清华大学自动化系

专业实践技术报告

题目:

频率学习中的高低频分离模型研究

班　　级: 自 76

姓　　名: 吾尔开希 · 阿布都克力木

学　　号: 2017011589

实践地点: 清华大学自动化系

实践单位: 自动化系工业智能与系统研究所

实践部门: 工业智能与系统研究所

指导教师: 黄高

2020 年 9 月 12 日

目录

1 课题背景及需求分析

1.1 概述

卷积神经网络已经在计算机视觉领域取得了革命性的成就，在不同的视觉任务上均有不俗的表现，比如图像分类 [8]、物体检测和语义分割等。

受计算资源和存储空间大小的限制，大部分卷积神经网络模型只接受较小分辨率的 RGB 图像，比如典型的 224×224 大小。然而，现代相机拍下的照片通常尺寸较大。比如高清晰度相机（HD 相机）的分辨率为 1920×1080 ，通常被认为是较小的分辨率。甚至是 ImageNet[2] 数据集的平均尺寸 482×415 ，也大概是 224×224 尺寸的四倍。因此，大部分真实图像在被送入卷积神经网络之前，都需要很大尺度地被压缩。然而，图片压缩会难以避免地导致信息损失和准确率的降低 [11]。已有研究者尝试通过使用可学习的降采样网络来减少降采样过程中信息的损失 [7, 12]，然而，这样的降采样网络是受任务制约的，同时也增加了计算量，这在实际应用中是很大的劣势。

阿里巴巴团队提出，使用频域的频率筛选的方法来代替空间域的降采样 [15]，此论文也是本文算法的基础，以下用“原论文”来代称。受到普遍应用的 JPEG 图片存储格式采用 DCT 算法，将原图转化到 DCT 频率域进行存储，读取时又从频率域转化回到 RGB 空间域 [5]。原论文将卷积神经网络 CNN 的输入从原来三通道的 RGB 图像，改为多通道的频率谱，并从全频谱中采用注意力机制挑选部分频率，来提高信息传输效率，同时保持较高的正确率。

原论文的频率替代方法可以广泛应用于图像分类、图像分割、目标检测等计算机视觉任务中，且达到了与 RGB 输入同等水平，甚至更高的性能。

实验表明，频率替代方法只需要较少的频率分量就能达到足够高的性能，比如在分类任务中只需要全频率的八分之一就能达到最高性能。这在一些应用场合能带来很大的好处，如图1，传统的 RGB 输入需要将图片的全频信息传输到 GPU 计算服务器端，而频率替代方法只需要传输很少的频率分量，对信道带宽的要求大大降低。如今深度学习在移动端设备的应用越来越广泛，移动端设备往往不具备很强的算力，需要将图片通过网络传输到 GPU 服务器进行计算，GPU 计算效率一般较高，这时系统的瓶颈就在于传输时间，于是减少所需传输的图片

信息量就具有很大的意义。

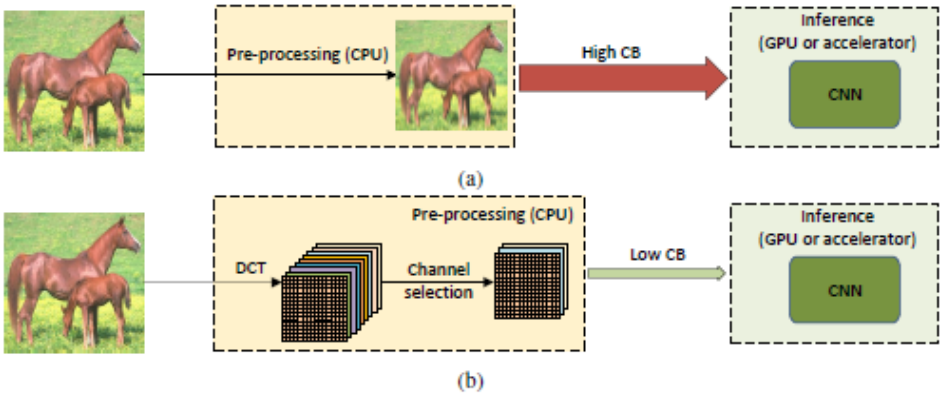


图 1: （a）传统 RGB 输入，需要将图片的全部信息传输到计算服务器。（b）频率替代方法，只需传输很少的频率分量

然而，原论文的方法为了适应频谱图较小的长宽尺寸，将 resnet 50 的池化操作减少了一次，这导致算法的运算量 FLOPs 大大增加。

为了进一步减少所使用的频率分量，同时减少算法的运算量和模型参数量，我们提出了高低频分离模型。对输入图片，首先使用极少的频率分量通过低频模型来进行预测，如果预测结果的置信概率满足条件，就将结果输出；否则再利用部分高频通过高频模型进行预测，高频模型计算过程中用到了低频模型的中间结果。实验结果表明，该方法在大大减少计算量和模型参数的同时，能保持较高的性能，同时进一步减少了对信道带宽的要求。我们认为该算法相比原算法有三点优势：

- 高低频分离法虽然用到两个 resnet50 模型，但每个模型的卷积通道数只有原模型的一半，因此总参数量和总运算量低于原模型，起到分组计算的效果。

- 高低频分离法适应不同的场景：当系统中的传输带宽成为系统计算时间的瓶颈时，使用高低频分离法可以优先传输低频分量，大部分数据（约 70% 的数据）只需要低频分量就能完成预测，对其他数据来说，在低频模型计算的同时进行高频分量的传输，在传输带宽一致的情况下并没有增大传输时间；当系统中的 GPU 运算成为系统计算时间的瓶颈时，可以使低频模型和高频模型同步运算，进一步减少运算时间。

- 在有着较低的运算量和参数量的同时保持了较高的准确率

1.2 相关工作

1.2.1 频域学习

除了我们作为算法基础的频率替代原论文 [15]，前人也有不少在频率学习方面的工作。在计算机视觉任务中，图片在频域中被压缩的信息往往包含着丰富的模式，[13] 设计了专用的自动编码器网络，来将频域转换和推理任务合并起来。[6] 从频域直接提取特征来进行图像分类。[3] 提出了一种模型转换算法，来将空域的模型参数直接转换到频域。本文的算法基础 [15] 与前述算法有两点不同，首先，频率替代法规避了复杂的模型转换过程，因此该模型有更广的应用空间；其次，频率替代法对神经网络在频域的偏向性提供了分析工具。

1.2.2 高效神经网络

随着对人工智能研究的深入开展，深度神经网络有着更深更广的发展趋势，模型的参数量和运算量都达到了空前水平。过多的参数量和运算量局限着深度神经网络的实际应用，特别是在移动端设备上的应用。为了减少模型的参数量和运算量，研究者致力于设计出高效深度神经网络。

[4, 10] 通过卷积核剪枝、量化的方式来压缩网络。另一类方法致力于在频率域压缩卷积网络。[1] 通过将卷积核参数转化到频域并用哈希表存储的方式，来减少存储空间。[14] 也将卷积核转换到频率域，并削减掉能量较低的频率因子，以此来压缩空间。这些方法所使用的 FFT 变换更适用于尺寸较大的卷积核，然而，目前性能较高的网络大多使用小卷积核。相比之下，频率替代法并没有对原卷积网络做本质的修改。

1.3 系统目标

原论文的频率替代方法可以实现高于传统算法 RGB 输入的准确率，但原论文的方法为了适应频谱图较小的长宽尺寸，将 resnet 50 的池化操作减少了一次，

这导致算法的运算量 FLOPS 大大增加，是传统方法的 3.3 倍，计算如公式1。

$$\frac{\text{频率替代法 Flops}}{\text{传统方法 Flops}} = \frac{13.5G}{4.08G} = 3.3 \tag{1}$$

此外，虽然原论文只筛选了图片的部分频率，但进行 DCT 变换前图片的尺寸是标准尺寸的四倍，因此，即使是筛选频率后的图片信息量也只是原图的 $\frac{1}{2}$ ，推理过程如公式2

$$\begin{gathered}\text{原图尺寸：} 3 \times 224 \times 224 \\ \text{频率筛选后尺寸：} 24 \times 56 \times 56 \\ \text{比例：} \frac{24 \times 56 \times 56}{3 \times 224 \times 224} = 1/2\end{gathered} \tag{2}$$

原论文的实验表明，选取输入图片的 24 组频率（共 192 组频率）来进行训练和测试时，模型性能较优。我们希望使用更少的频率分量（比如只用 6 组频率分量）来测试，尽可能多的减少所需传输的图片信息量（此时图片信息量是原图的 $\frac{1}{8}$ ），同时提高 GPU 运算效率。

然而，减少频率分量必然会导致准确率的降低，是否有办法在减少信道带宽要求，减少运算量的同时，保持较高的准确率呢? 我们为了达到此目标，提出高低频分离模型，部分图片只需要用低频分离通过低频模型来推理，如果推理结果的置信概率不满足要求，再用到高频分量通过高频模型来推理。实验表明，超过 70% 的图片只需 6 组频率分量通过低频模型来推理，就能达到满足要求的置信概率，同时达到 90% 以上的推理正确率。也就是说，大部分的输入样本只需要很少的信息传输和运算量，这表明频率分量模型的合理性。

2 数据集和评价指标

2.1 数据集介绍

2.1.1 ImageNet

ImageNet 项目是一个大型视觉数据库，旨在用于视觉对象识别软件研究。该项目已对超过 1400 万张图像进行了手动注释，以指示拍摄了哪些对象，并且在至少一百万张图像中，还提供了边框 [2]。ImageNet 包含 20000 多个类别，其中一些典型类别，例如 “气球” 或 “草莓”，由数百个图像组成。尽管实际图像不归 ImageNet 所有，但可以直接从 ImageNet 免费获得第三方图像 URL 批注数据库。自 2010 年以来，ImageNet 项目每年举办一次软件竞赛，即 ImageNet 大规模视觉识别挑战赛 (ILSVRC)，在此竞赛中，软件程序将竞争以正确地分类和检测对象和场景。该挑战选取 ImageNet 中的一千个非重叠类来进行。ImageNet 的部分类别图片如图2。

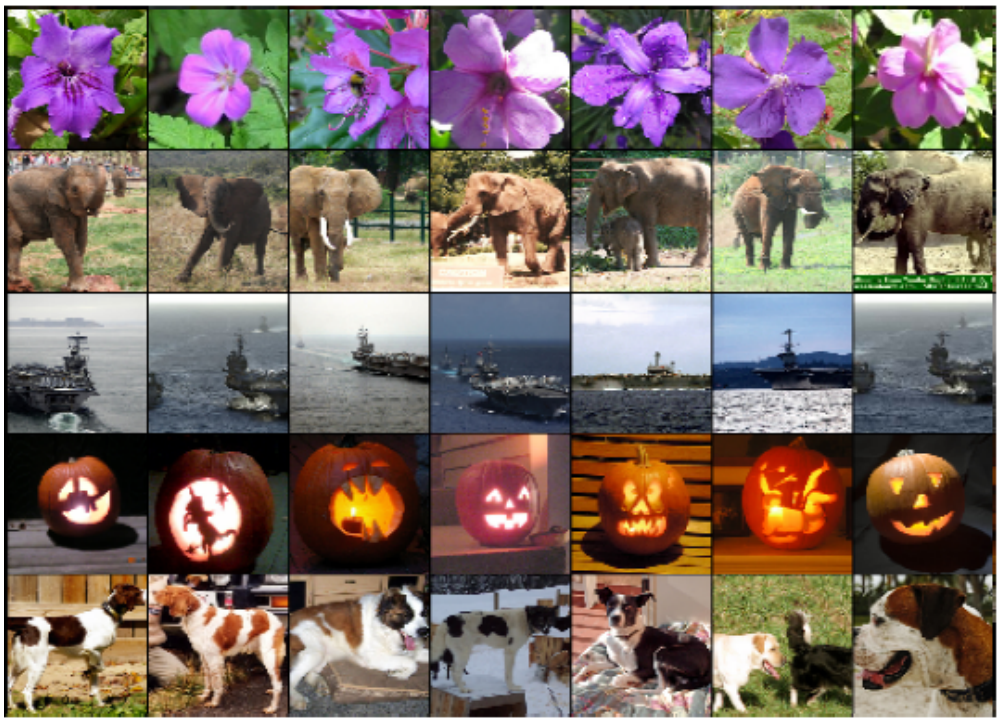


图 2: ImageNet 部分类别图片

2012 年，卷积神经网络 AlexNet[9] 在 ImageNet 2012 挑战赛中 top 5 错

误率达到 15.3%，比第二名低 10.8%。AlexNet 在训练期间使用了图形处理单元(GPU)，GPU 成为深度学习革命的重要组成部分。此后，不仅是在 AI 社区内部，而且是整个技术行业开始关注人工智能。在 2015 年，AlexNet 被 Microsoft 的深度卷积神经网络所击败，CNN 赢得了 ImageNet 2015 竞赛。

2.1.2 mini-ImageNet

mini-ImageNet 是 ImageNet 数据集的子集，常被用于小样本学习，以及普通的深度学习任务。mini-ImageNet 数据集包含 100 类共六万张彩色图片，其中每类有 600 个样本，图片和 ImageNet 原图一样，只是从 ImageNet 中抽取出一百类，方便研究者用更短的训练时间来测试自己的模型性能。

2.2 数据预处理

2.2.1 数据增广

算法主要采用随机纵横比、随机缩放、随机裁剪和随机水平翻转的增广方式。ImageNet 的图片尺寸较为多样，但经过预处理后均以 448×448 的大小进行 DCT 变换。

随机纵横比会将图片的长宽比重新设定为一个随机值，体现为纵向或横向的拉伸。随机值范围为原长宽比的 3/4 到 4/3，如图3中的（b)。

随机缩放会将图片的尺寸随机缩放，此时长宽比不变，随机缩放的比例范围在 0.08 到 1.0，如图3中的（c)。

随机裁剪会从图片随机区域裁剪下固定尺寸的块，裁剪大小为 448×448，如图3中的（d)。

随机水平翻转会以 50% 的概率将图片进行水平翻转，水平翻转的效果如图3中的（e)。

数据增广是应用于训练集图片的流程，针对测试集，需要先将输入缩放为 512×512 大小，再中心裁剪 448×448 大小的图片，再进行 DCT 变换。

图 3: 数据增广流程

2.2.2 DCT 变换

DCT 变换即离散余弦变换，通过将原图与一系列余弦系数进行卷积来得到图像的频域信号，DCT 变换多用于数据的压缩，如 JPEG 协议所用的就是 DCT 变换。为了压缩数据，DCT 变换得到的系数可以被阈值量化，即低于某能量阈值的系数可以被削减。DCT 变换也对其他很多在科学工程技术中应用很广，例如数字信号处理，通讯设备和网络通讯等。

之所以使用余弦系数而不是正弦系数，是因为使用余弦函数来拟合标准函数需要更少的系数。DCT 变换是傅里叶相关的变换，类似于离散傅里叶变换(DFT)，但只用到实数。DCT 变换的运算公式如图4。

$$F(u,v) = c(u)c(v)\sum_{i=0}^{N-1}\sum_{j=0}^{N-1} f(i,j)\cos\left[\frac{(i+0.5)\pi}{N}u\right]\cos\left[\frac{(j+0.5)\pi}{N}v\right]$$

$$c(u) = \begin{cases}\sqrt{\frac{1}{N}}, & u = 0\\ \sqrt{\frac{2}{N}}, & u \neq 0\end{cases}$$

图 4: DCT 变换运算公式

在实际运用 DCT 变换时，会有一些特殊设置，如 JPEG 技术。JPEG 算法的第一步是将输入图片转化到 YCrCb 域，再分割成 8×8 的小块，对每个小块单独使用 DCT 变换。由于人眼对 Y 分量的高频更敏感，对 CrCb 分量的高频相对不敏感，所以 JPEG 技术会舍弃 CrCb 分量的高频，三个分量的频域图尺寸比为 2：1：1. 8×8 的余弦系数如图5。

频率替代法 [15] 在使用 JPEG 算法进行 DCT 变换前，为了使 YCrCb 三个分量的频率尺寸一致，会将 CrCb 两个分量的尺寸增大一倍。再将图片分割成 8×8 的小块，对每个小块单独使用 DCT 变换，每个 8×8 的小块得到 8×8 大小的频率图。将不同小块的同一种频率拼成同一频率图，于是 YCrCb 三个分量各得到 64 种频率，一共 $3 \times 64 = 192$ 个通道，频率图尺寸为 $448/8 = 56$。最终得到的全频尺寸为 $192 \times 56 \times 56$。上述流程如图6。

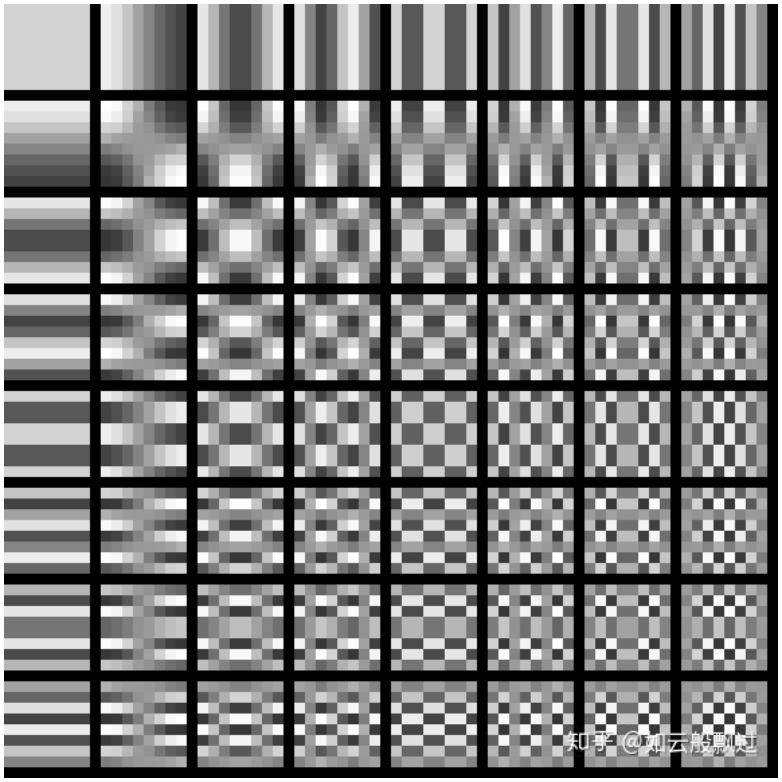


图 5: 8×8 的余弦系数

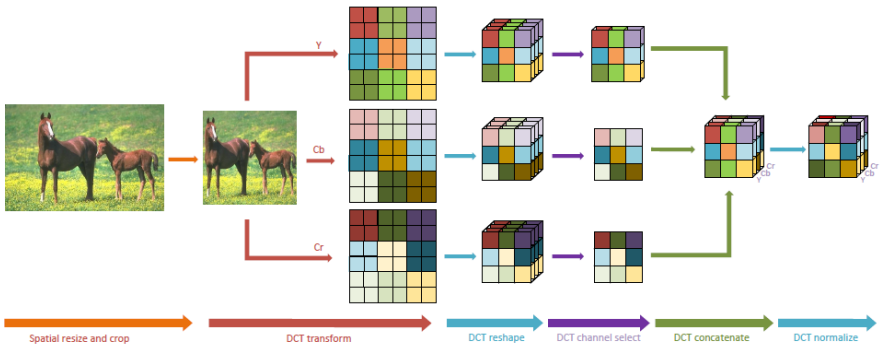


图 6: 频率替代法 DCT 变换流程

2.3 评价指标

模型的评价指标包括准确率、运算量和模型参数量三个指标。

模型输出可以理解为其对输入图片属于每个类的概率大小预测，比如 mini-ImageNet 有 100 类图片，因此模型会对每个输入输出长度为 100 的向量，其每个分量表示输入属于该类的概率大小。top1 accuracy 只取其中概率最大的类别，如果该预测类别与实际类别相同，则认为预测正确；top5 accuracy 取预测结果中概率最大的五个类别，如果实际类别包含在这五个类别中，则任务预测正确。top1 accuracy 和 top5 accuracy 的计算公式如3。

$$\begin{aligned} top1\ accuracy &= \frac{\text{测试图片中正确标签与最大分类概率相同的数量}}{\text{总的测试图片数量}} \\ top5\ accuracy &= \frac{\text{测试图片中正确标签包含在最大五个分类概率中的数量}}{\text{总的测试图片数量}} \end{aligned} \tag{3}$$

模型的运算量用 floating point operations （FLOPs）来衡量，指浮点运算数。以卷积层为例，不考虑偏置。假设输入通道数为 C_i，卷积核尺寸为 K，输入特征图长宽为 H 和 W，输出通道数为 C_o。在计算输出特征图的每个像素时，要先做 $C_i \cdot K^2$ 次乘法运算，再将这 $C_i \cdot K^2$ 个乘法结果相加，需要做 $C_i \cdot K^2 - 1$ 次加法运算，因此一个卷积层不考虑偏置时的运算数如公式4。模型总的运算量等于模型各个层运算量之和。

$$(C_i \cdot K^2 + C_i \cdot K^2 - 1) \cdot H \cdot W \cdot C_o = (2 \times C_i \times K^2 - 1) \times H \times W \times C_o \tag{4}$$

模型参数量指模型参数的个数，一般来说，参数量越多模型的表达能力越强，但也对运算设备的要求越高。无偏置的全连接层，若输入特征维度为 $N_features$，输出特征维度为 N_units，那么参数量为 $N_features \times N_units$; 无偏置的卷积层，若输入通道数为 C_i，卷积核尺寸为 K，输入特征图长宽为 H 和 W，输出通道数为 C_o，则参数量为 $K^2 \times C_i \times C_o$。

3 算法实现

3.1 频率替代法

频率替代法是本文的基础算法，[15] 提出了频域学习的一般方法，包括数据预处理和输入信息修剪的方法。图 1 展示了传统方法和频率替代法的区别。在传统方法里，高分辨率的 RGB 图像一般是在 CPU 上进行预处理并被传输到 GPU 加速器中用于实时计算。由于未压缩的图像在 RGB 模式下一般比较大，因此在 CPU 和 GPU 设备之间的传输带宽就可能成为实时计算的瓶颈。为此，频率替代法将 RGB 图像转化到频域，并留下特定的一部分频率。与传统方法相比，频率替代法需要更少的传输带宽，并能达到较高的准确率。

使用频率替代法只需对现有的 CNN 模型做最小的修改，具体来说，只需移除 CNN 的输入层并保留剩下的残差层。第一个残差层被用作输入层，且其输入维度按频率图的维度进行修改。如图7所示，当输入频率图尺寸为 56×56×64 时，resnet-50 的前三个输入层被移除，只剩下残差层，且残差层的输入维度被改为 64。

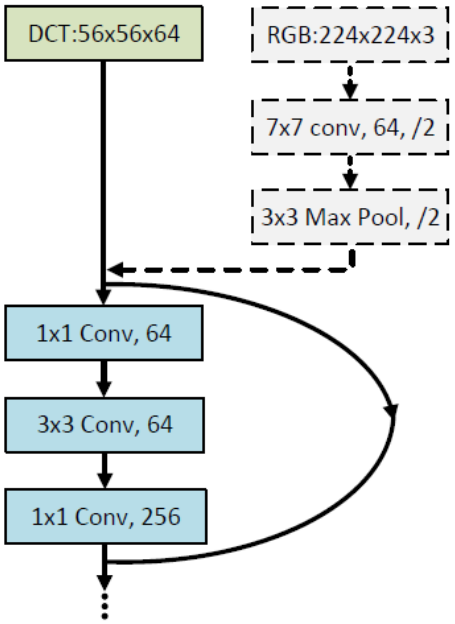


图 7: 频率替代法对已有 CNN 输入层的修改

3.1.1 频域转换

如图6所示，对一张图片首先在空域进行数据增广操作，如随机缩放、随机剪切和随机翻转。下一步，图片被转换到 YCbCr 颜色空间域再用 DCT 算法转换到频域，将二维的频域图中属于同一频率大小的系数拼到同一频道中，得到三维的频率图。频域转换时使用 JPEG 算法，会将将输入图片转化到 YCrCb 域，再分割成 8×8 的小块，对每个小块单独使用 DCT 变换。由于频域转换时是以 8×8 的小块为基本单位，所以 YCbCr 一个分量得到的频率数为 64，三个分量就是 192，频率图的长宽是原图的 1/8。如果原图的尺寸是 $H \times W \times C$，那么得到的频率图尺寸为 $H/8 \times W/8 \times 64C$，原图和全频频率图的信息量是相同的。

由于频率图的长宽较小，所以跳过 CNN 的输入层，其一般会缩小输入图的尺寸，比如 resnet-50，有 stride 为 2 的卷积层和一个 max-pooling 最大池化层。此外，作者发布的代码中，还将 resnet-50 第二个 layer 的 stride 从 2 改为 1，这其实会大大增加模型的 FLOPs。

在 ImageNet 分类问题中，传统方法一般将输入图缩放裁剪到 $224 \times 224 \times 3$ 大小。频率替代法将输入 RGB 图像缩放裁剪到 $448 \times 448 \times 3$ 大小，频域转换后得到的全频图大小为 $56 \times 56 \times 192$。

3.1.2 自适应频率筛选

假设在全部 192 种频率中，不是所有频率都会对图像分类、语义分割、实例分割等任务有很大的贡献，去除这些频率并不会对模型性能产生很大的影响，同时能减少频率图信息量。因此，[15] 提出自适应频率筛选法，来探究各个频率对每种任务的重要性。

自适应频率筛选方法在模型输入处采用动态门模型，对每种频率赋予二值分数，重要的频率分量被赋予 1，不重要的频率分量被赋予 0，且不参与后续的模型计算，动态门模型的结构如图8所示。

自适应频率筛选提供了对各个频率分量重要性的估计，在测试集对不同频率分量被挑选的概率进行统计，可以直观的看出各个频率分量对该数据集在该任务中所起作用的大小。在图像分类任务和语义分割任务中的统计结果如图9所示，每个方框里的数字代表频率种类，数字越大表示频率越高。方框的颜色表示

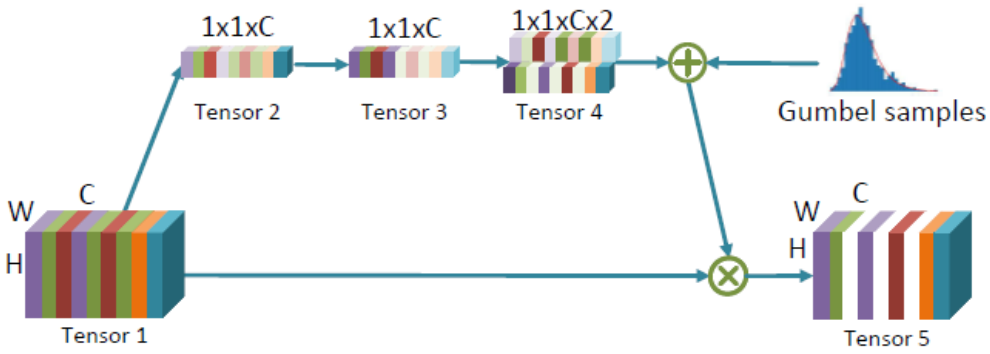


图 8: 动态门模型筛选频率

该频率给挑选的概率，颜色越深表示该分量被挑选的概率越大，即在该任务中起的作用越大。

从统计结果可以看出，低频通道被选择的更多，这说明低频通道比高频信息量更大；同时，颜色空间中 Y 分量的频率通道被选择的更多，这说明亮度信息在视觉推理中更重要；此外，图像分类任务和语义分割任务被选择的频率分量大致相同，这说明频率筛选的规律是比较普适的。

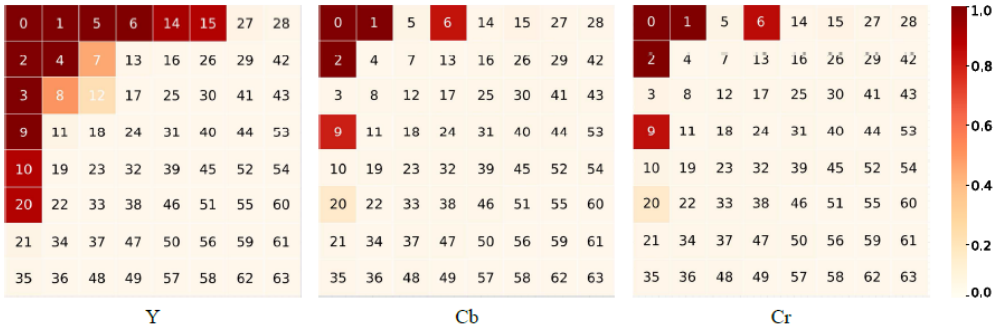
在实际训练时，并不采用动态门模型，而是根据频率选择的统计规律固定的选择若干频率作为输入，比如选择 6 个频率，24 个频率，48 个频率等。

3.2 高低频分离

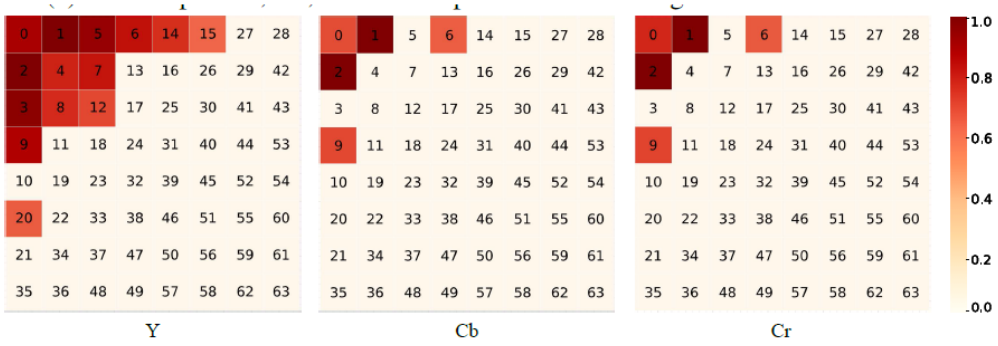
频率替代法将模型的输入从空域改到频域，并从全部频率中挑选一部分来减少传输的信息量。然而，原论文为了与频率图较小的尺寸相匹配，将 resnet50 中 layer2 的 stride 从 2 改为 1，大大增加了模型的 FLOPs，是传统方法的 3.3 倍，如公式1。此外，原论文虽然进行了频率筛选，但选择 24 个频率后图片信息量只是原图的 1/2，如公式2，并没有很显著的减小信息量，从而减小带宽。为减小模型运算量、参数量和传输带宽，我们提出高低频分离模型。

3.2.1 算法流程

高低频分离算法使用两个 0.5 的 resnet50 模型，0.5 意味着模型各个卷积层的输出维度为原模型的一半，这就保证了算法的运算量和参数不会因为有两个



(a) 在 image-net 图像分类数据集中各个频率分量被挑选的频率



(b) 在语义分割数据集中各个频率分量被挑选的概率

图 9: 不同频率在分类任务和语义分割任务中被挑选的概率大小

模型而增加，反而会减少。将一个 resnet50 称为低频模型，另一个称为高频模型，这就意味着输入图片的低频分量会被送入低频模型，高频分量会被送入高频模型，比如从输入图片的频率图中选择 24 个频率（与原论文的选择相同），将其中的 6 个低频分量送入低频模型，剩余 18 个高频分量送入高频模型。

低频模型计算出输入图片属于每个类的概率大小，取其中概率最大的类别作为低频模型的预测结果，将其对应概率视为置信度。如果置信度高于设定的阈值（如 90%），则认为低频模型的预测是可靠的预测，算法输出其预测结果；否则再使用高频模型计算，高频模型计算过程中会用到低频模型的部分中间结果，最终输出高频模型的预测结果。实际上，高频模型可以与低频模型同步计算，可以根据系统的时间瓶颈来决定。算法流程图如图10所示。

3.2.2 模型结构

上文提到算法使用两个 0.5 的 resnet50 模型，其中的高频模型在运算过程中会用到低频模型的部分运算结果。如何搭建两个模型之间的关系，就变得尤为重要，不仅影响到推理过程，也会对模型的训练过程产生影响。为此，我们尝试了三种不同的模型结构。

第一种结构如图11所示，高频模型最终得到的特征向量与低频模型对应的特征向量相拼接，再送入高频分类器中。这种结构的缺点是高频模型的卷积层在推理过程中只用到高频信息，而图像的特征更多的蕴含在低频中，这使得高频模型很难感知有用的图像特征。

第二种结构如图12所示，高频模型中间层的 feature map 就与低频模型对应部位的 feature map 相融合，具体为 resnet50 四个 layer 输出的 feature map 相融合。融合方式有两种，直接相加（由于两个 resnet50 在相同部位的 feature map 大小一样，所以可以直接相加）和在通道数维度进行拼接。四个 layer 输出有四个融合点，我们尝试了不同的组合连接方式，发现当只有 layer1 输出的 feature map 进行融合时，效果最佳，这说明模型深层部位的融合并没有带来更好的效果。我们猜测这是由于低频模型和高频模型在深层部分所感知的图像层次和结构不一样，这时两个模型的信息不再互相匹配。

结构二的实验表明：在模型前端的融合更有利于提升性能，于是我们提出第

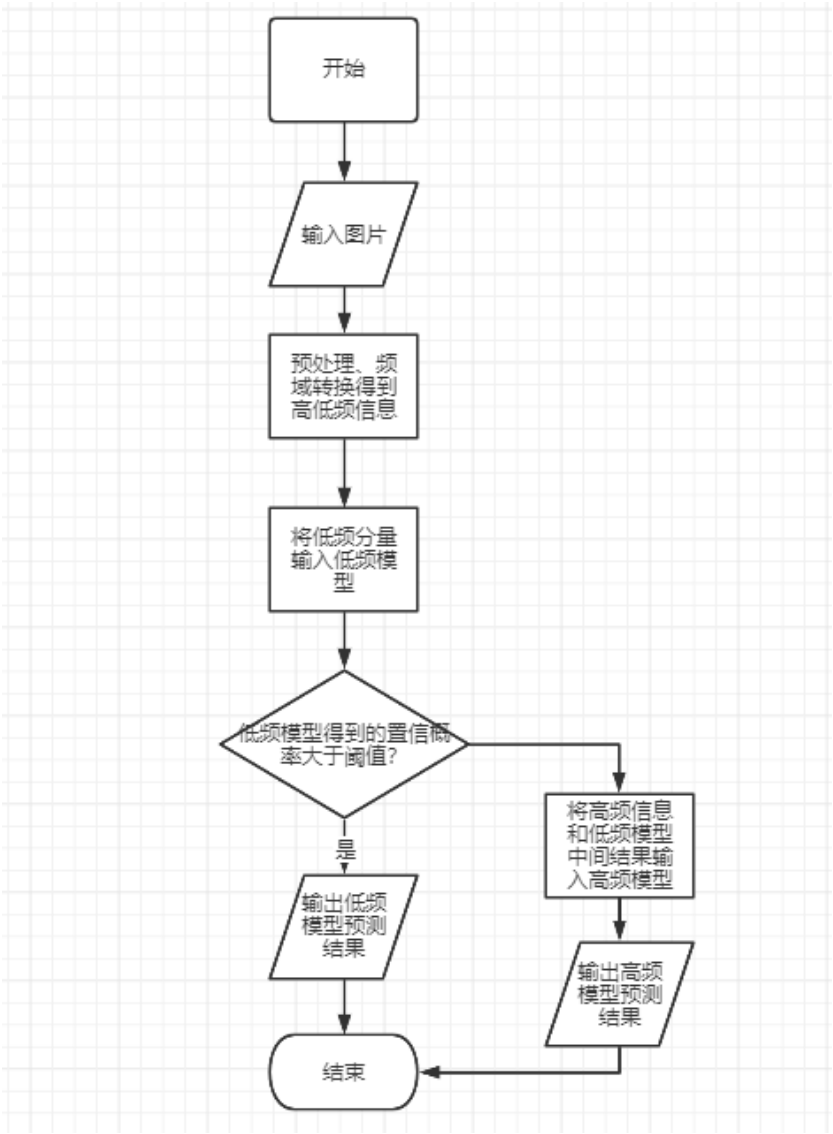

开始
输入图片
预处理、频域转换得到高低频信息
将低频分量输入低频模型
低频模型得到的置信概率大于阈值?
是
输出低频模型预测结果
将高频信息和低频模型中间结果输入高频模型
输出高频模型预测结果
结束

图 10: 算法流程图

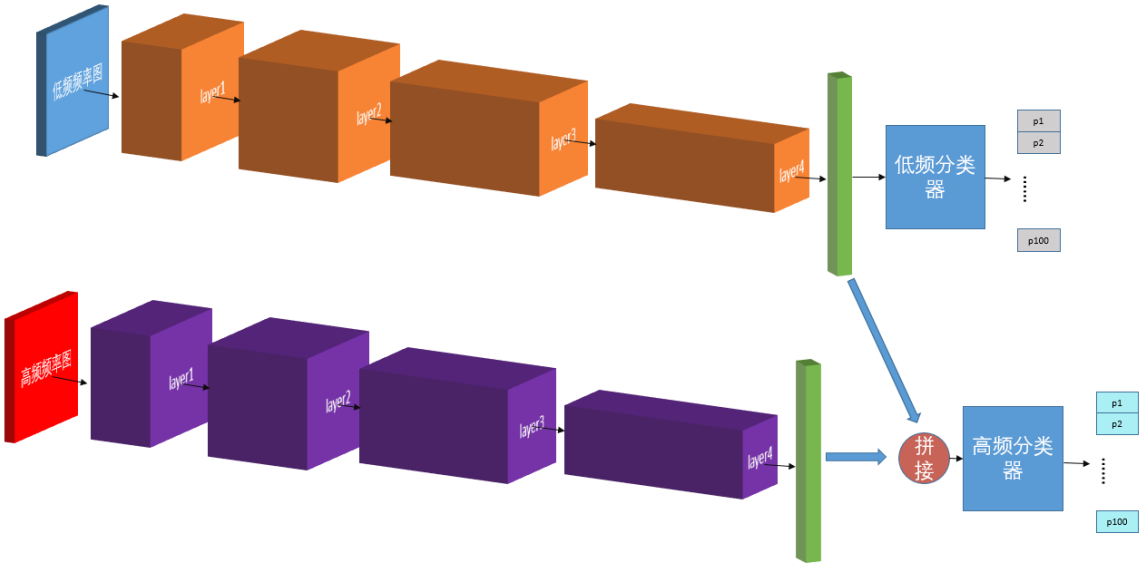


图 11: 模型结构一：特征向量拼接

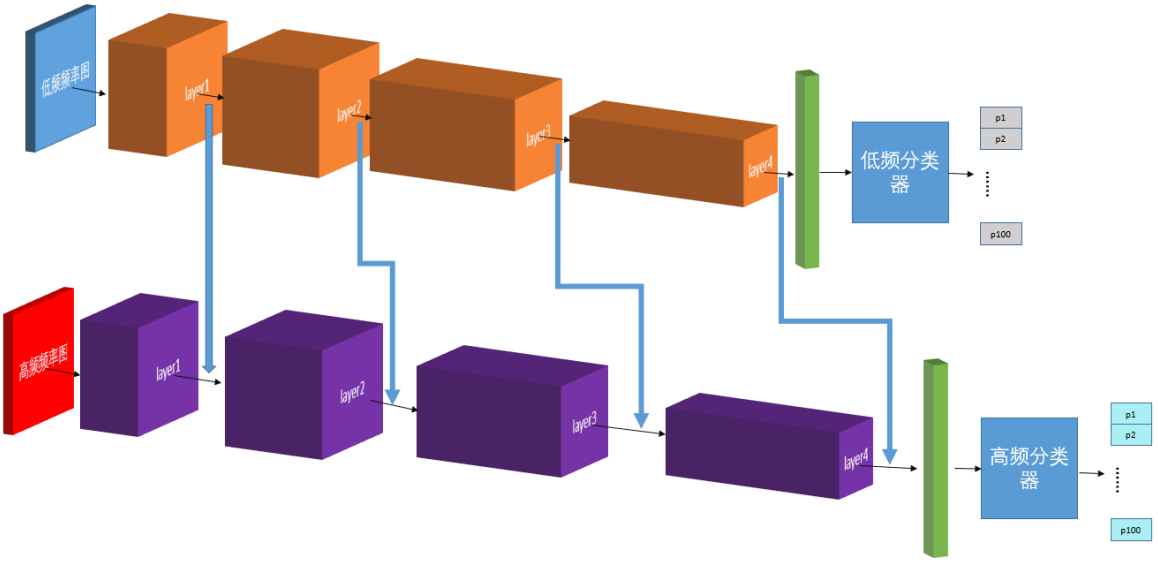


图 12: 模型结构二：中间层融合

三种结构，如图13所示，在两个 resnet 模型的残差结构前增加一个输入卷积层（二维卷积层 + 批正则化层 + 激活层），在高频模型中将卷积层的输出与低频模型卷积层输出进行拼接，再送入残差结构。实验表明结构三的性能优于结构二，且优于不融合的 resnet50。

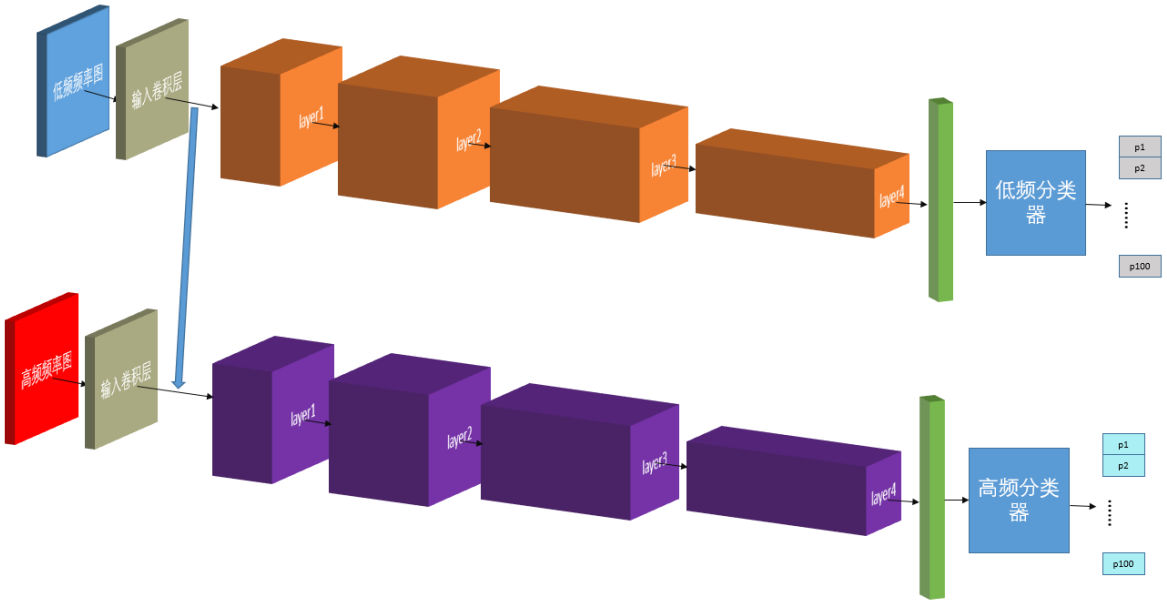


图 13: 模型结构三：输入卷积层融合

3.2.3 训练方法

由于我们的算法用到了两个 0.5 倍的 resnet50 模型，两个模型都有输出，因此需要两步训练。首先，将低频模型的残差层和分类器从模型中分离，只训练输入卷积层以及高频模型，这时模型的前向传播路径如图14所示。在第二步训练中，固定低频模型的输入卷积层参数，只训练低频模型的残差层和分类器。

每一步训练都使用 SGD 优化器，学习率设置为 0.5，使用余弦函数式的学习率调整方式，batch size 为 128，训练 90 个 epoch，原论文的算法以及 RGB 模型的训练也使用同样的训练参数，第二步训练相关数值的变化如图15所示。

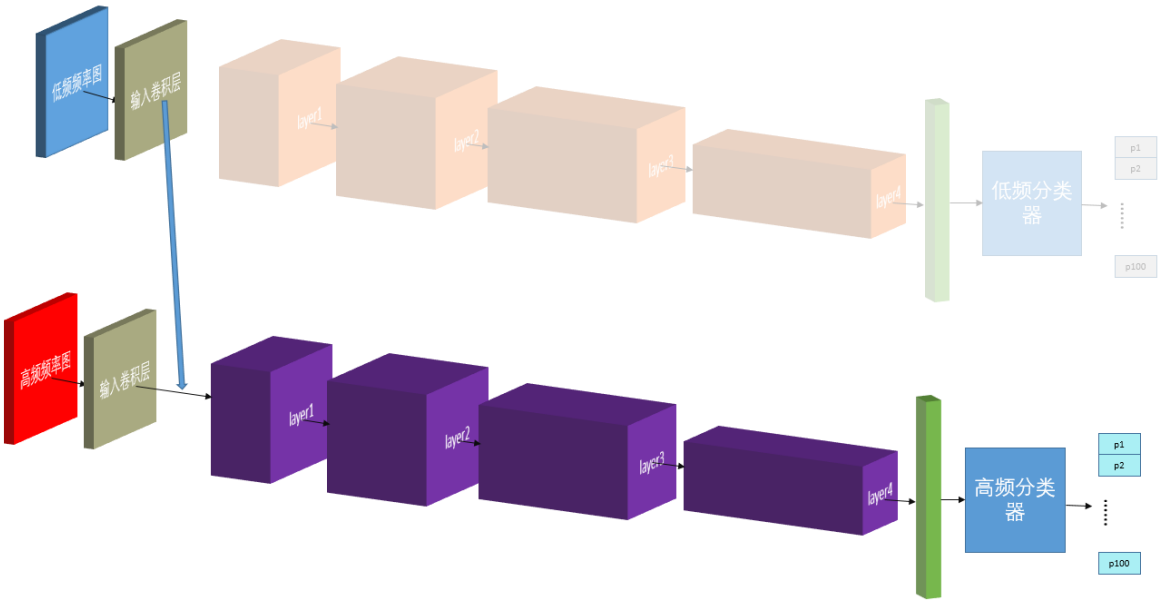


图 14: 模型第一步训练的前向传播路径

4 实验结果

4.1 与传统方法的比较

高低频分离法与传统方法的比较如表格1所示，作为基准线的有传统 RGB 输入法和频率替代法 [15]，两种方法都使用 resnet50 模型，高低频分离使用两个 0.5 倍的 resnet50 模型（卷积层通道数均为原模型的 0.5 倍）。三种方法训练时的参数都相等，保证训练至收敛。各种方法 resnet 50 模型的残差结构都相同，第二个 layer 的 stride 为 2。高低频分离法置信概率的阈值设置为 0.85，有 68.0% 的数据只需要通过低频模型。

表中高低频分离法的运算量 FLOPs 是指平均运算量，68.0% 的数据只需要通过低频模型，这些数据需要 0.99G FLOPs 的运算量，其他 32.0% 数据需要 2.00G FLOPs 的运算量，因此平均需要 $68.0\%\times 0.99G + (1-68.0\%)\times 2.00G = 1.31G$

从表中可以看出，虽然频率替代法的准确率最高，但高低频分离法参数量为频率替代法的 50%，运算量 FLOPs 为频率替代法的 33%，且高低频分离法准确率只比频率替代法低 0.3%。此外，从表中可以看出，当直接减少频率替代法的

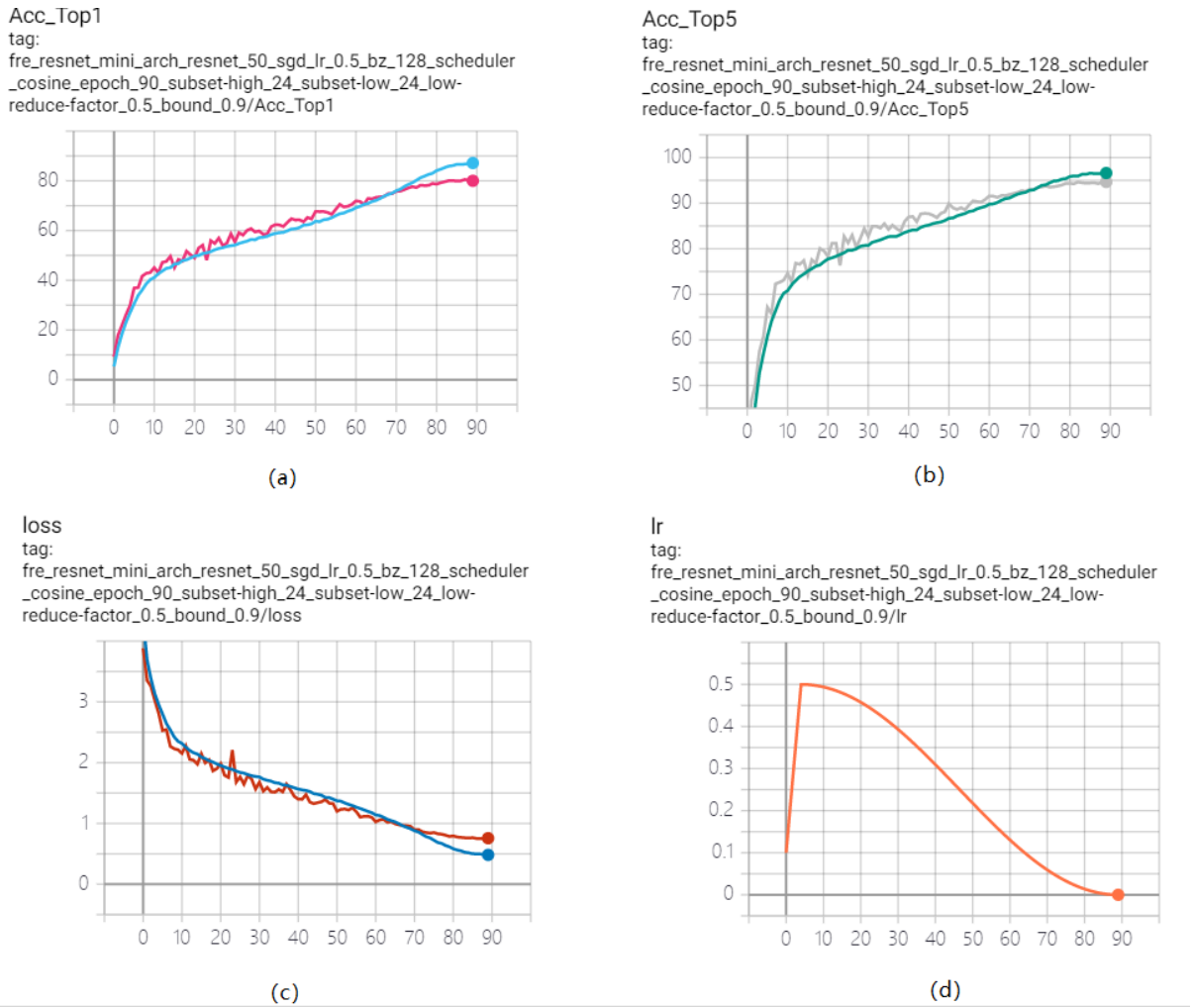


图 15: 模型第二步训练时的相关曲线：（a）top 1 准确率，浅蓝色曲线为训练集准确率，橘红色曲线为测试集准确率。（b）top 5 准确率，绿色曲线为训练集准确率，灰色曲线为测试集准确率。（c）损失函数值，绿色曲线为训练集损失，深红色曲线为测试集损失。（d）学习率，按余弦函数式进行调整。

表 1: mini-ImageNet 数据集中，高低频分离法在准确率、参数量和计算量三个方面与传统方法与传统方法的比较

算法	resnet 模型	top 1 准确率	参数量	FLOPs
RGB 输入	resnet50	82.67%	25.5M	4.08G
频率替代法 (24 个频率)	resnet50	**83.59%**	23.7M	3.93G
频率替代法 (6 个频率)	resnet50	81.53%	23.7M	3.93G
高低频分离法 (低频 6 个频率，高频 18 个频率)	2*(0.5 resnet50)	83.22%	11.88M	1.31G

表 2: mini-ImageNet 数据集中，高低频分离法不同模型结构的比较：连接位置的列表中，第 i 个分量表示在 resnet50 第 i 个 layer 后是否连接

结构	连接方式	连接位置	top 1 准确率
结构一	-	-	81.82%
结构二	直接相加	位置：[0,1,2,3]	81.93%
	通道拼接	位置：[0,1,2,3]	81.95%
		位置：[0,1,2]	81.78%
		位置：[0,1]	81.99%
		位置：[0]	82.76%
结构三	-	-	**83.22%**

输入频率数时，准确率大大降低。

4.2 不同模型结构的比较

在 3.2.2 中我们介绍了三种不同的结构，我们尝试了不同结构的不同连接方式，结果如表2所示。可以看出，结构三的性能最优。结构二中，连接位置越少越靠前，性能越好，这与结构三只在输入卷积层后连接的方法相符。

4.3 消融实验

高低频分离法有两个 resnet50 模型，分别为低频模型和高频模型，我们比较三种推理方法：1. 仅低频：只使用 6 个频率分量通过低频模型进行推理。2. 高低频：低频分量通过低频模型的输入卷积层，高频分量通过高频模型进行推理，如图14。3. 部分高低频：部分输入仅通过低频模型（置信概率达到阈值），其他

表 3: 两个 resnet50 模型的三种不同推理方式

推理方式	top1 准确率
仅低频	80.52%
高低频	82.87%
部分高低频	83.22%

输入通过低频模型后也通过高频模型。三种推理方法的性能比较如表3，可以看出仅有低频的方式准确率很低，单纯高低频结合的方式准确率也不如部分高低频的推理方式，且部分高低频的方式能使部分输入只需 6 个频率分量，减少传输信息量。

我们收集了低频模型预测结果中置信度较高的一些输入图片和较低的一些输入图片，来进行对比，如图16所示。可以看出，低频模型能得到较高置信度的输入图片，一般是图片所包含的类别信息一目了然，且图片中的其他干扰物体不多，类别物体占据了图片的大部分空间，即使让这些图片经过滤波模糊，也能看出物体的类别；而置信度较低的图片，可能是类别物体在图片中不处于显眼的位置，被其他物体干扰（如第一张图中的船），也可能是物体形状容易被误解为其他物体，需要纹理等细节信息来帮助判断类别（如第三张图中的刀）。

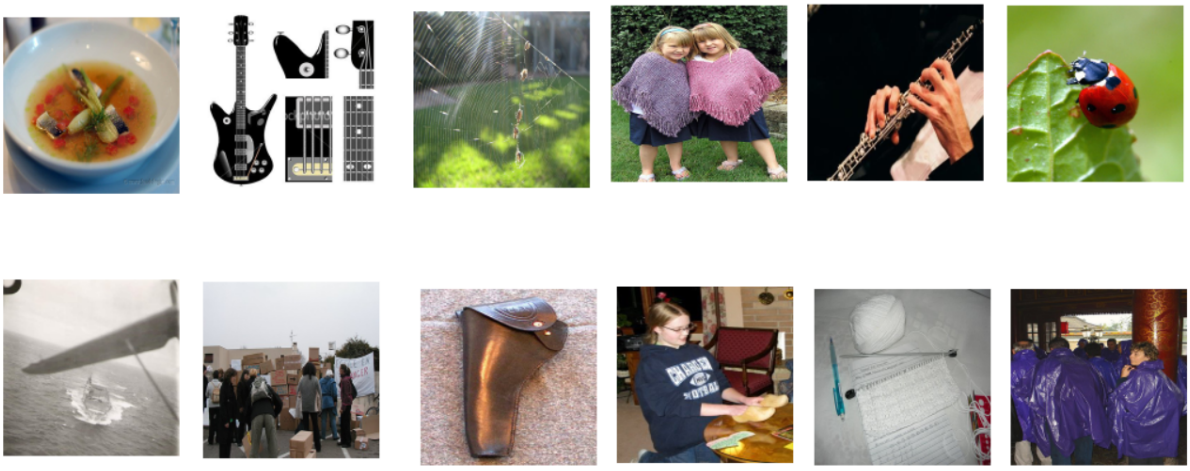


图 16: 低频模型预测得出置信度较高的图片（上排）和置信度较低的图片（下排）

其他关于训练细节的实验我们不再赘述，前述的训练参数已是经过实验验证最佳的参数。

5 总结与展望

高低频分离法大大降低了系统传输所需信道大小，约 70% 的数据只需 6 个低频分量就能完成预测，6 个频率分量的信息量是传统方法一张图片信息量的 1/8。此外，高低频分离法适用于不同的应用场景，当系统中的传输带宽成为系统计算时间的瓶颈时，使用高低频分离法可以优先传输低频分量；当系统中的 GPU 运算成为系统计算时间的瓶颈时，可以使低频模型和高频模型同步运算，进一步减少运算时间。其次，高低频分离法的参数量和运算量比传统方法以及频率替代法 [15] 都要低。在实现这些功能的同时高低频分离法能达到较高的性能，top 1 准确率只比原方法低 0.3%。

由于时间和设备的约束，我们只在 mini-ImageNet 上进行了分类实验，将来需要在 ImageNet 数据集以及语义分割数据集上补充进一步的实验，来充分说明高低频分离法的有效性。此外，两个模型的结构也是可以继续探索的方面，比如减少低频模型的规模，增大高频模型的规模，两个模型使用其他的连接方式等，使高频模型更充分的利用到低频模型的中间运算结果。

参考文献

[1] Wenlin Chen, James T. Wilson, S. Tyree, Kilian Q. Weinberger, and Yixin Chen. Compressing convolutional neural networks in the frequency domain. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2016.

[2] Jia Deng, W. Dong, R. Socher, L. Li, K. Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR 2009*, 2009.

[3] M. Ehrlich and L. Davis. Deep residual learning in the jpeg transform domain. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 3483–3492, 2019.

[4] Jonathan Frankle and Michael Carbin. The lottery ticket hypothesis: Finding sparse, trainable neural networks. *arXiv: Learning*, 2019.

[5] Jessica J. Fridrich. Feature-based steganalysis for jpeg images and its implications for future design of steganographic schemes. In *Information Hiding Workshop*, 2004.

[6] L. Gueguen, A. Sergeev, Ben Kadlec, Rosanne Liu, and J. Yosinski. Faster neural networks straight from jpeg. In *NeurIPS*, 2018.

[7] Heewon Kim, Myungsub Choi, Bee Lim, and Kyoung Mu Lee. Task-aware image downscaling. In *ECCV*, 2018.

[8] A. Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. Imagenet classification with deep convolutional neural networks. In *CACM*, 2017.

[9] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. ImageNet Classification with Deep Convolutional Neural Networks. In F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 25*, pages 1097–1105. Curran Associates, Inc., 2012.

[10] P. Molchanov, Arun Mallya, S. Tyree, I. Frosio, and J. Kautz. Importance estimation for neural network pruning. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11256–11264, 2019.

[11] Yanting Pei, Ya-Ping Huang, Qi Zou, X. Zhang, and S. Wang. Effects of image degradation and degradation removal to cnn-based image classification. *IEEE transactions on pattern analysis and machine intelligence*, 2019.

[12] Faraz Saeedan, Nicolas Weber, M. Goesele, and S. Roth. Detail-preserving pooling in deep networks. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9108–9116, 2018.

[13] Robert Torfason, Fabian Mentzer, Eirikur Agustsson, Michael Tschannen, R. Timofte, and L. Gool. Towards image understanding from deep compression without decoding. *ArXiv*, abs/1803.06131, 2018.

[14] Yunhe Wang, Chang Xu, C. Xu, and D. Tao. Packing convolutional neural networks in the frequency domain. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41:2495–2510, 2019.

[15] Kai Xu, Minghai Qin, Fei Sun, Yuhao Wang, Yen-Kuang Chen, and Fengbo Ren. Learning in the Frequency Domain. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1740–1749, 2020.