透明度，alpha=0 表示全透明。卷积神经网络，是一种多层的监督学习神经网络，隐含层的卷积层和池采样层是实现卷积神经网络特征提取功能的核心模块。卷积神经网络的低隐层是由卷积层和最大池采样层交替组成，高层是全连接层对应传统多层感知器的隐含层和逻辑回归分类器。第一个全连接层的输入是由卷积层和子采样层进行特征提取得到的特征图像。最后一层输出层是一个分类器，可以采用逻辑回归，Softmax 回归甚至是支持向量机对输入图像进行分类。CNN 中每一层的由多个图（map）组成，每个 map 由多个神经单元组成，同一个 map 的所有神经单元共用一个卷积核（即权重），卷积核往往代表一个特征，比如某个卷积核代表一段弧，那么把这个卷积核在整个图像上卷积一遍，卷积值较大的区域就很有可能是一段弧。CNN 一般采用卷积层与采样层交替设置，即一层卷积层接一层采样层，采样层后接一层卷积；当然也可以多个卷积层接一个采样层，这样卷积层提取出特征，再进行组合形成更抽象的特征，最后形成对图像对象的描述特征，CNN 后面还可以跟全连接层。ReLU 函数，其公式即为 $\text{ReLU}(x)=\max(0, x)$ ，从 ReLU 函数的图形可以看出 ReLU 与其他激活函数例如 sigmoid 函数相比，主要变化有三点：①单侧抑制；②相对宽阔的激活边界；③稀疏激活性。VGG 模型，VGG 模型结构简单有效，前几层仅使用 3×3 卷积核来增加网络深度，通过最大池化（Max Pooling）依次减少每层的神经元数量，最后三层分别是 2 个有 4096 个神经元的全连接层和一个 softmax 层。“16”和“19”表示网络中的需要更新权重（即 weight，要学习的参数）的卷积层和全连接层的层数，VGG16 模型和 VGG19 模型的权重都由 ImageNet 训练而来。

本实施例提供一种图像识别方法，图 1B 为本申请实施例图像识别方法的实现流程示意图，如图 1B 所示，所述方法包括以下步骤：

步骤 S101，获取待识别图像。这里，所述步骤 S101 可以由计算机设备实现的，进一步地，所述计算机设备可以是智能终端，例如可以是移动电话（比如，手机）、平板电脑、笔记本电脑等具有无线通信能力的移动终端设备，还可以是台式计算机等不便移动的智能终端设备。所述计算机设备用于进行图像识别或处理。

在本实施例中，所述待处理图像可以是外观复杂的图像，还可以是外观简单的图像。

步骤 S102，将所述待识别图像输入，经过训练得到的神经网络模型，得到所述待识别图像的识别结果。这里，所述步骤 S102 可以由计算机设备实现的。所述神经网络模型是对所述神经网络中卷积层输出的特征图进行 IN 和 BN 处理后得到的。在本实施例中，对所述神经网络中卷积层输出的特征图进行 IN 和 BN 处理，得到所述神经网络模型，即 IBN-Net。所述识别结果可以是图像的类别、图像的名称等。所述神经网络

可以是卷积神经网络, 比如, ResNet50、VGG 和 DenseNet 等。由于所述神经网络模型中同时采用了 IN 和 BN, 这样 IN 引入了外观不变性并改进了泛化, 而 BN 保留了内容信息的区分特征。所以, 在采用 IN 和 BN 相结合的神经网络模型中, 不仅能够提升神经网络的泛化能力, 还能够提高该神经网络识别图像的正确率。

5 步骤 S103, 输出所述待识别图像的识别结果。这里, 所述步骤 S103 可以由计算机设备实现的。在实际实现过程中, 所述计算机设备输出所述待识别图像的分析结果可以是在自身显示屏上输出所述待识别图像, 还可以是所述计算机设备将所述分析结果输出至其他设备, 也就是发送给其他设备, 例如所述其他设备可以是用户的智能终端上。

10 在本申请实施例提供的一种图像识别方法中, 通过将 IN 和 BN 结合后应用在神经网络中, 然后在经过 IN 和 BN 处理后的神经网络模型中对待识别的图像进行分析, 有效的提高了图像识别的正确率。

在实现的过程中, 经过训练的神经网络模型可以在计算机设备的本地, 也可以是在服务器端。

15 当经过训练的神经网络模型在计算机设备本地时, 可以是计算机设备安装客户端的时候, 即安装了经过训练的神经网络模型, 这样, 参见图 1C 所示, 计算机设备通过步骤 S101 获取待识别图像, 然后通过步骤 S102 获得待识别图像的识别结果, 最后通过步骤 S103 输出识别结果。从以上过程可以看出, 计算机设备在安装完客户端之后, 上述的步骤 S101 至步骤 S103 都在计算机设备本地执行, 最后, 计算机设备将得到识别结果输出给用户。

20 在一些实施例中, 经过训练的神经网络模型也可以位于服务器端, 参见图 1D 所示, 这样计算机设备将输入图像发送给服务器, 这样服务器接收计算机设备发送的输入图像, 这样服务器实现了步骤 S101, 换句话说, 如果上述的方法是在服务器端实现, 那么步骤 S101, 包括: 服务器接收计算机设备发送的输入图像, 即服务器确定待识别的图像, 然后服务器通过步骤 S102 获得所述待识别图像的输出结果, 最后通过步骤 S103 获得输出
25 的识别结果; 从以上过程可以看出, 上述的步骤 S101 至步骤 S103 都在服务器端执行, 最后服务器还可以将识别结果发送给计算机设备, 这样计算机设备接收到识别结果后, 输出识别结果给用户。本实施例中, 计算机设备在安装完客户端之后, 用户上传用户的待识别图像, 以及接收服务器发送的待识别图像, 并将识别结果输出给用户。

30 本实施例提供一种图像识别方法, 图 2 为本申请实施例图像识别方法的又一实现流程示意图, 如图 2 所示, 所述方法包括以下步骤:

步骤 S201, 从所述神经网络的卷积层中确定第一卷积层集合和所述第二卷积层集合。这里, 所述第一卷积层集合和所述第二卷积层集合组成的集合为所述神经网络的所有卷积层中的全部或部分。所述第一卷积层集合和所述第二卷积层集合组成的集合为所述神经网络的所有卷积层中的全部, 可以理解为, 该神经网络中所有的卷积层都经过了
35 IN 和/或 BN 处理。第一卷积层集合和所述第二卷积层集合组成的集合为所述神经网络的所有卷积层中的部分, 可以理解为, 该神经网络中的部分卷积层没有进行 IN 或 IN 结

合 BN 的处理。

步骤 S202, 从所述第一卷积层集合中的每一所述卷积层输出的特征图对应的通道中确定第一通道集合。这里, 所述第一通道集合为所述第一卷积层集合中的每一所述卷积层输出的特征图对应的所有通道的全部或部分。所述第一卷积层集合中不包括所述神经网络中的最后一层卷积层, 即对神经网络的最后一层(深层)不进行 IN 处理, 这样就不减少内容在深层特征中的区分度, 还能够降低由于图像外观变换引起的特征变化, 如此, 提升该神经网络模型图像识别的正确率。在实际使用过程中, 一般选择特征图对应的所有通道的一半进行 IN 处理, 另一半进行 BN 处理, 显然, 进行 IN 处理的通道的比例是可以调整的。在本实施例中, 为使神经网络的恒等路径较为清洁, 因此, 不在神经网络的恒等路径进行 IN 处理。

步骤 S203, 对所述第一通道集合进行 IN 处理。这里, 当第一通道集合为所述第一卷积层集合中的每一所述卷积层输出的特征图对应的所有通道的全部时, 对第一卷积层集合中的每一所述卷积层输出的特征图对应的所有通道进行 IN 处理; 当第一通道集合为所述第一卷积层集合中的每一所述卷积层输出的特征图对应的所有通道的部分时, 对所述部分通道 IN 处理, 然后对剩余的通道进行 BN 处理, 或者不做任何处理。

步骤 S204, 从第二卷积层集合中的每一卷积层输出的特征图对应的通道中确定第二通道集合。这里, 所述第二通道集合为所述第二卷积层集合中的每一所述卷积层输出的特征图对应的所有通道的全部或部分。

步骤 S205, 对所述第二通道集合进行 BN 处理。这里, 当第二通道集合为所述第二卷积层集合中的每一所述卷积层输出的特征图对应的所有通道的全部时, 对第二卷积层集合中的每一所述卷积层输出的特征图对应的所有通道进行 BN 处理; 当第二通道集合为所述第二卷积层集合中的每一所述卷积层输出的特征图对应的所有通道的部分时, 对所述部分通道 BN 处理, 然后对剩余的通道进行 IN 处理。

在一些实施例中, 所述第一卷积层集合与所述第二卷积层集合之间的关系包括以下三种情况: 情况一: 所述第一卷积层集合与所述第二卷积层集合不具有交集, 即第一卷积集合与第二卷积集合分别进行不同的归一化处理, 即对所述第一卷积层集合中的每一卷积层输出的特征图进行 IN 处理; 对所述第二卷积层集合中的每一卷积层输出的特征图进行 BN 处理, 如图 4(b) 所示, 只对求和操作之后得到的输出结果的一部分进行 IN 处理, 而对其余卷积层输出的特征图进行 BN 处理。情况二: 所述第一卷积层集合与所述第二卷积层集合具有交集, 即对第一卷积层集合进行了 IN、IN 结合 BN 的处理; 对第二卷积层集合进行了 BN、IN 结合 BN 的处理; 即, 如步骤 202 和步骤 203 所述的, 当第一通道为第一卷积层集合中的每一所述卷积层输出的特征图对应的所有通道的部分时, 对所述部分进行 IN 处理, 对其余部分进行 BN 处理。或者, 步骤 204 和步骤 205 所述的, 当第二通道为第二卷积层集合中的每一所述卷积层输出的特征图对应的所有通道的部分时, 对所述部分进行 BN 处理, 对其余部分进行 IN 处理; 如图 4(d) 所示, 对卷积层输出的特征图进行 BN、IN 结合 BN 的处理。情况三: 所述第二卷积层集合为

所述第一卷积层集合的子集合,当所述第二卷积层集合为所述第一卷积层集合的真子集时,对第一卷积集合进行 IN、IN 结合 BN 的处理,对第二卷积集合进行 IN 结合 BN 的处理。当所述第二卷积层集合和所述第一卷积层集合相同时,即是对第二卷积层集合和第一卷积层集合均进行 IN 结合 BN 的处理。即,第一通道第一卷积层集合中的每一所述卷积层输出的特征图对应的所有通道的部分,对所述部分进行 IN 处理,对其余部分进行 BN 处理,或者不处理(即第一卷积层集合层中包括了两种处理方式:IN、IN 结合 BN)。

在一些实施例中,所述方法还包括:对所述神经网络的两个块对应的特征图进行求和,得到输出结果,对所述输出结果进行 IN 处理;如图 3(c)所示,首先,对图 3(c)所呈现的残差块经过三层卷积得到的特征图和上一个残差块经过多层卷积得到的特征图求和,得到求和结果(即输出结果);然后对求和结果进行 IN 处理。由于在本实施例中,外观信息既可以保留在残留路径中,也可以保存在恒等路径中,为了更加有效的提升神经网络的泛化能力,在残差路径(即图 3 中的路径 1)和恒等路径(即图 3 中的路径 2)汇聚后,再进行 IN 处理,从而有效的提高了图像识别的正确率。

与相关技术中单独采用 IN 和 BN 的 CNN 结构不同,本实施例提供的图像识别方法中,通过深入研究 IN 和 BN 的学习能力来实现将 IN 和 BN 结合使用在同一个 CNN 中。例如,在许多先进的深度架构采用 BN 作为提高其高级视觉任务的学习能力的关键组件,而 IN 通常与 CNN 结合以消除低级视觉任务的图像的方差,如,图像风格变换等。但 IN 和 BN 的学习特征的不同特点及其组合的影响尚未在相关技术中得出结论。相反,本申请实施例提供的 IBN-Net 表明,以适当的方式组合 IN 和 BN 会提高 CNN 的学习和泛化能力。在 IBN-Net 中将 IN 和 BN 组合的特征保留在 CNN 的浅层,BN 特征保留在 CNN 的高层,以符合网络不同深度下的统计特征。由于在 CNN 中,图像外观相关的信息(比如,颜色,对比度,风格等)主要存在于浅层的特征中,而图像中物体类别相关的信息主要存在于深层的特征中,并且在浅层的特征中仍然存在。基于此,按照两条规则向 CNN 引入 IN 层:一是,为了减少在浅层由外观引起的特征变化,同时不干扰深层内容区分,因此只在 CNN 中低层的那一半添加 IN 层。二是,为了在浅层不损失图像内容信息,将原始的 BN 层中一半的特征替换为 IN 层,将剩下另一半仍然采用 BN;基于此,本实施例提供的在卷积神经网络中采用 BN 和 IN 的结合进行处理,形成 IBN-Net。

在卷积神经网络中采用本申请实施例提供的 IBN-Net,具有以下优点:

第一,本申请实施例提出的 IBN-Net,提高了卷积神经网络的性能和泛化能力。例如,IBN-Net50 和 ResNet50 有相似数量的参数和计算成本时,IBN-Net50 在图像网络数据库(ImageNet)的原始验证集上的精度分别达到第五和第一的正确率为 93.7%和 77.4%,分别比 ResNet50 的验证率高出 0.8%和 1.7%。IBN-Net50 在 ImageNet 的风格转换后的新验证集上的精度分别达到第五和第一的正确率为 72.9%和 48.9%,分别比 ResNet50 的验证率高出 2.2%和 2.9%。

第二,在本申请实施例提出的 IBN-Net 中,IN 提供视觉和外观不变性,同时 BN 能

够加速训练并保留区分性的特征。这个特征有助于设计 IBN-Net 的体系结构, 其中将 IN 设置在浅层中以消除外观变化, 而为了保持区分性, 应该减小 IN 在深层中的强度。IBN-Net 的模块可以用来重新开发许多最近正在研究的深度架构, 以提高该深度结构的学习和泛化能力, 但保持该深度结构的计算成本不变。例如, 通过在 VGG16, ResNet101, ResNeXt101 和挤压-激励网络(Squeeze-and-Excitation Residual Network, SE-ResNet101) 中使用 IBN-Net, 它们在 ImageNet 验证集上精度达到第一的正确率分别优于各自的原始版本的 0.8 %, 1.1 %, 0.6 % 和 0.7 %。

第三, IBN-Net 显著提高了跨域的性能。例如, 基于交通场景的真实数据集和虚拟数据集属于两种图像域, 其中, 真实数据集可以是城市景观(Cityscapes), 虚拟数据集可以是侠盗猎车手(Cityscapes Grand Theft Auto, GTA); 当在 GTA 上训练, 在 Cityscapes 上测试的情况下, IBN-Net 集成的 ResNet50 性能提高了 7.6 %。当使用 GTA 预训练的模型在 Cityscapes 上微调时, 还显著降低了需要的样本量。例如, 使用 Cityscapes 只有 30 % 的训练数据进行微调时, 本实施例提供的 IBN-Net 模型的分割准确率达到 65.5 %, 而使用所有训练数据来调整的 ResNet50 则只有 63.8 %。

为了更好的理解本实施例, 这里提供与 IBN-Net 相关的四个方面: CNN 的不变性, CNN 网络体系结构, 域适配方法和场景理解的方法。CNN 中的不变性: 在相关技术中提出的模块通常是为了提高 CNN 的建模能力, 或者减少过拟合来增强其在单个域上的泛化能力。这些方法通常通过在 CNN 的体系结构中引入特定的不变性来达到上述目的。例如, 最大池化和可变形卷积将空间不变性引入到 CNN 中, 从而增加了卷积神经网络对空间变化(例如仿射, 失真和视角变换) 的鲁棒性。丢弃(dropout) 层和 BN 在训练中的作用可以被视为正则化来减少样本噪音的影响。对于图像外观, 简单的外观变化例如颜色和亮度偏移可以通过用平均值和标准偏差对图像的每个 RGB 通道进行归一化来消除。对于更复杂的外观变化, 如风格变换, 最近的研究发现这些信息可以被编码在特征图的均值和方差中。因此, 实例归一化层就显示出了消除这种外观差异的潜力。CNN 网络体系结构: 自从 CNN 显示出比传统方法更强的性能, CNN 的架构已经经历了许多发展。其中, 使用最广泛的是 ResNet, ResNet 使用捷径来缓解非常深的网络的训练难度。此后, ResNet 的多种变体被相继提出。与 ResNet 相比, ResNeXt 通过增加 ResNet 的“基数”来提高模型性能这是通过使用组卷积来实现的。在实践中, 增加基数会增加深度学习框架的运行时间。此外, 挤压-激励网络(Squeeze-and-Excitation Network, SENet) 将通道维度的注意力机制引入 ResNet。与 ResNet 相比, SENet 在 ImageNet 上实现了更好的性能, 但也增加了网络参数和计算量。最近提出的密集连接网络(Densely Connected Networks, DenseNet) 使用堆叠操作来取代 ResNet 的捷径。DenseNet 被证明比 ResNet 更有效率。但是, 上述 CNN 体系结构有两个限制。首先, 有限的基本模块阻止 CNN 获得更具吸引力的属性。例如, 所有这些体系结构都是由卷积, BN、激励层(Rectified Linear Unit, ReLU) 和共享池组成的, 不同 CNN 唯一的区别是这些模块是如何组织的。然而, 这些层的组成天然地易受外观变化的影响。其次, 这些模型的设计目标是在一个单一领

域的单一任务中实现强大的性能,但是这些模型推广到新的领域的的能力仍然有限。在图像风格变换的领域,一些方法采用 IN 来帮助消除图像对比度。但是,图像外观的不变性还没有成功地引入到 CNN 中,特别是在图像分类或语义分割等高级任务中。这是因为 IN 丢失了特征中有用的内容信息,影响了模型的性能。域适配方法:缓解由不同域之间的偏差引起的性能衰落是一个重要问题。一种自然的方法是使用迁移学习,例如,在目标域上微调模型。但是,这需要目标域有人工标注结果,并且当网络模型应用于源域时,微调后的模型的性能会下降。有许多领域适应方法使用目标域的统计数据来促进自适应。一般需要通过精心设计的损失函数,如最大平均差异(Maximum Mean Discrepancy, MMD),相关性对齐(Correlation Alignment, CORAL)和对抗损失(Adversarial Loss, AL),通过减少两个域的偏差引起的特征差异来缓解性能衰落的问题。迁移学习和领域适应有两个主要限制:首先,在实际应用中很难获得目标域的统计信息。收集涵盖目标域中所有可能场景的数据也非常困难。其次,大多数最先进的方法对源域和目标域采用不同的模型以提高性能。但理想的情况是,一个模型可以适应所有领域。

针对这个问题的另一个范式是域推广,其目标是从许多相关的源域获取知识并将其应用于训练期间统计数据未知的新目标域。相关技术中通常设计算法来捕捉不同域中共有的因素。但是,对于实际应用,通常很难收集多个相关的源域的数据,并且最终的性能高度依赖于所收集的一系列源域。在这项工作中,本申请实施例通过设计新的具有外观不变性的 CNN 体系结构 IBN-Net 增加了模型性能和泛化能力。与域适应和域推广不同,本申请不需要目标域数据或相关源域。本实施例对于无法得到目标域数据的情况非常有用,这是相关技术不能达到的。

本实施例中,对于基于 BN 的 CNN,图像外观相关的信息(颜色,对比度,风格等)主要存在于浅层的特征中,而图像中物体类别相关的信息主要存在于深层的特征中,并且在浅层特征中仍然存在。因此,本实施例按照两条规则引入 IN。首先,为了不减少深层特征对图片内容信息的区分度,不会在 CNN 的最后一层或几层添加 IN。其次,为了在浅层中依然保存内容信息,我们保留了在浅层中对一部分特征进行 BN 处理。

本实施例将 IBN-Net 应用于 ResNet 中,图 3 为本申请实施例基于残差网络的组成结构图。ResNet 主要由 4 组残差块组成,如图 3(a)为原始的 ResNet 中一个残差块的结构图,图 3(b)和图 3(c)分别为对 ResNet 中不同的卷积层输出的特征图进行 IN 结合 BN 处理的结构图;其中,在图 3(a)中,路径 1 为残差路径,路径 2 为恒等路径;30 中(x, 256d)中 x 表示输入的特征,256d 表示输入的特征是 256 个通道,31 表示卷积核为 1*1,64 个通道的卷积层,32、34 和 36 表示激励层(ReLU);33 表示卷积核为 3*3,64 个通道的卷积层;35 表示卷积核为 1*1,256 个通道的卷积层;311 表示对 64 个通道进行批归一化(BN);312 对 64 个通道进行批归一化(BN);313 对 256 个通道进行批归一化(BN);在图 3(b)中,321 表示对卷积层输出的特征图对应的通道的一半(即 32 个通道)进行 IN 处理,另一半(即另外 32 个通道)进行 BN 处理;在图 3

(c)中, 331 表示对进行求和操作之后的结果, 进行 IN 处理, 所述求和操作是在 ResNet 中图 3(c) 所呈现的残差块经过三层卷积得到的特征图和上一个残差块经过多层卷积得到的特征图 (即输入的特征 x) 求和。

对于一个残差块, 为了利用 IN 的泛化潜力, 在第一个卷积层之后得到的特征图中, 将 BN 用于该特征图的一半的通道, 将 IN 用于其他通道, 如图 3(b) 所示, 有三个理由这样做: 首先, 干净的恒等路径对于优化 ResNet 是至关重要的, 因此我们将 IN 添加到残差路径而不是恒等路径。其次, 在残差学习函数 $y = F(x, \{W_i\}) + x$ 中残差函数 $F(x, \{W_i\})$ 在学习过程中是与恒等路径中的 x 对齐。因此, IN 应用于第一个归一化层 (即第一个卷积层输出的特征图) 而不是最后一个, 以减少发生 $F(x, \{W_i\})$ 与恒等路径中的 x 错位的概率。第三, 在卷积层输出的特征图中, 将一半通道进行 BN 处理, 一半通道进行 IN 处理, 满足了将图像内容信息保存在浅层的要求。

这种设计是对模型性能的追求。一方面, IN 使模型能够学习外观不变特征, 从而能够更好地利用一个数据集内具有高外观多样性的图像。另一方面, 以适度的方式添加 IN, 以便与内容相关的信息可以得到很好的保留。本实施例中将这个模型表示为 IBN-Net-a。此外, 本申请还提出另一种追求最大泛化能力的网络 IBN-Net-b。由于外观信息既可以保留在残差路径中, 也可以保存在恒等路径中, 因此, 为了保证神经网络的泛化能力, 在加法操作之后立即添加 IN, 如图 3(c) 所示。为了不使 ResNet 的性能出现劣化, 本实施例只在第一卷积层和前两个卷积组之后添加三个 IN 层。

如表 1.1 为原始的 ResNet50 及其对应的两种 IBN-Net 的整体网络结构, 从表 1.1 可以看出, 相比较原始的 ResNet50, 在 IBN-Net50-a 中前三组模块 (conv2_x-conv4_x) 被换成了图 3(b) 中 IBN-a 的结构, 而在 IBN-Net50-b 中前两组模块 (conv2_x-conv3_x) 的最后一个残差块被换成了图 3(c) 中 IBN-b 的结构, 并且第一个卷积层 conv1 后的 BN 被替换成了 IN。在表 1A 中, conv2_x 表示第一组残差块。

表 1.1 原始的 ResNet50 及其对应的两种 IBN-Net 的整体网络结构

不同的层	配置	归一化类型		
		ResNet50	IBN-Net50-a	IBN-Net50-b
卷积层 1	$7 \times 7, 64$, 步长为 2	BN	BN	IN
池化层	最大池化 3×3 , 步长为 2	-	-	-
conv2_x	$\begin{bmatrix} 1 \times 1, 64 \\ 3 \times 3, 64 \\ 1 \times 1, 256 \end{bmatrix} \times 3$	BN	IBN-a	第一个和第二个残差块进行 BN 处理, 第三个残差块基于模型 IBN-b 进行处理
conv3_x	$\begin{bmatrix} 1 \times 1, 128 \\ 3 \times 3, 128 \\ 1 \times 1, 512 \end{bmatrix} \times 4$	BN	IBN-a	前三个残差块进行 BN 处理, 第四个残差块基于模型 IBN-b 进行处理

Conv4_x	$\begin{bmatrix} 1 \times 1,256 \\ 3 \times 3,256 \\ 1 \times 1,1024 \end{bmatrix} \times 6$	BN	IBN-a	BN
Conv5_x	$\begin{bmatrix} 1 \times 1,512 \\ 3 \times 3,512 \\ 1 \times 1,2048 \end{bmatrix} \times 3$	BN	BN	BN
分类器	平均池化, 1000 维全连接层	-	-	-

上述两种 IBN-Net 不是唯一的在 CNN 中使用方法 IN 和 BN 的方法。在本实施例提供了一些有趣的变体, 如图 4 所示。图 4 为本申请实施例又一基于残差网络的组成结构图, 图 4 (a)、4 (b)、4 (c) 和 4 (d) 为残差神经网络中同一个块的结构图, 如图 4 (a)、4 (b)、4 (c) 和 4 (d) 分别为, 残差神经网络的不同卷积层输出的特征图 (即归一化层) 进行 IN 结合 BN 处理, 其中, 在图 4 (a) 中, 路径 3 为残差路径, 路径 4 为恒等路径; 40 中 (x, 256d) 中 x 表示输入的特征, 256d 表示输入的特征是 256 个通道 (可以理解为 256 张图像), 41 表示卷积核为 1*1, 64 个通道的卷积层, 42、44 和 46 表示激励层 (ReLU); 43 表示卷积核为 3*3, 64 个通道的卷积层; 45 表示卷积核为 1*1, 256 个通道的卷积层; 411 表示对 64 个通道进行批归一化 (BN); 412 对 256 个通道进行批归一化 (BN); 431 表示对第一个卷积层输出的特征图分别进行 IN 和 BN 处理; 47 表示将两种归一化方式处理之后的结果进行堆叠操作, 然后, 输出到下一层即激励层; 在图 4 (b) 中, 413 表示对 64 个通道进行批归一化 (BN); 431 表示表示对进行求和操作之后的结果的一半 (即对 256 个通道中的 128 个通道进行 IN 处理), 进行 IN 处理, 所述求和操作是在卷积神经网络中图 4 (b) 所呈现的神经网络的块经过三层卷积得到的特征图和上一个该神经网络的块经过多层卷积得到的特征图求和。在图 4 (c) 中, 441 表示对第一个卷积层输出的特征图对应的通道的一半 (即 32 个通道) 进行 IN 处理, 另一半 (即另外 32 个通道) 进行 BN 处理; 在图 4 (d) 中, 442 表示对第二个卷积层输出的特征图对应的通道的一半 (即 32 个通道) 进行 IN 处理, 另一半 (即另外 32 个通道) 进行 BN 处理。

20

表 1.2 原始 VGG16 及其 IBN-Net 版本的整体结构

不同的层	归一化类型	
	VGG16	IBN-VGG16-a
Conv3-64	BN-64	IN-32, BN-32
Conv3-64	BN-64	BN-64
最大池化	-	-
Conv3-128	BN-128	IN-64, BN-64
Conv3-128	BN-128	BN-128
最大池化	-	-
Conv3-256	BN-256	IN-128, BN-128
Conv3-256	BN-256	BN-256
Conv3-256	BN-256	BN-256
最大池化	-	-

Conv3-512	BN-512	IN-256,BN-256
Conv3-512	BN-512	BN-512
Conv3-512	BN-512	BN-512
最大池化	-	-
Conv3-512	BN-512	BN-512
Conv3-512	BN-512	BN-512
Conv3-512	BN-512	BN-512
最大池化	-	-
FC × 3	-	-

从表 1.2 可以看出，在本实施例通过的 IBN-Net 中，有多种实施方式，其中，部分参数可以在满足以下两个条件变化：一是不在网络的最后一层（根据对训练结果的需要，也可以将模型设置为不在最后两层或 3 层等不进行 IN 处理）使用 IN 进行处理；二是，在其余的层（除最后一层）将 IN 和 BN 结合使用或交替使用。例如：观察图 4 给出的四个方案即图 4（a）、4（b）、4（c）和 4（d），可以看出，IN 层的位置、利用 IN 进行处理的通道的数量均可以调整；另外，在同一卷积神经网络的不同模块中，图 4 给出的四种实施方案可以交替使用。

表 1.3 使用 ImageNet 验证集的图像外观变化对应的错误率

外观变换	ResNet50	IBN-Net50-a	IBN-Net50-b
	top1/top5 err	top1/top5 err	top1/top5 err
原图	24.27/7.08	22.60/6.28	23.58/6.87
RGB+50	28.22/9.64 (3.94/2.56)	25.61/7.96 (3.01/1.68)	23.83/6.99 (0.25/0.12)
R+50	27.53/8.78 (3.26/1.70)	25.03/7.59 (2.43/1.31)	25.09/7.36 (1.51/0.49)
contrast*1.5	40.01/19.08 (15.74/12.00)	35.73/16.24 (13.13/9.96)	23.64/6.87 (0.06/0.87)
Monet	54.51/29.32 (30.24/22.24)	51.62/27.15 (29.02/20.87)	50.42/25.41 (26.84/18.54)

在图 4（a）中，对第一个卷积层输出的特征图分别进行 IN 和 BN 处理，然后，将两种归一化方式处理之后的结果堆叠起来并输出到下一层即激励，保持 IN 和 BN 的高范化性能和高区分度的特征，但是这会引入更多参数。由于保留两种特征的想法也适用于 IBN-b，从而产生了图 4（b）。另外，还可以将图 4（b）和图 4（b）呈现的方案结合起来，如图 4（c）和 4（d）所示。关于这些变体的讨论将在下文的实验部分中给出。表 1.3 为采用 ImageNet 数据库作为待识别图像，输入到 IBN-Net 神经网络模型中，得到的基于 top1 和 top5 的错误率；其中，top1 的意思是预测出来最大概率的分类是正确的概率。top5 的意思是预测出来最大概率的 5 个分类里有正确的概率，top1/top5 err，分别表示基于 top1 和基于 top5 的错误率。RGB+50 表示，在原图的基础上，将图像的 R、G 和 B 三个通道各加 50；R+50 表示红色通道加 50，即图像更红；contrast*1.5 表示对比度乘以 1.5；Monet 表示使用 CycleGAN（图像风格转换工具）将图像转换成莫奈风格的

图像。因此，从表 1.3 可以看出，无论是基于哪种外观变化，采用模型 IBN-Net50-a（即图 3（b））得到的基于 top1 的错误率和基于 top5 的错误率分别低于未采用 IBN-Net 的原始的 ResNet50 基于 top1 的错误率和基于 top5 的错误率；而且采用模型 IBN-Net50-b（即图 3（c））得到的基于 top1 的错误率和基于 top5 的错误率分别低于未采用 IBN-Net 的原始的 ResNet50 基于 top1 的错误率和基于 top5 的错误率。因此，采用 IBN-Net 模块得到的神经网络的性能优于未采用 IBN-Net 模块（即原始 ResNet，如图 3（a）所示）得到的神经网络的性能。

表 2 在其他 CNN 模型中采用 IBN-Net，通过 ImageNet 验证集的错误率

模型	原始模型	re-implementation	IBN-Net-a 模型
	top1/top5 err.	top1/top5 err.	top1/top5 err.
VGG16	26.6/8.6	25.94/8.25	25.10/7.69(0.84/0.56)
ResNet50	24.7/7.8	24.27/7.08	22.60/6.32(1.67/0.75)
ResNet101	23.6/7.1	22.44/6.20	21.37/5.57(1.07/0.63)
ResNeXt101	21.2/5.6	21.31/5.70	20.75/5.32(0.56/0.38)
SE-ResNet101	22.38/6.07	21.68/5.88	20.99/5.49(0.69/0.39)

为了表明比传统 CNN 更强大的 IBN-Net 模型性能，本实施例将 IBN-Net 的性能与最初流行的原始 ImageNet 验证集中的 CNN 架构进行比较。如表 2 所示，IBN-Net 实现了对这些 CNN 一致的改进，表明了更强大的模型性能。具体而言，IBN-ResNet101 优于 ResNeXt101 和 SE-ResNet101，后两者需要更多的时间消耗或引入额外的参数。但是在本实施例中提供的 IBN-Net 模型中，不会带来额外的参数，而只会在测试阶段添加很少量的计算。实验结果显示，剔除特征中的一些均值和方差统计量有助于模型从具有高度外观多样性的图像中学习。

表 3 IBN-Net 的变体基于 ImageNet 验证集和莫奈风格图像的错误率

模型	原始模型	Monet
	top1/top5 err.	top1/top5 err.
ResNet50	24.26/7.08	54.51/29.32(30.24/22.24)
IBN-Net50-a	22.60/6.03	51.62/27.15(29.02/20.83)
IBN-Net50-b	23.58/6.87	50.42/25.41(26.84/18.54)
IBN-Net50-c	22.78/6.32	51.83/27.09(29.05/20.77)
IBN-Net50-d	22.86/6.48	50.80/26.16(27.94/19.68)
IBN-Net50-a&d	22.89/6.48	51.27/26.64(28.38/20.16)
IBN-Net50-a*2	22.81/6.46	51.95/26.98(29.14/20.52)

本实施例进一步研究 IBN-Net 的其他变体。表 3 显示了方法部分描述的 IBN-Net 变体的结果。在本实施例提供的所有的 IBN-Net 变体都比原始的 ResNet50 显示出更好的性能，并且在外观变换下性能下降更少。具体而言，IBN-Net-c 实现与 IBN-Net-a 类似的性能，提供了另一种特征组合的方法。IBN-Net-d 的性能和泛化能力位于 IBN-Net-a 和 IBN-Net-b 之间，这表明在卷积层输出的特征图的部分通道中保留一些 BN 特性有助

于提高性能，但同时会失去一定的泛化能力。IBN-Net-a 和 IBN-Net-b 的组合基本等同于 IBN-Net-d，这表明 IN 对 ResNet 主路径的影响将占主导地位。最后，向 IBN-Net-a 添加额外的 IBN 层不会带来好处，适量的添加 IN 层就足够了。

表 4 在 IBN-Net50-a 中的残差组中增加不同数量的 IN 层对应的错误率

残差组	不增加 IN	1	1-2	1-3	1-4
top1 err.	24.27	23.58	22.94	22.60	22.96
top5 err.	7.08	6.72	6.40	6.28	6.49

5 表 5 IN 在 IBN 层中所占比例的不同对错误率的影响

IN 占比	0	0.25	0.5	0.75	1	全部 BN
top1 err.	24.27	22.49	22.60	23.11	23.44	28.56
top5 err.	7.08	6.39	6.28	6.57	6.94	9.83

在本实施例中，研究了添加不同数量 IN 层的 IBN 网络。表 4 给出了 IBN-Net50-a 的性能，其中 IN 层添加到不同量的残差组。可以看出，随着更多 IN 层添加到浅层，性能得到改善，但是当将 IN 层添加到最后的残差组时，性能却降低。这表明将 IN 应用在浅层有助于提高模型性能，而在深层需要采用 BN 以保留重要的内容信息。此外，本实
10 施例还研究了 IN-BN 比率对性能的影响，如表 5 所示，当 IN 的比例为 0.25 至 0.5 时，基于 top1 的错误率和基于 top5 的错误率分别为最低，这就证明 IN 和 BN 在使用的过程中需要折衷进行。

表 6 基于 Cityscapes-GTA 数据集的结果

训练的图像库	测试的图像库	模型	mIoU(%)	Pixel Acc.(%)
城市景观	城市景观	ResNet50	64.5	93.4
		IBN-Net50-a	69.1	94.4
		IBN-Net50-b	67.0	94.3
	GTA5	ResNet50	29.4	71.9
		IBN-Net50-a	34.2	73.3
		IBN-Net50-b	38.0	78.8
GTA5	GTA5	ResNet50	63.0	92.1
		IBN-Net50-a	65.0	92.4
		IBN-Net50-b	63.8	92.4
	城市景观	ResNet50	22.4	54.6
		IBN-Net50-a	28.1	66.0
		IBN-Net50-b	30.0	67.5

本实施例中用采用了孔卷积（Hole Convolution）的 ResNet50 为基准，IBN-Net 遵
15 循相同的修改。在每个数据集上训练模型并对 IBN-Net 和 ResNet50 进行评估，评估结果如表 6 所示。在表 6 中，mIoU（%）表示均交并比（Mean Intersection over Union，mIoU）。本实施例的实验的结果与 ImageNet 数据集中的结果一致。IBN-Net 在一个数据集中显示出更强的模型性能，并且在不同域的数据集之间具有更好的泛化能力。具体而言，IBN-Net-a 显示出更强大的模型性能，在两个数据集上，IBN-Net-a 的正确率优于

ResNet50 4.6% 和 2.0%。当进行交叉评估时, IBN-Net-b 的泛化更好, 采用 IBN-Net-b 比原始的 ResNet50, 从 Cityscapes 到-侠盗猎车手 (Grand Theft Auto V, GTA5) 的性能提高了 8.6%, 从 GTA5 到 Cityscapes 的性能提高了 7.6%。值得一提的是, 本实施例提供的 IBN-Net 与域适应工作不同。域适应是面向目标领域的, 并且在训练期间需要目标领域数据, 而本实施例通过的方法则不需要。尽管如此, 本实施例的方法的性能增益与域适应方法的性能增益仍然相当, 而且本实施例的方法向更通用的模型迈出了重要的一步, 因为本实施例向模型引入了内置的外观不变性, 而不是强迫该模型适应于特定的数据域。

表 7 调整不同数据的百分比提升的性能

调整数据的比例	10	20	30	100
ResNet50	52.7	54.2	58.7	63.84
BN-Net50-a	56.5	60.5	65.5	-

在新数据域上应用模型的另一个常用方法是使用少量目标域标注对其进行微调。本实施例提供的模型有更强的泛化能力, 因此网络所需的数据可以显著减少。本实施例使用不同数量的 Cityscapes 数据和标注来微调在 GTA5 数据集上预训练的模型。初始学习速率和周期数分别设为 0.003 和 80。如表 7 所示, 在只使用了 30% 的 Cityscapes 训练数据的条件下, IBN-Net50-a 的性能优于使用了所有训练数据的 ResNet50。

为了便于理解本实施例 IBN-Net 如何实现更好的泛化, 这里分析域偏置引起的特征散度。特征散度量如下, 对于 CNN 中某一层的输出特征, 将一个通道的平均值表示为 F , F 基本上描述了这个通道激活了多少, 假设 F 为均值为 μ , 方差为 σ^2 的高斯分布。则在该通道上域 A 和域 B 之间的对称相对熵 (symmetric KL divergence), 可以表示为:

$$D(F_A \| F_B) = KL(F_A \| F_B) + KL(F_B \| F_A) \quad (1);$$

$$KL(F_A \| F_B) = \log \frac{\sigma_A}{\sigma_B} + \frac{\sigma_A^2 + (\mu_A - \mu_B)^2}{2\mu_B^2} - \frac{1}{2} \quad (2);$$

用 $D(F_{iA} \| F_{iB})$ 表示第 i 个通道的对称相对熵。对该层的特征的所有通道的对称相对熵取平均, 就可以得到该层上域 A 和域 B 特征差异的一个度量, 其表达式为:

$$D(L_A \| L_B) = \frac{1}{C} \sum_{i=1}^C D(F_{iA} \| F_{iB}) \quad (3);$$

在公式 (3) 中, C 是该层中通道的数量, 此度量提供了域 A 和域 B 的特征分布之间距离的度量标准。

为了捕获实例归一化对外观信息和内容信息的影响, 这里考虑三组域。前两组是 Cityscapes-GTA5 和原始图像-莫奈风格的图像 (Photo-Monet), 这两组域有明显的外观差异。为了构建具有不同内容的两个域, 将 ImageNet-1k 验证集分成两部分, 第一部分包含图像有 500 个对象类别, 第二个部分包含其余 500 个类别。然后计算 ResNet50 和 IBN-Net50 主路径上的 17 个 ReLU 层的输出特征的特征散度。对上述三组图像进行实验, 得到的实验结果是, 在 IBN-Net 中, 由图片外观不同所造成的特征散度明显减小。对于

IBN-Net-a, 散度适度减小, 而对于 IBN-Net-b, 在第 2、4 和 8 层的 IN 层之后遇到突然下降, 而且这种效应持续到深层, 这意味着深度特征中的外观造成的差异减少了, 因此对分类的干扰减少了。另一方面, 由于内容差异导致的特征散度在 IBN-Net 中没有下降, 表明特征中的内容信息在 BN 层中保留得很好。在本实施例提出的 IBN-Net 中, 将 IN 和 BN 应用在单独的一个深度网络层中, 以提高神经网络的性能和泛化能力。本实施例将 IBN-Net 应用在包括 VGG, ResNet, ResNeXt 和 SENet 上, 并在 ImageNet 数据集上实现了一致的正确率提升。此外, 即使不使用目标域数据, 由 IN 引入的内置外观不变性可提升神经网络模型跨图像域的泛化能力。因此, IN 和 BN 层在 CNN 中的作用可以总结为: IN 引入了外观不变性并提升了泛化能力, 而 BN 保留了内容信息的在特征中的区分度。

本申请实施例提供一种图像识别装置, 图 5 为本申请实施例图像识别装置的组成结构示意图, 如图 5 所示, 所述装置 500 包括: 第一获取模块 501、第一处理模块 502 和第一输出模块 503, 其中: 第一获取模块 501, 配置为获取待识别图像; 第一处理模块 502, 配置为将所述待识别图像输入, 经过训练得到的神经网络模型, 得到所述待识别图像的识别结果, 其中, 所述神经网络模型是对神经网络进行 IN 和 BN 处理得到的。第一输出模块 503, 配置为输出所述待识别图像的识别结果。

在一些实施例中, 所述装置 500 还包括: 第二处理模块, 配置为对所述神经网络中卷积层输出的特征图进行 IN 和 BN 处理, 到所述神经网络模型。

在一些实施例中, 所述第二处理模块, 包括: 第一确定模块, 配置为从所述神经网络的卷积层中确定第一卷积层集合和所述第二卷积层集合; 第一子处理模块, 配置为对所述第一卷积层集合中的每一卷积层输出的特征图进行 IN 处理; 第二子处理模块, 配置为对所述第二卷积层集合中的每一卷积层输出的特征图进行 BN 处理。

在一些实施例中, 所述第一卷积层集合和所述第二卷积层集合组成的集合为所述神经网络的所有卷积层中的全部或部分。在本申请实施例中, 所述第一卷积层集合与所述第二卷积层集合不具有交集, 或者, 所述第一卷积层集合与所述第二卷积层集合具有交集; 或者, 所述第二卷积层集合为所述第一卷积层集合的子集合。

在一些实施例中, 所述第一子处理模块包括: 第一子确定模块, 配置为从所述第一卷积层集合中的每一所述卷积层输出的特征图对应的通道中确定第一通道集合; 第三子处理模块, 配置为对所述第一通道集合进行 IN 处理。

在一些实施例中, 所述第二子处理模块, 包括: 第二子确定单元, 配置为从每一所述卷积层输出的特征图对应的通道中确定第二通道集合; 第四子处理模块, 配置为对所述第二通道集合进行 BN 处理。

在一些实施例中, 所述第一通道集合为所述第一卷积层集合中的每一所述卷积层输出的特征图对应的所有通道的全部或部分; 所述第二通道集合为所述第二卷积层集合中的每一所述卷积层输出的特征图对应的所有通道的全部或部分。

在一些实施例中, 所述装置还包括:

第二处理模块，配置为对所述神经网络的两个块对应的特征图进行求和，得到输出结果，对所述输出结果进行 IN 处理；其中，所述神经网络至少包括两个块，且每一块的最后一层输出的特征图对应的通道数与上一块的最后一层输出的特征图对应的通道数相同。

5 需要说明的是，以上装置实施例的描述，与上述方法实施例的描述是类似的，具有同方法实施例相似的有益效果。对于本申请装置实施例中未披露的技术细节，请参照本申请方法实施例的描述而理解。需要说明的是，本申请实施例中，如果以软件功能模块的形式实现上述的即时通讯方法，并作为独立的产品销售或使用，也可以存储在一个计算机可读取存储介质中。基于这样的理解，本申请实施例的技术方案本质上或者说对
10 现有技术做出贡献的部分可以以软件产品的形式体现出来，该计算机软件产品存储在一个存储介质中，包括若干指令用以使得一台即时通讯设备（可以是终端、服务器等）执行本申请各个实施例所述方法的全部或部分。而前述的存储介质包括：U 盘、移动硬盘、只读存储器（Read Only Memory，ROM）、磁碟或者光盘等各种可以存储程序代码的介质。这样，本申请实施例不限制于任何特定的硬件和软件结合。

15 相应地，本申请实施例再提供一种计算机程序产品，所述计算机程序产品包括计算机可执行指令，该计算机可执行指令被执行后，能够实现本申请实施例提供的图像识别方法中的步骤。相应地，本申请实施例再提供一种计算机存储介质，所述计算机存储介质上存储有计算机可执行指令，所述该计算机可执行指令被处理器执行时实现上述实施例提供的图像识别方法的步骤。相应地，本申请实施例提供一种计算机设备，图 6 为本
20 申请实施例计算机设备的组成结构示意图，如图 6 所示，所述设备 600 包括：一个处理器 601、至少一个通信总线 602、用户接口 603、至少一个外部通信接口 604 和存储器 605。其中，通信总线 602 配置为实现这些组件之间的连接通信。其中，用户接口 603 可以包括显示屏，外部通信接口 604 可以包括标准的有线接口和无线接口。其中所述处理器 601，配置为执行存储器中存储的病理图像识别程序，以实现上述实施例提供的图
25 像识别方法的步骤。

以上计算机设备和存储介质实施例的描述，与上述方法实施例的描述是类似的，具有同方法实施例相似的有益效果。对于本申请即时通讯设备和存储介质实施例中未披露的技术细节，请参照本申请方法实施例的描述而理解。应理解，说明书通篇中提到的“一个实施例”或“一实施例”意味着与实施例有关的特定特征、结构或特性包括在本申请的
30 的至少一个实施例中。因此，在整个说明书各处出现的“在一个实施例中”或“在一实施例中”未必一定指相同的实施例。此外，这些特定的特征、结构或特性可以任意适合的方式结合在一个或多个实施例中。应理解，在本申请的各种实施例中，上述各过程的序号的大小并不意味着执行顺序的先后，各过程的执行顺序应以其功能和内在逻辑确定，而不对本申请实施例的实施过程构成任何限定。上述本申请实施例序号为了描述，不代表实施例的优劣。需要说明的是，在本文中，术语“包括”、“包含”或者其任何其他
35 变体意在涵盖非排他性的包含，从而使得包括一系列要素的过程、方法、物品或者装置

不仅包括那些要素，而且还包括没有明确列出的其他要素，或者是还包括为这种过程、方法、物品或者装置所固有的要素。在没有更多限制的情况下，由语句“包括一个……”限定的要素，并不排除在包括该要素的过程、方法、物品或者装置中还存在另外的相同要素。

- 5 在本申请所提供的几个实施例中，应该理解到，所揭露的设备和方法，可以通过其它的方式实现。以上所描述的设备实施例仅仅是示意性的，例如，所述单元的划分，仅仅为一种逻辑功能划分，实际实现时可以有另外的划分方式，如：多个单元或组件可以结合，或可以集成到另一个系统，或一些特征可以忽略，或不执行。另外，所显示或讨论的各组成部分相互之间的耦合、或直接耦合、或通信连接可以是通过一些接口，设备
10 或单元的间接耦合或通信连接，可以是电性的、机械的或其它形式的。

- 上述作为分离部件说明的单元可以是、或也可以不是物理上分开的，作为单元显示的部件可以是、或也可以不是物理单元；既可以位于一个地方，也可以分布到多个网络单元上；可以根据实际的需要选择其中的部分或全部单元来实现本实施例方案的目的。另外，在本申请各实施例中的各功能单元可以全部集成在一个处理单元中，也可以是各
15 单元分别单独作为一个单元，也可以两个或两个以上单元集成在一个单元中；上述集成的单元既可以采用硬件的形式实现，也可以采用硬件加软件功能单元的形式实现。

- 本领域普通技术人员可以理解：实现上述方法实施例的全部或部分步骤可以通过程序指令相关的硬件来完成，前述的程序可以存储于计算机可读取存储介质中，该程序在执行时，执行包括上述方法实施例的步骤；而前述的存储介质包括：移动存储设备、只
20 读存储器（Read Only Memory, ROM）、磁碟或者光盘等各种可以存储程序代码的介质。

- 或者，本申请上述集成的单元如果以软件功能模块的形式实现并作为独立的产品销售或使用时，也可以存储在一个计算机可读取存储介质中。基于这样的理解，本申请实施例的技术方案本质上或者说对现有技术做出贡献的部分可以以软件产品的形式体现出来，该计算机软件产品存储在一个存储介质中，包括若干指令用以使得一台计算机设
25 备（可以是个人计算机、服务器、或者网络设备等）执行本申请各个实施例所述方法的全部或部分。而前述的存储介质包括：移动存储设备、ROM、磁碟或者光盘等各种可以存储程序代码的介质。

- 以上所述，仅为本申请的具体实施方式，但本申请的保护范围并不局限于此，任何熟悉本技术领域的技术人员在本申请揭露的技术范围内，可轻易想到变化或替换，都应
30 涵盖在本申请的保护范围之内。因此，本申请的保护范围应以所述权利要求的保护范围为准。

权利要求书

- 1、一种图像识别方法，其中，所述方法包括：
获取待识别图像；
将所述待识别图像输入，经过训练得到的神经网络模型，得到所述待识别图像的识别结果，其中，所述神经网络模型是对所述神经网络中卷积层输出的特征图进行实例归一化 IN 和批归一化 BN 处理后得到的；
输出所述待识别图像的识别结果
- 2、根据权利要求 1 所述的方法，其中，所述神经网络模型的训练过程，包括：
从所述神经网络的卷积层中确定第一卷积层集合和所述第二卷积层集合；
对所述第一卷积层集合中的每一卷积层输出的特征图进行 IN 处理；
对所述第二卷积层集合中的每一卷积层输出的特征图进行 BN 处理。
- 3、根据权利要求 2 中所述的方法，其中，所述第一卷积层集合和所述第二卷积层集合组成的集合为所述神经网络的所有卷积层中的全部或部分。
- 4、根据权利要求 2 中所述的方法，其中，所述第一卷积层集合与所述第二卷积层集合不具有交集，或者，所述第一卷积层集合与所述第二卷积层集合具有交集；或者，所述第二卷积层集合为所述第一卷积层集合的子集合。
- 5、根据权利要求 2 中所述的方法，其中，所述对所述第一卷积层集合中的每一卷积层输出的特征图进行 IN 处理，包括：
从所述第一卷积层集合中的每一所述卷积层输出的特征图对应的通道中确定第一通道集合；
对所述第一通道集合进行 IN 处理。
- 6、根据权利要求 2 中所述的方法，其中，所述对所述第二卷积层集合中的每一卷积层输出的特征图进行 BN 处理，包括：
从所述第二卷积层集合中的每一所述卷积层输出的特征图对应的通道中确定第二通道集合；
对所述第二通道集合进行 BN 处理。
- 7、根据权利要求 2 至 6 任一项中所述的方法，其中，所述第一通道集合为所述第一卷积层集合中的每一所述卷积层输出的特征图对应的所有通道的全部或部分；所述第二通道集合为所述第二卷积层集合中的每一所述卷积层输出的特征图对应的所有通道的全部或部分。
- 8、根据权利要求 2 至 7 任一项中所述的方法，其中，所述第一卷积层集合中不包括所述神经网络中的最后一层卷积层。
- 9、根据权利要求 2 至 8 任一项中所述的方法，其中，所述神经网络至少包括两个块，且每一块的最后一层输出的特征图对应的通道数与上一块的最后一层输出的特征图

对应的通道数相同，所述方法还包括：

对所述神经网络的两个块对应的特征图进行求和，得到输出结果；

对所述输出结果进行 IN 处理。

- 10、一种图像识别装置，其中，所述装置包括：第一获取模块、第一处理模块和第一输出模块，其中：

第一获取模块，配置为获取待识别图像；

第一处理模块，配置为将所述待识别图像输入，经过训练得到的神经网络模型，得到所述待识别图像的识别结果，其中，所述神经网络模型是对神经网络进行 IN 和 BN 处理得到的。

- 10 第一输出模块，配置为输出所述待识别图像的识别结果。

11、根据权利要求 10 所述的装置，所述装置还包括：

第二处理模块，配置为对所述神经网络中卷积层输出的特征图进行 IN 和 BN 处理，到所述神经网络模型。

12、根据权利要求 11 所述的装置，其中，所述第二处理模块，包括：

- 15 第一确定模块，配置为从所述神经网络的卷积层中确定第一卷积层集合和所述第二卷积层集合；

第一子处理模块，配置为对所述第一卷积层集合中的每一卷积层输出的特征图进行 IN 处理；

- 20 第二子处理模块，配置为对所述第二卷积层集合中的每一卷积层输出的特征图进行 BN 处理。

13、根据权利要求 12 所述的装置，其中，所述第一卷积层集合和所述第二卷积层集合组成的集合为所述神经网络的所有卷积层中的全部或部分。

- 14、根据权利要求 12 所述的装置，其中，所述第一卷积层集合与所述第二卷积层集合不具有交集，或者，所述第一卷积层集合与所述第二卷积层集合具有交集；或者，
25 所述第二卷积层集合为所述第一卷积层集合的子集合。

15、根据权利要求 12 所述的装置，其中，所述第一子处理模块包括：

第一子确定模块，配置为从所述第一卷积层集合中的每一所述卷积层输出的特征图对应的通道中确定第一通道集合；

第三子处理模块，配置为对所述第一通道集合进行 IN 处理。

- 30 16、根据权利要求 12 所述的装置，其中，所述第二子处理模块，包括：

第二子确定单元，配置为从每一所述卷积层输出的特征图对应的通道中确定第二通道集合；

第四子处理模块，配置为对所述第二通道集合进行 BN 处理。

- 17、根据权利要求 12 至 16 任一项中所述的装置，其中，所述第一通道集合为所述
35 第一卷积层集合中的每一所述卷积层输出的特征图对应的所有通道的全部或部分；所述第二通道集合为所述第二卷积层集合中的每一所述卷积层输出的特征图对应的所有通

道的全部或部分。

18、根据权利要求 12 至 17 任一项中所述的装置，其中，所述神经网络至少包括两个块，且每一块的最后一层输出的特征图对应的通道数与上一块的最后一层输出的特征图对应的通道数相同，所述装置还包括：

- 5 第二处理模块，配置为对所述神经网络的两个块对应的特征图进行求和，得到输出结果，对所述输出结果进行 IN 处理。

19、根据权利要求 12 至 18 任一项中所述的装置，其中，所述第一卷积层集合中不包括所述神经网络中的最后一层卷积层。

- 20、一种计算机存储介质，其中，所述计算机存储介质上存储有计算机可执行指令，
10 该计算机可执行指令被执行后，能够实现权利要求 1 至 9 任一项所述的方法步骤。

21、一种计算机设备，其中，所述计算机设备包括存储器和处理器，所述存储器上存储有计算机可执行指令，所述处理器运行所述存储器上的计算机可执行指令时可实现权利要求 1 至 9 任一项所述的方法步骤。

- 22、一种计算程序产品，其中，所述计算机程序产品包括计算机可执行指令，该计
15 算机可执行指令被执行后，能够实现权利要求 1 至 9 任一项所述的方法步骤。