

实战 | 用Python和OpenCV搭建一个老人跌倒智能监测系统 (步骤 + 源码)

原创 Alwaz Sheikh OpenCV与AI深度学习 2024-01-12 07:46 重庆

点击下方**卡片**，关注“**OpenCV与AI深度学习**”

视觉/图像重磅干货，第一时间送达!



OpenCV与AI深度学习

专注计算机视觉、深度学习和人工智能领域干货、应用、行业资讯的分享交流!

257篇原创内容

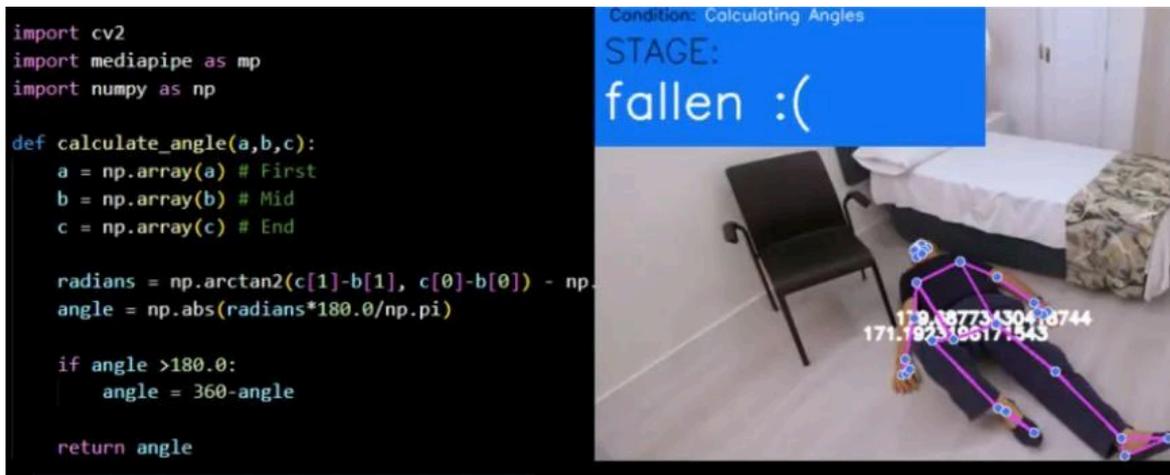
公众号

导读

本文将使用Python、OpenCV和MediaPipe搭建一个老人跌倒智能监测系统。（公众号：OpenCV与AI深度学习）

背景介绍

老人监测系统是一种智能检测系统，可以检测老人是否躺在床上或是否跌倒在地。这是一个解决实际问题的程序，可用于在您外出工作或外出时监控家中的老人，以便在出现任何问题时通知您。



实现步骤

【1】导入必要的模块：

在python中导入Numpy、MediaPipe和opencv

```
1 import cv2
2 import mediapipe as mp
3 import numpy as np
```

OpenCV与AI深度学习

【2】定义一个计算角度的函数：

由于我们将根据我们使用 OpenCV 获得的角度和坐标来假设一个人是在行走还是跌倒在地上，因此，我们必须计算角度，最简单的方法是定义一个函数，然后调用它在程序中。

```
def calculate_angle(a,b,c):
    a = np.array(a) # First
    b = np.array(b) # Mid
    c = np.array(c) # End

    radians = np.arctan2(c[1]-b[1], c[0]-b[0]) - np.arctan2(a[1]-b[1], a[0]-b[0])
    angle = np.abs(radians*180.0/np.pi)

    if angle >180.0:
        angle = 360-angle

    return angle
```

OpenCV与AI深度学习

【3】查找坐标：

我们还必须找到坐标，以便我们可以在条件下使用它们，也可以将它们与 calculate_angle 函数一起使用。

```
# Get coordinates
left_eye = [landmarks[mp_pose.PoseLandmark.LEFT_EYE.value].x,landmarks[mp_pose.PoseLandmark.LEFT_EYE.value].y]
left_hip= [landmarks[mp_pose.PoseLandmark.LEFT_HIP.value].x,landmarks[mp_pose.PoseLandmark.LEFT_HIP.value].y]
left_heel = [landmarks[mp_pose.PoseLandmark.LEFT_HEEL.value].x,landmarks[mp_pose.PoseLandmark.LEFT_HEEL.value].y]
right_eye = [landmarks[mp_pose.PoseLandmark.RIGHT_EYE.value].x,landmarks[mp_pose.PoseLandmark.RIGHT_EYE.value].y]
right_hip = [landmarks[mp_pose.PoseLandmark.RIGHT_HIP.value].x,landmarks[mp_pose.PoseLandmark.RIGHT_HIP.value].y]
right_heel=[landmarks[mp_pose.PoseLandmark.RIGHT_HEEL.value].x,landmarks[mp_pose.PoseLandmark.RIGHT_HEEL.value].y]
right_index=[landmarks[mp_pose.PoseLandmark.RIGHT_INDEX.value].x,landmarks[mp_pose.PoseLandmark.RIGHT_INDEX.value].y]
left_index = [landmarks[mp_pose.PoseLandmark.LEFT_INDEX.value].x,landmarks[mp_pose.PoseLandmark.LEFT_INDEX.value].y]
```

OpenCV与AI深度学习

【4】如何知道主体（老人）是安全的还是跌倒的？

我们将借助从 cv2 和 mediapipe 获得的坐标以及使用上述定义的函数获得的角度来找到这一点。

由于我们正在获取眼睛臀部和脚踝的坐标，因此我们知道当一个人平躺（倒下）时，他的眼睛、臀部和脚踝之间的角度在 170 到 180 度的范围内。因此，我们可以简单地提出一个条件，当角度在 170 -180 度之间时，我们可以说一个人摔倒了。

```
if angle1 != angle2 and (angle1>170 and angle2>170):
    if (((right_index[0]<0.70 and right_index[0]>0.20) and (right_index[1]<0.56 and
right_index[1]>0.15)) or ((left_index[0]<0.55 and left_index[0]>0.18) and
(left_index[1]<0.56 and left_index[1]>0.15))):
        stage="Hanging on !!"

    else:
        stage = "fallen :("

elif angle1 != angle2 and (angle1<140 or angle2<140) :
    stage = "Trying to Walk"
elif angle1!=angle2 and ((angle1<168 and angle1>140) and (angle2<168 and angle2>140)):
    stage="Barely Walking"
else:
    pass
```

OpenCV与AI深度学习

现在你的脑海中可能会出现一个问题，即如何确定这个人是否真的摔倒了，或者他是否只是躺在床上，因为在这两种情况下，角度都在相同的范围内。

我们也会回答它，所以请继续阅读:)

【5】如何区分床和地板？

我们将再次使用从 OpenCV 获得的坐标，然后使用它来找到床的坐标，然后在检查跌倒条件时引入一个新条件，即当受试者的坐标与床坐标一致时，这意味着一个人在床上时自然是安全的。这种情况将排除跌倒的情况，程序将显示安全。只有当此条件变为假时，才会检查跌倒条件和其他步行和尝试步行条件。

```
if ((left_eye[0]>=0.41 and left_eye[0]<=0.43) and (left_hip[0]>=0.44 and left_hip[0]<=0.46) and
(left_heel[0]>=0.41 and left_heel[0]<=0.43) or (right_eye[0]>=0.41 and right_eye[0]<=0.43) and
(right_hip[0]<=0.43 and right_hip[0]>=0.41) and (right_heel[0]>=0.37 and right_heel[0]<=0.39)):

    if ((left_eye[1]>=0.24 and left_eye[1]<=0.33) and (left_hip[1]<=0.35 and left_hip[1]>=0.45) and
(left_heel[1]<=0.74 and left_heel[1]>=0.72) or (right_eye[1]<=0.30 and right_eye[1]>=0.24) and
(right_hip[1]<=0.50 and right_hip[1]>=0.32) and (right_heel[1]>=0.37 and right_heel[1]<=0.39)):
        stage = "safe :)"
```

因此，通过引入带有床坐标的单一条件，我们也解决了上述问题。

【6】让我们在屏幕上打印结果：

现在打印摔倒和安全等的结果；我们可以很容易地使用 cv2 中的 putText 函数来显示保存在变量 stage 中的文本。

该函数的示例用法如下所示：

```
1 cv2.putText(image, 'Condition: ', (15,12), cv2.FONT_HERSHEY_SIMPLEX, 0.5, (0,0,0))
```

【7】添加图形用户界面：

我们还可以添加一点 GUI 来进一步增加整个程序的外观并使其更加用户友好。实现最简单 GUI 的代码片段示例如下：

```
root = Tk()
root.geometry("1920x1080+0+0")
root.state("zoomed")
root.config(bg="#3a3b3c")
root.title("Eldering Monitring")

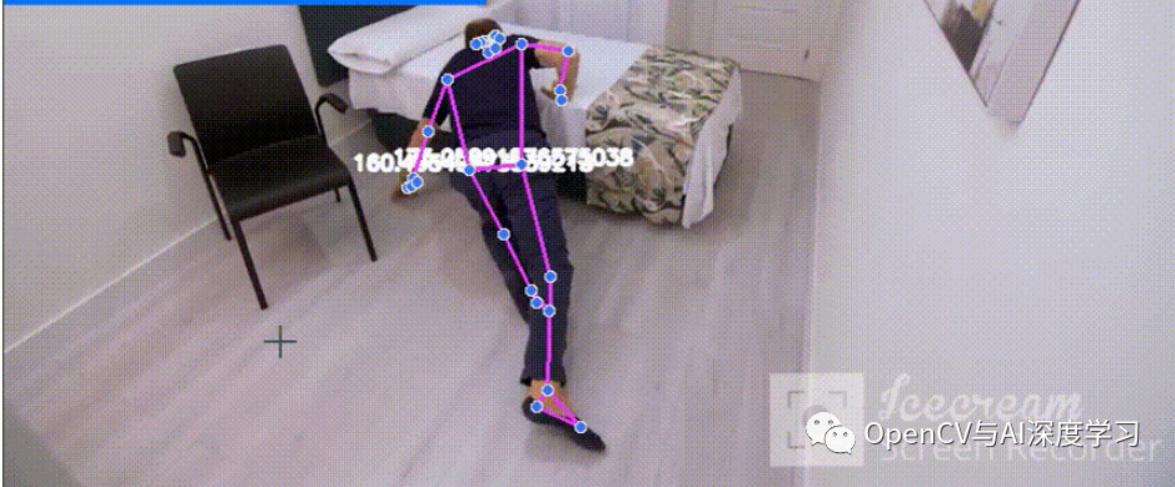
def path_select():
    global video_path,cap
    video_path = filedialog.askopenfilename()
    cap = cv2.VideoCapture(video_path)
    text = Label(root,text="Recorded Video ",bg="#3a3b3c",fg="#ffffff",font=("Calibri",20))
    text.place(x=250,y=150)
# For Live feed
def video_live():
    global video_path,cap
    video_path = 0
    cap = cv2.VideoCapture(video_path)
    text = Label(root,text="Live Video Feed",bg="#3a3b3c",fg="#ffffff",font=("Calibri",20))
    text.place(x=250,y=150)

live_btn = Button(root, height =1,text='LIVE', width=8, fg='magenta', font=("Calibri", 14, "bold"), command=lambda:vid
live_btn.place(x=1200,y=20)
text = Label(root,text=" For Live Video",bg="#3a3b3c",fg="#ffffff",font=("Calibri",20))
text.place(x=1000,y=30)
```

最终效果：

Condition: Calculating Angles

STAGE:
Trying to Walk



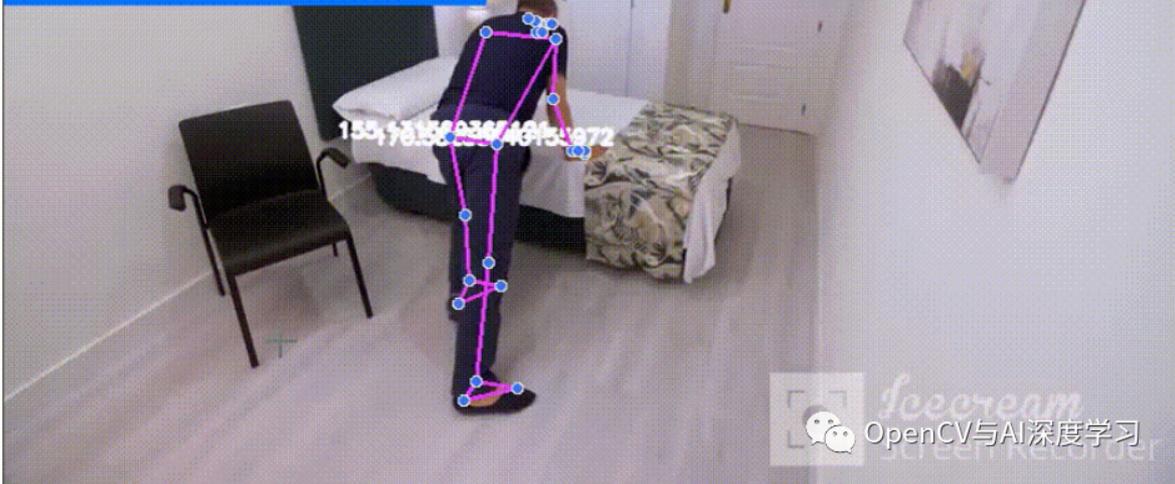
160.17 159.98 159.14 136.27 150.38



foccrenm
OpenCV与AI深度学习
screen recorder

Condition: Calculating Angles

STAGE:
Hanging on !!



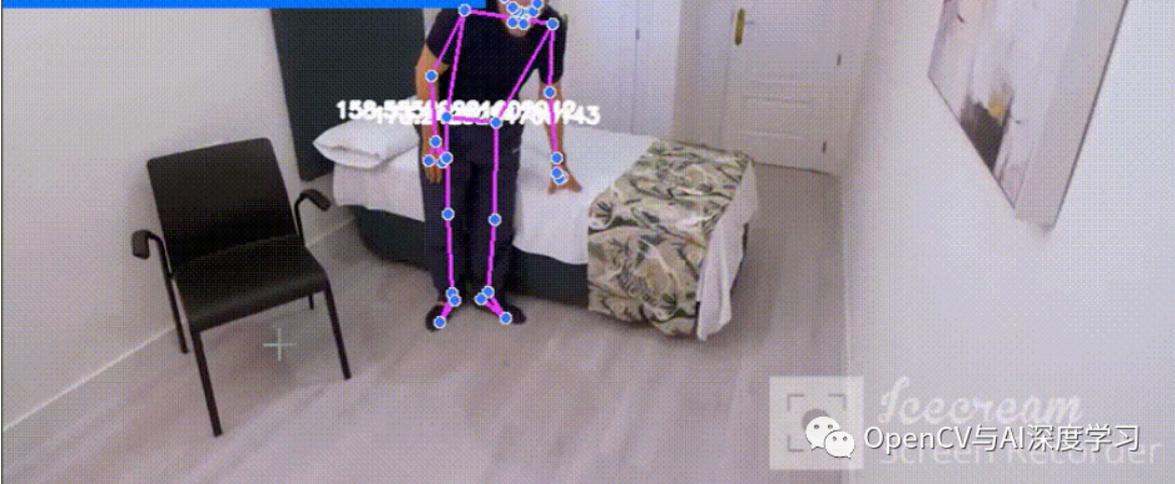
155.17 158.23 146.15 149.72



foccrenm
OpenCV与AI深度学习
screen recorder

Condition: Calculating Angles

STAGE:
Barely Walking



158.55 152.14 149.14 143



foccrenm
OpenCV与AI深度学习
screen recorder



完整代码下载链接：

<https://github.com/Alwaz-Sheikh/Alwaz>

—THE END—

下载1: Pytorch常用函数手册

在「**OpenCV与AI深度学习**」公众号后台回复：**Pytorch函数手册**，即可下载学习全网第一份Pytorch函数常用手册，包括**Tensors介绍**、**基础函数介绍**、**数据处理函数**、**优化函数**、**CUDA编程**、**多处理**等十四章内容。

下载2: 145个OpenCV实例应用代码

在「**OpenCV与AI深度学习**」公众号后台回复：**OpenCV145**，即可下载学习145个OpenCV实例应用代码（**Python和C++双语言实现**）。

CV相关内容讨论交流欢迎加入微信交流群！



MediaPipe 3 计算机视觉 402 深度学习 327 OpenCV 178 Python 30

MediaPipe · 目录

上一篇

实战 | 用Python和OpenCV搭建一个老人跌倒智能监测系统 (步骤 + 源码)

下一篇

基于手势识别的周边应用实例展示与实现教程 (含视频教程+源码)

喜欢此内容的人还喜欢

OpenCV实现基于边缘的模板匹配--适用部分遮挡和光照变化情形(附源码)

OpenCV与AI深度学习



基于OpenCV实现模糊检测 / 自动对焦

OpenCV与AI深度学习



用Python和OpenCV搭建自己的一维码和QRCode扫描仪 (步骤 + 源码)

OpenCV与AI深度学习

