

УДК 519.6

ПРО РОЗПІЗНАВАННЯ ОКРЕМИХ ОЗНАК ЧАСТИН ТІЛА ЛЮДИНИ З ВИКОРИСТАННЯМ ОБМЕЖЕНИХ ОБЧИСЛЮВАЛЬНИХ РЕСУРСІВ

С. Іванов, А. Музичук

*Львівський національний університет імені Івана Франка,
вул. Університетська, 1, Львів, 79000,
e-mail: serhii.ivanov@lnu.edu.ua, anatoliy.muzychuk@lnu.edu.ua*

Розглянуто загальний підхід до розпізнавання окремих частин людського тіла на відеозображеннях, які надходять з камер на різноманітних пристроях з обмеженими обчислювальними ресурсами. Зокрема, досліджується розпізнавання та аналіз жестів руки для розробки безконтактного графічного інтерфейсу користувача (GUI). Друга задача стосується контролю стану сонливості людини в режимі реального часу. Відео у цьому випадку містить зображення голови, наприклад, водія. Розроблено множину параметрів сонливості, які характеризують стан людини: тривалість і частота моргання очима, частота і тривалість нахилів голови, частота і глибина зівання.

Особливість зазначених задач – вимога до можливості виконання їхньої програмної реалізації на малопотужних пристроях, які можуть працювати автономно без використання додаткових серверних ресурсів. Для кожної задачі побудовано моделі, в основі яких є множини, що складаються з малої кількості ключових точок. Завдяки цьому вдалося розробити швидкі та високоефективні алгоритми детекції й аналізу.

Програмна реалізація алгоритмів першої задачі виконана з використанням нейронних мереж бібліотеки MediaPipe для операцій детекції та скелетонізації і SciKit для безпосереднього розпізнавання жестів. Завдяки малій кількості використаних ключових параметрів кисті руки на завершальному етапі виявилось ефективним використання простої повнозв'язної мережі. Наведено множину жестів, яка є достатньою для розробки безконтактного GUI.

Для реалізації алгоритмів другої задачі використано бібліотеки OpenCV і Dlib, зокрема згортова мережа ResNet-10 для розпізнавання обличчя, в якій останнім шаром також є повнозв'язна мережа. Подано порогові значення параметрів сонливості, які отримали експериментально. Наведені результати демонструють високу точність розпізнавання досліджуваних характеристик.

Ключові слова: розпізнавання жестів, розпізнавання стану сонливості, безконтактний інтерфейс, нейронні мережі.

1. ВСТУП

Останнім часом спостерігається бум розвитку підходів і технологій штучного інтелекту (AI), зокрема його підрозділу – комп'ютерного зору (CV) та їхнього впровадження у різні сфери сучасної науки і техніки. Такий розвиток є виявом трьох взаємодоповнювальних тенденцій: постійного прогресу алгоритмів машинного навчання, доступності величезних обсягів даних і зростаючої обчислювальної потужності комп'ютерної техніки.

Розпізнавання та автоматизований аналіз зображень – це найважливіший напрям розвитку технологій CV. Він супроводжується інтенсивним розвитком алгоритмів розпізнавання, а також їхньою реалізацією для різних платформ у рамках високопродуктивних бібліотек програмного забезпечення (ПЗ), значна частина яких є вільним і відкритим ПЗ. Зазначимо значний внесок у їхній розвиток сучасних технологічних гігантів і великої кількості невеликих ІТ-компаній. У підсумку

спостерігаємо тенденцію до швидкого впровадження найновіших досягнень AI у різних сферах повсякденної діяльності.

Цікавим прикладом впровадження AI та CV є історія технологій розпізнавання облич. Значний поштовх до розвитку нових технологій відбувся, зокрема, після застосування компанією Apple в своїй продукції системи FaceID, а також після впровадження алгоритмів виявлення та ідентифікації користувачів мережі Facebook на поширених у мережі фотозображеннях. Про рівень досягнень у цій області свідчить високий відсоток правильності розпізнавання. Водночас, при безумовній важливості таких технологій, вони породжують застереження щодо безпеки та свободи в суспільстві. З огляду на вплив цих новітніх технологій на суспільні процеси, виглядає цілком закономірним останнє рішення Facebook про дезактивацію функції розпізнавання облич на фотоматеріалах мережі [2]. Однак розробку нових алгоритмів AI та CV ця компанія виводить на новий рівень, зокрема у проєкті Ego4D [3]. Результати цих досліджень будуть основою систем AI нового покоління, в яких машинне навчання виконуватиметься на абсолютно інших даних – відео, які показують світ із центру події, а не з фотозображень, виконаних з позиції стороннього спостерігача за подією.

Широкий клас задач опрацювання зображень стосується визначення стану людини та параметрів її поведінки у певних процесах повсякденної діяльності. На відміну від розпізнавання облич, тут важливим є розпізнавання лише певних ознак, характерних для окремих частин тіла. Причому реалізація таких завдань, зазвичай, відбувається на обмежених обчислювальних ресурсах. У цій публікації стосовно таких підходів AI та CV ми розглядаємо дві задачі, які мають важливе застосування:

- 1) розпізнавання жестів з використанням кисті руки;
- 2) виявлення ознак втомленості людини за параметрами сонливості, до яких зараховуємо положення голови, частоту кліпання очима та ін.

Перша задача стосується розробки безконтактного графічного інтерфейсу користувача (GUI). Така задача є в центрі уваги багатьох досліджень, зокрема як частина комплексного розпізнавання активності людини (див., наприклад, [4]). Зауважимо, що в умовах пандемії безконтактні GUI знаходять все більше застосування у пристроях для зважування в супермаркетах, у всіляких можливих платіжних терміналах, банкоматах тощо. Зараз GUI таких пристроїв традиційно функціонують на основі дотиків користувача до відповідних ділянок екрана чи клавіатури. Розробити нове покоління пристроїв з безконтактним інтерфейсом можна за допомогою сучасних технологій розпізнавання зображень і пов'язаних з ними методів машинного навчання.

Загальна вимога до GUI приладів у місцях масового використання – зменшення вартості технічних пристроїв та забезпечення їхнього функціонування в автономному режимі, без дорогого серверного обладнання. Зазначимо, що в цих задачах невисокі вимоги до швидкодії та точності розпізнавання, бо для користувача візуалізують результати його дій у реальному часі.

Сферою застосування системи розпізнавання сонливості є місця, які потребують постійної уваги людини, наприклад, місце водія автомобіля. У цьому випадку вимоги до швидкодії та точності розпізнавання високі, оскільки вони життєво важливі.

Одне із завдань у зазначених задачах – виявити певну множину ознак, які достатньо інформативно характеризують стан частин тіла людини чи її активність. Зазвичай, намагаються мінімізувати кількість елементів у таких множинах, щоб

послабити вимоги до обчислювальних ресурсів для їхнього виявлення та опрацювання. Для виявлення та опрацювання зазначених характеристик ми використали бібліотеки OpenCV [12], Dlib [1] та MediaPipe [10].

Ця публікація складається з чотирьох розділів. Спочатку у розділі 1 розглянуто моделі кисті руки й обличчя, визначено параметри, які характеризують стан сонливості людини. У розділі 2 описано нейронні мережі, за допомогою яких реалізовано алгоритми розпізнавання різних частин тіла. У розділах 3 і 4 описано програмну реалізацію обох задач і наведено отримані результати.

2. ВИЗНАЧЕННЯ ОСНОВНИХ ПОНЯТЬ І МОДЕЛЕЙ

Розглянемо коротко предметну область зазначених задач і відповідні моделі. **Модель кисті руки** є основою функціонування безконтактного GUI. Вона заснована на множині з невеликою кількістю елементів, визначених ключовими точками кисті руки [5]. Схема ключових точок цієї моделі зображена на рис. 1.

Зазначимо, що жести, які розпізнаватимуться під час використання такої моделі, звичні для користувачів сучасних пристроїв з контактним GUI, тому перехід до безконтактного інтерфейсу на становитиме проблем для його освоєння.

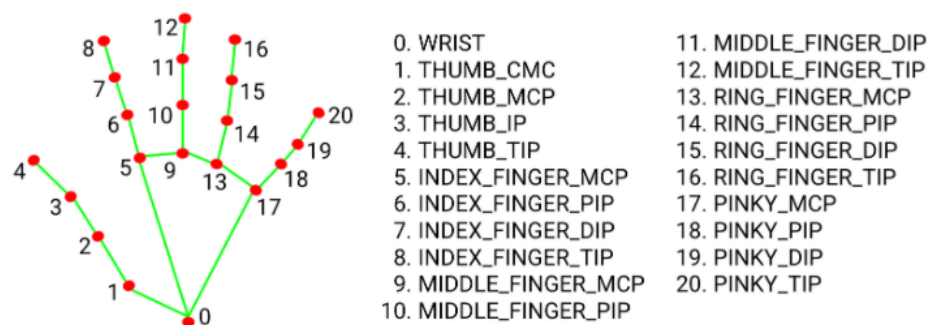


Рис. 1. Модель кисті руки та позначення ключових точок

Модель сонливості заснована на множині параметрів (ознак), значення яких можуть змінюватися у певних допустимих діапазонах [17]. Головне завдання контролю за поведінкою водія – ідентифікація з плином часу таких моментів, коли значення цих параметрів досягають певних критичних (порогових) величин.

Ми досліджуємо такі параметри, які характеризують стан сонливості:

- тривалість моргання;
- частота моргання;
- частота і тривалість нахилів голови;
- частота і глибина зівання.

Надалі називатимемо їх параметрами сонливості.

Під час руху автомобіля поведінку водія фіксують на відеокамеру, яка встановлена на рівні його голови, щоб отримувати її повне відеозображення. Відповідно задається освітлення, якого має бути достатньо для розпізнавання основних елементів обличчя. Для уникнення впливу зовнішніх джерел світла можна також використовувати інфрачервону камеру. З отриманого відеозображення в режимі реального часу отримують й аналізують параметри сонливості.

В моделях обличчя відповідні алгоритми працюють з ключовими точками, відомими як “face landmarks”. Традиційно використовують множину з 68 точок, які характеризують різні елементи обличчя: підборіддя, краї очей та брів, ніс тощо. Кількість точок у разі потреби можна змінювати, наприклад, використовувати 3D модель [10] з 468 точок. Для виявлення стану сонливості обмежуємося контролем лише за кількома елементами обличчя, а саме, важливими є точки, які моделюють очі (див. рис. 2) та губи.

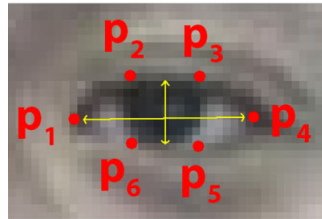


Рис. 2. Ключові точки зображення очей

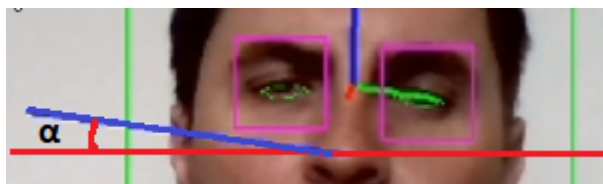
Відношення сторін очей (Eyes Aspect Ratio, EAR) є обчислюваним параметром стану сонливості. Його знаходять за формулою [15]

$$EAR = \frac{dist(p_2, p_6) + dist(p_3, p_5)}{2dist(p_1, p_4)}, \quad (1)$$

де $dist(p_i, p_j)$ – відстань між ключовими точками p_i і p_j на зображенні очей. За допомогою EAR визначають інший параметр сонливості – частоту моргань.

Для розпізнавання очей на відеозображенні голови використано метод Хаара [16], суть якого полягає в аналізі елементів зображення за яскравістю пікселів, уникаючи безпосередньої ресурсозатратної роботи з кольором пікселя. Це є запорукою розробки високоефективної програмної реалізації, придатної до використання в режимі реального часу.

Ще одним обчислюваним параметром стану сонливості є кут нахилу голови (див. рис. 3).

Рис. 3. Кут α нахилу голови

Це кут між прямою, що з'єднує середини очей, та горизонтальною прямою. Перша пряма визначається через ключові точки p_i на зображенні очей.

3. НЕЙРОННІ МЕРЕЖІ ТА АЛГОРИТМИ ПРОЦЕСІВ РОЗПІЗНАВАННЯ

Інформація для розпізнавання надходить кадрами відео. У випадку розпізнавання жестів – це потік від відеокамери, встановленої на пристрої, з яким взаємодіє

користувач через безконтактний GUI. У випадку контролю за станом водія камеру встановлюють в салоні автомобіля.

Процеси розпізнавання стосовно отриманого зображення в обох завданнях виконують у режимі конвейера за такими етапами:

- 1) видалення фонового зображення;
- 2) детекція потрібної частини тіла на зображенні;
- 3) розпізнавання ключових точок виділеної частини та знаходження обчислювальних параметрів.

У процесі експлуатації на відео потрапляє зображення певного стаціонарного фону, зокрема, приміщення, де встановлений банкомат, чи кабіни автомобіля. Його виділення відбувається засобами бібліотеки OpenCV шляхом аналізу останніх кількох кадрів у відео. Після знаходження частин зображення, які не змінюються, з їхніх даних генерується маска динамічних об'єктів. У подальшому цю маску віднімають від поточних кадрів попіксельно і результат порівнюють з певним пороговим значенням THRESHOLD. Пікселі, які перевищують заданий THRESHOLD, формують підготовлене зображення. Далі з підготовленого зображення виділяють за допомогою нейронних мереж зображення потрібної частини тіла. У наших задачах – кисті руки чи обличчя.

Нейронні мережі та процеси розпізнавання жестів. Розпізнавання жестів реалізовано з використанням високопродуктивних бібліотек для роботи з нейронними мережами MediaPipe [10] та SciKit [13]. Для опрацювання зображень використано конвейер з трьох нейронних мереж:

- 1) мережа, яка працює з даними усього вхідного зображення, виконуючи операцію детекції – локалізацію зображення кисті руки;
- 2) мережа для скелетонізації – відтворення тривимірних позицій ключових елементів кисті;
- 3) повнозв'язна мережа, яка визначає жест по нормованих ключових координатах скелету кисті.

Дві перші мережі, які є частиною MediaPipe, працюють у парі. Вихідними даними цього конвейера є нормовані координати 21 ключового елемента кисті руки. Це величини у діапазоні від 0 до 1 по осях абцис та ординат. За цими даними відбувається розпізнавання та побудова скелетона лівої та правої рук. Нейронна мережа, яку використовує MediaPipe, була тренувана на 30 тисячах реальних тривимірних зображень. Оскільки кількість параметрів на виході цього фреймворку невелика, то для розпізнавання жесту виявляється достатньо використати просту повнозв'язну нейронну мережу.

Повнозв'язна мережа побудована на підставі методу логістичної регресії. На вхід вона отримує координати ключових елементів кисті, відповідно на виході отримуємо вектор наближень до ключових жестів. Навчання цієї мережі є простим і не потребує багато вхідних даних через її простоту та використання механізму “оберненого розповсюдження похибки”.

Розглянутий підхід дає змогу досягати до 96% вдалого розпізнавання. Крім того, отримуємо достатньо високу швидкість обробки зображень, зокрема, програма може ефективно працювати на мобільних пристроях.

Нейронні мережі та процеси контролю параметрів сонливості. Під час руху автомобіля опрацювання параметрів сонливості проходить в реальному часі такі основні етапи:

- 1) виділення з відео ключових ознак обличчя;

2) обчислення на підставі даних з відео нових значень параметрів вищого рівня: співвідношення сторін очей під час моргання та рота під час позіхання, кута нахилу голови;

3) аналіз змін у часі значень параметрів вищого рівня для контролю за наближенням цих величин до критичних значень, тобто виявлення стану надмірної сонливості.

Зазначимо, що порогові значення зазначених параметрів сонливості визначалися емпірично. Програмна реалізація контролю сонливості виконана з використанням бібліотек Dlib [1] та OpenCV [12]. Зокрема, це засоби для роботи зі згортковими нейронними мережами в задачах розпізнавання самого обличчя та його частин – очей, рота, які є важливими для аналізу сонливості.

Для розпізнавання обличчя використовують згорткову мережу (convolution network) ResNet-10, архітектуру і функціональність якої зображено на рис. 4.

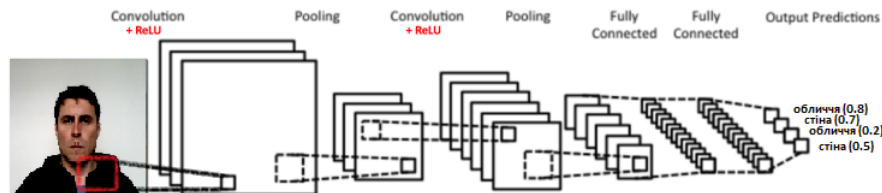


Рис. 4. Архітектура і функціональність згорткової мережі ResNet-10 щодо розпізнавання обличчя

Мережа складається з послідовних шарів “Convolution” та “Pooling”, які по чергово виконують перетворення даних. Зокрема, згортковий шар виділяє мітки та орієнтири обличчя, а об’єднувальний шар виділяє одну з конкретних домінуючих ознак в певному сегменті зображення та видаляє усі інші ознаки.

Останній шар – повнозв’язна мережа. Результатом його роботи є набір ймовірностей наявності того чи іншого об’єкта у початковому відеозображенні. Якщо об’єктів на початковому зображенні кілька, то воно розбивається на сектори і до кожного з них застосовується згорткова нейронна мережа. Так реалізовано основну ідею згорткової мережі, яка полягає у перенесенні опрацювання абстрактних елементів зображення на вищі шари мережі.

Зауважимо, що при використанні класичних згорткових мереж точність розпізнавання об’єктів спочатку зростає з додаванням нових шарів мережі, а потім починає помітно знижуватися. Щоб уникнути цієї проблеми, у ResNet додано з’єднання так званого швидкого доступу [11] для безпосередньої передачі даних з певного шару нейронної мережі на пряму у наступні шари, оминаючи проміжні. Експериментально з’ясовано, що такий прийом дає змогу зберігати точність перетворень.

Розпізнавання ключових точок на обличчі виконується за допомогою алгоритму “Вирівнювання обличчя з використанням дерев регресії” [9], який реалізований у бібліотеці dlib [1].

Розпізнавання очей вирізняється серед інших ключових параметрів обличчя застосуванням додаткового алгоритму на підставі методу Хаара. Це зумовлено підвищеними вимогами щодо точності розпізнавання очей, бо, крім моргання, ще обчислюємо кут нахилу голови через середню лінію очей. З цим завдання метод

Хаара справляється успішно, причому його застосовуємо на малих за площею прямокутниках очей (див. рис. 5), що не потребує значних затрат обчислювальних ресурсів.

Контроль параметрів сонливості, як вже було зазначено, відбувається програмно у реальному часі. Для візуалізації динаміки цих параметрів під час проведення експериментів розроблено відповідний GUI. Частина цього інтерфейсу відображено на скріншоті з демонстраційного відео на рис. 5.

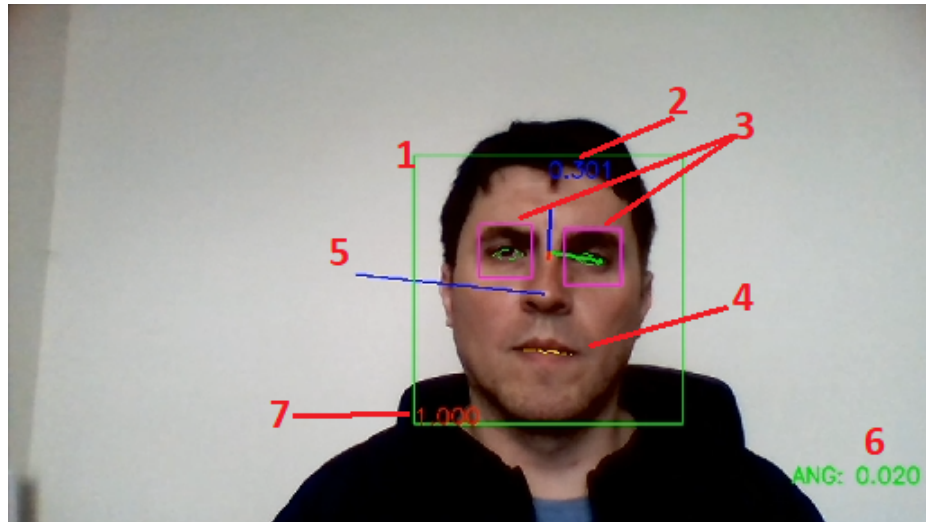


Рис. 5. Демонстрація параметрів стану водія: 1– область розпізнаного обличчя (зелений прямокутник); 2 – відношення сторін очей (EAR); 3 – область розпізнаних очей (рожеві прямокутники) та область самих очей (помічено зеленим); 4 – область розпізнаного рота (помічено жовтим); 5 – вектор орієнтування обличчя; 6 – кут нахилу голови; 7 – точність розпізнавання обличчя

Це зображення створене з оригінального відеозображення шляхом нанесення додаткової інформації для візуального відстежування розпізнавання програмою ділянок самого обличчя та різних його ділянок. Зокрема виводяться значення параметрів EAR і кута нахилу голови, а також біжуча точність розпізнавання обличчя. Нагадаємо, що під точністю розуміємо ймовірність розпізнавання обличчя, для цього прикладу вона становить 1.

4. АРХІТЕКТУРА ПЗ КОНТРОЛЮ СТАНУ СОНЛИВОСТІ ТА ЙОГО ДЕМОНСТРАЦІЯ

Програмна реалізація описаного вище алгоритму контролю за станом водія виконана мовою Python з використанням бібліотек OpenCV [12] і dlib [1]. На рис. 6 схематично зображена діаграма класів програми. Тут синіми стрілками показано використання екземплярів класу. Червоні стрілки вказують “workflow” – порядок виконання коду та рух даних між відповідними екземплярами для їх подальшого аналізу.

Клас **DrowsinessDetectionApp** є основним класом програми. Він відповідає за створення і взаємодію різних компонентів: графічного інтерфейсу, читання вхідних

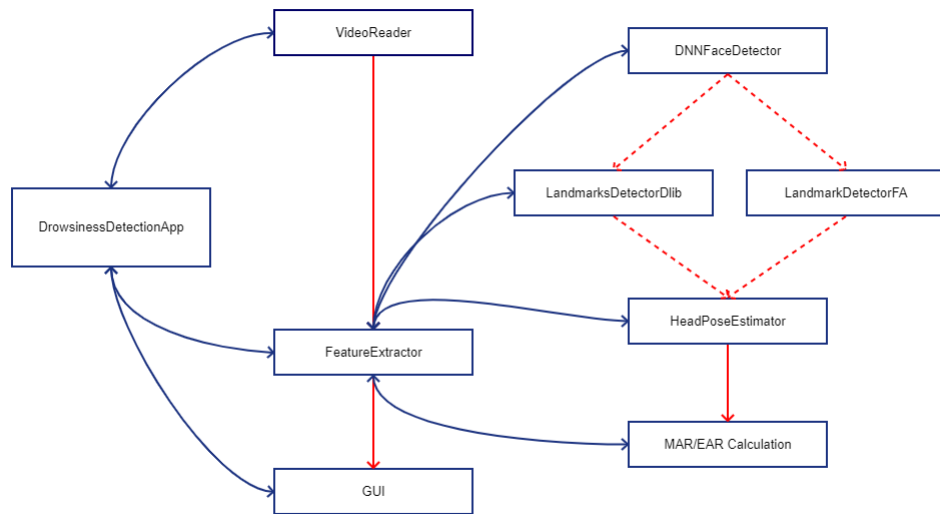


Рис. 6. Діаграма класів програми та їхні асоціації

даних, визначення ознак, агрегацію та візуалізацію даних.

Після читання відео за допомогою класу **VideoReader** відбувається визначення характерних ознак **FeatureExtractor** за таким алгоритмом:

- 1) визначається присутність обличчя на зображенні через використання згорткових нейронних мереж (**DNNFaceDetector**);
- 2) обирається основне обличчя (це обличчя найбільшої площі, якщо на зображенні їх кілька);
- 3) визначаються ключові точки частин обличчя (**LandmarksDetectorDlib** або **LandmarksDetectorFA**);
- 4) визначення положення голови та кута її нахилу (**HeadPoseEstimator**);
- 5) обчислення відношення сторін очей (**EAR**);
- 6) обчислення відношення сторін рота (**MAR**).

Зазначимо, що моделі бібліотеки **dlib** є швидшими при виконанні обчислень на CPU, а моделі **openCV** дають точніші результати і підтримують обчислення в реальному часі на графічних процесорах.

Отримані значення параметрів сонливості та зображення повертаються в **DrowsinessDetectionApp.process**, де вони записуються в буфери для подальшого аналізу та візуалізації. Аналіз даних щодо визначення моргання, перевищення ознак порогових значень, визначення рівня сонливості відбувається в методі **DrowsinessDetectionApp.detect_drowsiness**. Після чого уся інформація візуалізується з допомогою класу **GUI** шляхом відмальовування поверх оригінального зображення.

Зазначимо, що для побудови графічного інтерфейсу **GUI** програми використано модулі **pyqtgraph** та **PyQt5**, а для зчитування вхідних даних задіяно клас **VideoReader**, який може працювати з камерами та з файлами. На останньому етапі початкового встановлення класу ініціалізації екземпляру **DrowsinessDetectionApp** запускається покадрове опрацювання вхідного потоку даних методом **pro-**

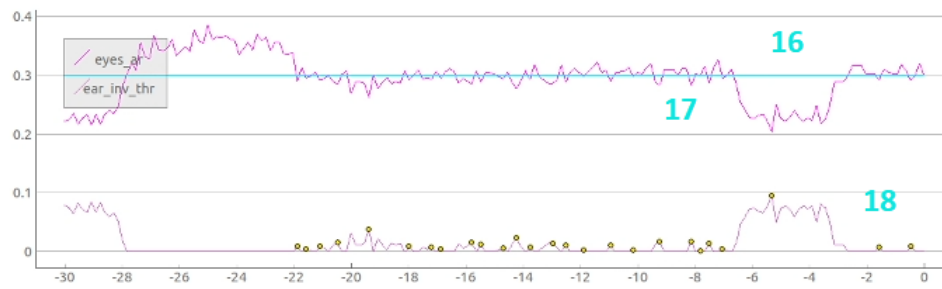


Рис. 7. Демонстрація зміни в часі параметра EAR (відношення сторін очей): 16 – ділянка, яка відповідає нормально відкритим очам, коли $EAR > 0.3$, 17 – ділянка, яка відповідає закритим очам, коли $EAR < 0.3$, 18 – допоміжний графік EAR для контролю закритих очей при $EAR < 0.3$

cess.

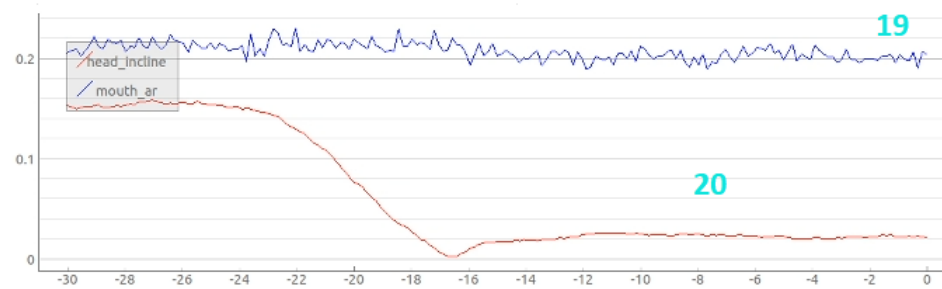


Рис. 8. Демонстрація зміни в часі параметрів кута нахилу голови і MAR: 19 – графік кута нахилу голови, 20 – графік відношення сторін рота (MAR)

Для візуалізації роботи процедури контролю GUI користувача, крім відтворення відеозображення, містить ще панелі (див. рис. 7, 8), на яких програма в реальному часі відображає графіки біжучих значень параметрів моделі. Демонстраційне відео контролю сонливості див. [6].

Як було зазначено вище, порогові значення деяких параметрів сонливості визначені емпіричним шляхом, наприклад, для EAR таким є число 0.3. На рис. 7 зображено, що за таким значенням виділяють випадок нормально чи широко відкритих очей (область 16) і примружених чи закритих (область 17). Для зручності також будується додатковий графік (область 18) з відображенням моментів моргання, які можна трактувати як стан із закритими очима. Нахил голови та позіхання можна спостерігати на графіках рис. 8.

Отож запропонований підхід і його програмна реалізація дають змогу контролювати стан водія в реальному часі.

5. АРХІТЕКТУРА ПЗ РОЗПІЗНАВАННЯ ЖЕСТІВ І ЙОГО ДЕМОНСТРАЦІЯ

Основні етапи роботи програми зображено на рис. 9. Як видно з діаграми, після видалення фону інформація покадрово аналізується фреймворком MediaPipe [10].

Його перша частина локалізує зображення рук, а друга відтворює тривимірну модель скелета руки та передає двовимірні координати, отримані з зображення на картинці. Для цього послідовно використовується кожен п'ятнадцятий кадр відео.

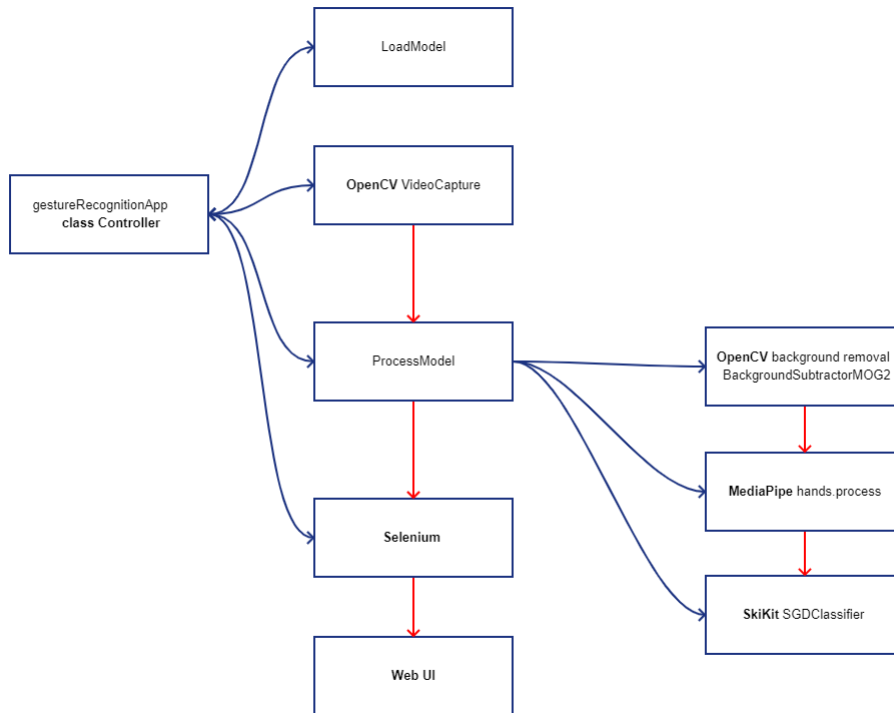


Рис. 9. Діаграма виклику функцій у програмі й обробки даних

Скріншот з демонстраційного відео подано на рис.10. Зазначимо, що повне відео можна переглянути за адресою [7]. На картинці зліва відображається кадр оригінального відео, посередині – локалізована рука користувача, а справа – кадр з відео, промаркований ключовими точками і скелетом руки зеленим та червоним кольором, відповідно.

Зверху над зображенням з відео імітується обробка команд прототипу користувачького інтерфейсу. Активацію відповідних елементів інтерфейсу виконують керуванням жестами. Перехід між елементами інтерфейсу відбувається за рахунок жесту, заданого порухом лише пальця, тоді як перехід на наступну сторінку відбувається за допомогою жесту, де для задання напрямку гортання сторінок задіяна вся кисть руки. Опис усіх жестів і відповідних команд, які зараз розпізнає програма, подано у табл. 1.

Після опрацювання зображень на кадрі отримані дані надходять у повнозв'язну нейронну мережу. Якщо був розпізнаний один з керівних жестів, то викликаються методи фреймворку Selenium [14] для керування прототипом інтерфейсу у браузері. У випадку прикладної програми, що працюватиме на конкретному фізичному пристрої, останній програмний компонент буде замінений викликами конкретних

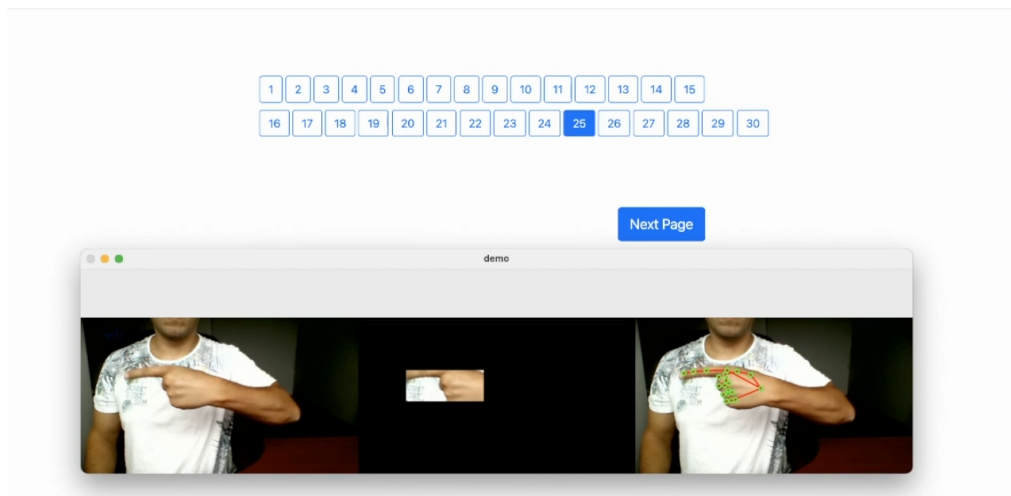


Рис. 10. Скріншот роботи програми, зверху – імітація GUI користувача, яка моделює роботу з меню за допомогою розпізнаних жестів

Таблиця 1

Жести та відповідні команди безконтактного графічного інтерфейсу користувача

Жест	Команда GUI відносно поточного елемента
Палець показує вліво	Переміститись вліво
Палець показує вправо	Переміститись вправо
Палець показує вверх	Переміститись вгору
Палець показує вниз	Переміститись вниз
Рука показує вправо	Переміститись до наступної сторінки
Рука показує вліво	Переміститись до попередньої сторінки
Зображена долоня	Активувати поточний елемент

функцій, що є обробниками команд користувацького інтерфейсу для такого застосування.

Для тренування регресійної нейронної мережі використовувалось відео тривалістю 1 хв по 8 секунд на кожен жест. Точність, якої вдалось досягти стосовно зазначених жестів на цьому етапі досліджень, становить 89%.

6. ВИСНОВКИ

Результати, отримані під час апробації розробленого програмного забезпечення, свідчать про ефективність застосування підходів комп'ютерного зору та машинного навчання для розпізнавання елементів частин тіла людини та її активності. Причому

му відповідні програми можуть працювати на пристроях з обмеженими обчислювальними потужностями в режимі реального часу. Розроблені моделі та програмне забезпечення утворюють основу для подальших досліджень і розробок у сферах безконтактної взаємодії користувачів з пристроями загальної доступності та контролю параметрів, які характеризують стан людини.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Electronic Sources: Dlib C++ Library. – Available from: <http://dlib.net/>.
2. Electronic Sources: Facebook app: An Update On Our Use of Face Recognition. – Available from: <https://about.fb.com/news/2021/11/update-on-use-of-face-recognition/>
3. Electronic Sources: Facebook AI: Ego4D. – Available from: <https://ai.facebook.com/research/publications/ego4d-unscripted-first-person-video-from-around-the-world-and-a-benchmark-suite-for-egocentric-perception>.
4. Electronic Sources: Google AI Blog: MediaPipe Holistic – Simultaneous Face, Hand and Pose Prediction, on Device. – Available from: <https://ai.googleblog.com/2020/12/mediapipe-holistic-simultaneous-face.html>.
5. Electronic Sources: Google AI Blog: On-Device, Real-Time Hand Tracking with MediaPipe. – Available from: <https://ai.googleblog.com/2019/08/on-device-real-time-hand-tracking-with.html>.
6. Ivanov S.A. Drowsiness Detection with CV and ML: demo. – Available from: <https://www.dropbox.com/s/b8fy2d8rjtac61h/2020-06-0118-41-55.mp4?dl=0>.
7. Ivanov S.A. Touchless GUI with CV and ML: demo. – Available from: <https://www.dropbox.com/s/vtfuwcssr0mlr7q/demo.mp4?dl=0>
8. Jabbara R. Real-time Driver Drowsiness Detection for Android Application Using Deep Neural Networks Techniques / R. Jabbara, K. Al-Khalifaa, M. Kharbechea, W. Alhajjaseena, M. Jafaric, S. Jiangc // 2020 IEEE International Conference on Informatics, IoT, and Enabling Technologies (ICIoT), 2-5 Feb. – 2020. – DOI: 10.1109/ICIOT48696.2020.9089484.
9. Kazemi V. One millisecond face alignment with an ensemble of regression trees / Vahid Kazemi, Josephine Sullivan // IEEE Conference on Computer Vision and Pattern Recognition. – 2014. – Available from: <https://ieeexplore.ieee.org/document/6909637>.
10. Electronic Sources: MediaPipe. Open Source Google Machine Learning Library. – Available from: <https://google.github.io/mediapipe/>.
11. Ng A. Convolutional Neural Networks / Andrew Ng. – Available from: <https://www.coursera.org/learn/convolutional-neural-networks>.
12. Electronic Sources: OpenCV: Open Source Computer Vision Library. – Available from: <https://docs.opencv.org/master/d9/df8/tutorialroot.html>.
13. Electronic Sources: Scikit-learn: Machine Learning in Python. Classification. – Available from: <https://scikit-learn.org/stable/>
14. Electronic Sources: Selenium IDE: Open source record and playback test automation for the web. – Available from: <https://www.selenium.dev/selenium-ide/>
15. Soukupova T. Real-Time Eye Blink Detection using Facial Landmarks / T. Soukupova, J. Cech // 21-st Computer Vision Winter Workshop, Luka Cehovin, Rok Mandeljc, Vitomir Struc (eds.), Rimske Toplice, Slovenia, February 3-5. – 2016.
16. Viola P. Rapid Object Detection using a Boosted Cascade of Simple Features / Paul Viola, Michael Jeffrey Jones // CVPR, IEEE Computer Society Conference on Computer Vision and Pattern Recognition. IEEE Computer Society Conference on Computer Vision and Pattern Recognition 1:I-511-I-518. – Vol. 1. – DOI: 10.1109/CVPR.2001.990517.

17. Zhong G. Towards Data Science: Drowsiness Detection with Machine Learning / Grant Zhong, Rui Ying, He Wang, Aurangzaib Siddiqui, Gaurav Choudhary. – Available from: <https://towardsdatascience.com/drowsiness-detection-with-machine-learning-765a16ca208a>.

Стаття: надійшла до редколегії 08.09.2021

доопрацьована 03.11.2021

прийнята до друку 24.11.2021

On the recognition of separate features of a person's body using restricted computing resources

S. Ivanov, A. Muzychuk

*Ivan Franko National University of Lviv,
Universytetska str., 1, Lviv, 79000, e-mail: hrefserhii.ivanov@lnu.edu.ua,
anatoliy.muzychuk@lnu.edu.ua*

A general approach to the recognition of separate features of a person's body on video images which may be obtained from cameras on different devices and used for their further analysis was considered. In particular, the recognition and analysis of hand gestures are investigated to develop a touch-less graphic user interface (GUI). The second task concerns controlling drowsiness of a person in real time. Videos in this case contain images of the head of a driver. A set of parameters of drowsiness that characterizes the state of a person has been devised: duration and frequency of blinking, frequency and duration of the head tilt, frequency and depth of yawning.

A peculiarity of tasks considered is a demand for their program implementation to be executed on devices which are limited in their computing resources and can work offline without involving additional server resources. Models based on sets of a small quantity of keypoints are designed for each task. Due to this, fast and highly efficient algorithms for detection and analysis were developed.

The program implementation of the first task algorithms is carried out using neural networks of MediaPipe library for gestures detection and their skeleton modeling and SciKit library for the direct recognition of gestures. Due to a small quantity of keypoints of a hand at the stage of completion, using a simple fully connected network turned out to be efficient. A set of gestures are proposed which is sufficient for touch-less GUI development.

The program implementation of the second task algorithms is carried out using OpenCV and Dlib libraries, ResNet-10 network for the faces recognition in particular, in which fully connected network is the last layer. Thresholds of drowsiness that were obtained during tests are presented. The given results demonstrate high accuracy of the recognition of the features under study.

Key words: gesture recognition, drowsiness recognition, touch-less interface, neural networks.