

基于人工智能技术的电脑用眼疲劳监测与提醒系统

作品设计思路：

目前，电脑使用量急剧增加，人眨眼的目的就是不停地在眼表形成泪膜保护眼睛。正常情况下人每分钟眨眼 15-20 次，眨眼次数长时间保持在较低水平会给人眼造成生理上的伤害。温州医科大学附属眼视光医院杭州院区的试验结果显示：当眨眼次数减少为正常次数的两三倍时，泪膜破裂时间与干眼症患者相当。生活中白领、程序员、玩游戏的青少年等人群往往会由于各种原因不自觉的减少眨眼次数，长时间睁眼注视电脑，造成眼表的泪膜蒸发，角膜干燥，视力下降、人眼疲劳恶性循环等不良影响，容易形成角膜炎。人工智能技术主要通过模仿和拓展人的智能来开发智能系统，实现系统的智能行为，为人类解决各类复杂问题提供便利。

基于上述背景，我们团队将通过 python3.7 环境开发一套基于人工智能技术的电脑用眼疲劳监测与提醒系统，利用人工智能技术中的机器视觉与模式识别，根据人眼生理上正常睁一次眼所持续的时间进行睁眼延迟实时监测和智能化弹窗提醒，为上述人群提供在长时间使用电脑时提供实时睁眼延迟监测与智能提醒，防止因为长时间注视电脑不眨眼睛而造成的泪膜蒸发、角膜干燥、视力下降、人眼疲劳恶性循环等不良影响，督促他们养成健康的用眼习惯。

团队成员介绍和工作分工说明：

队长：杨汉

山东大学电气工程学院本科生；

科创经验丰富，拥有优秀的逻辑处理能力；（团队设计 logo）
具有优秀的组织和统筹能力，并对 AI 的技术革新充满热情与期待；熟练使用 MATLAB, Python 语言。



队伍成员：

程仁乾：山东大学电气工程学院本科生；优秀的信息检索以及文本语言编辑能力，团队间良好的配合是一个优秀团队的基础；熟练掌握硬件组配。

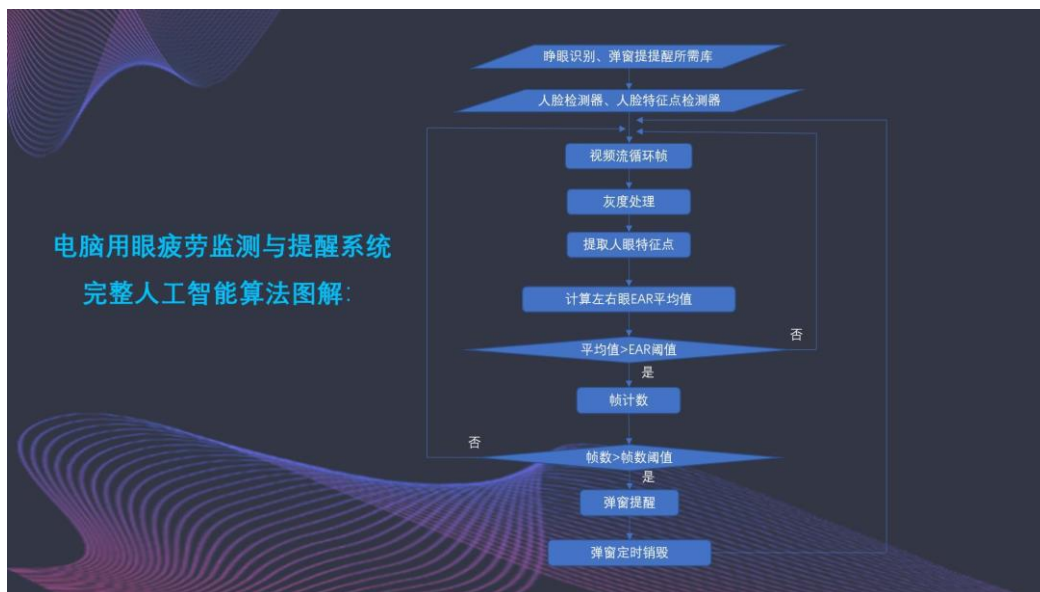
田云鹏：山东大学控制科学与工程学院本科生；熟练掌握 MATLAB、Python、C 语言等，充分融入团队，负责代码优化。

李哲：山东大学电气工程学院本科生；熟练各方面硬件输出，拥有各种科创活动的经验，负责团队项目完善。

我们团队具有全方面的能力，富有创新精神以及团队合作精神，具有市场调查的能力，熟练掌握 python 语言和其他计算机语言的编写，对人工智能技术有着浓厚兴趣，富有责任心并且动手能力强，坚持理论与实际相结合。

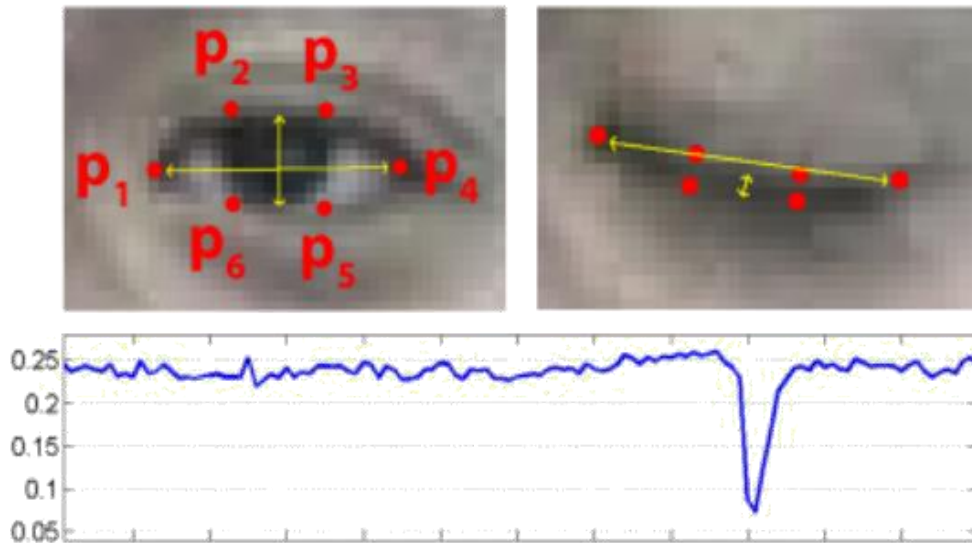
团队产品人工智能算法设计

基于 python3.7 环境下的电脑桌面眼疲劳检测与提醒实现



电脑用眼疲劳监测与提醒 Python 设计算法

人脸有 68 个特征点，其中左眼和右眼分别对应了 6 个特征点，每只眼睛由 6 个 (x, y) 坐标表示，从眼睛的左角开始，然后围绕该区域的其余部分顺时针显示：p1, p2, p3, p4, p5, p6 六个特征点，眼睛的长宽比在眼睛张开的时候大致是恒定的，但是在发生闭眼时会迅速下降到零，引入眼睛纵横比 (EAR) 的概念，由 Soukupová 和 Čech 在其 2016 年的论文“使用面部标志实时眼睛眨眼检测”中介绍。由于垂直眼睛的特征点有两组，而眼睛水平方向的特征点只有一组，所以通过加权，得到眼睛纵横比 (EAR) 的计算公式：



人眼纵横比（EAR）在不同状态下的变化

$$EAR = ((|p2 - p6|) + (|p3 - p5|)) / (2 * (|p1 - p4|))$$

基于人工智能技术的电脑用眼疲劳监测与提醒系统基于 python3.7 环境进行开发，能够在视频流中实时监测人每次睁眼所持续的连续帧数，每一帧所用时间是相同的，通过监测每次睁眼所持续的连续帧数就能间接的监测睁眼时间的长短，当电脑使用者每次睁眼持续时间过长，系统将进行弹窗提醒。

```

Python 3.7.0 Shell
File Edit Shell Debug Options Window Help
FishTEAR = 0.30337600935550613
LeftEAR = 0.32668153786023735
RightEAR = 0.3166240622999997
LeftEAR = 0.30566135663068732
RightEAR = 0.2954867420612471
LeftEAR = 0.321245154968971
RightEAR = 0.327920094904366273
LeftEAR = 0.32249030993194194
RightEAR = 0.30337600935550613
LeftEAR = 0.35
RightEAR = 0.25063051794381147
LeftEAR = 0.270974579204999
RightEAR = 0.23152512446070806
LeftEAR = 0.2703975365587607
RightEAR = 0.25063051794381147
LeftEAR = 0.3100858364730211
RightEAR = 0.2909032840790431
LeftEAR = 0.25925925925925924
RightEAR = 0.23302705699402712
LeftEAR = 0.3030977360722638
RightEAR = 0.212152964970497
LeftEAR = 0.13784910325911582
RightEAR = 0.13443739878693848
LeftEAR = 0.1802473185742597
RightEAR = 0.16712236430100124
LeftEAR = 0.1517149297799906
RightEAR = 0.17006687036131
LeftEAR = 0.216694430303705
RightEAR = 0.2068034255720007
LeftEAR = 0.2693643984530026
RightEAR = 0.2693967943213487
LeftEAR = 0.25
RightEAR = 0.25110897307471086
LeftEAR = 0.308889720826443
RightEAR = 0.250838214910045
LeftEAR = 0.3269230769230769
RightEAR = 0.2960952540790431

```

实时记录的 EAR 值



团队产品人工智能程序测试效果图

```
if ear < EYE_AR_THRESH:  
    pass  
else:  
    frame_counter += 1  
    if frame_counter >= EYE_AR_CONSEC_FRAMES:  
        tk.messagebox.showwarning('警告', '你已经睁眼较长时间, 请及时进行眼部休息')  
        frame_counter = 0  
    else:  
        pass
```

当视频流中实时监测的左右眼平均 EAR 超过我们设定的 EAR 阈值时，系统将判断为睁眼，帧数从 0 开始连续计数，

如果睁眼持续的帧数已经超过设定的帧数阈值，则系统将判断为睁眼延迟，此时将进行弹窗提醒。



提醒窗口设定时间自动进行销毁，不影响电脑的正常使用

在 python3 环境  中导入睁眼监测和弹窗提醒所需要的各种库：numpy, cv2, dlib, scipy.spatial, os, imutils, tkinter,time 。 OpenCV 定义 eye_aspect_ratio 函数，用来返

回 EAR 的值。eye_aspect_ratio 函数接受单一的参数，即给定的眼睛面部标志的 (x, y) 坐标 。计算两组垂直眼睛标志之间的距离之和作为分子，计算水平眼睛标志之间的距离的两倍作为分母。将分子和分母相结合，得出最终的眼睛纵横比。然后将眼睛纵横比返回给调用函数。

shape_predictor_68_face_landmarks.dat 为 68 个人脸特征点模型文件，导入人脸检测器和人脸特征点检测器后将调用 68 个人脸特征点模型文件。由于 list 从 0 开始，为保持一致，将人脸特征点中对应于眼睛的那几个特征点序号减一。

在 EAR 的实时监测中，需要设定 EAR 阈值，和连续睁眼所持续的帧数阈值，并将每次睁眼所进行的帧计数赋初值为 0。

通过 while 循环在视频流中循环帧，将每一帧读取并进行灰度处理，通过 dlib 内置的人脸检测器检测灰度帧中的人脸，并进行面部标志检测。分别取出左眼和右眼的特征点，计算左右眼的 EAR 并取平均值。每一帧对应于一段相同的时间，通过监测睁眼所持续的帧计数的次数就能间接的判定睁眼时间长短。当人眼长时间盯着电脑，会不自觉的发生睁眼延迟，损伤眼睛造成视力下降。

当视频流中实时监测的左右眼平均 EAR 超过我们设定的 EAR 阈值时，

系统将判断为睁眼，然后帧数从 0 开始连续计数，如果睁眼持续的帧数已经超过设定的帧数阈值，则系统将判断为不正常的睁眼状态，即睁眼已经发生延迟。此时系统将进行弹窗提醒“你已经长时间睁眼，请及时进行眼部休息”。两次判断通过 if 语句嵌套进行实现。弹窗提醒窗口将设置在电脑右下角，并在指定时间自动进行销毁，不影响电脑的正常使用。

发行：

通过 pyinstaller 将 python 程序与 68 个人脸特征点模型打包为 exe 可执行文件进行发布，实现在任何一台普通电脑上都可以方便的获取和使用本套系统。