

(19)中华人民共和国国家知识产权局



## (12)发明专利申请



(10)申请公布号 CN 111429350 A

(43)申请公布日 2020.07.17

(21)申请号 202010210142.6

(22)申请日 2020.03.24

(71)申请人 安徽工业大学

地址 243002 安徽省马鞍山市湖东路59号

(72)发明人 刘恒 吴文雪

(74)专利代理机构 安徽知问律师事务所 34134

代理人 平静

(51)Int.Cl.

G06T 3/40(2006.01)

G06N 3/04(2006.01)

G06N 3/08(2006.01)

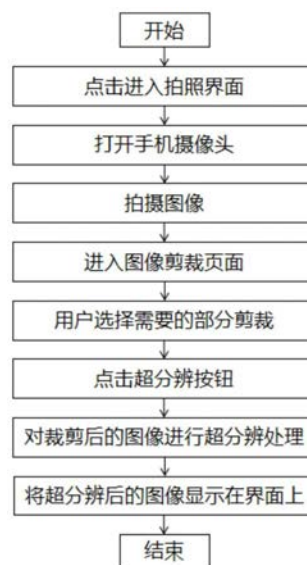
权利要求书2页 说明书8页 附图2页

(54)发明名称

一种手机拍照快速超分辨处理方法

(57)摘要

本发明公开了一种手机拍照快速超分辨处理方法,属于图像处理技术领域。本发明提出一个新型的快速的轻量化的图像超分辨模型,并采用Tensorflow lite通用技术将模型应用在手机端,对拍摄后的图片不清晰的部分进行裁剪,选取需要的部分并对其进行图像超分辨处理从而达到局部放大清晰化的效果,模型运行速度在3-5s左右,从而为深度学习模型在Android等移动端的应用提供了广阔的应用前景。



CN 111429350 A

1. 一种手机拍照快速超分辨率处理方法,其特征在于,其步骤为:

步骤1、利用常用图像处理数据集,制作高分辨率图像块与低分辨率图像块训练集;

步骤2、构建一个快速轻量化的图像超分辨率模型;

步骤3、依据步骤1获得的训练集和步骤2构建的模型网络,进行网络训练,调整网络参数;

步骤4、将训练好的模型保存为pb格式;

步骤5、对步骤4所得训练模型进行测试;

步骤6、将pb格式的模型移植到Android端,实现手机拍照后快速进行图像超分辨率处理。

2. 根据权利要求1所述的一种手机拍照快速超分辨率处理方法,其特征在于:步骤1中使用DIV2K数据集制作高分辨率图像块与低分辨率图像块训练集。

3. 根据权利要求2所述的一种手机拍照快速超分辨率处理方法,其特征在于:步骤1制作高分辨率图像块与低分辨率图像块训练集的过程为:

对图像处理数据集中所有原始彩色图像进行裁剪,得到高分辨率图像块;然后对高分辨率图像进行采样,得到与高分辨率图像对应的低分辨率图像,最后得到用于深度网络训练的高分辨率图像块与低分辨率图像块的集合。

4. 根据权利要求1-3中任一项所述的一种手机拍照快速超分辨率处理方法,其特征在于:步骤2中构建的图像超分辨率模型,由一个卷积后接32个残差单元,后接一个卷积,再接一个上采样单元,最后再接一个卷积构成;其中,每个残差单元使用反向残差,且反向残差块中使用的激活函数是ReLU6;网络输入为低分辨率图像块,输出为超分辨率图像块,该网络将学习低分辨率图像块和超分辨率图像块之间的映射关系,如公式(1)所示:

$$y=F(x, \Phi) \quad (1)$$

其中, $x, y$ 分别表示低分辨率图像块和超分辨率图像块, $\Phi$ 为学习到的模型参数,用于之后的高分辨率图像重建。

5. 根据权利要求4所述的一种手机拍照快速超分辨率处理方法,其特征在于:步骤2中每个卷积层得到的输出用公式(2)表示:

$$y_i=f(W_i x_i+b_i), i=1,2,\dots,10 \quad (2)$$

其中, $W_i$ 表示第*i*层的权重, $b_i$ 表示第*i*层的偏置, $x_i$ 表示第*i*层的输入, $y_i$ 表示第*i*层的输出。

ReLU6激活层的激活函数用公式(3)表示如下:

$$f(x)=\min(\max(0,x),6) \quad (3)$$

其中, $x$ 为ReLU6激活函数的输入, $f(x)$ 为ReLU6激活函数的输出。

6. 根据权利要求5所述的一种手机拍照快速超分辨率处理方法,其特征在于:步骤2中残差单元使用带线性瓶颈的反向残差;每个残差单元有三个卷积层,首先由一个1\*1的卷积接后激活函数ReLU6,然后接一个3\*3的Depthwise卷积后接激活函数ReLU6,最后接一个1\*1的卷积和非线性的激活函数;残差单元的表现如公式(4)所示:

$$\begin{aligned} y_i &= h(x_i) + F(x_i, w_i) \\ x_{i+1} &= f(y_i) \end{aligned} \quad (4)$$

其中, $x_i$ 表示为第*i*层的残差单元的特征输入, $w_i$ 表示为第*i*层权重和偏置项的设置, $F$ 代表的是残差函数, $f$ 则代表激活函数ReLU6, $h$ 是作为恒等式映射: $h(x_i)=x_i$ 。

7. 根据权利要求6所述的一种手机拍照快速超分辨率处理方法, 其特征在于: 步骤3中采用Tensorflow深度学习平台对网络进行训练; 对步骤2中构建的快速轻量化超分辨率神经模型, 首先采用Xavier方式初始化超分辨率网络, 偏置全部初始化为0; 具体过程为:

1) 在超分辨率网络中采用Xavier方式初始化权重W后, W满足以下高斯分布:

$$W \sim G\left[0, \sqrt{\frac{2}{n}}\right] \quad (5)$$

其中, n表示该层网络输入单元数目, 即卷积层输入特征图数量;

2) 在整个网络中, 偏置全部初始化为0, 即 $b_i = 0$ 。

8. 根据权利要求7所述的一种手机拍照快速超分辨率处理方法, 其特征在于: 步骤3中使用Adam优化算法优化网络模型, 模型训练的损失函数采用均方误差表示, 如公式(6)所示:

$$L_1 = \min_{\Phi} \frac{1}{N} \|F(y^i, \Phi) - x^i\|^2 \quad (6)$$

其中, N为步骤1所得训练集中样本数量,  $x^i, y^i$ 分别为第i个高分辨率图像块和对应的低分辨率图像块; 在训练过程中, 通过指定迭代次数来进行网络参数更新。

9. 根据权利要求8所述的一种手机拍照快速超分辨率处理方法, 其特征在于: 步骤5的测试过程为: 将一幅低分辨率图像作为网络输入, 并利用步骤3学习的参数重建出一幅高分辨率图像作为输出; 步骤6中使用Tensorflow lite通用技术将模型移植到Android端。

## 一种手机拍照快速超分辨率处理方法

### 技术领域

[0001] 本发明属于图像处理技术领域,更具体地说,涉及一种手机拍照快速超分辨率处理方法。

### 背景技术

[0002] 现有比较常见的拍照后的图片清晰化方法,如采用ps对图片进行锐化处理等等,真正处理出来效果一般,随着深度学习在图像处理方面的应用的深入,图像超分辨率算法的不断发展,使得图像清晰化处理有了质的飞跃。作为计算机视觉领域中的经典问题,图像超分辨率(super resolution,SR)技术是指通过一幅或者多幅低分辨率图像或图像序列,恢复为较高分辨率图像的过程。该技术主要被用于对图像空间分辨率的增强,其可以突破原有的系统成像硬件条件的限制,重新获得的高分辨率图像具有更高的分辨率,更多的细节信息,更高质量的画质等特点,是目前获取高精度图像最有效、成本较低的途径之一。图像超分辨率技术从它被提出,直至发展到今天,有许多开发者对它进行研究,各种算法模型也不断产生,其无论在我们日常生活娱乐中还是在军事、天文、教育等方面都扮演着非常重要的角色。

[0003] 相较于低分辨率图像,高分辨率图像通常包含更大的像素密度、更丰富的纹理细节及更高的可信度。但是在实际处理过程中,由于受到采集设备与环境、网络传输介质与带宽、图像退化模型本身等诸多因素的影响,我们通常并不能直接得到具有边缘锐化、无成块模糊的理想高分辨率图像。而提升图像分辨率最直接的做法便是对采集系统中的光学硬件进行改进,但是,这种做法受制造工艺难以大幅改进、制造成本高昂等约束。因此,从软件和算法的角度着手,实现图像超分辨率重建的技术逐渐成为了图像处理和计算机视觉等多个领域的重点研究对象。

[0004] 基于深度学习的图像超分辨率算法体系结构中通常包含有巨大的参数和操作数,这些参数和操作数通常被认为是获得满意性能的关键要素,而过大的参数对于运算能力要求较高,通常这种图像超分辨率处理在计算机上较为常见,因为需要足够强的计算机硬件设施来支持相关的工作。由于移动端计算能力与速度的不足,手机端很少有深度学习相关的应用。然而,随着广泛使用手机拍摄和润色照片,手机端图像超分辨率也成为需要,这些包含了巨大的参数和操作数的算法体系结构大多体积都过于庞大或者要求的计算资源过多,占用内存较高、计算量大,可移植性相对较差,从而极大地阻碍了深度学习超分辨率模型在移动设备上的部署。

[0005] 经检索,申请号为:201810535634.5,申请日为:2018年05月29日,发明名称为:一种基于改进的稠密卷积神经网络的超分辨率重建方法。该申请案中将稠密卷积神经网络结构(Dense ConvolutionalNetwork,DenseNet)的思想应用到单帧图像的超分辨率重建,并在DenseNet结构的基础上改进了网络结构,减小了一定的参数,但此方法占用内存较高,计算量大,因此,不适合在一般的电脑或者移动端使用。

[0006] 又如,申请号为:201910460263.3,申请日为:2019年05月30日,发明名称为:一种



多前端深度图像超分辨率系统及其数据处理方法。该申请案公开的图像超分辨率系统及其数据处理方法，主要包括深度图像超分辨率、微信小程序前端、网页前端、Android客户端、服务器后台五个模块，深度图像超分辨率模块用于实现对深度图像进行分辨率放大操作，前端和客户端模块用于向后台提交图像，服务器后台模块将图像转交给深度图像超分辨率模块。Android客户端、Web网页或微信小程序构造发往服务器后台模块的HTTP请求并发送，开始等待处理响应；服务器后台模块接收到请求后，首先验证数据的有效性；若是有效数据，即有一张深度图像和一张彩色引导图像，后者的分辨率是前者的超分辨率尺度倍，则调用深度图像超分辨率模块进行处理，得到结果作为返回数据；若不是有效数据，则返回数据为对应的错误信息；随后服务器后台模块根据返回消息构造HTTP响应并发送回请求端。但是该申请案存在以下缺陷点：1. 必须是以向服务器发送请求HTTP获得相应响应这种方式，来获得后台的超分辨率处理，这种处理方式依赖网络流量及其性能，且耗时较久；2. 可移植性差，难以移植到移动设备终端进行使用。

[0007] 基于以上分析，现有技术需要一种能够更好进行快速轻量化的手机端图像超分辨率的方法。

## 发明内容

### [0008] 1. 要解决的问题

[0009] 本发明的目的在于克服传统深度学习模型可移植性相对较差，难以在移动端使用的不足，提供了一种手机拍照快速超分辨率处理方法。本发明提出一个新型的快速的轻量化的图像超分辨模型，并采用Tensorflow lite通用技术将模型应用在手机端，对拍摄后的图片进行裁剪，选择需要的部分，对其进行图像超分辨率处理达到部分放大清晰化的效果，模型运行速度在3-5s左右，从而为深度学习模型在Android等移动端的应用提供了广阔的应用前景。

### [0010] 2. 技术方案

[0011] 为了解决上述问题，本发明所采用的技术方案如下：

[0012] 本发明的一种手机拍照快速超分辨率处理方法，其步骤为：

[0013] 步骤1、利用常用图像处理数据集，制作高分辨率图像块与低分辨率图像块训练集；

[0014] 步骤2、构建一个快速轻量化的图像超分辨模型；

[0015] 步骤3、依据步骤1获得的训练集和步骤2构建的模型网络，进行网络训练，调整网络参数；

[0016] 步骤4、将训练好的模型保存为pb格式；

[0017] 步骤5、对步骤4所得训练模型进行测试；

[0018] 步骤6、将pb格式的模型移植到Android端，实现手机拍照后快速进行图像超分辨率处理。

[0019] 更进一步的，步骤1中使用DIV2K数据集制作高分辨率图像块与低分辨率图像块训练集。

[0020] 更进一步的，步骤1制作高分辨率图像块与低分辨率图像块训练集的过程为：

[0021] 对图像处理数据集中所有原始彩色图像进行裁剪，得到高分辨率图像块；然后对

高分辨率图像进行采样,得到与高分辨率图像对应的低分辨率图像,最后得到用于深度网络训练的高分辨率图像块与低分辨率图像块的集合。

[0022] 更进一步的,步骤2中构建的图像超分辨模型,由一个卷积后接32个残差单元,后接一个卷积,再接一个上采样单元,最后再接一个卷积构成;其中,每个残差单元使用反向残差,且反向残差块中使用的激活函数是ReLU6;网络输入为低分辨率图像块,输出为超分辨率图像块,该网络将学习低分辨率图像块和超分辨率图像块之间的映射关系,如公式(1)所示:

$$[0023] \quad y = F(x, \Phi) \quad (1)$$

[0024] 其中, $x, y$ 分别表示低分辨率图像块和超分辨率图像块, $\Phi$ 为学习到的模型参数,用于之后的高分辨率图像重建。

[0025] 更进一步的,步骤2中每个卷积层得到的输出用公式(2)表示:

$$[0026] \quad y_i = f(W_i x_i + b_i), i = 1, 2, \dots, 10 \quad (2)$$

[0027] 其中, $W_i$ 表示第*i*层的权重, $b_i$ 表示第*i*层的偏置, $x_i$ 表示第*i*层的输入, $y_i$ 表示第*i*层的输出。

[0028] ReLU6激活层的激活函数用公式(3)表示如下:

$$[0029] \quad f(x) = \min(\max(0, x), 6) \quad (3)$$

[0030] 其中, $x$ 为ReLU6激活函数的输入, $f(x)$ 为ReLU6激活函数的输出。

[0031] 更进一步的,步骤2中残差单元使用带线性瓶颈的反向残差;每个残差单元有三个卷积层,首先由一个1\*1的卷积接后激活函数ReLU6,然后接一个3\*3的Depthwise卷积后接激活函数ReLU6,最后接一个1\*1的卷积和非线性的激活函数;残差单元的表现如公式(4)所示:

$$[0032] \quad y_i = h(x_i) + F(x_i, w_i)$$

$$[0033] \quad x_{i+1} = f(y_i) \quad (4)$$

[0034] 其中, $x_i$ 表示为第*i*层的残差单元的特征输入, $w_i$ 表示为第*i*层权重和偏置项的设置, $F$ 代表的是残差函数, $f$ 则代表激活函数ReLU6, $h$ 是作为恒等式映射: $h(x_i) = x_i$ 。

[0035] 更进一步的,步骤3中采用Tensorflow深度学习平台对网络进行训练;对步骤2中构建的快速轻量化超分辨神经模型,首先采用Xavier方式初始化超分辨率网络,偏置全部初始化为0;具体过程为:

[0036] 1) 在超分辨网络中采用Xavier方式初始化权重 $W$ 后, $W$ 满足以下高斯分布:

$$[0037] \quad W \sim G\left[0, \sqrt{\frac{2}{n}}\right] \quad (5)$$

[0038] 其中, $n$ 表示该层网络输入单元数目,即卷积层输入特征图数量;

[0039] 2) 在整个网络中,偏置全部初始化为0,即 $b_i = 0$ 。

[0040] 更进一步的,步骤3中使用Adam优化算法优化网络模型,模型训练的损失函数采用均方误差表示,如公式(6)所示:

$$[0041] \quad L_1 = \min_{\Phi} \frac{1}{N} \|F(y^i, \Phi) - x^i\|^2 \quad (6)$$

[0042] 其中, $N$ 为步骤1所得训练集中样本数量, $x^i, y^i$ 分别为第*i*个高分辨率图像块和对应的低分辨率图像块;在训练过程中,通过指定迭代次数来进行网络参数更新。

[0043] 更进一步的,步骤5的测试过程为:将一幅低分辨率图像作为网络输入,并利用步骤3学习的参数重建出一幅高分辨率图像作为输出;步骤6中使用Tensorflow lite通用技术将模型移植到Android端。

[0044] 3.有益效果

[0045] 相比于现有技术,本发明的有益效果为:

[0046] (1) 本发明的一种手机拍照快速超分辨处理方法,通过构建一种快速轻量化的图像超分辨模型,并采用Tensorflow lite通用技术将模型应用在手机端,从而对手机拍摄后的图片进行裁剪,然后选择需要清晰化的部分,进行图像超分辨处理,以达到图像部分放大清晰化的效果。同时,本发明通过对图像超分辨模型的构建进行优化设计,尤其是残差单元使用了带线性瓶颈的反向残差,一方面能够有效保证对图片进行超分辨处理的效果;另一方面,能够大大减少图像超分辨模型的参数和操作数,有利于减小内存和计算量,因此,在确保其具有满意的性能基础上,也为模型在Android等移动端的应用提供了巨大的便利,使得图像超分辨率技术在移动端也能得到广泛应用。

[0047] (2) 本发明的一种手机拍照快速超分辨处理方法,通过使用Tensorflow lite通用技术将神经网络移植到移动Android端,通过先在高性能的机器上对模型训练、测试,然后将该模型转换为pb格式,通过Tensorflow在Android端的接口,最终使得本发明的模型成功加载到android端,从而将有助于提高传统深度神经网络的实用性,提高其研究的现实意义。

[0048] (3) 本发明的一种手机拍照快速超分辨处理方法,通过将所得模型保存为pb文件类型,一方面,它具有语言独立性,可独立运行,其封闭的序列化格式,使得任何语言都可以解析它;另一方面,其便于其他语言和深度学习框架读取、继续训练和迁移Tensorflow的模型。同时,将模型保存为pb文件类型时,模型的变量都会变成固定的,从而有利于进一步减小模型的大小,进而便于模型在手机端运行。此外,真正离线测试使用的时候,pb格式的数据能够保证数据不会更新变动,不会进行反馈调节。

[0049] (4) 本发明的一种手机拍照快速超分辨处理方法,对图像进行超分辨处理时,可以在较快的速度内达到较好的超分辨效果,由于减少了参数和操作数,模型的大小已经得到有效的减小,而更小的模型也会耗费更少的计算资源和加快计算速度,因此在Android端进行超分辨处理时,模型运行速度在3-5s左右,这为深度学习模型在Android等移动端的应用提供了广阔的应用前景。同时,将深度神经超分辨网络应用在移动端的方案均适用于其他深度学习模型,因此,在图像处理、语音识别及手势识别等方面均拥有广阔的发展前景。

[0050] (5) 本发明的一种手机拍照快速超分辨处理方法,通过运用深度神经模型对图像进行超分辨率操作来提高深度图像的分辨率,因此该系统不需要昂贵的深度摄像头就能实现拥有足够清晰度的高分辨率深度图像,这将有效改善手机拍照图像不清晰的问题,加快深度图像相关技术的发展。

## 附图说明

[0051] 图1是本发明的一种快速轻量化的图像超分辨模型架构图;

[0052] 图2是本发明的图像超分辨处理的界面流程图;

[0053] 图3是本发明中对手机拍照所得图片快速进行超分辨处理的流程图。



## 具体实施方式

[0054] 现有图像超分辨处理在计算机上较为常见,因为需要足够强的计算机硬件设施来支持相关的工作。但是由于移动端计算能力与速度的不足,移动端很少有深度学习相关的应用。然而,在Android和iOS设备上,Tensorflow Lite都提供了C++API的支持,并且在Android平台还额外提供了Java API的支持。这样一来,开发者就能非常方便得使用这些Tensorflow Lite API进行设计和开发。Tensorflow Lite具有的特点如下:(1)轻量级:允许小binary size和快速初始化/启动的设备端机器学习模型进行推断。(2)跨平台:运行时的设计使其可以在不同的平台上运行,如目前支持的android和iOS。(3)快速:专为移动设备进行优化,包括大幅提升模型加载时间,支持硬件加速。由于TensorFlow Lite通用技术的支持,因此可以通过在手机端部署Tensorflow的接口,从而达到在手机端进行图像超分辨的目的。

[0055] 但是,用于超分辨的深度学习体系结构通常包含巨大的参数,这被认为是获得满意性能的关键优势,而过大的参数对于运算能力要求较高。随着广泛使用手机拍摄和调色照片,手机端图像超分辨也成为需要,因此深度学习的过大参数这一特性极大地阻碍了超分辨模型在移动设备上的部署。为了解决传统深度学习模型难以应用在移动端的问题,我们需要构建一个快速的新型的轻量级模型。

[0056] 一般来说,为了建立一个准确快速的轻量级模型,需要一个设计良好的超参数的合适的模型体系结构。其中两个主要因素的关系:参数和操作数,要促进超分辨方法在移动设备上的应用,关键问题是减少模型体系结构的参数和操作数,从而保持满意的性能。因此,本发明提出了一个新的轻量化的超分辨模型,可以在参数和操作都有限的情况下,以较快的速度获得较好的性能。

[0057] 具体的,本发明中通过将模型保存为pb文件类型,pb文件具有语言独立性,可独立运行,封闭的序列化格式,因此任何语言都可以解析它,它允许其他语言和深度学习框架读取、继续训练和迁移Tensorflow的模型。它的主要使用场景是实现创建模型与使用模型的解耦,使得前向推导inference的代码统一。本发明中将模型保存为pb文件时,注意给模型中每个参数一个特定的名称,再保存,以便后续TensorFlow Lite的fetch()函数寻找。同时,保存为pb文件时,模型的变量都会变成固定的,从而进一步降低模型的大小,使其适合在手机端运行。并且,在真正离线测试使用时,pb格式的数据能够保证数据不会更新变动,即不会进行反馈调节。

[0058] 本发明通过手机拍照,将拍照后的图片模糊地方进行选择裁剪,然后采用深度学习图像超分辨方法,对剪裁后的图片进行超分辨处理,解决了手机拍摄图片部分不清晰的问题(例如远处的景物)。本发明通过构建一个新型轻量化的深度学习图像超分辨模型,然后利用Tensorflow lite通用技术将模型转化为手机端可用形式,将深度学习方法应用在android端,最终以较快的速度获得较好的图像超分辨性能。

[0059] 本发明将深度神经超分辨网络应用在移动端的方案均适用于其他深度学习模型,因此,在图像处理、语音识别及手势识别等方面均拥有广阔的发展前景。

[0060] 为进一步了解本发明的内容,结合附图和具体实施例对本发明作详细描述。

[0061] 实施例1

[0062] 结合图1,本实施例的一种手机拍照快速超分辨处理方法,其步骤为:



[0063] 步骤1、利用常用图像处理数据集，制作高分辨率图像块与低分辨率图像块训练集；具体过程为：

[0064] 首先，使用常规已有的DIV2K数据集，将常用图像处理数据集中原始每张彩色图像，首先以步长为27进行裁剪，最终得到100\*100大小的高分辨率图像块，接着对高分辨率图像进行4倍下采样，得到与高分辨率图像对应的25\*25大小的低分辨率图像；由此得到用于深度网络训练的高分辨率图像块与低分辨率图像块的集合。

[0065] 步骤2、构建一个快速轻量化的图像超分辨模型；

[0066] 2-1、构建的快速轻量化的超分辨网络模型，如图2所示，该模型是由一个卷积后接32个残差单元，后接一个卷积，再接一个上采样单元，最后再接一个卷积构成。每个残差单元是使用了带线性瓶颈的反向残差，反向残差块中使用的激活函数是ReLU6；网络输入为低分辨率图像块，输出为超分辨率图像块，该网络将学习低分辨率图像块和超分辨率图像块之间的映射关系，如公式(1)所示：

$$[0067] \quad y = F(x, \Phi) \quad (1)$$

[0068] 其中， $x$ 为低分辨率图像块， $y$ 为超分辨率图像块， $\Phi$ 为学习到的模型参数，用于之后的高分辨率图像重建。

[0069] 每个卷积层得到的输出用公式(2)表示：

$$[0070] \quad y_i = f(W_i x_i + b_i), i = 1, 2, \dots, 10 \quad (2)$$

[0071] 其中， $W_i$ 表示第 $i$ 层的权重， $b_i$ 表示第 $i$ 层的偏置， $x_i$ 表示第 $i$ 层的输入， $y_i$ 表示第 $i$ 层的输出

[0072] ReLU6激活层的激活函数用公式(3)表示如下：

$$[0073] \quad f(x) = \min(\max(0, x), 6) \quad (3)$$

[0074] 其中， $x$ 为ReLU6激活函数的输入， $f(x)$ 为ReLU6激活函数的输出。

[0075] 2-2、为了减少参数和操作数，采用了带线性瓶颈的反向残差。每个残差单元有三个卷积层，首先由一个1\*1的卷积接后激活函数ReLU6，之后接一个3\*3的常规已有的Depthwise卷积后接激活函数ReLU6，最后接一个1\*1的卷积和非线性的激活函数。

[0076] 由于中间的3x3卷积变为了Depthwise的了，计算量很少了，所以通道可以多一点，效果更好。因此，通过1x1卷积先提升通道数，再使用Depthwise的3x3空间卷积，再用1x1卷积降低维度。两端的通道数都很小，所以1x1卷积升通道或降通道计算量都并不大，而中间通道数虽然多，但是Depthwise的卷积计算量也不大。所以称之为带线性瓶颈的反向残差，两边窄中间宽，因此较小的计算量得到较好的性能。

[0077] 残差单元的表现如公式(4)所示：

$$[0078] \quad y_i = h(x_i) + F(x_i, w_i)$$

$$[0079] \quad x_{i+1} = f(y_i) \quad (4)$$

[0080] 其中， $x_i$ 表示为第 $i$ 层的残差单元的特征输入， $w_i$ 表示为第 $i$ 层权重和偏置项的设置， $F$ 代表的是残差函数， $f$ 则代表激活函数ReLU6， $h$ 是作为恒等式映射： $h(x_i) = x_i$ 。

[0081] 一般认为神经网络的每一层分别对应于提取不同层次的特征信息，有低层、中层和高层，而网络越深的时候，提取到的不同层次的信息会越多，而不同层次间的层次信息的组合也会越多。但是实验结果表明，在网络层数达到一定的深度之后，结果会变差，这就是退化问题。因此本实施例提出了采用残差学习来解决退化问题。对于一个堆积层结构(几层

堆积而成)当输入为 $x$ 时其学习到的特征记为 $H(x)$ ,现在我们希望其可以学习到残差 $F(x) = H(x) - x$ ,这样其实原始的学习特征是 $H(x)$ 。之所以这样是因为残差学习相比原始特征直接学习更容易。当残差为 $F(x) = 0$ 时,此时堆积层仅仅做了恒等映射,至少网络性能不会下降,实际上残差不会为0,这也会使得堆积层在输入特征基础上学习到新的特征,从而拥有更好的性能。

[0082] 步骤3、依据步骤1获得的训练集和步骤2构建的网络,进行网络训练,调整网络参数;

[0083] 3-1、利用Tensorflow深度学习平台对网络进行训练,对步骤2中构建的快速轻量化超分辨率神经模型,首先采用Xavier方式初始化超分辨率网络,偏置全部初始化为0。具体过程为:

[0084] 1) 在超分辨率网络中采用Xavier方式初始化权重 $W$ 后, $W$ 满足以下高斯分布:

$$[0085] \quad W \sim G\left[0, \sqrt{\frac{2}{n}}\right] \quad (5)$$

[0086] 其中, $n$ 表示该层网络输入单元数目,即卷积层输入特征图数量。

[0087] 2) 在整个网络中,偏置全部初始化为0,即 $b_i = 0$ 。

[0088] 3-2、使用通用的Adam优化算法来优化网络模型,模型训练的损失函数采用均方误差表示,如公式(6)所示:

$$[0089] \quad L_1 = \min_{\Phi} \frac{1}{N} \|F(y^i, \Phi) - x^i\|^2 \quad (6)$$

[0090]  $N$ 为步骤1所得训练集中样本数量, $x^i, y^i$ 分别为第 $i$ 个高分辨率图像块和对应的低分辨率图像块。

[0091] 在训练过程中,通过指定迭代次数来进行网络参数更新。

[0092] 步骤4、将训练好的模型保存为pb格式;

[0093] 4-1、将训练好的模型保存为pb格式,先通过传入CKPT模型的路径得到模型的图和变量数据;导入模型中的图;并从模型中恢复图中各个变量的数据;最后将模型持久化。

[0094] 4-2、训练结束后,利用学习到的模型参数进行测试。将一幅低分辨率图像作为网络输入,得到的输出就是超分辨率重建后的高分辨率图像。模型测试正常后,随后可以开启Android手机端的工作。

[0095] 步骤5、对步骤4所得训练模型进行测试,将一幅低分辨率图像作为网络输入,利用步骤3学习的参数重建出一幅高分辨率图像作为输出;

[0096] 步骤6、将pb格式的模型移植到Android端、实现手机拍照后快速进行图像超分辨率处理;

[0097] 在Android studio平台,首先将超分辨模型的接口在android端配置好以后,再将pb模型导入项目,最后编写代码完成一个软件,来实现手机端超分辨图片的效果。将之前训练好的模型应用在手机端,其所使用的技术就是Tensorflow Lite框架通用技术,是为将模型应用在手机端而进行操作而准备。

[0098] 结合图3,通过编写代码实现,打开系统相机,拍摄图片,并对拍摄后的图片,用户自主选择需要放大的部分进行剪裁;最后对裁剪后的部分图片,采用超分辨模型,进行图像放大操作,以较快速度得到清晰图片。

[0099] 在实际使用软件时,Android端,首先启动系统相机进程,点击拍摄按钮,使用者自主选定内容进行拍摄,然后对拍摄后的图片,以手指触碰的方式对裁剪框进行大小的控制,以框定选择需要放大清晰化的部分,最后将选定的图片送入后台,通过编辑代码,将输入的图片转化为RGB格式的数据,然后将数据feed给模型的输入节点,即送入pb格式的模型进行超分辨处理,最终获得输出节点的信息,将输出的数据进行格式的转换后,用ImageView控件接收输出的图片,并显示在界面上,最终得到高清的超分辨后的图片。

[0100] 值得注意的是,与以往常规的超分辨模型对比,例如EDSR模型,至少需要1-2分钟才能在手机上完成超分辨的过程并将图片显示出来。本实施例提出的新型的快速的轻量化的超分辨模型,仅需要3-5s就可以完成图片的超分辨过程,且效果较好。

[0101] 以上示意性的对本发明及其实施方式进行了描述,该描述没有限制性,附图所示的也只是本发明的实施方式之一,实际的结构并不局限于此。所以,如果本领域的普通技术人员受其启示,在不脱离本发明创造宗旨的情况下,不经创造性的设计出与该技术方案相似的结构方式及实施例,均应属于本发明的保护范围。



图1

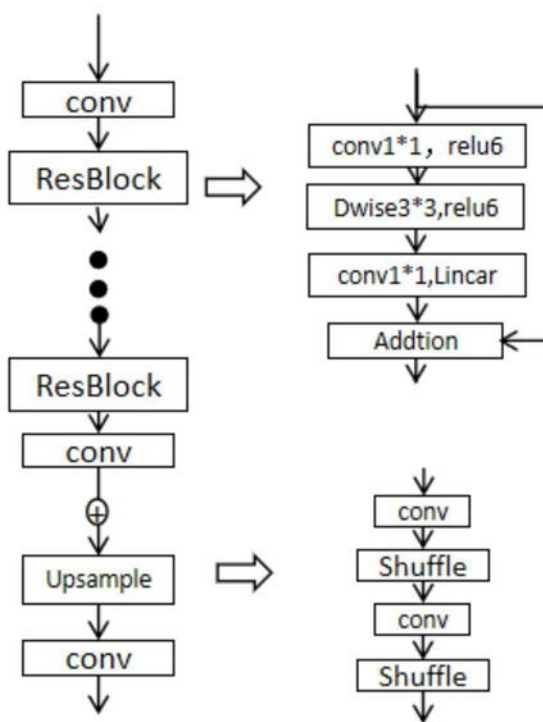


图2



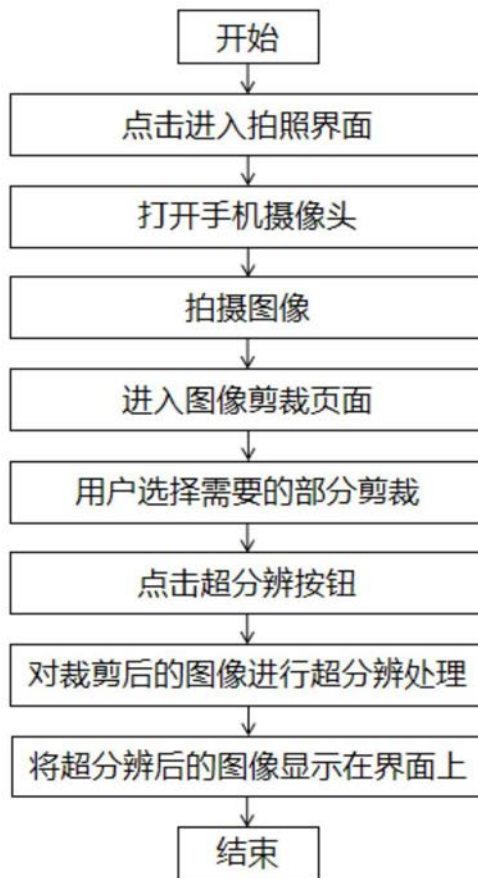


图3