

Dlib 在人脸识别技术中的运用

刘兆丰

(南宁高新区技术办, 广西南宁, 530007)

摘要: 随着人脸识别技术的应用推广, 越来越多的技术工具正在运用于人脸识别的创建, 作为构造人脸识别技术应用的工具之一, Dlib 正以它的实用性和通用性, 快速推广至各个领域的人脸识别, 基于上述背景, 本文主要介绍 Dlib 人脸对齐的三种函数, Dlib 在识别人脸过程中的原理和技术方法。

关键词: Dlib 人脸对齐; Dlib 人脸识别; 残差学习模块

DOI:10.16589/j.cnki.cn11-3571/tn.2020.21.015

1 Dlib 简介

Dlib 是一个基于 c++ 开发的开源数据工具库, 其中包含了不少的机器学习的成熟算法与模型, 相对于 tensorflow 和 PyTorch, 它用于图像处理以及人脸面部特征提取、分类及对比这几个方面比较具有通用性和优越性, 因此, **Dlib 正在越来越广泛地应用在人脸识别技术领域。**

Dlib 具有独立使用的可移植代码。Dlib 中的代码使用 c++ 语言进行开发而成, **使用独立封装, 在不借助第三方数据库的情况下, 可以直接移植到自己所需要设计的项目中进行使用。**

Dlib 拥有全面的文档说明。作为一个开源的人脸数据库训练集, **Dlib 中有很多功能齐全的程序和文件**, 从人性化的角度而言的, Dlib 在这一点上做的是非常不错的, 因为它为每一个程序文档和文件都做了相对应的注释, 这样开发者就可以迅速准确的调集程序文档来完成自己所需要的项目功能。

Dlib 涵盖了支持功能完备的深度学习以及图像处理的各类算法。Dlib 为开发者提供了机器深度学习的**各类成熟的完备算法, 并且在图像处理方面也带来了能够解决大多数实质问题的优良算法。**例如基于 SVM 的递归和分类算法, 以及专门用于面对大规模分类和递归的降维算法。当然还有能够对未知函数进行预分类和预测的相关向量机, 其分类和预测训练是基于贝叶斯框架。

Dlib 运用于人脸识别技术过程, 简单而言, 其主要过程分别是: 首先, 通过人脸检测获取目标人脸图像; 然后, 对目标人脸图像进行特征点检测; 接下来, 对完成特征点检测的人脸图像进行对齐处理, 使人脸图像变化成特征点形式图像, 并且将其对齐到基准人脸, 通俗地比喻就是摆正和校正人脸; 最后, 将特征点形式的人脸图像放入人脸数据训练集进行遍历对比, 完成识别人脸。对于以上过程, 在下文中, 着重地说明 Dlib 人脸对齐的基本函数、Dlib 完成识别人脸过程的基本原理与方法。

2 Dlib 人脸对齐函数

■ 2.1 extract_image_chip(img,dlib::get_face_chip_details(shape,150,0.25),face_chip)

在 `dlib::get_face_chip_details(shape, 150, 0.25)` 这一部分中, details 后的 shape 指的是我们所要提取的 N (68) 个人脸特征点; 补充说明一点, 原来的特征点形式人脸图像是 68 个特征点, 在 Dlib 中, Dlib 会将其处理成 51 个特征点。在使用对齐功能之后, 我们可以预测需要的人脸大小, 如果不满足实际需要操作者可自己调节, 这是 details 后第 2 个函数参数的意义。details 后的第 3 个函数参数 (可自己调节) 是人脸对齐区域的比例。Padding: Dlib 人脸数据训练集对所有一共 51 个人脸特征点进行对齐的过程, 编号从 0 到 50, 见图 1、图 2。



图 1 参数是 0.1 的对齐比例结果

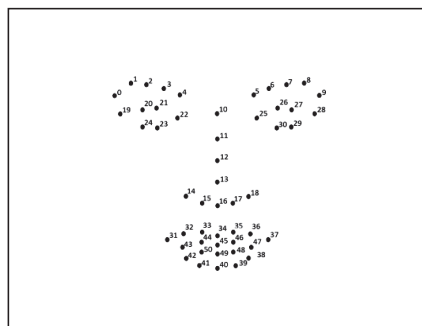


图 2 参数是 0.25 的对齐比例结果

将 details 后的第三个参数设置为 0.1, 则进行对齐过程, 对齐比例效果如图 1 所示。将 details 后的第三个参数设置

为 0.25, 则进行对齐过程, 对齐比例效果如图 2 所示。

通过对上面两幅图的对比分析可知, 这两幅图的实际整体大小都是 150×150 。但是设置 details 其后的第三个参数 0.25 且使用 padding 的人脸对齐过程, 明显发现图 1 的人脸面积占整幅图片面积的比例缩小, 也就是收敛, 最终使我们获得了一张涵盖更大面部区域的对齐图像如图 2 所示。

■ 2.2 get_face_chip_details(shape, 150, 0.25)

接下来使用该函数方法进行下面的计算过程。将上面已经变化好的收敛人脸图像进行取特征点的操作, 我们所取的特征点主要是分布在眉毛、双眼、鼻子以及嘴唇。在计算之前, 首先要记录好这些特征点的编号, 也就是上图 1、2 中的展示编号, 将这些编号作为参数, 用相应的函数计算出每个关键点的相对应的位置。接下来做完上面的工作之后, 利用下面中展示的相似变换公式 (1)、(2), 对每两个特征点计算它们之间的相似变换矩阵, 再由该矩阵, 我们可以得到目标人脸图像的人脸区域的 rect 坐标以及基于与所要对准的人脸图像之间的缩放系数旋转角度等重要基础数据。

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} s \cos \theta & -s \sin \theta & t_x \\ s \sin \theta & -s \cos \theta & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \quad (1)$$

$$x' = H_s x = \begin{pmatrix} sR & t \\ 0 & 1 \end{pmatrix} x \quad (2)$$

■ 2.3 voidextract_image_chip(constimage_type1&img, constchip_details&location, image_type2&chip, constinterpolation_type&interp)

根据上述的相似变换公式 (1)、(2), 使用该代码所表示的函数方法完成计算经过相似变化的矩阵的相对应位置, 再对原始图像做相对应的变化得到图像块, 上述这一步使用插值法完成处理。在函数中, 位置是 location, 原始图像是 img, 图像块是 chip, 插值法是 interp。

由上面所有的分析, 我们可以得到这么一个基本的结论, 通过对眼睛鼻子嘴唇的相似变换, 实现 Dlib 的人脸对齐。而且在这个对齐变化中, 不能够变化基础重要五官的近似几何形状。做到上述的要求就可以不错地完成人脸对齐的过程。

3 Dlib 在识别人脸过程中的技术应用

在 Dlib 中构造每一个能够进行人脸识别的人脸数据训练集都需要对提取的人脸面部特征数据进行对比分类识别, 而上述对比分类的过程都要在神经网络中完成, Dlib 中的 Resnet 就是承担这一工作, 接下来着重地介绍 Resnet。

Resnet 英文全称为 Residual Neural Network, 是一种基于残差学习思想的深度神经网络。如图 3 所示, 其中的 x

identity (Resnet 中全称为 identity mapping) 是指本身的输入变量, 通过 skip connection (图 3 中的旁路曲线) 到达比较点, 会在比较点与 residual mapping 的输出量进行比较计算, residual mapping 就是 skip connection (图 3 中的旁路曲线) 半包围的过程部分, 其输出量就是图中的 $F(x)$ 。上述 identity mapping 的变量与 residual mapping 的输出量最后比较计算, 结果得到图 3 中的 $F(x)+x$ 。由此就可获知, 残差其实指的是 $[F(x)+x]-x$, 实质就是图 3 中的 $F(x)$, 而整个图 3 展示的结构就是 Resnet 的残差学习模块。

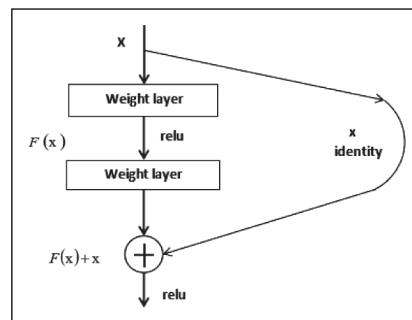


图3 残差学习模块结构

基于上述的残差学习模块, Resnet 可以很好地克服在过去的神经网络中出现的数据信息及变量在网络中传输损耗损失的问题, 并且能够确保梯度的完整性, 使用残差学习模块完成对输入变量的 skip connection 绕行输出或传递, 从而使得数据的完整度得到保障。而且, 残差学习模块使得整个 Resnet 神经网络几乎只需要针对输入与输出的不同部分以及出现的残差进行训练与学习, 这就降低神经网络的分析问题的复杂度。

Dlib 中的 Resnet 在实际应用时, 主要有两种类型的残差学习模块, 分别是如图 4 所示的由两个 $3 \times 3, 64$ 维卷积层拼接的模块, 以及由 $1 \times 1, 64$ 维、 $3 \times 3, 64$ 维、 $1 \times 1, 256$ 维这三个卷积层构成的模块。上述第一个模块是主要运用在 Resnet34 中, 第二个是主要运用在 Resnet50 以及拥有更多网络层数的 Resnet 中。第二个模块是第一个模块的进阶, 在第二个模块中, $3 \times 3, 64$ 维卷积层与 $1 \times 1, 64$ 维卷积层相互作用减少了各类数据参与计算的计算量, $3 \times 3, 64$ 维卷积层又与 $1 \times 1, 256$ 维卷积层相互作用完成对数据及参数的维度还原, 相对于第一个模块, 第二个模块在确保网络中的各类数据的分析与计算准确度的情况下, 通过减少数据计算量降低了问题分析运算的复杂度。

Resnet 可以以相对于其他的神经网络更快的速度完成对神经网络的训练, 并且在加深网络深度时, Resnet 可以确保网络中的数据以及计算稳定准确而且可以随着网络深

度的加深不断优化整个神经网络的准确率，因此，相比较于其它的神经网络，Resnet 在响应速度和准确度这两个方面更胜一筹。

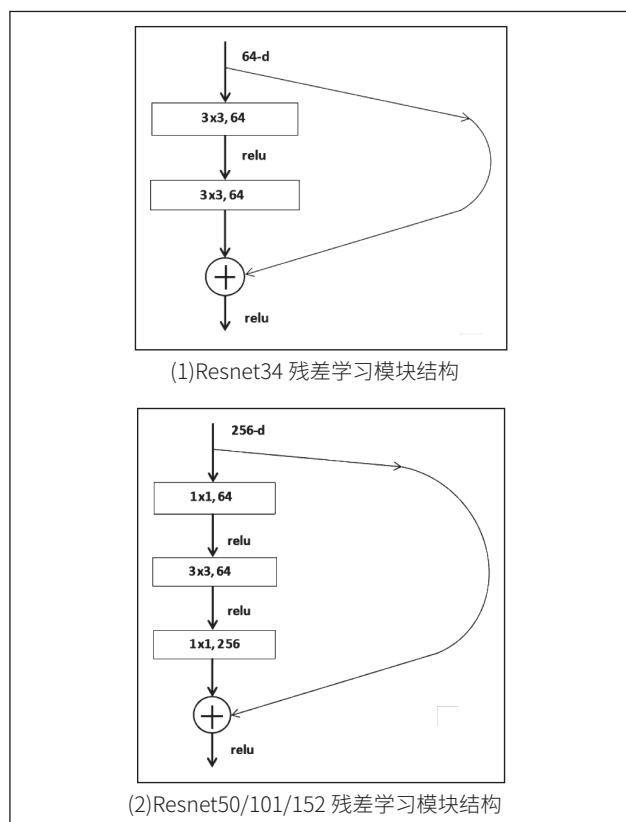


图 4 两种类型的残差学习模块

综上所述，正是基于上述的 Resnet 神经网络，Dlib 才能实用准确地完成人脸识别过程，因此，可以肯定地说，Resnet 是 Dlib 的核心关键部分。

在这里，以 Resnet34 说明 Resnet 在 Dlib 完成识别人脸过程中的作用以及 Dlib 完成最后识别人脸的过程。在此之前，会向 Dlib 中添加与需要识别的人脸图像相同或不同的一定数量人脸图像，这一定数量的人脸图像在提取人脸面部特征数据后，所有的人脸面部特征数据都会在 Resnet34 完成遍历对比分类训练，Dlib 中就能产生一个成熟的人脸数据训练集。需要识别的人脸图像在经过人脸检测、特征点检测以及人脸对齐之后，接下来就为这个人脸图像创造一个 128 维的向量，使用 Resnet34 创造上述的向量。下面这一段代码概括地说明了 Resnet34。

```
template<typename SUBNET> using level1 = res<512,
res<512, res_down<512, SUBNET>>>;
```

```
template<typename SUBNET> using level2 =
res<256, res<256, res<256, res<256, res<256, res_
own<256, SUBNET>>>>>;
```

```
template<typename SUBNET> using level3 = res<128,
res<128, res<128, res_down<128, SUBNET>>>>;
```

```
template<typename SUBNET> using level4 = res<64, r
es<64, res<64, SUBNET>>>>;
```

下面的图 5 会更清晰的介绍这种 Resnet34。

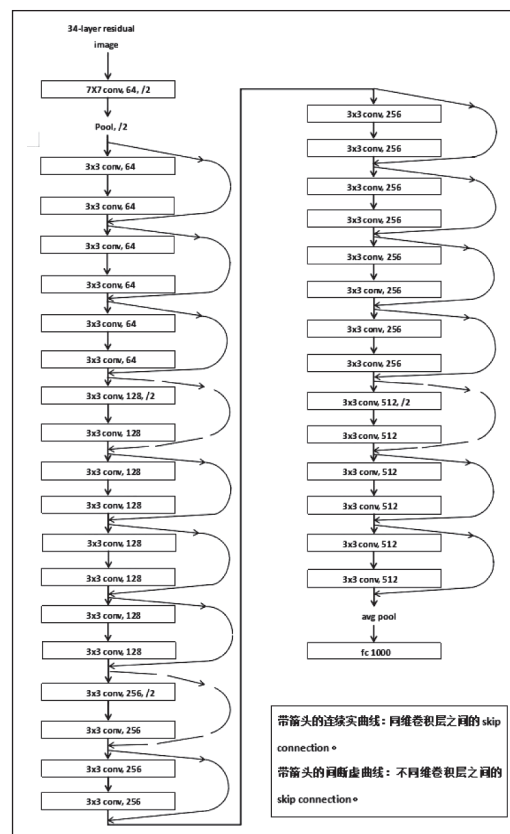


图 5 Resnet34 神经网络的结构

由图 5 我们可以清楚地看到，在 resnet34 的最后一层，拥有 1000 个神经元，在这 1000 个神经元之后再追加一个 Dense(128)，即可达到使用 Resnet 为目标人脸图像创建一个 128 维向量的目的。接下来，在人脸数据训练集中，计算需要识别的人脸图像的 128 维向量与数据库中已有的不同或相同人脸图像的 128 维向量它们的欧式距离，Dlib 用百分比形式在人脸数据训练集中设置一个相似程度，当计算出来的一定数量的欧式距离超出设定阈值（在 Dlib 的训练分类人脸面部特征时已完成设定）且该数量占欧式距离总数量的百分比大于相似程度时，Dlib 的人脸数据训练集或者就是 Dlib 就能认定它们之中的某一些人脸图像是同一张人脸，完成人脸识别。

4 结语

虽然 Dlib 已经在人脸识别技术的很多方面应用，但是随着人脸识别技术的继续发展和开拓，在之后一定会有包涵

（下转第 7 页）

www.ele169.com | 41

下延迟一个周期,此种做法能保证数据的同步。其次,输入并例化所需的三行数据,通过移位存储后,即可得到想要的 3×3 矩阵了。

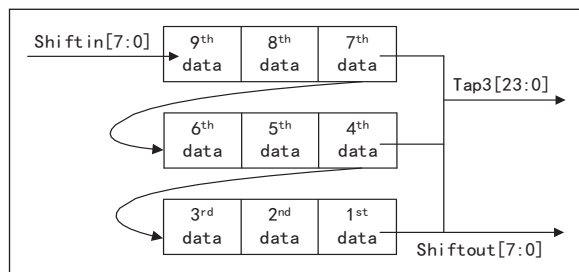


图3 移位存储器原理

2.3 实现 Sobel 边界确定算法的步骤

为了加快图像算法的效率,体现 FPGA 并行处理图像数据的优点,可以将索贝尔边界确定算法的实现过程分成以下步骤:

- (1) 输入原始 RGB 图像 P;
- (2) 将原始 RGB 图像转换为灰度值,从 RGB 到灰度转换的标准公式为 $Y=0.299R+0.587G+0.114B$;
- (3) 计算原始灰度图像与 Sobel 算子 X 方向的卷积,即 $G_x=P * \text{Sobel}_X$;
- (4) 计算原始灰度图像与 Sobel 算子 Y 方向的卷积,即 $G_y=P * \text{Sobel}_Y$;
- (5) 计算梯度的大小,即 $G=\sqrt{G_x^2+G_y^2}$;
- (6) 根据精度要求与输入阈值比较,从而决定像素边缘点,最终确定边界。

根据以上的步骤,最后检测所得的模型仿真波形如图4所示。

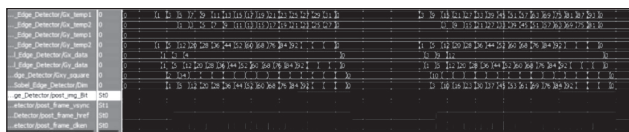


图4 索贝尔边界检测的波形图

3 板级验证与调试

经过仿真验证之后,将生成的 JIC 固件下载到 Alientek

(上接第41页)

更优良算法和技术方法的工具,更好地实现人脸识别在各个领域的构建与运用。所以,人脸识别技术在整个人工智能领域的地位依然十分重要,未来可期。

参考文献

- * [1] 寿翌辰. 基于机器视觉技术的排队人数识别系统[J]. 电子制

Pionner 开发板中进行板级测试,得到的实验结果如图5所示。

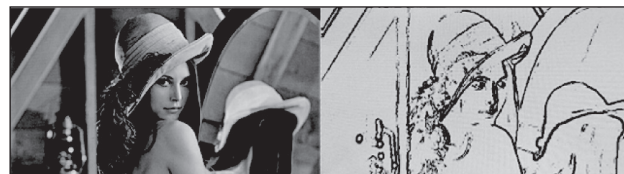


图5 原始图与算法实现效果的对比

通过以上的仿真结果与板级验证得到的图像来看,证实了在 FPGA 平台能够高效稳定地完成算法的加速功能。

4 结论

边缘检测在数字图像处理领域扮演重要角色,是诸多重要应用的基础。本文以边缘检测算法的硬件加速为背景,实现了使用 FPGA 完成 Sobel 边缘检测算法的加速。实验结果表明,与同类设计相比,利用 FPGA 的流水线和并行处理的结构优势对图像进行边界锁定的效率比使用传统软件程序处理提升了9倍左右,可以大大减少重复性工作,提升开发效率^[7]。

参考文献

- * [1] 温海滨. 计算机图像处理技术在选矿在线粒度检测中的应用[D]. 昆明理工大学,2005.
- * [2] 林彬彬,邱新法,何永健,朱晓晨,张阳. 茶树病害智能诊断识别算法研究[J]. 江苏农业科学,2019,47(06):85-91.
- * [3] 李强,张娟. 一种改进的基于模板匹配的污损车牌识别方法[J]. 智能计算机与应用,2019,9(03):113-117.
- * [4] 胡秀敏,何志琴. 无人驾驶汽车中的车道线检测研究[J]. 新型工业化,2018,8(12):57-60.
- * [5] 卢静,李华. 家庭视频监控系统中异常情况的检测研究[J]. 河南工程学院学报(自然科学版),2015,27(04):73-76.
- * [6] 张皓. 基于FPGA的图像处理加速器研究[D]. 武汉大学,2015.
- * [7] 郑玉. 基于HLS的Sobel边缘检测算法的FPGA实现[D]. 北京工业大学,2017.
- * [8] 金大超. 基于FPGA的实时监控系统和边缘检测的研究与实现[D]. 天津理工大学,2017.

作. 2020, 6(12): 6-8.

- * [2] 李国防, 阙中强. 智慧人脸识别系统中间件的实现和优化[J]. 电脑知识与技术. 2019, 12(36): 185-186.
- * [3] 周光朕, 杜姗姗, 冯瑞. 基于残差量化卷积神经网络的人脸识别方法[J]. 计算机系统应用. 2018, 8(08): 21-22.