

高加瑞, 杜洪波, 蔡锦, 等. 基于 MediaPipe 的手势识别算法研究与应用[J]. 智能计算机与应用, 2024, 14(4): 157-161. DOI: 10.20169/j.issn.2095-2163.240424

基于 MediaPipe 的手势识别算法研究与应用

高加瑞, 杜洪波, 蔡锦, 李佳翰

(沈阳工业大学 理学院, 沈阳 110870)

摘要: 随着人机交互的飞速发展, 手势识别技术逐渐引起研究人员的关注。然而复杂的环境会对手势识别的准确性产生很大的影响, 本文旨在探索一种基于 MediaPipe 的高精度手势识别算法, 算法采用 MediaPipe 的手部关键点检测模块, 对手部姿态进行检测和识别, 利用本文的手势识别算法对手势进行分类识别。通过实验证明, 该算法可以有效地克服光照变化、背景干扰和手部遮挡等问题, 并可同时对多只手进行识别, 提高了手势识别算法的准确性和鲁棒性。

关键词: MediaPipe; 手势识别; OpenCV; 手势跟踪; 人机交互

中图分类号: TP301 **文献标志码:** A **文章编号:** 2095-2163(2024)04-0157-05

Research and application of gesture recognition algorithm based on MediaPipe

GAO Jiarui, DU Hongbo, CAI Jin, LI Jiahao

(School of Science, Shenyang University of Technology, Shenyang 110870, China)

Abstract: With the rapid development of human-computer interaction, gesture recognition technology has gradually attracted the attention of researchers. However, complex environments can significantly affect the accuracy of gesture recognition. This paper aims to explore a high-precision gesture recognition algorithm based on MediaPipe. The hand pose is detected and recognized using MediaPipe's hand keypoint detection module. The gesture recognition algorithm proposed in this paper is employed for gesture classification and identification. Through experiments, when compared with traditional gesture recognition algorithms, this algorithm effectively overcomes issues such as lighting variations, background interference, and hand occlusions. Additionally, it can simultaneously recognize multiple hands, thereby enhancing the accuracy and robustness of the gesture recognition algorithm.

Key words: MediaPipe; gesture recognition; OpenCV; gesture tracking; human-computer interaction

0 引言

随着人机交互技术的不断发展, 手势识别作为其中重要的一种方式, 逐渐引起了人们的关注。手势识别可以通过人体姿态和动作来实现对计算机的控制和交互, 具有广泛的应用前景。如: 虚拟现实^[1]、车辆安全^[2]、医疗康复^[3]等方面。然而, 手势识别算法面临着很多的挑战, 例如光照变化、背景干扰、手部遮挡等因素, 都会影响算法的准确性和鲁棒性。

近年来, 已有很多研究者针对手势识别算法进行了探索和研究, 如传统的图像处理方法、深度学习算法等。随着 Google 开源项目 MediaPipe 的推出,

MediaPipe 是一个用于处理视频、音频等时间序列数据构建机器学习管道的工具包。基于 MediaPipe 的手势识别算法也逐渐成为了研究的热点之一。

本文基于 MediaPipe 工具包, 旨在研究一种可以有效克服光照变化、背景干扰和手部遮挡问题^[4]的手势识别算法, 从而提高手势识别的准确性和鲁棒性。

1 相关工作

手势识别作为一种重要的人机交互方式, 已经受到了广泛的关注。在过去的几十年里, 已经有很多研究者对手势识别算法进行了探索和研究, 其中较有代表性的研究成果包括传统的图像处理方法和

基金项目: 辽宁省教育厅高等学校基本科研项目的(LJKZ0157); 辽宁省大学生创新训练计划项目(S202210142057)。

作者简介: 高加瑞(2003-), 男, 本科生, 主要研究方向: 机器学习与计算机视觉; 蔡锦(2002-), 男, 本科生, 主要研究方向: 机器学习与计算机视觉; 李佳翰(2002-), 女, 本科生, 主要研究方向: 机器学习与计算机视觉。

通讯作者: 杜洪波(1977-), 男, 硕士, 副教授, 硕士生导师, 主要研究方向: 数据挖掘理论与应用, 深度学习优化与识别。Email: 1833435353@qq.com

收稿日期: 2023-08-07

深度学习算法等。

传统的图像处理方法主要包括特征提取和分类器两个步骤^[5]。在特征提取方面,通常采用手工设计的特征,如颜色、纹理和形状等;而在分类器方面,通常采用 SVM^[6]、KNN^[7] 等传统的分类器。然而,这种方法的局限性在于特征的设计和选择方面存在一定的主观性和局限性,而且对于复杂的手势识别任务,这种方法的准确率很难得到保证。

基于深度学习的方法是近年来非常流行的一种手势识别方法。通过使用深度神经网络进行特征提取和分类^[8],可以避免手工设计特征的问题,并且能够自动学习更具有区分性的特征。目前常用的深度学习模型包括卷积神经网络(CNN)^[9]、循环神经网络(RNN)^[10]和注意力机制(Attention)^[11]等。虽然这种方法在手势识别方面取得了很好的效果,但是需要大量的数据和计算资源,并且对于复杂手势的处理仍然存在挑战。

本文的主要创新在于使用 MediaPipe 工具包中的手部关键点检测模块对手部动作进行检测和识别,获取手部的关键节点信息,然后利用该算法构建的分类器对手势进行分类。通过实验,验证了其有效性,并对本文算法的优缺点进行了分析。

2 模型建立

2.1 MediaPipe 模型

MediaPipe 是一款非常流行的机器学习框架,其提供了一系列预训练的神经网络模型,其中包括了手部关键节点检测算法,算法中包含的 21 个手部关键节点如图 1 所示。利用 MediaPipe 手部关键节点检测模型,来构建本文算法中的手部检测模型、手势识别算法以及手势库。

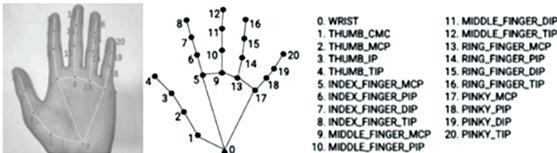


图 1 手部的 21 个关键结点

Fig. 1 The 21 key points of the hand

2.2 获取待检测信息

首先,需要获取待检测的手部图像信息。在实际应用中,这些图像信息可以通过多种途径获取,例如从摄像头、图像库或视频流中获取。不同的获取途径需要不同的接口和处理方式,但都需要将图像信息转化为适合模型处理的数据格式。在 MediaPipe 中,可以使用 ImageFrame 类来处理图像

数据,其可以将常见的图像格式转化为 TensorFlow 可以处理的格式,获取待检测信息的流程如图 2 所示。循环读取摄像头输入视频流的每一帧,经过处理后在窗口中显示,并检查窗口状态以决定是否继续循环,最终释放摄像头资源。

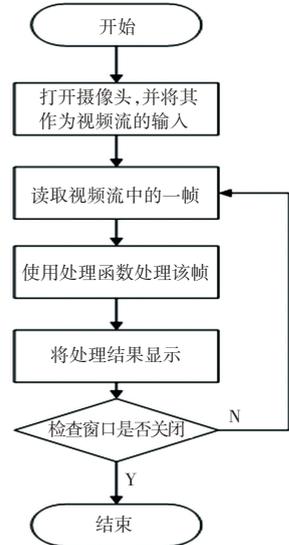


图 2 获取待检测信息

Fig. 2 Obtaining detection information

2.3 构建手部检测模型

MediaPipe 手部姿态估计模型的核心是一个卷积神经网络(CNN)模型,其可以对手部图像进行特征提取和分类,从而推断出手部的姿态信息。为了实现手势识别,需要在手部姿态估计模型的基础上构建一个手部检测模型。该模型可由 MediaPipe 模型中的 MediaPipe Hands 模块直接实现。该模型在手掌检测过程中平均准确率可达 95.7%,而使用常规方法^[12]识别的准确率仅为 86.22%。

2.4 手势识别算法的建立

在手部检测模型的基础上,可以进一步实现手势识别算法。手势识别算法是一种分类算法,可以将检测到的手部图像区域识别为不同的手势类别,例如识别简单的数字,以及简单的手势等。根据获得的输入信息中被检测出的手的集合(集合中包含手的 21 个关键节点坐标)。通过节点的坐标,构建手部的特征向量,通过特征向量之间的夹角关系^[13],判断当前手掌表示的动作。

算法 1 构建手部角度列表

输入 2.3 小节输出的手的集合

输出 包含所有手指角度信息的列表

1 Function *vetorialAngle*

2 *angle_list* = [] # 存放手指角度信息

```

3  DEF vectorAngle(v1, v2):
4      # 计算向量夹角
5      v1_x, v1_y ← v1[0], v1[1]
6      v2_x, v2_y ← v2[0], v2[1]
7      angle ← arccos(cos < v1, v2 >)
8      RETURN angle
9  END vectorAngle
10  angleIndex = vectorangle(
11  # 计算食指的角度, 其余类似
12  # 计算出代表食指坐标的向量, 将其输入
    ((int(handSet[0][0]) - int(hand_[6][0])),
    (int(hand_[0][1]) - int(hand_[6][1])),
13  ((int(hand_[7][0]) - int(hand_[8][0])),
    (int(hand_[7][1]) - int(hand_[8][1])))
14  angle_list.append(angleIndex)
15  RETURN anagle_list
16  END vectorial Angle
    
```

算法 2 构建手势库

输入 算法 1 输出的角度列表 anagle-list
 输出 识别出的手势信息

```

1  Function gesture
2      OherFingerAngle ← 65.
3      thumbAngle ← 53.
4      openAngle ← 49.
5      gesture ← "Unknown"
6      IF anagle-list NOT NULL
7  IF (angle_list[0] > thumbAngle) and (angle_list[1] >
    OherFingerAngle) and (angle_list[2] >
    OherFingerAngle) and (angle_list[3] >
    OherFingerAngle) and (angle_list[4] >
    OherFingerAngle):
8          gesture_str = "0"
9      #根据经验与手指角度相结合设计出相关
    动作, 复杂动作需考虑关节的弯曲角度
10  ELIF (angle_list[0] > thumbAngle) and (angle_
    list[1] < OherFingerAngle) and (angle_list[2] >
    OherFingerAngle) and (angle_list[3] >
    OherFingerAngle) and (angle_list[4] >
    OherFingerAngle):
11          gesture_str = "1"
12  END IF
13  END IF
14  RETURN gesture
15  END gesture
    
```

算法 3 手势识别算法

```

输入 手的集合
输出 手势识别出的语义字符串 gesture
1  frame = getInformation() # 2.2 小节
2  WHILE TRUE
3      frame = cv2.cvtColor(frame, cv2.COLOR_
    BGR2RGB)
4      hands ← HandGesture() #2.3 小节
5      frame = cv2.cvtColor(frame, cv2.COLOR_
    RGB2BGR)
6      IF hands NOT NULL
7          绘制手部节点以及关节点的连线
8      ENDIF
9      anagle ← vectorialAngle(hands) #算法 1
10     gestureStr ← gesture(anagle) # 算法 2
11  ENDWHILE
12  END
    
```

算法 3 结合算法 1、2, 实现了手势动作的识别, 识别的流程如图 3 所示。首先获得视频输入流, 识别出视频中的一帧画面, 使用 2.3 小节中的手部检测模型检测手部的关键节点信息, 接着使用算法一计算手指角度的信息, 输入到算法二构建的手势库中, 根据手势库中的信息将识别结果作为输出返回。

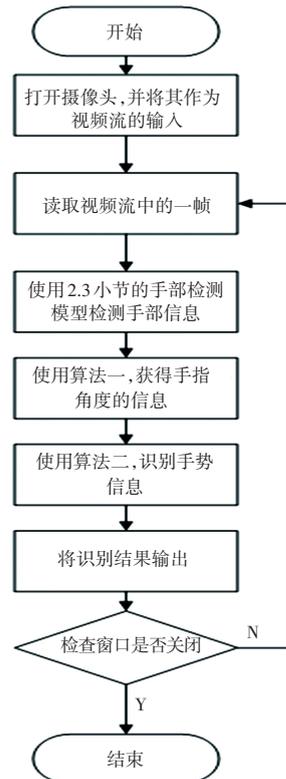


图 3 手势识别流程

Fig. 3 Gesture recognition process

3 基于 MediaPipe 手势识别系统的设计与实现

3.1 PyQt5 框架介绍

Qt 是一个用 C++ 编写的框架^[14]。其提供了在网络、数据库、OpenGL、Web 技术等领域的跨平台开发的模块,其中 API 可以访问移动端与现代化的桌面。PyQt5 是对于 Qt5 版本全方面的 Python 跨平台语言实现。在 Python 中实现了超过 35 个扩展模块,每个 PyQt 控件都对应一个 Qt 控件,使所有支持 Python 的平台都可以开发如 C++ 平台相同的应用程序,设计软件方便、准确、快捷。基于此框架开发的手势识别系统,实现了算法的可视化。

3.2 系统结构

基于本文提出的算法设计系统,实现对手势动作的简单识别功能以及可视化的人机交互,该系统实现了图像手势识别、视频手势识别、摄像头下的实时手势识别,以及手势跟踪、手势控制鼠标、手势控制音量等功能。

系统按照功能进行分区,采用核心代码与系统 UI 界面分离的方式进行开发,并按照以下功能进行分区构建:

- (1) mainWindows: 负责控制主窗口的运行,以及与其他窗口的连接;
- (2) pictureHandWindows: 负责图像中的手部检测功能,以及功能界面的建立;
- (3) pictureWindows: 负责批量图像人体识别功能,以及 UI 界面的建立;
- (4) videoHandwindows: 负责对视频中的手部检测,以及 UI 界面的建立;
- (5) videoWindows: 负责对视频中的人体行为做出检测,以及功能界面的建立;
- (6) shiBieDaiMa: 软件包中手势识别部分的核心代码,用于匹配手势。

3.3 图像手势识别模块

图像手势识别的主要流程,首先使用 openCV 模块从本地导入需要检测的图片,由于 openCV 导入的图片为 BGR 格式,而可视化部分需要使用 RGB 格式,故需要将 BGR 格式的图片转换为 RGB 格式^[15]。接着创建手部检测模型,见算法 1。该模型可以准确的识别出手掌的整体边缘,以及手掌的关键节点信息,具有较大的覆盖范围以及比较高的识别准确性,可适用于不同场景中不同大小信息部分缺失的手势识别。手势识别界面以及识别的结果如图 4 所示,界面中左侧展示原始的图象,右侧展示处

理后的图象结果。



图 4 图像手势识别界面

Fig. 4 Image gesture recognition interface

3.4 视频手势识别模块

视频中的手势识别,就是将视频转化为对图象的处理。首先对视频进行分帧,使用 openCV 中的方法 CV2.VideoCapture() 获取视频总帧数,之后对每一帧视频按照处理图象的方法进行处理,最后将视频帧进行拼接,获取检测后的视频,视频检测界面如图 5 所示,图中左侧展示原始视频,右侧展示处理后的视频,两个窗口画面同步播放。



图 5 视频手势识别界面

Fig. 5 Video gesture recognition interface

4 实验结果分析

4.1 实验环境

本系统的开发软件为 PyCharm,基于 Python3.8 版本搭建。Python 中需安装 PyQt5、OS、OpenCV、MediaPipe-tools、Autopy、Matplotlib、Math。实验所用的平台为实验室的 DELL 工作站,其处理器为 Intel(R) Xeon(R) Silver 4210 CPU @ 2.20 GHz 2.19 GHz, GPU 是两块 NVIDIA GeForce RTX 3070,机带 RAM 32.0 GB (31.7 GB 可用)。

4.2 手势识别准确性实验

为验证该算法的准确性,在强光环境、背景较复杂场景,以及手部被遮挡的场景下,进行三组测试实验。实验参数设置最小置信度为 0.75,最小追踪阈值为 0.75;每组实验对各种手势分别进行 100 次实验,总计 1 000 次。

在强光环境下,识别的准确率见表 1,在背景较为复杂的场景(如有其他无关人员)识别准确率见表 2,在手部被遮挡的场景下,识别的准确率见表 3。

表1 强光环境识别结果

Table 1 Strong lighting conditions

手势类别	样本数目	正确数目	错误数目	识别率/%
0	100	92	8	92.0
1	100	96	4	96.0
2	100	98	2	98.0
3	100	93	7	93.0
4	100	95	5	95.0
5	100	97	3	97.0
6	100	1	9	91.0
8	100	90	10	90.0
Bye	100	88	12	88.0
ok	100	83	17	83.0
总计	1 000	923	77	92.3

表2 复杂场景识别结果

Table 2 Complex scenes

手势类别	样本数目	正确数目	错误数目	识别率/%
0	100	93	7	93.0
1	100	95	5	95.0
2	100	96	4	96.0
3	100	91	9	91.0
4	100	97	3	97.0
5	100	94	6	94.0
6	100	93	7	93.0
8	100	93	7	93.0
Bye	100	90	10	90.0
ok	100	86	14	86.0
总计	1 000	928	72	92.8

表3 手部被遮挡识别结果

Table 3 Hand occlusion status

手势类别	样本数目	正确数目	错误数目	识别率/%
0	100	90	10	90.0
1	100	93	7	93.0
2	100	94	6	94.0
3	100	91	9	91.0
4	100	89	11	89.0
5	100	90	10	90.0
6	100	88	12	88.0
8	100	89	11	89.0
Bye	100	87	13	87.0
ok	100	81	19	81.0
总计	1 000	892	109	89.2

4.3 结果分析

通过实验表明,在强光环境下算法识别准确率为92.3%,在复杂场景下识别的准确率为92.8%,在手部分被遮挡的情况下识别的准确率为89.2%,算法的平均准确率为91.43%,识别的准确率比较高,具有一定的稳定性,但对于较为复杂的手势识别的准确率较低,有待提高。

5 结束语

本文基于 MediaPipe 框架提出了一种手势识别算法。通过实验表明,该算法可以有效地克服光照变化、背景干扰和手部遮挡等问题,提高了手势识别算法的准确性和鲁棒性。根据此算法设计出手势识别系统,进行可视化设计,可移植性高。虽然改进的算法识别准确率有一定的提高,但仍然存在识别不准确的场景,后续将结合深度学习、扩散模型等算法作为下一步的研究方向。

参考文献

- [1] LIAO Ting. Application of gesture recognition based on spatiotemporal graph convolution network in virtual reality interaction[J]. Journal of Cases on Information Technology (JCIT), 2022, 24(5): 79-83.
- [2] 陈义,马云林. 基于视觉的手势识别技术在车载主机上的应用[J]. 电子设计工程, 2016, 24(8): 141-144.
- [3] 钱涛. 基于 Kinect 的动态姿势识别方法在医疗康复中的应用[D]. 杭州:浙江工业大学, 2020.
- [4] 任海兵,祝远新,徐光祐,等. 复杂背景下的手势分割与识别[J]. 自动化学报, 2002(2): 256-261.
- [5] 黄凯奇,任伟强,谭铁牛. 图像物体分类与检测算法综述[J]. 计算机学报, 2014, 37(6): 1225-1240.
- [6] 丁世飞,齐丙娟,谭红艳. 支持向量机理论与算法研究综述[J]. 电子科技大学学报, 2011, 40(1): 2-10.
- [7] 桑应宾. 基于 K 近邻的分类算法研究[D]. 重庆:重庆大学, 2009.
- [8] 沈雅婷. 基于深度学习的手势识别研究[J]. 软件导刊, 2019, 18(11): 25-29.
- [9] 周楠. 基于卷积神经网络的手势识别算法研究[D]. 桂林:桂林理工大学, 2022.
- [10] 杨丽,吴雨茜,王俊丽,等. 循环神经网络研究综述[J]. 计算机应用, 2018, 38(S2): 1-6, 26.
- [11] 朱张莉,饶元,吴渊,等. 注意力机制在深度学习中的研究进展[J]. 中文信息学报, 2019, 33(6): 1-11.
- [12] 刘德发. 基于 MediaPipe 的数字手势识别[J]. 电子制作, 2022, 30(14): 55-57.
- [13] 范嘉义. 基于骨架信息的人体动作识别[D]. 合肥:中国科学技术大学, 2017.
- [14] 陶文玲,侯冬青. PyQt5 与 Qt 设计师在 GUI 开发中的应用[J]. 湖南邮电职业技术学院学报, 2020, 19(1): 19-21.
- [15] 赵健,张冬泉. 基于 OpenCV 的数字手势识别算法[J]. 计算机应用, 2013, 33(S2): 193-196.