面向触发器约束的人脸识别

刘瑜[[1]](#footnote-0)[1]光禄宋\*满园张\*海吉昊刘\*郁聪周骏捷燕yuliu@ee.cuhk.edu.hk[[2]](#footnote-1)

{ Songguanglu，zhangmanyuan，liujihao } @ sensetime.com

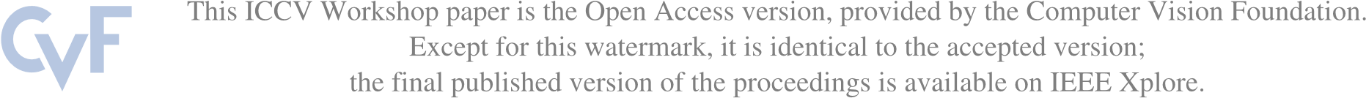
香港中文大学

SenseTime研究

摘要

*大规模的面部识别具有挑战性，尤其是在计算预算有限的情况下。给定FLOPS的上限，关键是找到最佳的神经网络架构和优化方法。在本文中，我们的团队主要专注于两个“大”轨道，分别基于图像和基于视频。这两个路径的提交必须是一个单一模型，其计算预算不得超过30 GFlops。我们介绍了一种网络架构“高效PolyFace”，一种新颖的损失函数“ ArcNegFace”，一种新颖的帧聚合方法“ QAN ++”，以及我们实施中的一系列有用技巧（增强效果，规则的人脸，标签平滑，锚点微调等） ）。我们的基本模型“ Efficient PolyFace”用于基于“ deepglint-large”的基于图像的轨道需要28.25 Gflops，而“ PolyFace + QAN ++”解决方案用于基于“ iQiyi-large”的基于视频的轨道需要24.12 Gflops。这两个解决方案在两个轨道上分别达到94.198％@ 1e-8和72.981％@ 1e4，这是最新的结果*[[3]](#footnote-2)[2][[4]](#footnote-3) *在这场比赛中。*

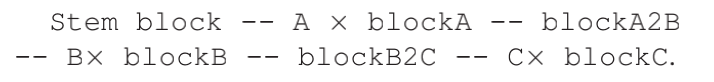
# ~~1.轻量级人脸识别挑战~~

~~ICCV19-轻量级面部识别挑战[ 2 ]是开放式面部识别中最严格的竞赛之一。它要求训练数据[ 4 ]，面部检测器[ 3 ]和不同提交之间的对齐方法具有严格的一致性。这项比赛有四条赛道：基于小型图像，基于大型图像，基于小型视频和大型视频。计算小径和大径分别为1Gflops和30Gflops。~~

# 2.基于图像的基线模型

我们采用两种不同的CNN架构R100 [ 1 ]和提出的PolyFace作为我们的基线模型。根据挑战[ 2 ]的要求，两种基本架构的输入大小均为112×112 。

**PolyFace：**与PolyNet [ 11 ] 的结构相似，基本PolyFace通过重复其基本块进行设计。基本块的详细信息如图1 所示。在拟议的PolyFace的主干块中，空间大小首先通过一个上采样和一个卷积层（我们称为“主干富集块”）上采样到235×235 ，然后缩小到112×112 。整个PolyFace中的数据流为：



在主干的末尾，采用具有256个外通道的完全连接层来生成表示，然后是BatchNorm1d 层。基本模型中[A，B，C]的block number为[10,20,10]。

**训练细节**。在基本模型的训练过程中，使用16个GPU来实现1,024的全局批处理大小。同步BN用于组大小1。总训练迭代次数设置为100,000，初始学习率为0.001，并且在前10,000迭代期间预热到0.4。权重衰减设置为1e-5，动量设置为0.9。最终嵌入的落差率为0.4的落差用于防止过度拟合。

表1 中显示了测试服务器上的[ 2 ] 两个基本模型的结果。

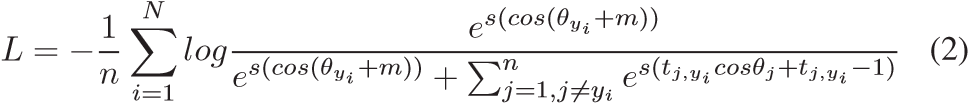
|  |  |  |  |
| --- | --- | --- | --- |
| 模型 | 人字拖 | 失利 | TPR @ FPR = 1e-8 |
| R100 | 24.22克 | 弧面 | 90.972 |
| 宝利通 | 16.62克 | 弧面 | 90.829 |

|  |
| --- |
| 图1. PolyFace中块的详细信息。*Conv* + *BN* + *ReLU*块中的数字表示输入通道，输出通道，内核大小，步幅和填充。*Max Pooling*块中的数字表示内核大小和步幅。*Conv* + *BN*块中的数字表示输入通道，输出通道，内核大小，步幅和填充。 |

表1.不同基本模型之间的比较。触发器由https：// github中的公共工具计算。com / Swall0w / torchstat （公共工具中的总MAdd）。

# 3.新的损失函数：ArcNegFace

在本节中，我们将引入一个名为ArcNegFace的新的鲁棒的损失函数。与最近的大多数新颖损失不同，ArcNegFace考虑了类锚点之间的距离。



其中，t(没看)）函数表示函数*G*（· *，* ·）是高斯函数，公式为：

 （3）

其中*α*，*μ*和*σ*分别设置为1.2、0和1。选项卡2中显示了ArcNegFace的性能。

|  |  |  |
| --- | --- | --- |
| 模型 | 失利 | TPR @ FPR = 1e-8 |
| 宝利通 | 弧面 | 90.829 |
| 宝利通 | ArcNegFace | 91.639 |

表2.不同损失函数之间的比较。

# 4.高效的PolyFace

受高效网络[ 10 ] 想法的启发，我们启动了NAS处理，以扩展基本模型的深度和

|  |  |  |
| --- | --- | --- |
| 块号 | 频道号 | TPR @ FPR = 1e-8 |
| [3,13,30,3] | [64,128,256,512] | 88.652 |
| [3,13,30,3] | [72,144,288,576] | 90.243 |
| [3,16,37,3] | [65,130,260,520] | 90.188 |
| [3,20,46,3] | [59,118,236,472] | 89.954 |
| [3,25,57,3] | [53,106,212,424] | 89.875 |
| [3,13,50,3] | [61,122,244,488] | 89.789 |
| [3,9,19,3] | [84,168,336,672] | 89.734 |
| [3,9,31,3] | [74,148,296,592] | 89.699 |

表3.不同修改的R100型号的性能。

宽度受计算预算的约束。选项卡3 中显示了R100上的一些选定结果。请注意，所有实验都是在相同的基本设置下进行的。最终，我们发现扩展后的PolyFace模型之一的性能优于所有具有相同Flops（约28 Gflops）的候选者，因此我们将其称为Efficient PolyFace，作为最终的骨干[[5]](#footnote-4)[3]。选项卡7 中显示了一些选定的结果。[[6]](#footnote-5)

|  |  |  |
| --- | --- | --- |
| 模型 | 阿达酸 | TPR @ FPR = 1e-8 |
| 高效的PolyFace  高效的PolyFace ABN | √ | 93.801 94.198 |

表4. AdaBN的性能。成绩94.198是排行榜上的最后提交。

|  |  |  |
| --- | --- | --- |
| 模型 | 余量 | TPR @ FPR = 1e-8 |
| 宝利通 | 0.5 | 90.829 |
| 宝利通 | 0.3 | 91.332 |

表5.基于ArcFace的不同边距的性能。

# 5.绝招

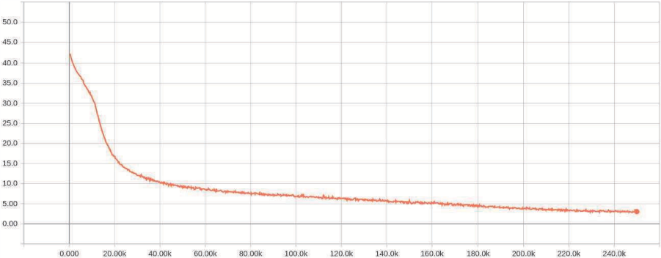
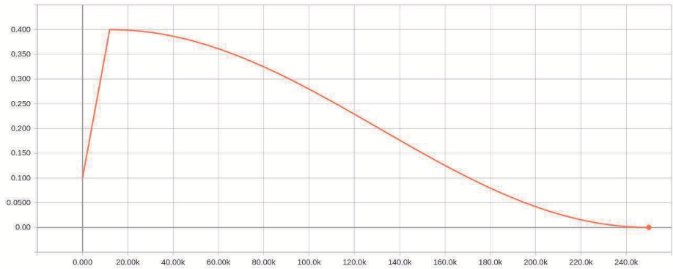
## 5.1。锚点微调

我们引入了一个新的正则化术语，称为“ *锚点微调”*。给定一个收敛模型，我们提取训练集的特征，并通过相应身份的均值特征在分类层中重新初始化权重*W。*然后，将基于此对模型进行微调，如表6 所示。

## 5.2。缩放和移位增强

在训练过程中，所有设置都使用数据扩充。原始图像将被重新缩放并随机在±1％范围内移动。表6中显示了性能。5.3。色彩抖动

添加颜色抖动时，亮度，对比度和饱和度设置为0.125。



濟濸濴瀅瀁濼瀁濺澳濥濴瀇濸

濟瀂瀆瀆

濜瀇濸滢濴瀇泺瀂瀁

图2.余弦衰减的细节。

## 5.4。翻转策略

在训练阶段采用翻转策略。在推断阶段，我们提取原始图像和翻转图像的特征。最终特征是它们的平均值。结果显示在选项卡6中。

## 5.5。普通脸

常规人脸[ 12 ]适用于限制类间距离，但是我们发现它在消耗大量内存的同时很少带来改善。

## 5.6。标签光滑

我们探索了标签平滑策略，该策略在ImageNet分类中得到了广泛使用。结果显示在选项卡6中。

## 5.7。阿达酸

考虑到训练集和测试集之间的域偏移，我们对收敛模型执行AdaBN [ 7 ]，以提高其性能。结果显示在选项卡4中。

## 5.8。保证金的修改

我们在ArcFace中修改了边距，如表5 所示，它带来了一些改进。

## 5.9。余弦学习率和随机深度

我们探索余弦学习速率衰减和随机深度[ 6 ]，以实现进一步的增益。在所有实验中，随机深度的保持率均设置为0.8。学习速率的函数*WRT*迭代示于图2 ，并且结果显示在选项卡显示7 。基本训练中的损失

PolyFace如图2 所示。

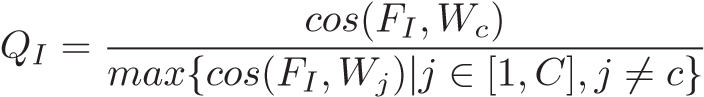
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | | 模型 | ArcNegFace | 缩放和移位 | 翻转 | 普通脸[ ] | 标签光滑 | Fc微调 | 拱微调[ 5 ] | TPR @ FPR = 1e-8 | | R100  R100  R100  R100  R100  R100  R100  R100 | √  √  √  √  √  √  √ | √  √  √  √  √ | √  √ | √ | √  √√√ | √ | √  √ | 81.503 80.59  81.628 80.819 81.085 81.272 81.922  81.638 |   表6.不同培训策略的比较。请注意，在不清除错误标签的情况下，对deepglint-large的性能进行了评估。   |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | 模型 | 人字拖 | 积木 | 余弦衰变 | 随机深度 | 色彩抖动 | TPR @ FPR = 1e-8 | | PolyNet [ 11 ]  PolyFace  高效的PolyFace | 16.62克24.04克  28.25克 | [10,20,10]  [20,30,20]  [23,38,23] | √  √  √ | √  √  √ | √  √  √ | 93.066 93.729  93.801 |   表7.基于ArcNegFace的余弦衰减和随机深度的性能。在*规模*和*转移八月*和 *翻转* 中采用 |

这些实验。使用边距为0.3的ArcNegFace。

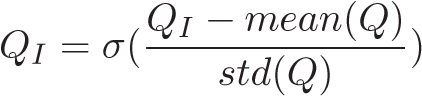
# 6.增强的质量感知网络，用于视频面部识别

为了产生在爱奇艺轨道设定toset识别鲁棒视频表示[ 2 ]，由QAN和RQEN [启发8 ，9 ]，我们提出了称为增强的质量感知网络（QAN ++）一个新的质量估计策略来近似每个质量图片。图像集的表示可以在图像质量的辅助下通过帧表示的加权总和进行汇总。

与图像的主观质量判断不同，我们的方法根据特征识别的特征分配图像质量。定义数据集*d*与*Ç*身份和重量锚*Wˉˉ 我*，*我* ∈[1 *，C* ] 在最后的分类层，图像的质量*我*与ID *Ç*可以通过计算：

 （4）

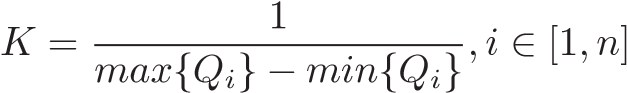
在训练集上计算图像质量，为了在推理阶段获得图像质量，我们添加了一个轻量级的质量生成分支以回归在训练集上计算出的质量值。为了更好地降低质量，我们将其标准化为：

 （5）

其中*σ*（· ），*均值*（*Q*）和*std*（*Q*）分别表示整个训练集中的S型函数，均值和标准差值。L2损失被用作训练损失。

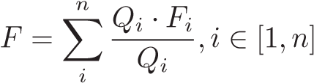
在推论阶段，给定视频*我我，我* ∈[1 *，N* ] ，其中n的装置的总的图像数目和相应的特征表示*˚F 我*，我们提取质量值*Q 我*的*我我*。质量值将通过以下方式重新缩放：

*Q i* = *K* · *Q i* + *B* （6）

 （7）

*B* = 1 − *K* · *max* { *Q i* } *，* i∈[1 *，n* ] （8）

最后，可以通过以下方式汇总视频级功能：

 （9）

如果图像集中的图像编号n小于3，我们直接采用等式9对其进行汇总，而无需重新缩放质量值。

## 6.1。不同聚合策略的表现

我们评估在LFR中对IQIYI提出的质量估计策略的有效性。结果显示在选项卡8中。我们在PolyFace中嵌入了一个新的质量分支。新分支看起来像是ResNet-18的小版本。每级中的块号为[2 *，* 2 *，* 2 *，* 2] 和在每级中的信道数被设定为[8,16,32,48]。在全局平均池之后，我们添加输出数为1的完全连接层以降低质量。质量网的触发器为81 *。*9 Mflops，输入与PolyFace相同。

|  |  |  |  |
| --- | --- | --- | --- |
| 型号（不含ABN） | 深闪 | 聚合 | 爱奇艺 |
| R100 | 92.433 | 平均 | 65.843 |
| R100 | 92.433 | 加权总和 | 67.381 |
| R100 | 92.433 | 顶级质量 | 65.217 |
| R100 | 92.433 | QAN ++ | 69.048 |
| PolyFace | 93.729 | QAN ++ | 72.981 |

表8.在LFR中对IQIYIlarge磁道使用不同质量策略的比较。成绩72.981是排行榜上的最终作品。

# 7.结论

在本文中，我们显示了针对ICCV19-LRF挑战的解决方案的详细信息。对于基于图像和基于视频的轨道，我们引入了新的主干Efficient PolyFace和新的损失函数ArcNegFace。对于基于视频的轨道，我们提出了一种新颖的质量估算器QAN ++来为每个帧生成质量得分。此外，我们还探索了面部识别模型中的一些有用技巧。挑战测试服务器上的结果证明了所提出方法的有效性。

4

原文

提供更好的翻译建议

1. 他们为这项工作做出了同等贡献 [↑](#footnote-ref-0)
2. [↑](#footnote-ref-1)
3. [↑](#footnote-ref-2)
4. [↑](#footnote-ref-3)
5. 模型架构和参数将是开源的 [↑](#footnote-ref-4)
6. [↑](#footnote-ref-5)