Документација за систем за детекција на поспаност

Оваа документација дава детално објаснување на **системот за детекција на дремливост** имплементиран со Python, OpenCV и MediaPipe. Програмата користи фацијални маркери за да детектира поспаност преку следење на односот на аспектот на окото (EAR) и дава визуелна повратна информација на лицето на корисникот. Подолу е детално објаснување на кодот, неговите карактеристики и како функционира.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Содржина:**

**1. Преглед**

**2. Зависности**

**3. Објаснување на кодот**

* + 3.1. Пресметка на односот на аспектот на окото (EAR)
  + 3.2. Иницијализација на MediaPipe Face Mesh
  + 3.3. Иницијализација на веб-камерата
  + 3.4. Логика за детекција на дремливост
  + 3.5. Визуелизација на фацијалните маркери
  + 3.6. Главна јамка и обработка на фрејмови
  + 3.7. Услов за излез

**4. Како функционира програмата**

**5. Прилагодување и подобрувања**

**6. Заклучок**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

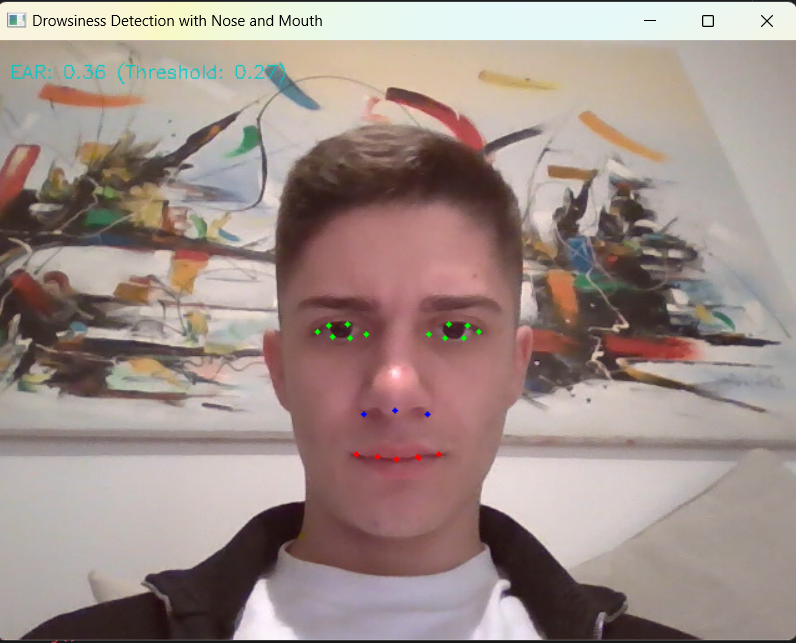
## **1. **Преглед****

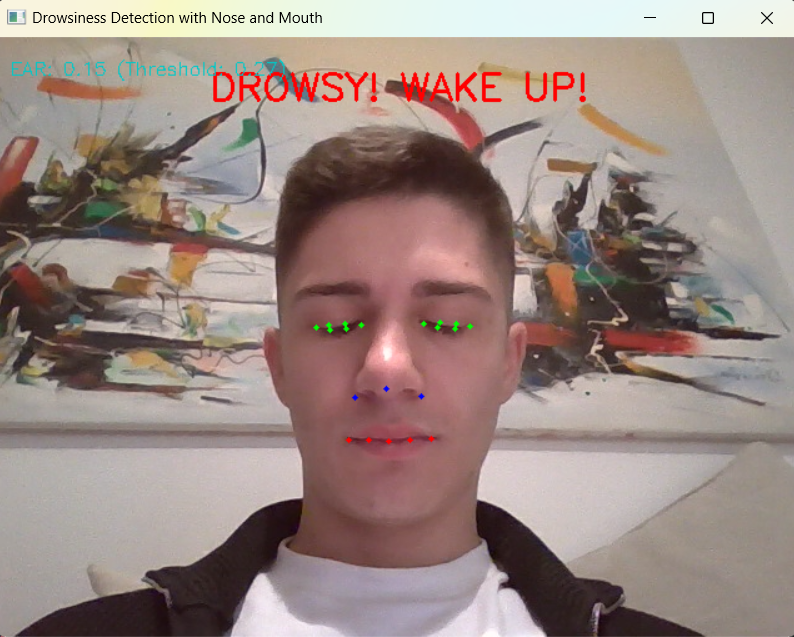
Системот за детекција на поспаност е дизајниран да ги следи очите на корисникот во реално време со користење на веб-камера. Тој го пресметува **аспектот на окото (EAR)** за да одреди дали очите на корисникот се затворени или отворени. Ако очите останат затворени за одреден број на последователни фрејмови, системот активира аларм за дремливост. Дополнително, програмата ги визуелизира фацијалните маркери (очи, нос и уста) на видео-излезот за подобро разбирање и дебагирање.

Системот користи напредна технологија за компјутерска визија преку MediaPipe Face Mesh, која овозможува прецизна детекција на 468 точки на лицето во реално време. Оваа технологија е особено корисна бидејќи обезбедува стабилна детекција дури и при различни услови на осветлување и различни агли на главата. За секоја рамка од видео стримот, системот ги анализира релативните позиции на специфични точки околу очите за да пресмета EAR вредност.

EAR метриката е клучна компонента на системот, базирана на геометриски односи помеѓу вертикалните и хоризонталните растојанија на точките околу окото. Кога окото е отворено, EAR вредноста е типично околу 0.3, додека при затворено око таа паѓа на околу 0.2. Системот користи праг од 0.27 за да направи разлика помеѓу отворени и затворени очи, при што последователни рамки под овој праг активираат предупредување за поспаност.

За да се избегнат лажни аларми предизвикани од нормално трепкање, системот бара очите да бидат затворени за минимален број на последователни рамки (поставено на 8 рамки) пред да активира предупредување. Оваа временска компонента е критична за разликување помеѓу нормално трепкање и вистинска поспаност. Дополнително, системот ги следи и другите фацијални карактеристики како носот и устата, што овозможува идно проширување на функционалноста за детекција на прозевање или други индикатори на замор.





\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

## **2. **Зависности****

Програмата зависи од следните Python библиотеки:

* **NumPy**: За нумерички пресметки, особено за пресметување на EAR.
* **OpenCV**: За снимање на видео од веб-камерата, обработка на фрејмови и приказ на излезот.
* **MediaPipe**: За детекција на маркери на лицето со користење на Face Mesh моделот.

За инсталација на зависностите, извршете:



## 3. Објаснување на кодот

### *3.1. Пресметка на односот на аспектот на окото (EAR)*

**Аспектот на окото (EAR)** е метрика што се користи за да се утврди дали очите се отворени или затворени. Се пресметува со користење на вертикалните и хоризонталните растојанија помеѓу специфичните маркери на окото.

Формулата за пресметка на EAR е базирана на евклидските растојанија помеѓу специфични точки околу окото. За секое око, се користат 6 клучни точки кои го дефинираат неговиот облик. Овие точки се избрани стратешки за да обезбедат најпрецизна репрезентација на состојбата на окото. Формулата го пресметува односот помеѓу вертикалните отвори на окото (измерени на две места) и хоризонталната должина на окото.

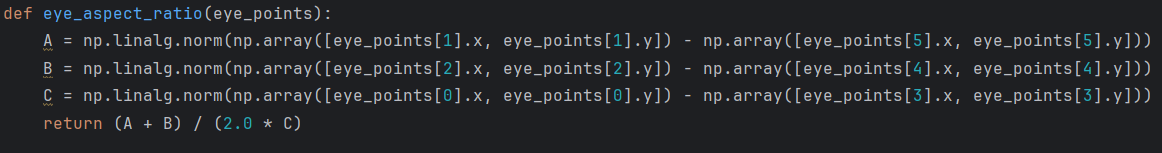
Математички, EAR се изразува како: ***EAR = (||p2 - p6|| + ||p3 - p5||) / (2 \* ||p1 - p4||)***

p1 до p6 се координатите на 6-те точки околу окото

||x|| означува евклидско растојание

p2 - p6 и p3 - p5 се вертикални растојанија

p1 - p4 е хоризонтално растојание



* **A:** Вертикално растојание помеѓу горниот и долниот маркер на окото.
* **B**: Вертикално растојание помеѓу средниот горен и средниот долен маркер на окото.
* **C**: Хоризонтално растојание помеѓу најлевиот и најдесниот маркер на окото.
* **EAR** се пресметува како (A + B) / (2.0 \* C). Понизок EAR укажува на затворени очи, додека повисок EAR укажува на отворени очи.

Оваа метрика е особено корисна бидејќи:

* Е независна од ротацијата на главата во рамнината на сликата
* Е нормализирана во однос на големината на окото во сликата
* Е робусна на различни израз на лицето
* Е едноставна за пресметка во реално време

Системот го пресметува EAR за двете очи и користи нивна средна вредност за поголема прецизност. Ова помага да се надминат потенцијални проблеми како што се нееднакво трепкање или делумно затворање на очите.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

### *3.2. Иницијализација на MediaPipe Face Mesh*

Програмата користи **Face Mesh** моделот на MediaPipe за детекција на фацијални маркери. Овој моќен алат, развиен од Google, овозможува прецизно следење на лицеви карактеристики во реално време.

MediaPipe Face Mesh е напредно решение за детекција на лице кое има неколку клучни карактеристики:

**Висока прецизност:**

Детектира 468 3D точки на лицето

Работи со брзина од 30 рамки во секунда

Обезбедува стабилно следење дури и при брзи движења

**Робустни карактеристики:**

* Работи при различни агли на главата
* Справување со делумно покриено лице
* Прецизна детекција при различно осветлување



* **min\_detection\_confidence**: Праг за доверба во детекцијата на лицето (поставен на 0.7).
* **min\_tracking\_confidence**: Праг за доверба во следењето на маркерите (поставен на 0.7).

Face Mesh користи машинско учење за да ги лоцира точките на лицето и ги следи нивните движења во реално време. Моделот е оптимизиран за работа на CPU, што го прави идеален за апликации што бараат реално-временско процесирање без потреба од специјализиран хардвер.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

### *3.3. Иницијализација на веб-камерата*

Програмата го снима видеото од стандардната веб-камера со користење на OpenCV. Оваа иницијализација е критичен чекор за системот бидејќи од неговата правилна конфигурација зависи квалитетот на детекцијата.

Процесот на иницијализација се врши со следниот код:



0 го означува индексот на првата достапна камера

За дополнителни камери може да се користат индекси 1, 2, итн.

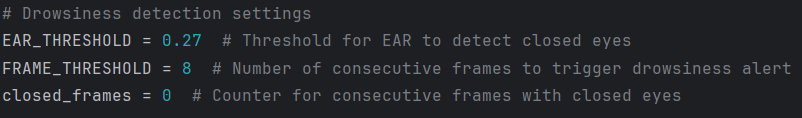
Враќа објект кој ќе се користи за читање на рамки од камерата

Правилната иницијализација на камерата е клучна за добра работа на системот за детекција на поспаност, бидејќи квалитетот на влезното видео директно влијае на прецизноста на детекцијата.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

### *3.4. Логика за детекција на дремливост*

Програмата го користи EAR за да ја детектира дремливоста. Ако EAR падне под прагот за одреден број на последователни фрејмови, се активира алармот за дремливост.



**EAR\_THRESHOLD** (0.27):

* Емпириски одредена вредност базирана на истражување
* Ако EAR < 0.27: очите се сметаат за затворени
* Ако EAR > 0.27: очите се сметаат за отворени
* Оваа вредност може да се прилагоди според потребите на корисникот

**FRAME\_THRESHOLD (8):**

* Број на последователни рамки со затворени очи
* Поставен на 8 рамки (~0.27 секунди при 30 FPS)
* Доволно долг период за да се избегнат лажни аларми од трепкање
* Доволно краток за навремена детекција на поспаност

**closed\_frames:**

* Динамички бројач што се ажурира во реално време
* Се зголемува кога EAR е под прагот
* Се ресетира на 0 кога очите се отвораат
* Служи за следење на времетраењето на затворени очи

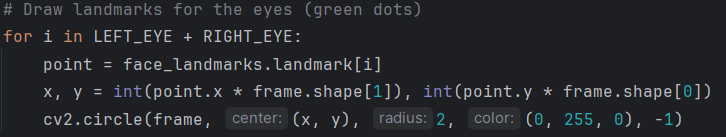
Оваа логика обезбедува балансиран пристап помеѓу чувствителноста на детекција и избегнувањето на лажни аларми, што е клучно за практична примена на системот.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

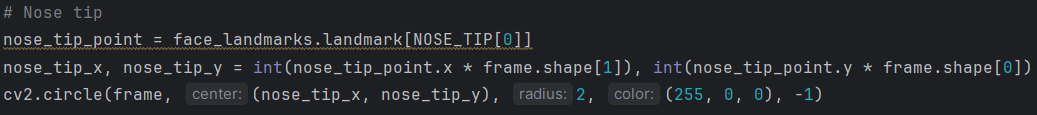
### *3.5. Визуелизација на фацијалните маркери*

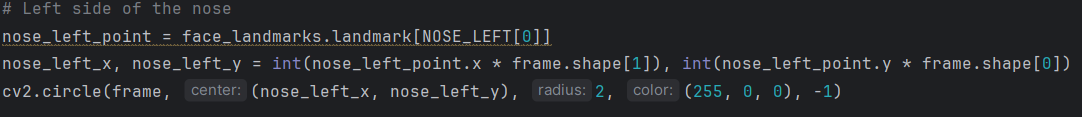
Програмата ги визуелизира фацијалните маркери за очите, носот и устата со користење на обоени точки:

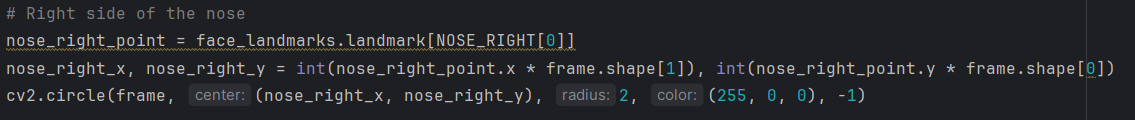
* 1. ***Очи****: Зелени точки.*
* **LEFT\_EYE + RIGHT\_EYE** ги спојува двете листи со **индекси ([362, 385, 387, 263, 373, 380] и [33, 160, 158, 133, 153, 144])**
* **point.x и point.y** се релативни координати (од 0 до 1)
* **frame.shape[1]** е ширината на рамката, **frame.shape[0]** е висината
* **(0, 255, 0)** е BGR код за зелена боја
* **2** е радиусот на кругот
* **-1** значи исполнет круг



***2. Нос****: Сини точки.*





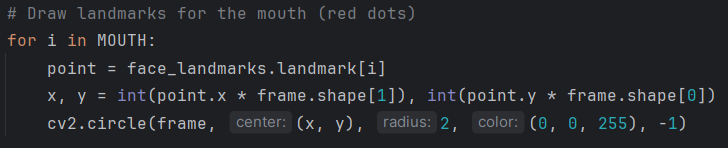


Слична логика како за очите

(255, 0, 0) е BGR код за сина боја

Се црта точка за врвот на носот, левата и десната страна

***3. Уста****: Црвени точки.*



Иста логика како за очите

(0, 0, 255) е BGR код за црвена боја

Важни детали:

* Точките се цртаат со различни бои за подобра визуелизација:
  + Очи: Зелена боја
  + Нос: Сина боја
  + Уста: Црвена боја
* Координатите се конвертираат од релативни (0-1) во пиксели со множење со димензиите на рамката
* Сите точки се цртаат како мали исполнети кругови со радиус 2 пиксели

Овој дел од кодот е важен за визуелна потврда дека системот правилно ги детектира и следи точките на лицето.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

### *3.6. Главен циклус и обработка на фрејмови*

Главен циклус ги обработува секој фрејм од веб-камерата:

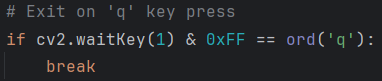
* Го превртува фрејмот хоризонтално за ефект на огледало.
* Го конвертира фрејмот во RGB за обработка со MediaPipe.
* Ги детектира фацијалните маркери со користење на MediaPipe Face Mesh.
* Го пресметува EAR и проверува за дремливост.
* Ги визуелизира маркерите и го прикажува EAR вредноста.



\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

### *3.7. Услов за излез*

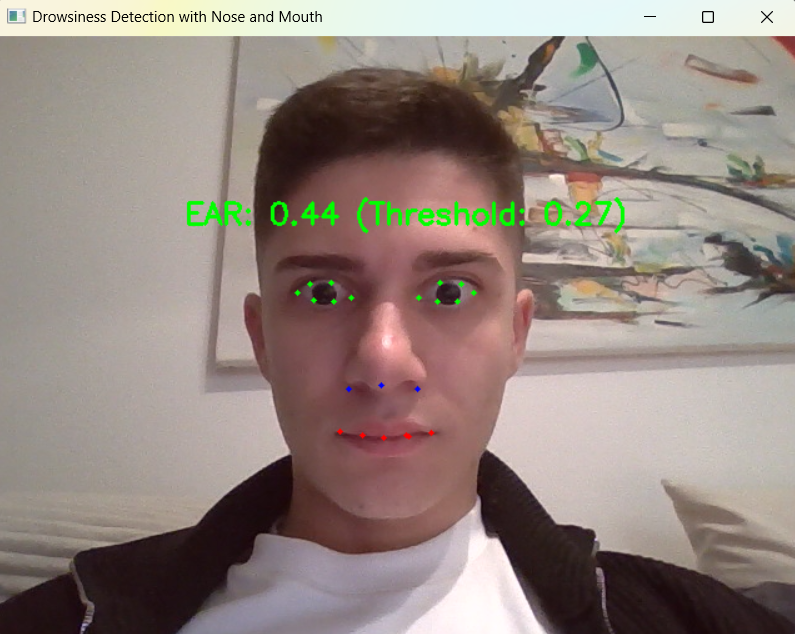
Програмата излегува кога корисникот ќе го притисне копчето **'q'**



**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

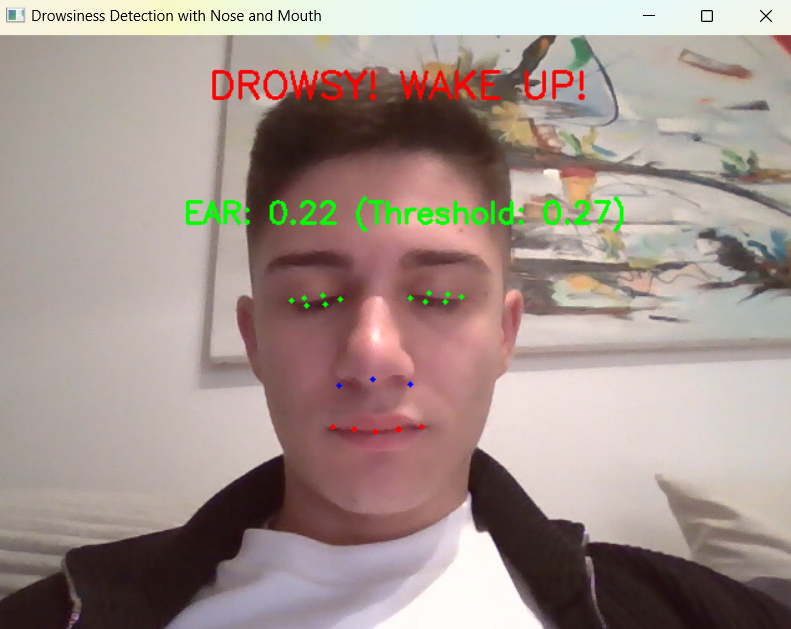
## *****4. Како функционира програмата*****

1. Програмата ја иницијализира веб-камерата и MediaPipe Face Mesh.
2. За секој фрејм:
   * Ги детектира фацијалните маркери.
   * Го пресметува EAR за двете очи.



*Тука може да се примети дека за многу отворени очи EAR индексот изнесува 0.44*

* + Проверува дали EAR е под прагот за одреден број на фрејмови.
  + Прикажува аларм за поспаност доколку е потребно.



*Тука може да се примети дека при затварање на очите EAR индексот се намалува на 0.22 што паѓа под опсегот од 0.27 и со тоа се активира пораката DROWSY! WAKE UP!*

* + Ги визуелизира фацијалните маркери на фрејмот.

1. Програмата излегува кога корисникот ќе го притисне копчето 'q'.

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

## *5. Прилагодување и подобрувања*

* **Прилагодете ги праговите**: Променете ги **EAR\_THRESHOLD** и **FRAME\_THRESHOLD** за подобар резултат.
* **Додадете звучен аларм**: Интегрирајте звучен аларм кога ќе се детектира дремливост.
* **Подобрете ја перформансата**: Оптимизирајте ја програмата за пониски перформанси со намалување на резолуцијата на фрејмовите или користење на полесен модел.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

## *6. Заклучок*

Системот за детекција на поспаност е практична примена на компјутерски вид и детекција на маркери на фацата. Тој обезбедува следење и аларми во реално време за поспаност, што го прави корисен за возачи, студенти или било кој што треба да остане буден. Програмата е многу прилагодлива и може да се прошири со дополнителни карактеристики за подобрена функционалност.

Имплементираниот систем демонстрира ефективна употреба на современи технологии за компјутерска визија, комбинирајќи ги можностите на MediaPipe Face Mesh за прецизна детекција на лицеви карактеристики со робустен алгоритам за анализа на состојбата на очите. Преку користење на Eye Aspect Ratio (EAR) како клучна метрика, системот успешно разликува помеѓу нормално трепкање и состојби на поспаност.

Главните придобивки од системот вклучуваат:

1. Работа во реално време со минимално доцнење
2. Висока прецизност во детекција на поспаност
3. Робустност при различни светлосни услови
4. Прилагодливост на различни кориснички потреби
5. Лесна интеграција со други системи

Потенцијални области за примена вклучуваат:

* Транспортна индустрија за следење на возачи
* Работни места со долги смени
* Образовни институции за следење на внимание
* Истражувачки лаборатории за студии на сон
* Индустриски постројки каде што е критична будноста

Идни насоки за развој би можеле да вклучуваат:

1. Интеграција на машинско учење за персонализирана детекција
2. Додавање на напредни алгоритми за препознавање на замор
3. Развој на мобилна апликација
4. Имплементација на cloud базирана аналитика
5. Подобрување на системот со дополнителни сензори

Технолошките предизвици што беа надминати вклучуваат:

* Оптимизација на перформансите за работа во реално време
* Прецизна калибрација на праговите за детекција
* Имплементација на робустна визуелизација
* Ефикасно управување со системските ресурси

Со продолжен развој и подобрување, овој систем има потенцијал да стане важна алатка во превенцијата на несреќи предизвикани од заспивање при возење или работа. Неговата отворена архитектура овозможува понатамошни модификации и подобрувања според специфичните потреби на различни индустрии и апликации.

Изработиле:

Борис Поцев 221209

Јована Силјановска 221021