

Development of the computer vision system based on machine learning for educational purposes

Serhiy O. Semerikov^{1,2,3,4[0000–0003–0789–0272]},
Tetiana A. Vakaliuk^{1,3,5[0000–0001–6825–4697]},
Iryna S. Mintii^{1,3[0000–0003–3586–4311]},
Vita A. Hamaniuk^{1[0000–0002–3522–7673]},
Vladimir N. Soloviev^{1[0000–0002–4945–202X]},
Olga V. Bondarenko^{1[0000–0003–2356–2674]},
Pavlo P. Nechypurenko^{1[0000–0001–5397–6523]},
Svitlana V. Shokaliuk^{1[0000–0003–3774–1729]},
Natalia V. Moiseienko^{1[0000–0002–3559–6081]},
and **Vitalii R. Ruban**¹

¹ Kryvyi Rih State Pedagogical University, 54, Gagarina Ave.,
Kryvyi Rih, 50086, Ukraine

² Kryvyi Rih National University, 11, Vitalii Matusevych Str.,
Kryvyi Rih, 50027, Ukraine

³ Institute for Digitalisation of Education of the NAES of Ukraine, 9,
M. Berlynskoho Str., Kyiv, 04060, Ukraine

⁴ University of Educational Management of NAES of Ukraine, 52-A, Sich
shooters Str., Kyiv, 04053, Ukraine

⁵ Zhytomyr Polytechnic State University, 103, Chudnivska Str.,
Zhytomyr, 10005, Ukraine

semerikov@gmail.com, tetianavakaliuk@gmail.com,
irina.mintiy@kdpu.edu.ua, vitana65@gmail.com,
vnsoloviev2016@gmail.com, bondarenko.olga@kdpu.edu.ua,
acinonyxleo@gmail.com, shokalyuk@kdpu.edu.ua,
n.v.moiseenko@gmail.com, vitas.uno@gmail.com

Abstract. The article provides an overview of the origins and current state of machine vision systems, examples of machine vision problems. The article describes the use of computer vision systems in education in both conventional

and pandemic conditions. The COVID-19 pandemic has triggered changes in education that have modified existing educational applications of computer vision systems and spawned new ones, including social distancing, facial mask recognition, detection of infiltration into universities and schools, prevention of vandalism and detection of suspicious objects, attendance monitoring, recognition of emotions on faces in and without masks. Computer vision systems can also be used in education to introduce immersive educational resources. On the basis of the analysis of autonomous libraries for the identification of dynamic objects, it is concluded that in the creation of computer vision systems for educational purposes it is advisable to use computer vision libraries based on in-depth learning (in particular, the implementation of convolutional neural networks). A prototype computer vision system developed on the basis of Microsoft Cognitive Toolkit and deployed in the Microsoft Azure cloud is described. The system allows you to perform with a high degree of reliability the main functions: identification of emotions and the presence of a mask on the face, as well as allows you to determine sex, age, hair color, smile intensity, the presence of makeup, glasses, etc.

Keywords: computer vision, COVID-19, mask detection, education.

1 Освітні застосування систем машинного зору

1.1 Витоки та сучасний стан систем машинного зору

Перші згадки про комп'ютерний зір (computer vision) або машинний зір (machine vision) з'являються наприкінці 1950-х рр., й на сьогодні в англомовних джерелах друге поняття є суттєво частіше згадуваним (рис. 1).

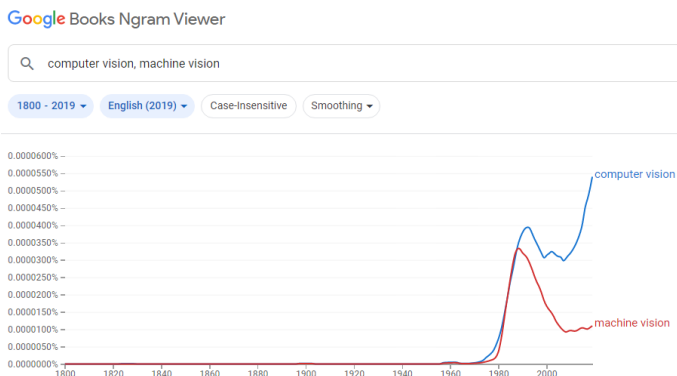


Рис. 1. Згадування про машинний зір у англомовних джерелах [1]

Г.Р.Грейп (Gunnar Rutger Grape) в дисертації [8] згадує роботу «розробника Інтернет» Л.Г.Робертса (Lawrence Gilman Roberts) 1963 року “Machine perception of three-dimensional solids” [14] як

піонерську. Виконана за сприянням К.Шеннона у М.І.Т. Lincoln Laboratory, ця робота, так само, як робота А.Е. Сазерленда (Ivan Edward Sutherland) [17], заклала основи комп'ютерної графіки.

Л.Г.Робертс указує, що в основі машинного розпізнавання простих образів уже в 1960-х рр., традиційним є застосування штучних нейронних мереж типу перцептронів, що розділяють множину вхідних образів (шаблонів) на класи шляхом добору коефіцієнтів зв'язку (вагових коефіцієнтів) перцептрону [17, с. 11]. Проте тодішні розробки для розпізнавання літер були непридатними для розпізнавання зображень, побудованих за допомогою відрізків прямих: проблема стосувалась як плоских (двовимірних), так і просторових (тривимірних) зображень — саме останнім і була присвячена дисертація [17].

Л.Г.Робертс спирався на психологічні дослідження сприйняття людиною глибини та розпізнавання фігур, насамперед — Дж.Дж.Гібсона (James Jerome Gibson) [6]. Результатом роботи Л.Г.Робертса стала система для перетворення фотографічних зображень на тривимірні моделі (рис. 2).

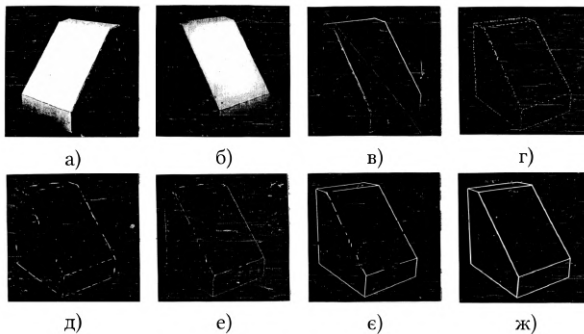


Рис. 2. Перетворення зображення за Л.Г.Робертсом: а) оригінальне зображення; б) оцифроване зображення; в-е) перетворення; ж) результуюча модель [17, с. 35–36]

За 10 років після роботи Л.Г.Робертса стан справ у галузі машинного зору суттєво змінився. Г.Р.Грейп у [8] представив систему машинного зору на основі двовимірних прототипів, що використовує ієрархію функцій для проектування. Входом системи було зображення з ТВ-камери, і проблема, поставлена дослідником — отримання тривимірної проєкції. Оцифроване зображення сканувалося для виявлення градієнтів високої інтенсивності (граней), які, ймовірно,

з'являться на стиках регіонів і об'єктів. Г. Р. Грейп продемонстрував перспективно узгоджені 2D-моделі (прототипи) уявлень про тривимірні об'єкти, а інтерпретації сцен-репрезентацій засновані на встановленні зв'язків елементів сцени із шаблонами прототипів. Система навчалась прототипів шляхом аналізу ідеальних екземплярів та їх узагальнення (рис. 3).

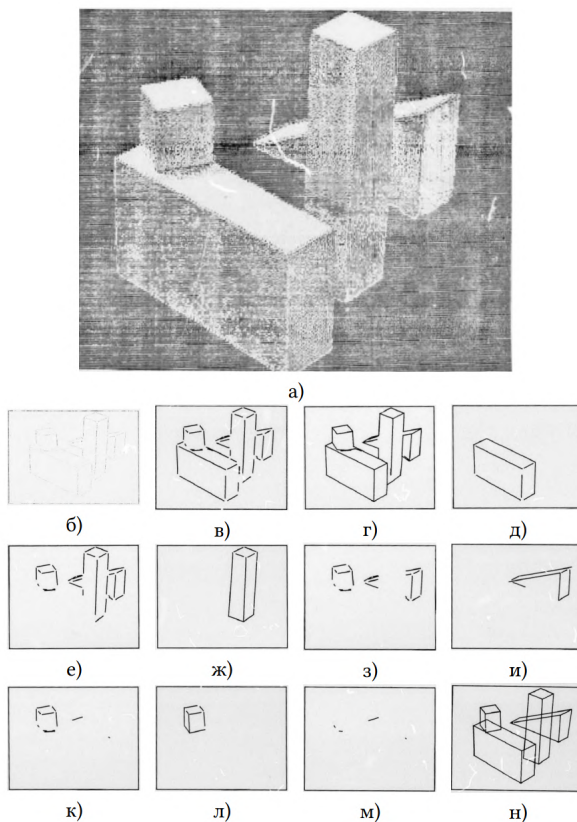


Рис. 3. Перетворення зображення за Г. Р. Грейпом: а) зображення з ТВ-камери; б, в) виділення граней; г, д) виділення першого об'єкту; е, ж) виділення другого об'єкту; з, и) виділення третього об'єкту; к, л) виділення четвертого об'єкту; м, н) фінальна інтерпретація [8, с. 174–181]

Визначаючи майбутні напрямки розвитку систем машинного зору,

Г. Р. Грейп запропонував концептуальну модель опрацювання потоків даних у тривимірній системі (рис. 4).

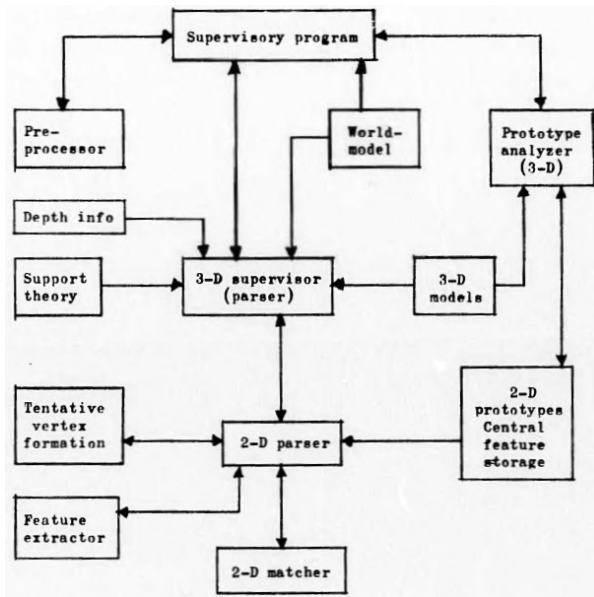


Рис. 4. Опрацювання потоків даних у 3D-системі [8, с. 242]

Через 10 років після роботи Г.Р.Грейпа було виконано перше узагальнення поточного стану справ у галузі машинного зору. Автори першого підручника з машинного зору — “Computer vision” — Д.Г.Баллард (Dana Harry Ballard) та К.М.Браун (Christopher M. Brown) розглядали методи формування зображення, попередньої обробки (відновлення внутрішньої структури, фільтрування зображення, пошук локальних границь, орієнтація поверхні за моделями відбиття, оптичний потік та піраміди роздільної здатності), виявлення границь, об’єднання регіонів, текстурування, розпізнавання руху, подання двовимірних геометричних структур (границь, регіонів), подання тривимірних структур (твердих тіл, поверхонь, циліндричних фігур, об’ємних зображень), подання та використання знань (бази знань, семантичні мережі), виявлення збігів, виведення, досягнення мети [4].

Д. Г. Баллард та К. М. Браун запропонували 4 категорії подання зображень у системах машинного зору (рис. 5). У таблиці 1 наведено

основні галузі застосування систем машинного зору так, як їх бачили автори у 1982 році.

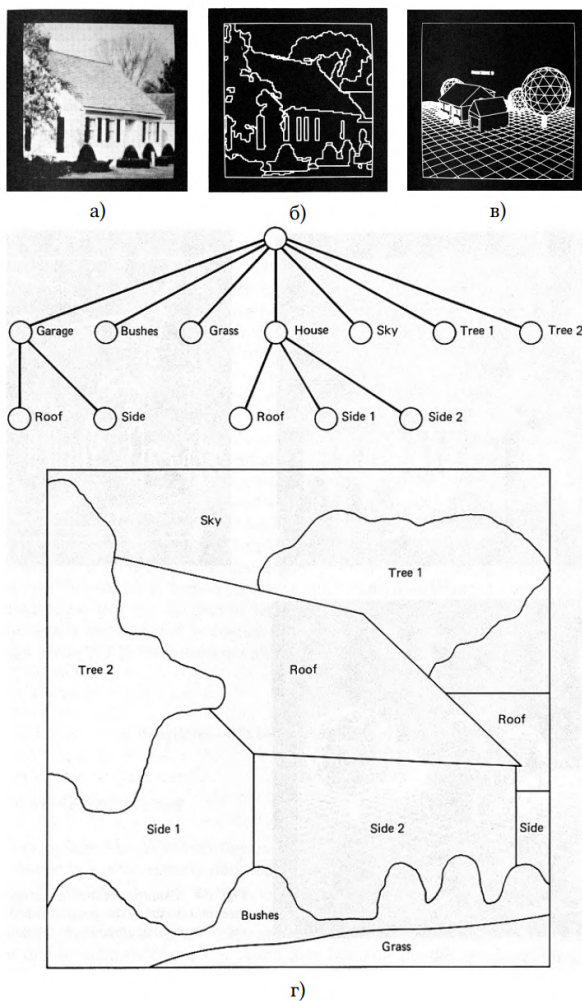


Рис. 5. Категорії подання зображень у системах комп'ютерного зору: узагальнене (а), сегментоване (б), геометричне (в) та реляційне (г) [4, с. 7–8]

Таблиця 1. Приклади задач машинного зору [3, с. 11]

Галузь	Об'єкти	Модальність	Задачі	Джерела інформації
Робототехніка	Тривимірні зовнішні та внутрішні сцени Механічні частини	Світло Х-промені	Ідентифікація чи опис об'єктів на сцені Промислові задачі	Моделі об'єктів Моделі відбиття світла від об'єктів
Аерозйомка	Будова Землі тощо	Світло Інфрачервоне випромінювання Радари	Покращення зображень Аналіз ресурсів Прогнозування погоди Шпіонаж Супровід запусків Тактичний аналіз	Карти Геометричні моделі форм Моделі формування зображень
Астрономія	Зорі Планети	Світло	Хімічний склад Покращення зображень	Геометричні моделі форм
Медицина (макро)	Частини тіла	Х-промені Ультразвук Ізотопи Тепло	Діагностика відхилень Планування операцій та лікування	Анатомічні моделі Моделі формування зображень
Медицина (мікро)	Клітини Ланцюги протеїнів Хромосоми	Електронна мікроскопія Світло	Патологія, цитологія Каріотипування	Моделі форм
Хімія	Молекули	Електронна густина	Аналіз молекулярного складу	Хімічні моделі Структурні моделі
Нейроанатомія	Нейрони	Світло Електронна мікроскопія	Визначення просторової орієнтації	З'єднання нейронів

Продовження таблиці 3

Галузь	Об'єкти	Модаль- ність	Задачі	Джерела інформації
Фізика	Треки частинок	Світло	Знаходження нових частинок Ідентифікація треків	Атомна фізика

«Комп'ютерний зір і машинне навчання одружилися, і ця книга — їхня дитина» — такий відгук на сайті Amazon надав В. Т. Фріман на книгу С. Дж. Д. Прінса (Simon J. D. Prince) «Computer Vision: Models, Learning, and Inference» [10], яка дає основи машинного навчання, необхідні для підтримки поточних застосувань і досліджень у галузі машинного зору.

Отже, за 60 років розвитку системи машинного зору проникли у значну кількість галузей, і насамперед — в освітню, де вони застосовуються для реалізації імерсивних освітніх ресурсів. Проте зміни в освітній галузі, спричинені пандемією COVID-19 [18], не могли не відобразитись на класичних освітніх застосуваннях систем машинного зору, модифікувавши тодішні та породивши нові напрями.

1.2 Зміни освітніх застосувань систем машинного зору, спричинені поширенням COVID-19

Н. Клінглер (Nico Klingler), співзасновник хмаро орієнтованого конструктора для розробки інтелектуальних систем машинного зору *viso.ai*, у популярному огляді [11] виокремив 8 основних освітніх застосувань систем машинного зору в освіті. Зосередимось на тих із них, що є актуальними в умовах поточної пандемії COVID-19.

Дотримання соціального дистанціювання — ключова стратегія боротьби з поширенням COVID-19 у громадських закладах, як от школи й університети. Системи глибокого навчання можна використовувати для моніторингу груп в аудиторії, натовпів на поверхах, щоб аналізувати соціальне дистанціювання, виявляти вузькі місця та запускати сповіщення в разі постійних порушень (рис. 6).

Засоби, необхідні для соціального моніторингу, наприклад, недорогі камери спостереження, уже наявні в окремих школах.

Головна перевага моніторингу соціального дистанціювання за допомогою систем машинного зору — безконтактність та автоматизованість.

У [25] визначені наступні можливості моніторингу соціального дистанціювання за допомогою систем машинного зору:



Рис. 6. Аналіз соціального дистанціювання за допомогою <https://viso.ai/application/social-distancing-monitoring/>

- автоматичне виявлення та відстеження кількох динамічних об'єктів (осіб) у режимі реального часу;
- аналіз траєкторії руху людей та рівня порушення соціального дистанціювання;
- визначення зон підвищеного ризику з найбільшою небезпекою можливого поширення коронавірусу.

Останнє набуває додаткової значущості в умовах поширення високозаразних штамів COVID-19, як-от «омікрон» [16].

Розпізнавання обличчя в масці — спосіб контролювати дотримання та дотримання масок у людних місцях, наприклад, в університетах чи школах. Алгоритми глибокого навчання автоматично виявляють людей без масок і відстежують порушення. Модель розпізнавання обличчя в масках на основі комп'ютерного зору є неінвазивною, масштабованою та порівняно легкою в реалізації, оскільки можна використовувати відеозапис будь-якої камери (рис. 7).

У [13] визначені наступні переваги моніторингу маскового режиму за допомогою інтелектуальних систем машинного зору:

- автоматичний моніторинг дотримання маскового режиму є більш послідовним і точним, ніж перевірка людиною;
- виявлення людей без масок підвищує безпеку, знижуючи ризик передачі COVID-19 та інших інфекційних захворювань;

- респіратори або маски для обличчя на робочому місці можуть запобігти спалахам заражень в школі та університеті, особливо при поширенні штамів високої заразності.

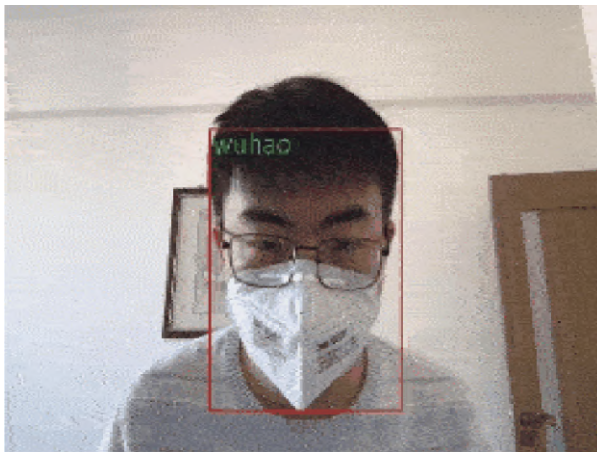


Рис. 7. Розпізнавання обличчя в масці за допомогою
<https://viso.ai/application/mask-detection/>

М. Резай (Mahdi Rezaei) та М. Азармі (Mohsen Azarmi) розроблено DeepSOCIAL [18] — гібридну систему машинного зору на основі глибокого навчання для автоматизованого виявлення людей у натовпі в приміщеннях та на вулиці за допомогою звичайних камер відеоспостереження (рис. 8). Запропонована авторами модель у поєднанні з адаптованою технікою відображення зворотної перспективи й алгоритмом відстеження SORT веде до надійного виявлення людей та моніторингу соціального дистанціювання. Модель навчена на основі двох наборів даних — Microsoft Common Objects in Context (MS COCO) і Google Open Image. Оцінку моделі проведено в складних умовах, включаючи часткову видимість та при змінах освітлення із середньою точністю 99.8% та швидкість в реальному часі 24,1 кадр/с. DeepSOCIAL також надає онлайн-схему оцінки ризику зараження шляхом статистичного аналізу просторово-часових даних про траєкторії пересування людей і частоти порушень соціального дистанціювання, визначаючи зони високого ризику з найбільшою ймовірністю поширення вірусу й інфікування. Це може допомогти органам влади змінити планування громадських місць або вжити

запобіжних заходів, щоб пом'якшити зони високого ризику. Розроблену модель можна застосувати в багатьох інших областях, як-от автономні транспортні засоби, розпізнавання людських дій, виявлення аномалій.

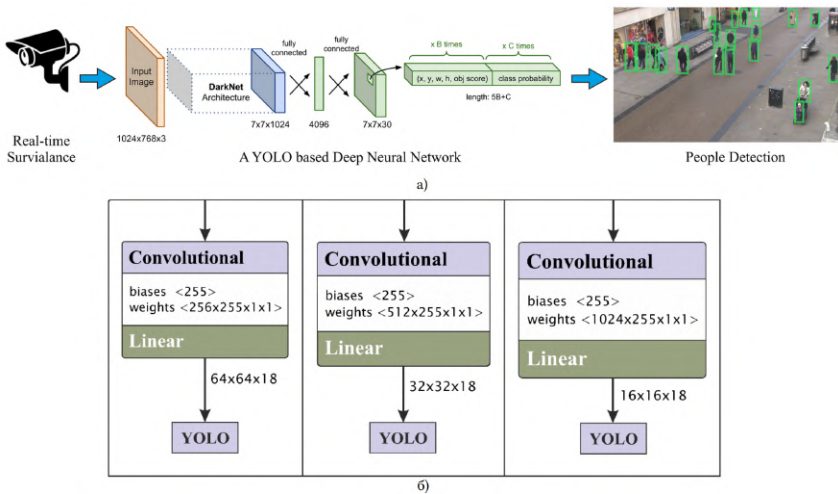


Рис. 8. Загальна структура системи машинного зору DeepSOCIAL (а) та архітектура нейронної мережі (б) [18, с. 7, 10]

К.Шортен (Connor Shorten), Т.М.Хошгофтаар (Taghi M. Khoshgoftaar) та Б.Фурхт (Borko Furht) в огляді [20] наводять граф застосувань глибокого навчання, серед яких чільне місце займають застосування машинного зору (рис. 9).

Системи управління паркуванням для виявлення заповненості автостоянок при школах або університетах є ще одним класичним застосуванням машинного зору, корисним й у пандемічний період. Камери відеоспостереження забезпечують відеозапис, який можна використовувати для автоматичного визначення та відстеження зайнятості кількох паркувальних місць. Інформацію про доступні місця для паркування можна візуалізувати на інформаційних панелях і надсилати до сторонніх систем, щоб надавати дані в режимі реального часу студентам та викладачам.

У [24] визначені наступні переваги системи управління паркуванням для виявлення заповненості автостоянок при школах або університетах:

- виявлення зайнятості автостоянки на основі зору з глибоким навчанням економить витрати порівнюючи з дорогими

сенсорними методами;

- оптимізація паркування допомагає зменшити затори в години пік і транспортні потоки в містах, викликані людьми, які шукають місця для паркування (близько 30%);
- для рішень на основі камер не потрібна додаткова інфраструктура, за умови, що об'єкт оснащений камерами відеоспостереження, які охоплюють місця для паркування;
- рішення на основі машинного зору забезпечують точне розташування вільного паркувального місця.

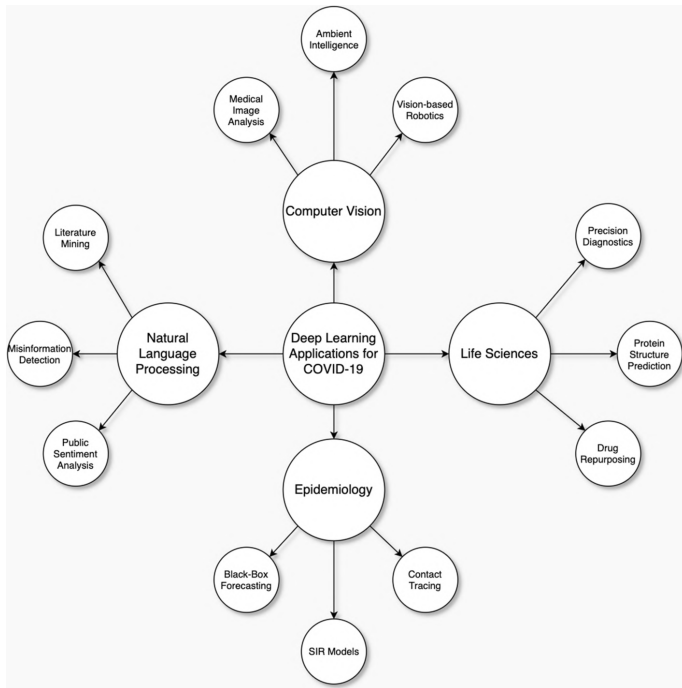


Рис. 9. Застосування глибокого навчання для подолання наслідків COVID-19 [20, с. 18]

Виявлення вторгнень в університетах і школах — застосування глибокого навчання зі звичайними камерами спостереження для здійснення моніторингу периметра й автоматичного виявлення зломисників (рис. 10).

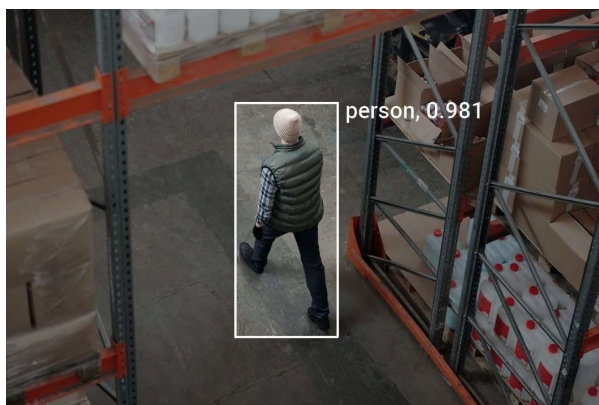


Рис. 10. Виявлення вторгнень за допомогою <https://viso.ai/application/intrusion-detection/>

Переваги застосування систем виявлення вторгнень [22]:

- наявні алгоритми виявлення об'єктів у режимі реального часу для виявлення людей та об'єктів на відео з кількох камер;
- підозрілі області можна візуально визначити, намалювавши та назвавши їх у кадрах камери;
- ініціювання сповіщень на основі того, як об'єкти виявляються в цих зонах (наприклад, після того, як людина входить у зону більше ніж на 5 секунд);
- можливість застосування Edge AI — машинного навчання на мобільному пристрої для збереження конфіденційності.

Системи запобігання вандалізму та виявлення підозрілих об'єктів без нагляду використовуються для виявлення потенційно небезпечних предметів (валіз або сумок, які можуть містити вибухівку або біологічні засоби) у громадських місцях (рис. 11).

Переваги застосування систем запобігання вандалізму та виявлення підозрілих об'єктів без нагляду [19]:

- класифікація предметів для визначення типу предмета (валіза, сумка тощо);
- фокусування на певних областях у потоках камери (платформи, поверхні, ліфти тощо);

- масштабований та об'єктивний моніторинг громадських місць, громадського транспорту, шкіл чи лікарень;
- раннє виявлення в реальному часі, щоб допомогти людям розпізнати підозрілі предмети й організувати евакуацію.

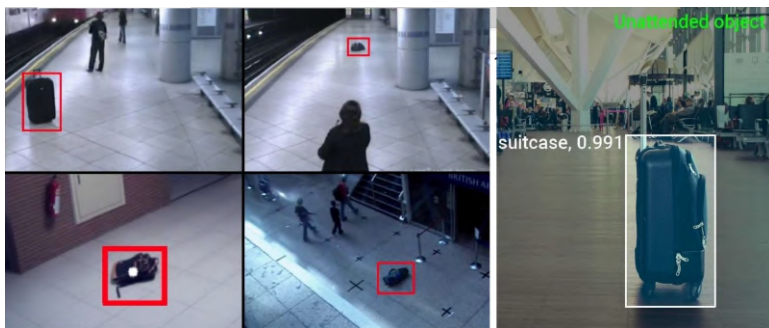


Рис. 11. Виявлення підозрілих об'єктів без нагляду за допомогою <https://viso.ai/application/abandoned-luggagedetection/>

Відеопотік можна використовувати для ідентифікації емоційних станів однієї або кількох осіб, застосовуючи алгоритми глибокого навчання, зокрема для *розпізнавання емоцій на обличчях у масках та без масок* (рис. 12).

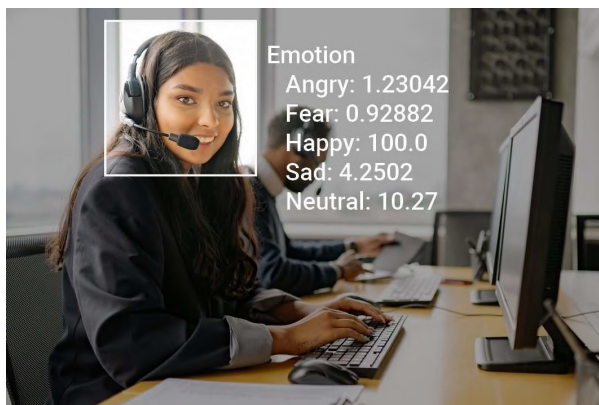


Рис. 12. Розпізнавання емоцій за допомогою <https://viso.ai/application/emotion-analysis/>

Системи розпізнавання емоцій надають можливості [21]:

- виявлення емоційних станів: смуток, гнів, щастя, страх, здивування, нейтральний стан;
- визначення зміни емоційних станів залежно від конкретних умов навчання;
- визначення оцінки впевненості для розпізнаних емоцій.

Моніторинг відвідуваності — розпізнавання та пошук обличчя в базі даних зображень для ідентифікації студентів (рис. 13).



Рис. 13. Розпізнавання обличч за допомогою <https://viso.ai/application/face-recognition/>

Переваги систем моніторингу відвідуваності [20]:

- автоматизована та неінвазивна ідентифікація однієї або кількох людей;
- розумне відеоспостереження для захисту інфраструктури установи;
- наявні моделі глибокого навчання від Google, Facebook, стійкі до виразу обличчя, освітлення та пози.

Такі системи можуть бути реалізовані апаратно на основі Raspberry-рі [16] із застосуванням згорткових нейронних мереж — класу глибинних нейронних мереж, найбільш часто застосовуваних для аналізу візуальних зображень [3, 12, 16] та реалізованих у таких бібліотеках, як Caffe, Deeplearning4j, Dlib, Microsoft Cognitive Toolkit, TensorFlow, Theano, Torch.

2 Проектування прототипу системи машинного зору освітнього призначення

2.1 Бібліотеки машинного зору для ідентифікації динамічних об'єктів

Для ідентифікації динамічних об'єктів найчастіше застосовуються наступні автономні бібліотеки:

- `fastai` — бібліотека глибокого навчання із функціями комп'ютерного зору (Python);
- `IPSDK` — бібліотека обробки 2D і 3D зображень (C++, Python);
- `Imutils` — бібліотека комп'ютерного зору, надбудова над `OpenCV` (C++, Python);
- `Keras` — бібліотека нейронних мереж високого рівня (Python), що включає підтримку згорткових і рекурентних мереж для розпізнавання зображень (Python);
- `OpenCV` — бібліотека комп'ютерного зору, орієнтована на програми реального часу й аналіз відео (C++ та ін.);
- `PyTorchCV` — фреймворк на основі `PyTorch` для задач машинного зору: класифікації зображень, сегментації, виявлення й оцінки пози. До складу входять класичні моделі, включаючи `AlexNet`, `ResNet`, `ResNeXt`, `PyramidNet`, `SparseNet`, `DRN-C/DRN-D` тощо (Python);
- `Scikit-Image` — бібліотека для обробки зображень, надбудова `SciPy` (Python);
- `SimpleCV` — бібліотека машинного зору, що забезпечує інтерфейс до камери, маніпулювання зображеннями або відеопотоками (Python).

Їх спільними рисами є вільні ліцензії, використання Python для швидкого прототипування, нейронних мереж для розпізнавання та можливість використання веб-платформ для розгортання. Останнє часто є платною опцією, як і опрацювання даних у хмарі розробника. Попри широкий спектр можливостей, що їх надає `viso.ai` ([19, 21] та ін.), цей сервіс для конструювання систем машинного зору із машинним навчанням має високу вартість, тому є малопридатним для вітчизняної системи освіти. Подібна ситуація спостерігається й з іншими комерційними рішеннями. Так, фахівці `Adaptive Vision`, порівнюючи функціональні можливості `OpenCV`, `MVtec HALCON`

та Adaptive Vision Library (AVL) [2], роблять висновки не на користь першої — особливо значущі відмінності за функціональністю на користь комерційних продуктів HALCON та AVL у таких суттєвих для ефективності розпізнавання компонентах, як аналіз областей, робота з об'єднанням, двовимірною графікою та машинне навчання.

При застосуванні хмарних фреймворків для розпізнавання, як-от як Google Cloud [7], вартість визначається за одиницю (одне зображення або сторінка багатосторінкового зображення). Наприклад, при обмеженні до 1000 одиниць на місяць Cloud Vision надає наступні можливості:

- CROP_HINTS — визначення вершин для області кадрування на зображенні;
- DOCUMENT_TEXT_DETECTION — оптичне розпізнавання зображень із щільним друкованим текстом, як-от документи (PDF/TIFF), та зображень із рукописним текстом;
- FACE_DETECTION — визначення обличчя на зображенні;
- IMAGE_PROPERTIES — визначення набору властивостей зображення, як-от домінуючі кольори;
- LABEL_DETECTION — додавання міток на основі вмісту зображення;
- LABEL_DETECTION — додавання міток на основі вмісту зображення;
- LOGO_DETECTION — визначення логотипу компанії на зображенні;
- OBJECT_LOCALIZATION — виявлення кількох об'єктів на зображенні;
- TEXT_DETECTION — оптичне розпізнавання зображень із розрідженим друкованим текстом;
- WEB_DETECTION — виявлення на зображенні актуальних об'єктів, наприклад, із новин, подій чи знаменитостей, і пошук подібних зображень в Інтернеті за допомогою пошуку зображень Google.

Будь-які нестандартні дії, зокрема, побудова власних моделей машинного навчання для ідентифікації з використанням Vertex AI, потребує оплати, проте для початківців часто надається певний кредит — такої політики наразі дотримуються Google Cloud, AWS,

Microsoft Azure, що надає можливість їх розгляду як технологічної основи для побудови систем машинного зору освітнього призначення.

2.2 Розробка прототипу системи машинного зору освітнього призначення

2.2.1 Розпізнавання статичних облич у Microsoft Azure

Практичне заняття із розпізнавання емоцій “Happy, Sad, Angry Workshop”, розроблене Дж. Беннеттом (Jim Bennett) [5], надає можливість створення прототипу відповідної системи машинного зору з використанням Azure Face API — програмного інтерфейсу, що дозволяє визначити положення обличчя на зображенні, 27 точок на зображенні включно із позицією очей, а також маски.

Для доступу до API є необхідним підписний ключ (subscription key), який можна отримати через вільну реєстрацію на <https://azure.microsoft.com/en-us/free/students> (Azure for Students Starter). При цьому не потребується використання кредитної карти, хоча й надається кредит на послуги. На жаль, поштові адреси в домені другого рівня (зокрема, Криворізького державного педагогічного університету) не розпізнаються як такі, що належать академічним установам. У цьому випадку рекомендується звичайна реєстрація (рис. 14).

Після отримання ключа його необхідно перевірити за допомогою тестового коду (рис. 15) або у веб-формі (рис. 16). У випадку неправильного ключа результатом буде повідомлення про помилку 400, 401, 403, 408, 415 або 429:

```
b'{"error":{"code":"401","message":"Access denied due to invalid subscription key or wrong API endpoint. Make sure to provide a valid key for an active subscription and use a correct regional API endpoint for your resource."}}'
```

Результат, що повертається у форматі JSON, надає можливість отримати велику кількість атрибутів, пов'язаних із обличчям.

Слід звернути увагу, що для різних моделей визначення (detectionModel) доступні різні атрибути обличчя: для detection_03 — headpose, mask, qualityforrecognition, для detection_01 — accessories, age, blur, emotion, exposure, facialhair, gender, glasses, hair, headpose, makeup, noise, occlusion, qualityforrecognition, smile.

Рекомендована модель розпізнавання (recognitionModel) — recognition_04: саме за цією моделлю, уведеною в лютому 2021 року, відбувається розпізнавання масок.

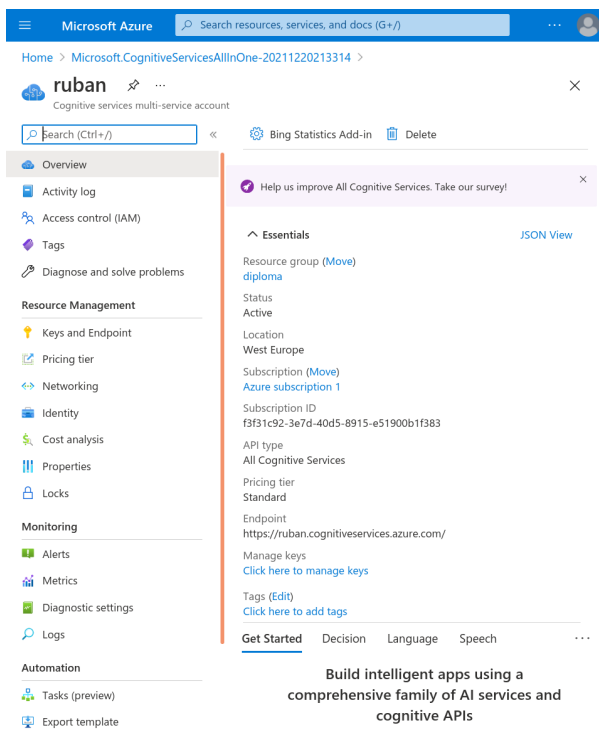


Рис. 14. Реєстрація у Cognitive Services

Azure надає можливість звернення й безпосередньо до кінцевої точки (endpoint) — у нашому випадку нею є `https://ruban.cognitiveservices.azure.com`.

Розглянемо код для отримання властивостей правого зображення з рис. 16:

```
import json, os, requests
subscription_key = "секретний підписний ключ"
face_api_url = "https://ruban.cognitiveservices.azure.com" +
'/face/v1.0/detect'
image_url =
'https://kdpu.edu.ua/images/ipm/%D0%90%D1%81%D0%BF%D1%96%D1%8
0%D0%0%D0%BD%D1%82%D1%83%D1%80%D0%B0/%D0%B3%D0%B0%D1%80%D0%B0
%D0%BD%D1%82%D1%81%D1%81_2.jpg'
```

```
+ Code + Text
##### Python 3.2 #####
import http.client, urllib.request, urllib.parse, urllib.error, base64

headers = {
    # Request headers
    'Content-Type': 'application/json',
    'Ocp-Apim-Subscription-Key': '663cf5499f3748b4812146c23913f7e8',
}

params = urllib.parse.urlencode({
    # Request parameters
    'returnFaceId': 'true',
    'returnFaceLandmarks': 'false',
    'returnFaceAttributes': '{string}',
    'recognitionModel': 'recognition_04',
    'returnRecognitionModel': 'false',
    'detectionModel': 'detection_03',
    'faceIdTimeToLive': '86400',
})

body= urllib.parse.urlencode({
    "uri": "https://kdpu.edu.ua/semerikov/cc.jpg"
})

try:
    conn = http.client.HTTPSConnection('westus.api.cognitive.microsoft.com')
    conn.request("POST", "/face/v1.0/detect?%s" % params, body, headers)
    response = conn.getresponse()
    data = response.read()
    print(data)
    conn.close()
except Exception as e:
    print("[Errno {0}] {1}".format(e.errno, e.strerror))

#####

b'{"error":{"code":"401","message":"Access denied due to invalid subscription"}}
```

Рис. 15. Перевірка ключа за допомогою тестового коду

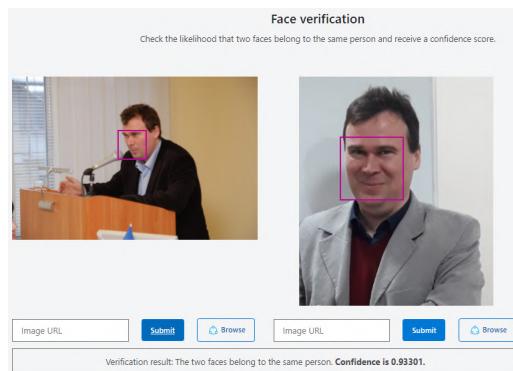


Рис. 16. Результати ідентифікації статичного зображення

```

headers = {'Ocp-Apim-Subscription-Key': subscription_key}
params = {
    'detectionModel': 'detection_03',
    'returnFaceId': 'true',
    'returnFaceLandmarks': 'true',
    'returnFaceAttributes': 'headpose,mask,qualityforrecognition',
    #для detection_03
# 'returnFaceAttributes':
# 'accessories,age,blur,emotion,exposure,facialhair,gender,'+
# 'glasses,hair,headpose,makeup,noise,occlusion,'+
# 'qualityforrecognition,smile', #для detection_01
'recognitionModel': 'recognition_04',
'returnRecognitionModel': 'false',
#'detectionModel': 'detection_01',
'faceIdTimeToLive': '86400',
}
response = requests.post(face_api_url, params=params,
headers=headers, json={"url": image_url})
print(json.dumps(response.json()))

```

Для визначення наявності маски на обличчі скористаємося наведеним кодом. Результат запиту для другого обличчя з рис. 14 зображення подано у табл. 2.

Таблиця 2. Результат запиту до Face API для моделі виявлення detection_03

Атрибут	Значення	Коментар
"faceId"	"1b98e2a9-0c6c-4864-91d7-b8281e9377cc"	унікальний ідентифікатор виявленого обличчя, створений за допомогою Face API, яким можна скористатись протягом 24 годин з моменту отримання
"faceRectangle"	"top": 77, "left": 92, "width": 144, "height": 191	область прямокутника, у якому на зображенні розташовано обличчя

Атрибут	Значення	Коментар
"faceLandmarks"	"pupilLeft": "x": 122.8, "y": 156.0, "pupilRight": "x": 180.7, "y": 157.2, "noseTip": "x": 143.7, "y": 196.2, "mouthLeft": "x": 123.8, "y": 219.1, "mouthRight": "x": 179.4, "y": 220.6, "eyebrowLeftOuter": "x": 102.5, "y": 147.1, "eyebrowLeftInner": "x": 136.1, "y": 147.3, "eyeLeftOuter": "x": 113.8, "y": 156.1, "eyeLeftTop": "x": 123.1, "y": 152.9, "eyeLeftBottom": "x": 122.4, "y": 158.6, "eyeLeftInner": "x": 131.9, "y": 156.4, "eyebrowRightInner": "x": 161.3, "y": 147.0, "eyebrowRightOuter": "x": 202.7, "y": 150.0, "eyeRightInner": "x": 171.4, "y": 157.4, "eyeRightTop": "x": 180.3, "y": 153.9, "eyeRightBottom": "x": 180.5, "y": 159.8, "eyeRightOuter": "x": 190.6, "y": 157.6, "noseRootLeft": "x": 141.0, "y": 159.6, "noseRootRight": "x": 157.1, "y": 160.2, "noseLeftAlarTop": "x": 134.5, "y": 182.5, "noseRightAlarTop": "x": 161.7, "y": 183.0, "noseLeftAlarOutTip": "x": 128.6, "y": 194.4, "noseRightAlarOutTip": 30	набір орієнтирів із 27 точок, що вказують на важливі позиції компонентів обличчя

Продовження таблиці 3

Атрибут	Значення	Коментар
	"x": 167.7, "y": 196.1, "upperLipTop": "x": 148.5, "y": 218.2, "upperLipBottom": "x": 147.8, "y": 222.1, "underLipTop": "x": 147.7, "y": 226.0, "underLipBottom": "x": 147.5, "y": 233.0	
"faceAttributes"		атрибути обличчя
"headPose"	"pitch": -14.2, "roll": 1.3, "yaw": -9.3	3-D кути тангажу/ нахилу/повороту обличчя
"mask"	"type": "noMask" "noseAndMouthCo- vered": false	визначає наявність маски на обличчі. Можливі варіанти: 'noMask', 'faceMask', 'otherMaskOrOcclusion' або 'uncertain'. Логічне значення 'noseAndMouthCovered' вказує, чи закриті ніс і рот
"qualityForRecognition"	"medium"	загальна якість зображення: низька, середня або висока. Для реєстрації особи рекомендовано лише зображення високої якості, а для сценаріїв ідентифікації — середньої або вище. Цей атрибут доступний лише при використанні будь-яких комбінацій моделей виявлення detection_01 або detection_03 і моделей

Продовження таблиці 3

Атрибут	Значення	Коментар
		розпізнавання recognition_03 або recognition_04

Встановивши параметр returnFaceAttributes у 'accessories, age, blur, emotion, exposure, facialhair, gender, glasses, hair, head-pose, makeup, noise, occlusion, qualityforrecognition, smile' та detectionModel у 'detection_01', отримаємо наступні атрибути для того ж обличчя (Табл. 3).

Таблиця 3. Результат запиту до Face API для моделі виявлення detection_01

Атрибут	Значення	Коментар
"smile"	0.999	інтенсивність посмішки, число між [0,1]
"gender"	"male"	стать: чоловіча або жіноча
"age"	44.0	приблизне число «зорового віку» в роках (на скільки років виглядає людина, а не фактичний біологічний вік)
"facialHair"	"moustache": 0.1, "beard": 0.1, "sideburns": 0.1	довжина волосся в трьох зонах обличчя: вуса, борода та бакенбарди – число між [0, 1]: 0 для відсутності волосся на обличчі в цій області, 1 для довгого або дуже густого волосся на обличчі в цій області
"glasses"	"NoGlasses"	наявність окулярів: 'NoGlasses',

Продовження таблиці 3

Атрибут	Значення	Коментар
		'ReadingGlasses', 'Sunglasses', 'SwimmingGoggles'
"emotion"	"anger": 0.0, "contempt": 0.001, "disgust": 0.0, "fear": 0.0, "happiness": 0.999, "neutral": 0.0, "sadness": 0.0, "surprise": 0.0	інтенсивність емоцій, включаючи гнів, презирство, огиду, страх, щастя, нейтральні, смуток і здивування
"blur"	"blurLevel": "medium "value": 0.48	розмиття обличчя: blurLevel повертає низький, середній або високий рівень, value — число між [0, 1]: чим більше воно, тим більше розмиття
"exposure"	"exposureLevel": "goodExposure "value": 0.69	рівень експозиції обличчя: exposureLevel повертає 'GoodExposure', 'OverExposure' або 'UnderExposure', value — число між [0, 1]
"noise"	"noiseLevel": "low "value": 0.0	рівень шуму пікселів обличчя; noiseLevel повертає низький, середній і високий рівень, value — число між [0, 1]: чим воно більше, тим більш зашумлене зображення
"makeup"	"eyeMakeup": "lipMakeup": false	false, макіяж: нафарбовані області очей та губ чи ні

Продовження таблиці 3

Атрибут	Значення	Коментар
"accessories"	[]	аксесуари навколо обличчя, включаючи головні убори (headwear), окуляри (glasses) та маску (mask); порожній масив означає, що аксесуари не виявлені
"occlusion"	"foreheadOccluded": false, "eyeOccluded": false, "mouthOccluded": false	чи кожна область обличчя закрита, включаючи лоб, очі та рот
"hair"		група значень для волосся
"bald"	0.22	частка лисини
"invisible"	false	вказує, чи є волосся видиме
"hairColor"	["color": "black "confidence": 0.98, "color": "brown "confidence": 0.97, "color": "gray "confidence": 0.41, "color": "other "confidence": 0.16, "color": "blond "confidence": 0.08, "color": "red "confidence": 0.04, "color": "white "confidence": 0.0]	визначає колір волосся, якщо його видно

Встановлення detectionModel у 'detection_01' призвело до отримання високого значення qualityForRecognition — для одного й того самого зображення це відповідає різним алгоритмам, що застосовуються для аналізу зображення.

2.2.2 Розпізнавання динамічних облич у Microsoft Azure

Розширення [5] розпочнемо зі зміни операційної системи з Windows/macOS на Linux — остання є широко поширеною у

вітчизняних закладах вищої освіти. Для роботи із засобами Azure може бути використане будь-яке середовище програмування, зокрема і спеціалізований текстовий редактор Visual Studio Code, у якому необхідно встановити модуль розширення Python Extension for Visual Studio Code та Azure App Service Extension for Visual Studio Code (рис. 17).

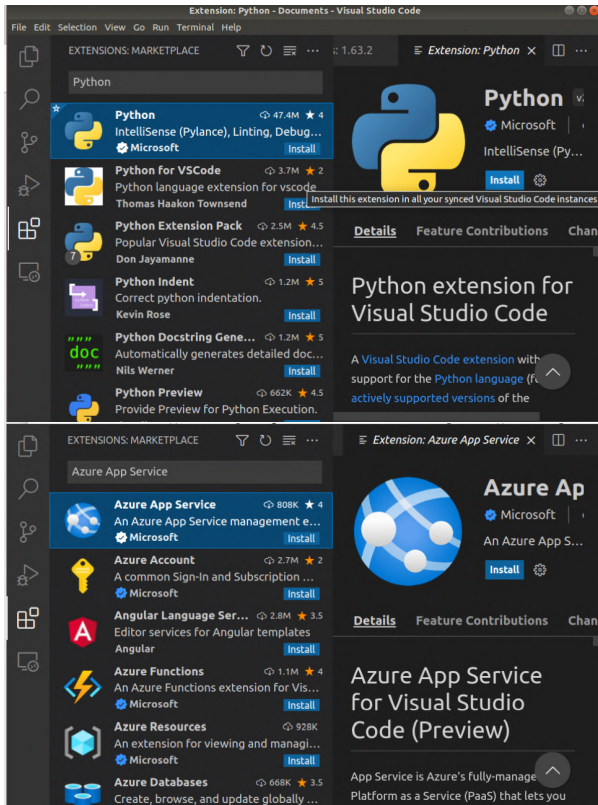


Рис. 17. Розширення Visual Studio Code, необхідні для розробки

Мікрофреймворк Flask для створення веб-додатків встановлюється командою

```
pip3 install flask
```

Якщо утиліта `pip3` не встановлена, вона може бути додана викликом пакетного менеджера. Наприклад:

```
sudo apt install python3-pip
```

Використання Python 2.7 можливе, але не рекомендується, хоча Azure надає змогу застосування різних версій Python (наразі — до 3.9 включно). Налаштування Visual Studio Code для використання Flask виконаємо за процедурою, описаною Дж. Беннеттом (рис. 18). Це дозволяє налагодити серверну частину системи. У процесі налаштування виявилось, що для правильної роботи Flask у терміналі необхідно налаштувати змінну оточення `FLASK_APP`, встановивши її в повний шлях до серверної частини системи — файлу `app.py` (рис. 19, додаток В), а також увести зміни до файлу конфігурації Flask у Visual Studio Code (рис. 20).

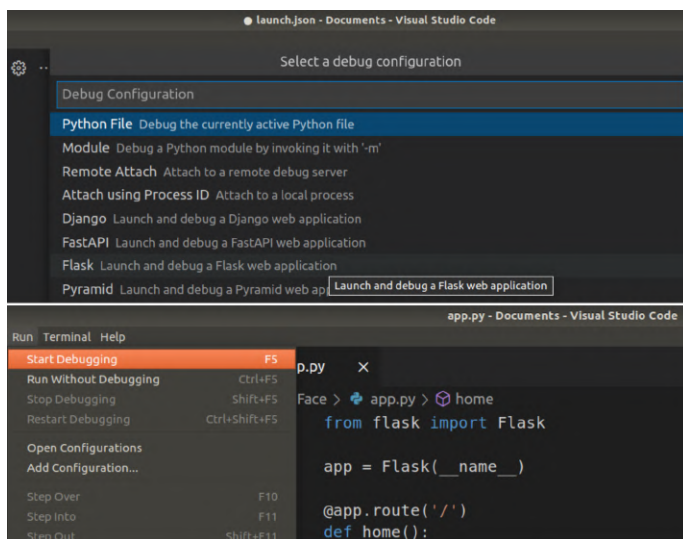


Рис. 18. Налаштування Visual Studio Code для запуску серверної частини з використанням Flask

```

/home/cc/Documents/MaskFace$ export FLASK_APP=/home/cc/Documents/MaskFace/app.py
/home/cc/Documents/MaskFace$ flask run
* Serving Flask app "app"
* Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)

```

Рис. 19. Запуск Flask у терміналі Linux

У процесі налагодження серверної частини системи її код можна змінювати. При цьому відбувається автоматичний перезапуск Flask.

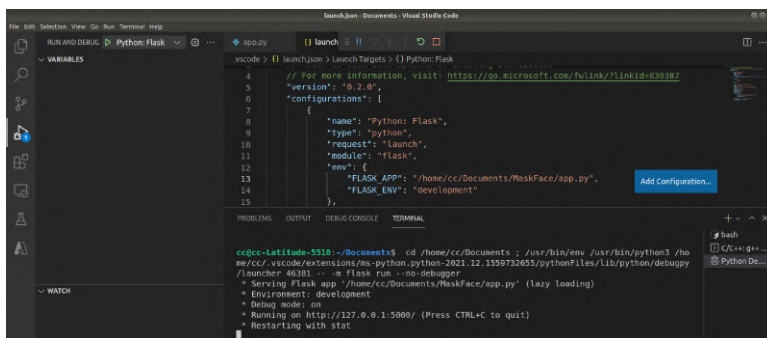


Рис. 20. Запуск Flask у терміналі Visual Studio Code

Клієнтська частина системи представлена одним файлом `home.html` (додаток А), розміщеному в каталозі `templates`.

Для забезпечення доступності розроблюваної системи доцільним є її розміщення у хмарі Azure. Для цього застосовується такий компонент Azure App Service, як `Deploy to Web App` (рис. 21). Це потребує налаштування входу до Azure безпосередньо з Visual Studio Code, визначення імені системи (першої частини доменного імені), вибору версії Python та тарифного плану (рис. 22).

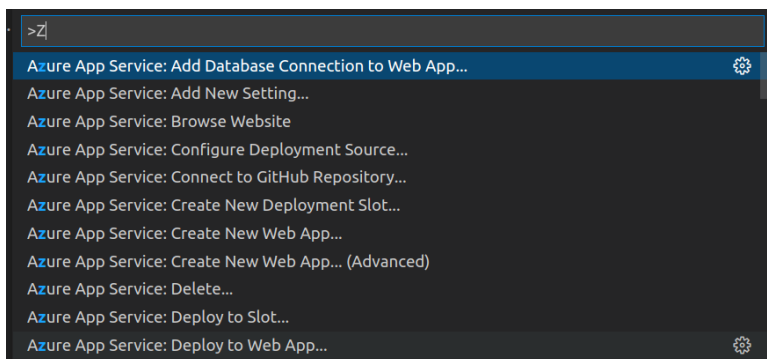


Рис. 21. Налаштування програмного забезпечення у хмарі Azure

На рис. 23 подано протокол розгортання системи у хмарі.

Моделі машинного навчання можуть бути створені як повністю самостійно, так і на основі моделей, створених іншими. Microsoft пропонує ряд попередньо навчених моделей, які й називаються `Cognitive Services`. Ці моделі включають розпізнавання зображень, розпізнавання

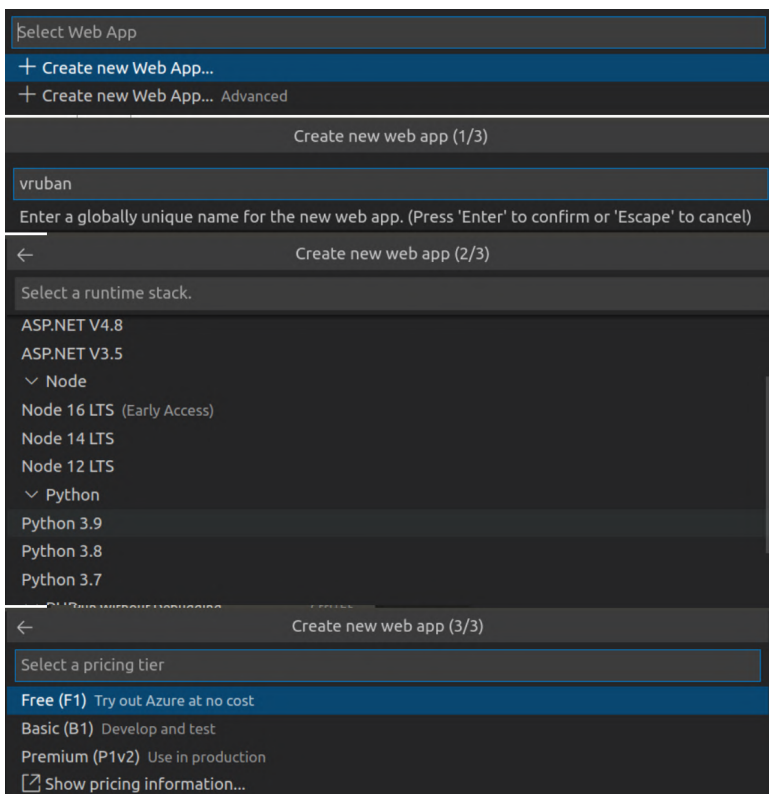


Рис. 22. Кроки розгортання системи у хмарі

мовлення та переклад між різними мовами. Face API реалізує моделі машинного навчання для пошуку облич на зображенні. Для доступу до нього є необхідним ключ підписки, який можна отримати на порталі Azure (рис. 24).

Ім'я створюваного ресурсу повинно бути глобально унікальним, оскільки воно стане частиною URL-адреси, до якої необхідно звернутись для ідентифікації обличчя на зображенні. Місце для запуску цього коду (регіон Azure — група центрів обробки даних) доцільно обирати найближчим.

Ціновий рівень має суттєве значення для програми освітнього спрямування. Якщо за допомогою програми буде здійснюватись менше 2037 викликів API на хвилину та менше 30000 викликів на місяць,

можна обрати безкоштовний рівень F0 — лише один для кожної служби Azure.

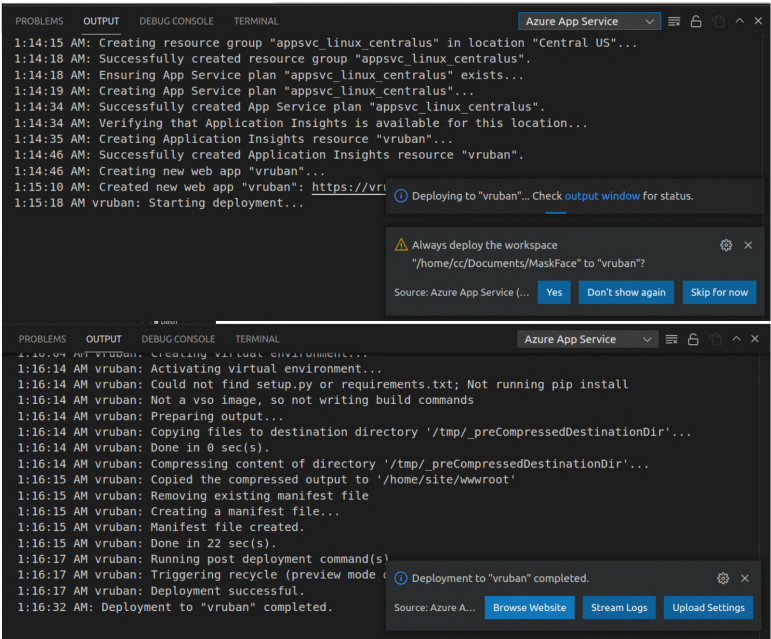


Рис. 23. Протокол розгортання системи у хмарі

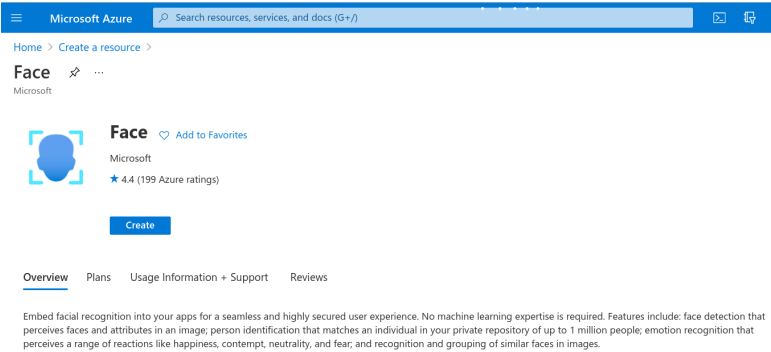


Рис. 24. Створення ресурсу для використання Face API

За умови успішного налаштування (рис. 25) можна буде отримати

секретні ключі та кінцеву точку для доступу до Face API (рис. 26).

Basics

Network

Identity

Tags

Review + create

Configure network security for your cognitive service resource.

17

Type *

☒ All networks, including the internet, can access this resource.

☐ Selected networks, configure network security for your cognitive service resource.

☐ Disabled, no networks can access this resource. You could configure private endpoint connections that will be the exclusive way to access this resource.

Validation Passed

By clicking "Create", I (a) agree to the legal terms and privacy statement(s) associated with the Marketplace offering(s) listed above; (b) authorize Microsoft to bill my current payment method for the fees associated with the offering(s), with the same billing frequency as my Azure subscription; and (c) agree that Microsoft may share my contact, usage and transactional information with the provider(s) of the offering(s) for support, billing and other transactional activities. Microsoft does not provide rights for third-party offerings. See the [Azure Marketplace Terms](#) for additional details.

Basics

Subscription

Azure subscription 1

Region

Central US

Name

ruban1

Pricing tier

Free F0 (20 Calls per minute, 30K call per month)

Network

Type

All networks, including the internet, can access this resource.

Identity

Identity type

None

Рис. 25. Налаштування ресурсу для використання Face API

Microsoft Azure

Search resources, services, and docs (2/11)

Home

>

Microsoft.CognitiveServicesFace-20211221012324

>

ruban1

ruban1 | Keys and Endpoint

Face API

Search (Ctrl+V)

Regenerate key1

Regenerate key2

Overview

Activity log

Access control (IAM)

Tags

Diagnose and solve problems

Resource Management

Quick start

Keys and Endpoint

Pricing tier

Networking

Identity

Cost analysis

Properties

These keys are used to access your Cognitive Service API. Do not share your keys. Store them securely– for example, using Azure Key Vault. We also recommend regenerating these keys regularly. Only one key is necessary to make an API call. When regenerating the first key, you can use the second key for continued access to the service.

Show Keys

KEY 1

.....

KEY 2

.....

Location/Region

Centralus

Endpoint

https://ruban1.cognitiveservices.azure.com/

Рис. 26. Отримання секретних ключів та адреси кінцевої точки для доступу

~ 39 ~

Обидва ключі є рівноправними — можна використати будь-який із них. Створена кінцева точка може бути застосована для доступу до Face API з багатьох програм, зокрема у спосіб, описаний у п. 2.2.1. Дж. Беннетт наголошує на недоцільності їх зберігання у програмному коді, вбачаючи альтернативою зберігання у сторонніх файлах, як-от .env.

У процесі доопрацювання коду [5] були виконані наступні основні зміни у клієнтській частині:

- (1) підвищено роздільну здатність до 1024x768 для кращого розпізнавання обличч — Face API визначає мінімальний розмір обличчя 36x36;
- (2) через те, що деталі, пов'язані з маскою, не є сумісними з іншими атрибутами, виконано розділення пов'язаних із ними дій на два різні обробники, асоційовані з відповідними кнопками «Чи я у масці?» (рис. 27) та «Проаналізувати обличчя».

Уведення кнопок, за якими здійснюється захоплення динамічного зображення, пов'язане з обраним тарифним планом: автоматичне звернення до Face API навіть 15 разів на секунду вичерпує місячний ліміт за півгодини.

Серверна частина зазнала найбільших змін:

- (1) поряд із функцією best_emotion була уведена best_color для визначення кольору волосся;
- (2) функція check_results, що опрацьовує натискання кнопки «Чи я у масці?», застосовує моделі розпізнавання recognition_04 та detection_03 для отримання атрибутів маски та повернення відповідної відповіді ('На Вашому обличчі немає маски', 'На Вашому обличчі є маска', 'Ваше обличчя чимось закрито', 'Неможливо визначити, чи є на Вашому обличчі маска', 'Ваші рот та ніс закриті', 'Ваші рот та ніс не закриті');
- (3) за кнопку «Проаналізувати обличчя» відбувається отримання таких атрибутів обличчя, як 'emotion', 'accessories', 'age', 'facialhair', 'gender', 'glasses', 'hair', 'makeup' та 'smile'.

2.2.3 Тестування розробленого прототипу

На рис. 27 подано результати тестування системи у режимі визначення маски: із 4 тестів 3 виконано правильно, а один (рис. 27 г) — частково правильно (визначено, що рот маскою не закритий).

У додатку С наведено результати тестування системи на мобільних та стаціонарних пристроях. Узагальнення результатів тестування



Рис. 27. Тестування системи у режимі визначення маски

надало можливість зробити наступні висновки щодо ідентифікації облич за допомогою навчених моделей Face API:

- (1) «видимий вік» облич суттєво залежить від настрою їх власників — чим він кращий, тим вік ближчий до біологічного: це пов'язано з тим, що навчання відбувалось на множині переважно усміхнених облич;
- (2) найгірше ідентифікується світле (русяве) волосся — залежно від освітлення воно може бути «будь-якого кольору» (хибно ідентифікуватись як сіре, блондинисте, чорне);
- (3) нерівномірне освітлення та тіні породжують хибні ефекти закриття лоба й одягання головного убору;

- (4) окуляри для читання та сонцезахисні окуляри з додатковими декоративними елементами можуть ідентифікуватись як окуляри для плавання;
- (5) поширеною є помилкова ідентифікація макіяжу очей обличчя у сонцезахисних окулярах.

3 Висновки

У процесі розв'язання проблеми нових освітніх застосувань систем машинного зору в умовах пандемії COVID-19 були отримані наступні результати:

1. Системи машинного (комп'ютерного) зору з початку 1960-х рр. пройшли тривалу еволюцію та широко застосовуються в різних сферах, зокрема, в освіті для реалізації імерсивних освітніх ресурсів. На сучасному етапі їх розвитку ідентифікація динамічних об'єктів у системах машинного зору виконується насамперед засобами машинного навчання. При створенні систем машинного зору освітнього призначення доцільно спиратись на бібліотеки машинного зору на основі глибокого навчання (зокрема, різні реалізації згорткових нейронних мереж).
2. Системи машинного зору для ідентифікації динамічних об'єктів можуть бути застосовані в освіті як за нормальних умов, так і за пандемічних. Зміни в освітній галузі, спричинені пандемією COVID-19, відобразились на класичних освітніх застосуваннях систем машинного зору, модифікувавши існуючі та породивши нові напрями, а саме: дотримання соціального дистанціювання, розпізнавання обличчя в масці, виявлення вторгнень в університетах і школах, запобігання вандалізму, розпізнавання емоцій на обличчях у масках та без масок, моніторинг відвідуваності.
3. Серед великої кількості бібліотек машинного зору загального призначення найбільш доцільними в освітньому процесі є застосування автономних бібліотек (зокрема, OpenCV), спільними рисами яких є вільні ліцензії та можливість використання Python для швидкого прототипування та нейронних мереж для розпізнавання. Комерційні засоби типу MVTec HALCON та Adaptive Vision Library надають суттєво більші функціональні можливості, а застосування viso.ai дозволяє непрограмуємим користувачам конструювати. Розгортання систем машинного

програмного зору забезпечення можливе на різноманітних веб-платформах, найбільш доцільними з яких є хмари провідних постачальників хмарних сервісів — Amazon Web Services, Google Cloud, Microsoft Azure. Останнє є платною опцією, як і опрацювання даних у хмарі розробника, проте компенсується доступом до великої кількості навчених моделей машинного навчання, як-от Cognitive Services.

4. Розроблений на основі Microsoft Cognitive Toolkit і розгорнутий у хмарі Microsoft Azure прототип системи машинного зору інтегрує в собі розпізнавання емоцій учасників освітнього процесу та виявлення порушень маскового режиму, додатково надаючи можливість із високим ступенем надійності визначати стать, інтенсивність посмішки та наявність окулярів та із середнім — вік, колір волосся, наявність головного убору, макіяжу, тощо.

References

1. 2021. Google Ngram Viewer. https://books.google.com/ngrams/graph?content=computer+vision%2C+machine+vision&year_start=1800&year_end=2019&corpus=26&smoothing=3&direct_url=t1%3B%2Ccomputer%20vision%3B%2Cc0%3B.t1%3B%2Cmachine%20vision%3B%2Cc0#t1%3B%2Ccomputer%20vision%3B%2Cc0%3B.t1%3B%2Cmachine%20vision%3B%2Cc0
2. Adaptive Vision. Libraries comparison. https://docs.adaptive-vision.com/avl/technical_issues/LibrariesComparison.html
3. Lakshya Agarwal, Manan Mukim, Harish Sharma, Amit Bhandari, and Atul Mishra. 2021. Face Recognition Based Smart and Robust Attendance Monitoring using Deep CNN. In 2021 8th International Conference on Computing for Sustainable Global Development (INDIACom), 699–704 (2021). doi: 10.1109/INDIACom51348.2021.00124
4. Dana, H.: Ballard and Christopher M. Brown. Computer Vision. Prentice Hall, Englewood Cliffs. <https://archive.org/details/computervision0000ball> (1982). Accessed 13 Nov 2021
5. Bennett, J.: Happy, Sad, Angry Workshop. <https://github.com/jimbobbennett/HappySadAngryWorkshop> (2020). Accessed 13 Nov 2021

6. Gibson, J. J.: The Perception of the Visual World. Boston (1950)
7. Google Cloud. Vision API Product Search pricing. <https://cloud.google.com/vision/product-search/pricing>48 (2021). Accessed 13 Nov 2021
8. Gunnar Rutger Grape. Model Based (Intermediate-Level) Computer Vision. Ph.D. Dissertation. Stanford University. <https://apps.dtic.mil/sti/pdfs/AD0763673.pdf>46 (1973). Accessed 13 Nov 2021
9. Klingler, N.: Top 8 Applications of Computer Vision in the Education Sector. <https://viso.ai/applications/computer-vision-in-education> (2021). Accessed 13 Nov 2021
10. Simon, J. D.: Prince. Computer Vision: Models, Learning, and Inference. Cambridge University Press (2012). Accessed 13 Nov 2021
11. Juliet, R. C.: Pulliam, Cari van Schalkwyk, Nevashan Govender, Anne von Gottberg, Cheryl Cohen, Michelle J. Groome, Jonathan Dushoff, Koleka Mlisana, and Harry Moultrie. Increased risk of SARS-CoV-2 reinfection associated with emergence of the Omicron variant in South Africa. medRxiv (2021). doi: 10.1101/2021.11.11.21266068
12. Ashwin Raj, Aparna Raj, and Imteyaz Ahmad. Smart Attendance Monitoring System with Computer Vision Using IOT. Journal of Mobile Multimedia **17** (1–3), 115–125 (2021). doi: 10.13052/jmm1550-4646.17135
13. Rezaei, M., Azarmi, M.: 2020. DeepSOCIAL: Social Distancing Monitoring and Infection Risk Assessment in COVID-19 Pandemic. Applied Sciences **10**, 21 (2020). doi: 10.3390/app10217514
14. Lawrence, G. R.: Machine perception of three-dimensional solids. Thesis (Ph.D.). Massachusetts Institute of Technology. <https://dspace.mit.edu/bitstream/handle/1721.1/11589/47> (1963). Accessed 13 Nov 2021
15. Shorten, C., Khoshgoftaar, N. M., Furht, B.: 2021. Deep Learning applications for COVID-19. Journal of Big Data **8** (1), 18 (2021). doi: 10.1186/s40537-020-00392-9
16. Sivakumar, S. A., John, J. T., Selvi, G. T., Madhu, B., Shankar, S. U, Arjun, K. P.: 2021. IoT based Intelligent Attendance Monitoring with Face Recognition Scheme. In 2021 5th International Conference on Computing Methodologies and Communication (ICCMC), 349–353 (2021). doi: 10.1109/ICCMC51019.2021.9418264

17. Ivan Edward Sutherland. Sketchpad, a man-machine graphical communication system. Ph.D. Dissertation. Massachusetts Institute of Technology. http://images.designworldonline.com.s3.amazonaws.com/CADhistory/Sketchpad_A_Man-Machine_Graphical_Communication_System_Jan63.pdf (1963). Accessed 13 Nov 2021
18. Tkachuk, V., Yechkalo, Yu., Semerikov, S., Kislova, M, Hladyr, Y.: Using Mobile ICT for Online Learning During COVID-19 Lockdown. In Information and Communication Technologies in Education, Research, and Industrial Applications, Andreas Bollin, Vadim Ermolayev, Heinrich C. Mayr, Mykola Nikitchenko, Aleksander Spivakovsky, Mykola Tkachuk, Vitaliy Yakovyna, and Grygoriy Zholtkevych (eds.). Springer International Publishing, Cham, 46–67 (2021). doi: 10.1007/978-3-030-77592-6_3
19. viso.ai. Abandoned Luggage. <https://viso.ai/application/abandonedluggage-detection> (2021). Accessed 13 Nov 2021
20. viso.ai. Face Recognition. <https://viso.ai/application/face-recognition> (2021). Accessed 13 Nov 2021
21. viso.ai. Facial Emotion Analysis. <https://viso.ai/application/emotionalanalysis> (2021). Accessed 13 Nov 2021
22. viso.ai. Intrusion Detection. <https://viso.ai/application/intrusion-detection> (2021). Accessed 13 Nov 2021
23. viso.ai. Mask Detection: Automatically detect unmasked people in public spaces or indoors. <https://viso.ai/application/mask-detection> (2021). Accessed 13 Nov 2021
24. viso.ai. 2021. Parking Lot Occupancy. <https://viso.ai/application/parking-lotoccupancy-detection> (2021). Accessed 13 Nov 2021
25. viso.ai. Social Distancing Monitoring. <https://viso.ai/application/social-distancing-monitoring> (2021). Accessed 13 Nov 2021

А КЛІЄНТСЬКА ЧАСТИНА ПРОТОТИПУ (TEMPLATES/ HOME.HTML)

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8">
    <title>Машинний зір у часи пандемії</title>
  </head>
  <body>
    <video id="video" autoplay></video>
    <br/>
    <button id="capture">Чи я у масці?</button>
    <button id="capture2">Проаналізувати обличчя</button>
    <p id="message"></p>

    <script type="text/javascript">
      window.addEventListener("DOMContentLoaded", function() {
        var video = document.getElementById('video');

        if (navigator.mediaDevices &&
            navigator.mediaDevices.getUserMedia) {
          const getImage = async () => {
            video.srcObject = await
              navigator.mediaDevices.getUserMedia({ video: true })
            video.play();
          }
          getImage()
        }

        var message = document.getElementById('message');
        document.getElementById('capture').addEventListener(
          'click', function() {
            var canvas = document.createElement('canvas');
            canvas.width = 1024;
            canvas.height = 768;

            var context = canvas.getContext('2d');
            context.drawImage(video, 0, 0, canvas.width, canvas.height);

            var data = {
              'image_base64': canvas.toDataURL("image/png"),
            }
          }
        )
      })
    </script>
  </body>
</html>
```

```

const getResult = async () => {
  var result = await fetch('result', {
    method: 'POST',
    body: JSON.stringify(data),
    headers: { 'Content-Type': 'application/json' }
  })

  var jsonResult = await result.json()
  message.textContent = jsonResult.message
}
getResult()
});

document.getElementById('capture2').addEventListener(
  'click', function() {
    var canvas = document.createElement('canvas');
    canvas.width = 1024;
    canvas.height = 768;

    var context = canvas.getContext('2d');
    context.drawImage(video, 0, 0, canvas.width, canvas.height);

    var data = {
      'image_base64': canvas.toDataURL("image/png"),
    }

    const getResult = async () => {
      var result = await fetch('result2', {
        method: 'POST',
        body: JSON.stringify(data),
        headers: { 'Content-Type': 'application/json' }
      })

      var jsonResult = await result.json()
      message.textContent = jsonResult.message
    }
    getResult()
  });
})
</script>
</body>
</html>

```


В СЕРВЕРНА ЧАСТИНА ПРОТОТИПУ (APP.PY)

```
import random, os, io, base64
from flask import Flask, render_template, request, jsonify
from azure.cognitiveservices.vision.face import FaceClient
from msrest.authentication import CognitiveServicesCredentials

credentials = CognitiveServicesCredentials('секретний ключ')
face_client = FaceClient(
    'https://ruban1.cognitiveservices.azure.com/',
    credentials=credentials)

emotions = ['anger', 'contempt', 'disgust', 'fear', 'happiness',
            'sadness', 'surprise']

def best_emotion(emotion):
    emotions = {}
    emotions['гнів'] = emotion.anger
    emotions['презирство'] = emotion.contempt
    emotions['орада'] = emotion.disgust
    emotions['страх'] = emotion.fear
    emotions['щастя'] = emotion.happiness
    emotions['нейтральний'] = emotion.neutral
    emotions['смуток'] = emotion.sadness
    emotions['здивування'] = emotion.surprise
    return max(zip(emotions.values(), emotions.keys()))[1]

def best_color(color):
    maxcolor = ''
    maxconfidence = 0
    for i in range(len(color)):
        if color[i].confidence > maxconfidence:
            maxconfidence = color[i].confidence
            maxcolor = color[i].color
    if maxcolor == 'unknown':
        return 'невідомий'
    if maxcolor == 'white':
        return 'білий'
    if maxcolor == 'gray':
        return 'сірий'
    if maxcolor == 'blond':
        return 'блондинистий'
    if maxcolor == 'brown':
        return 'коричневий'
```

```

if maxcolor == 'red':
    return 'рудий'
if maxcolor == 'black':
    return 'чорний'
if maxcolor == 'other':
    return 'не білий, не сірий, не блондинистий, не коричневий, '+
        ' не рудий та не чорний'

app = Flask(__name__)

@app.route('/')
def home():
    page_data = {
        #'emotion': random.choice(emotions)
    }
    return render_template('home.html', page_data = page_data)

@app.route('/result', methods=['POST'])
def check_results():
    body = request.get_json()
    image_bytes =
        base64.b64decode(body['image_base64'].split(',')[1])
    image = io.BytesIO(image_bytes)
    faces = face_client.face.detect_with_stream(image,
        recognition_model='recognition_04',
        detection_model='detection_03',
        return_face_attributes=['mask'])
    if len(faces) == 1:
        result = ''
        if faces[0].face_attributes.mask.type == 'noMask':
            result += 'На Вашому обличчі немає маски'
        if faces[0].face_attributes.mask.type == 'faceMask':
            result += 'На Вашому обличчі є маска'
        if faces[0].face_attributes.mask.type ==
            'otherMaskOrOcclusion':
            result += 'Ваше обличчя чимось закрито'
        if faces[0].face_attributes.mask.type == 'uncertain':
            result += 'Неможливо визначити, чи є'
            result += ' на Вашому обличчі маска'
        result += ', Ваші піт та ніс '
        if faces[0].face_attributes.mask.nose_and_mouth_covered:
            result += 'закриті'
        else:

```

```

        result += 'не закрити'
    return jsonify({ 'message': result })
else:
    return jsonify({
        'message': 'Помилка: Обличчя не знайдено'
    })

@app.route('/result2', methods=['POST'])
def check_results2():
    body = request.get_json()
    image_bytes =
        base64.b64decode(body['image_base64'].split(',')[1])
    image = io.BytesIO(image_bytes)
    faces = face_client.face.detect_with_stream(image,
        return_face_attributes=['emotion', 'accessories', 'age',
            'blur', 'exposure', 'facialhair', 'gender', 'glasses',
            'hair', 'headpose', 'makeup', 'noise', 'occlusion',
            'smile'])
    if len(faces) == 1:
        result = ''
        detected_emotion =
            best_emotion(faces[0].face_attributes.emotion)
        result += 'Ваш настрій - ' + detected_emotion + ', '
        result += 'Ви посміхаєтесь на ' + \
            str(faces[0].face_attributes.smile*100) + '%, '
        result += 'Ви виглядаєте на ' + \
            str(faces[0].face_attributes.age) + ' років, '
        result += 'Ви вусати на ' + \
            str(faces[0].face_attributes.facial_hair.moustache*100) + '%, '
        result += 'бородати - на ' + \
            str(faces[0].face_attributes.facial_hair.beard*100) + '%, '
        result += 'та \"бакенбардати\" на ' + \
            str(faces[0].face_attributes.facial_hair.sideburns*100) + '%, '
        result += 'Ваша стать - '
        if faces[0].face_attributes.gender == 'male':
            result+="чоловіча, "
        else:
            result+="жіноча, "
        if faces[0].face_attributes.glasses == 'noGlasses':
            result+="на Вас немає окулярів, "
        if faces[0].face_attributes.glasses == 'readingGlasses':
            result+="на Вас окуляри для читання, "
        if faces[0].face_attributes.glasses == 'sunglasses':

```

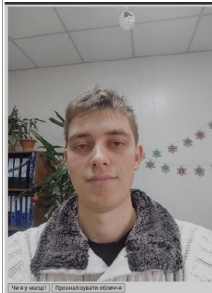
```

        result+="на Вас сонцезхисні окуляри, "
    if faces[0].face_attributes.glasses == 'swimmingGoggles':
        result+="на Вас окуляри для плавання, "
    result+="Ваші очі "
    if not faces[0].face_attributes.makeup.eye_makeup:
        result+="не "
    result+="нафарбовані, "
    result+="Ваші губи "
    if not faces[0].face_attributes.makeup.lip_makeup:
        result+="не "
    result+="нафарбовані, "
    if faces[0].face_attributes.occlusion.mouth_occluded:
        result+="Ваш рот чимось закритий, "
    else:
        result+="Ваш рот нічим не закритий, "
    if faces[0].face_attributes.occlusion.eye_occluded:
        result+="Ваші очі чимось закриті, "
    else:
        result+="Ваші очі нічим не закриті, "
    if faces[0].face_attributes.occlusion.forehead_occluded:
        result+="Ваш лоб чимось закритий, "
    else:
        result+="Ваш лоб нічим не закритий, "
    if len(faces[0].face_attributes.accessories) != 0:
        result+="на Вас "
        for attr in faces[0].face_attributes.accessories:
            if attr.type == 'headWear':
                result+="головний убір, "
            if attr.type == 'glasses':
                result+="окуляри, "
            if attr.type == 'mask':
                result+="маска, "
    if faces[0].face_attributes.hair.invisible:
        result+="Ваше волосся для камери невидиме."
    else:
        result+="Ваше волосся для камери видиме, "
        result+="густина Вашого волосся - " +\
            str((1-faces[0].face_attributes.hair.bald)*100) + "%, "
        detected_color =
            best_color(faces[0].face_attributes.hair.hair_color)
        result+="колір Вашого волосся - " + detected_color + "."
    return jsonify({ 'message': result })
else:

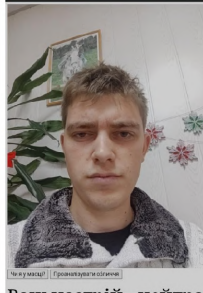
```

```
return jsonify({
    'message': 'Помилка: Обличчя не знайдено'
})
```

С ТЕСТУВАННЯ ПРОТОТИПУ



Ваш настрій - щастя, Ви посміхаєтесь на 86.8%, Ви виглядаєте на 21.0 років, Ви вусаті на 10.0%, бородаті - на 10.0% та "бакенбардаті" на 10.0%, Ваша стать - чоловіча, на Вас немає окулярів, Ваші очі не нафарбовані, Ваші губи не нафарбовані, Ваш рот нічим не закритий, Ваші очі нічим не закриті, Ваш лоб чимось закритий, на Вас головний убір, Ваше волосся для камери невидиме.



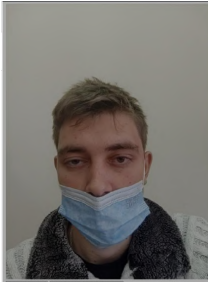
Ваш настрій - нейтральний, Ви посміхаєтесь на 0.0%, Ви виглядаєте на 23.0 років, Ви вусаті на 10.0%, бородаті - на 10.0% та "бакенбардаті" на 10.0%, Ваша стать - чоловіча, на Вас немає окулярів, Ваші очі не нафарбовані, Ваші губи не нафарбовані, Ваш рот нічим не закритий, Ваші очі нічим не закриті, Ваш лоб нічим не закритий, Ваше волосся для камери видиме, густина Вашого волосся - 86.0%, колір Вашого волосся - сірий.



Ваш настрій - нейтральний, Ви посміхаєтесь на 12.1%, Ви виглядаєте на 22.0 років, Ви вусаті на 0.0%, бородаті - на 0.0% та "бакенбардаті" на 0.0%, Ваша стать - жіноча, на Вас немає окулярів, Ваші очі нафарбовані, Ваші губи не нафарбовані, Ваш рот нічим не закритий, Ваші очі нічим не закриті, Ваш лоб нічим не закритий, Ваше волосся для камери видиме, густина Вашого волосся - 79.0%, колір Вашого волосся - блондинистий.



Ваш настрій - нейтральний, Ви посміхаєтесь на 29.299999999999997%, Ви виглядаєте на 23.0 років, Ви вусаті на 10.0%, бородаті - на 10.0% та "бакенбардаті" на 10.0%, Ваша стать - чоловіча, на Вас немає окулярів, Ваші очі не нафарбовані, Ваші губи не нафарбовані, Ваш рот нічим не закритий, Ваші очі нічим не закриті, Ваш лоб чимось закритий, на Вас головний убір, Ваше волосся для камери невидиме.



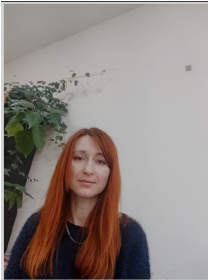
На знімку: Прогнозуйте обличчя

Ваш настрій - нейтральний, Ви посміхаєтесь на 8.7%, Ви виглядаєте на 26.0 років, Ви вусаті на 0.0%, бородаті - на 0.0% та "бакенбардаті" на 0.0%, Ваша стать - чоловіча, на Вас немає окулярів, Ваші очі нафарбовані, Ваші губи не нафарбовані, Ваш рот чимось закритий, Ваші очі нічим не закриті, Ваш лоб нічим не закритий, на Вас маска, Ваше волосся для камери видиме, густина Вашого волосся - 77.0%, колір Вашого волосся - чорний.



На знімку: Прогнозуйте обличчя

Ваш настрій - смуток, Ви посміхаєтесь на 0.0%, Ви виглядаєте на 27.0 років, Ви вусаті на 0.0%, бородаті - на 0.0% та "бакенбардаті" на 0.0%, Ваша стать - чоловіча, на Вас немає окулярів, Ваші очі нафарбовані, Ваші губи не нафарбовані, Ваш рот чимось закритий, Ваші очі нічим не закриті, Ваш лоб нічим не закритий, на Вас маска, Ваше волосся для камери видиме, густина Вашого волосся - 89.0%, колір Вашого волосся - блондинистий.



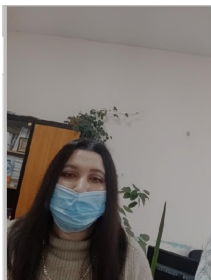
На знімку: Прогнозуйте обличчя

Ваш настрій - щастя, Ви посміхаєтесь на 60.099999999999994%, Ви виглядаєте на 34.0 років, Ви вусаті на 0.0%, бородаті - на 0.0% та "бакенбардаті" на 0.0%, Ваша стать - жіноча, на Вас немає окулярів, Ваші очі нафарбовані, Ваші губи нафарбовані, Ваш рот нічим не закритий, Ваші очі нічим не закриті, Ваш лоб нічим не закритий, Ваше волосся для камери видиме, густина Вашого волосся - 99.0%, колір Вашого волосся - рудий.



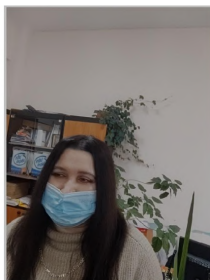
На знімку: Прогнозуйте обличчя

Ваш настрій - нейтральний, Ви посміхаєтесь на 12.6%, Ви виглядаєте на 37.0 років, Ви вусаті на 0.0%, бородаті - на 0.0% та "бакенбардаті" на 0.0%, Ваша стать - жіноча, на Вас немає окулярів, Ваші очі нафарбовані, Ваші губи нафарбовані, Ваш рот нічим не закритий, Ваші очі нічим не закриті, Ваш лоб нічим не закритий, Ваше волосся для камери видиме, густина Вашого волосся - 98.0%, колір Вашого волосся - чорний.



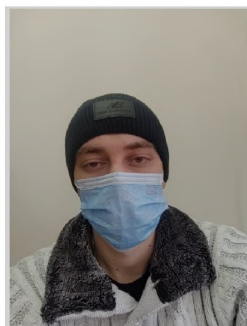
Чи я у масці? | Проаналізувати обличчя

Ваш настрій - нейтральний, Ви посміхаєтесь на 0.6%, Ви виглядаєте на 30.0 років, Ви вусаті на 0.0%, бородаті - на 0.0% та "бакенбардаті" на 0.0%, Ваша стать - жіноча, на Вас немає окулярів, Ваші очі нафарбовані, Ваші губи не нафарбовані, Ваш рот чимось закритий, Ваші очі нічим не закриті, Ваш лоб чимось закритий, на Вас головний убір, маска, Ваше волосся для камери невидиме.



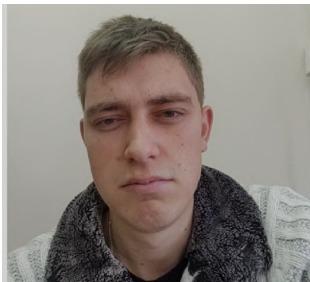
Чи я у масці? | Проаналізувати обличчя

Ваш настрій - нейтральний, Ви посміхаєтесь на 0.3%, Ви виглядаєте на 28.0 років, Ви вусаті на 0.0%, бородаті - на 0.0% та "бакенбардаті" на 0.0%, Ваша стать - жіноча, на Вас немає окулярів, Ваші очі нафарбовані, Ваші губи не нафарбовані, Ваш рот чимось закритий, Ваші очі нічим не закриті, Ваш лоб чимось закритий, на Вас головний убір, маска, Ваше волосся для камери невидиме.



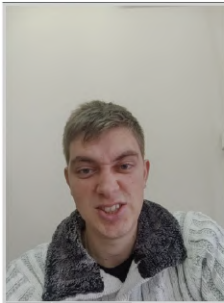
Чи я у масці? | Проаналізувати обличчя

Ваш настрій - нейтральний, Ви посміхаєтесь на 6.6000000000000005%, Ви виглядаєте на 30.0 років, Ви вусаті на 0.0%, бородаті - на 0.0% та "бакенбардаті" на 0.0%, Ваша стать - чоловіча, на Вас немає окулярів, Ваші очі не нафарбовані, Ваші губи не нафарбовані, Ваш рот чимось закритий, Ваші очі нічим не закриті, Ваш лоб чимось закритий, на Вас головний убір, маска, Ваше волосся для камери невидиме.



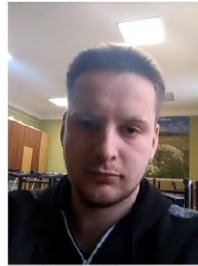
Чи я у масці? | Проаналізувати обличчя

Ваш настрій - смуток, Ви посміхаєтесь на 0.0%, Ви виглядаєте на 30.0 років, Ви вусаті на 0.0%, бородаті - на 10.0% та "бакенбардаті" на 10.0%, Ваша стать - чоловіча, на Вас немає окулярів, Ваші очі не нафарбовані, Ваші губи не нафарбовані, Ваш рот нічим не закритий, Ваші очі нічим не закриті, Ваш лоб чимось закритий, на Вас головний убір, Ваше волосся для камери невидиме, густина Вашого волосся - 0.0%.



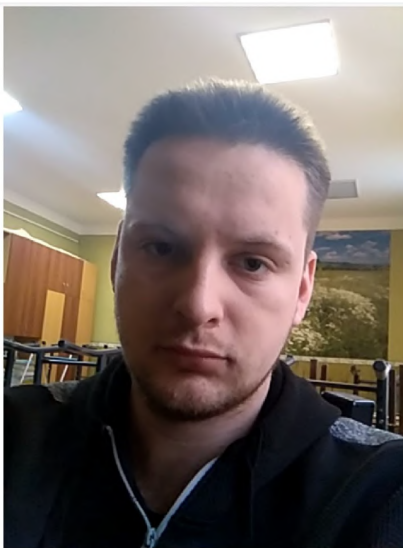
Чи я у масці? | Проаналізувати обличчя

Ваш настрій - гнів, Ви посміхаєтесь на 38.1%, Ви виглядаєте на 27.0 років, Ви вусаті на 10.0%, бородаті - на 10.0% та "бакенбардаті" на 10.0%, Ваша стать - чоловіча, на Вас немає окулярів, Ваші очі не нафарбовані, Ваші губи не нафарбовані, Ваш рот нічим не закритий, Ваші очі нічим не закриті, Ваш лоб нічим не закритий, Ваше волосся для камери видиме, густина Вашого волосся - 40.0%, колір Вашого волосся - сірий.



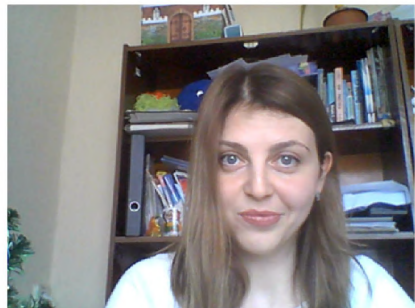
Чи я у масці? | Проаналізувати обличчя

Ваш настрій - нейтральний, Ви посміхаєтесь на 0.3%, Ви виглядаєте на 24.0 років, Ви вусаті на 10.0%, бородаті - на 10.0% та "бакенбардаті" на 10.0%, Ваша стать - чоловіча, на Вас немає окулярів, Ваші очі не нафарбовані, Ваші губи не нафарбовані, Ваш рот нічим не закритий, Ваші очі нічим не закриті, Ваш лоб нічим не закритий, Ваше волосся для камери видиме, густина Вашого волосся - 49.0%, колір Вашого волосся - сірий.



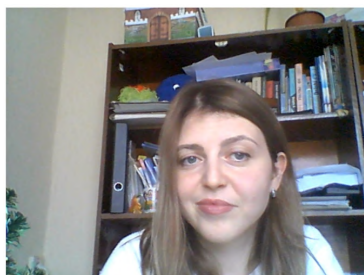
Чи я у масці? | Проаналізувати обличчя

На Вашому обличчі немає маски, Ваші рід та ніс не закриті



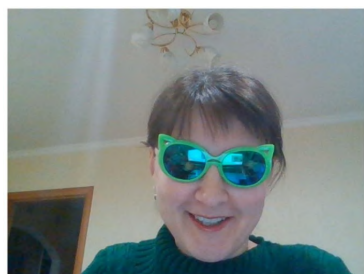
Чи я у масці? | Проаналізувати обличчя

На Вашому обличчі немає маски, Ваші рід та ніс не закриті



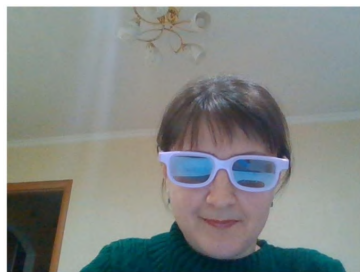
Чи я у масці? | Проаналізувати обличчя

Ваш настрій - приємно. Ви посміхаєтесь на 5.4%. Ви виглядаєте на 32.0 років. Ви вусати на 0.0%, бородаті - на 0.0% та "бакибардаті" на 0.0%. Ваша стать - жіноча, на Вас немає окулярів. Ваші очі не фарбовані, Ваші губи не фарбовані. Ваш рот нічим не закритий. Ваші очі нічим не закриті. Ваш лоб нічим не закритий. Ваше волосся для камери виглядає, густина Вашого волосся - 91.0%, колір Вашого волосся - коричневий.



Чи я у масці? | Проаналізувати обличчя

Ваш настрій - шаста. Ви посміхаєтесь на 100.0%. Ви виглядаєте на 41.0 років. Ви вусати на 0.0%, бородаті - на 0.0% та "бакибардаті" на 0.0%. Ваша стать - жіноча, на Вас окуляри для плавання. Ваші очі не фарбовані. Ваші губи не фарбовані. Ваш рот нічим не закритий. Ваші очі нічим не закриті. Ваш лоб нічим закритий, на Вас окуляри, головний убір. Ваше волосся для камери невидимі.



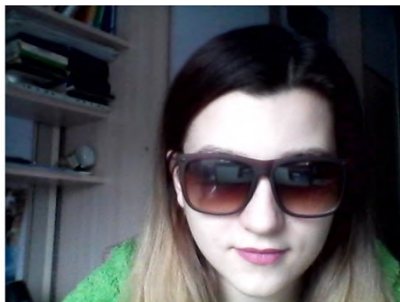
Чи я у масці? | Проаналізувати обличчя

Ваш настрій - шаста. Ви посміхаєтесь на 100.0%. Ви виглядаєте на 35.0 років. Ви вусати на 0.0%, бородаті - на 0.0% та "бакибардаті" на 0.0%. Ваша стать - жіноча, на Вас сонцезахисні окуляри. Ваші очі не фарбовані. Ваші губи не фарбовані. Ваш рот нічим не закритий. Ваші очі нічим не закриті. Ваш лоб нічим не закритий, на Вас окуляри. Ваше волосся для камери виглядає, густина Вашого волосся - 84.0%, колір Вашого волосся - чорний.



Чи я у масці? | Проаналізувати обличчя

Ваш настрій - нейтральний. Ви посміхаетесь на 5.3%. Ви виглядаєте на 37.0 років. Ви вусати на 0.0%, бородати - на 0.0% та "бакенбардати" на 0.0%. Ваша стать - жіноча, на Вас сонцезахисні окуляри, Ваші очі нафарбовані, Ваші губи нафарбовані, Ваш рот нічим не закритий, Ваші очі нічим не закриті, Ваш лоб нічим не закритий, на Вас окуляри, Ваше волосся для камери видиме, густина Вашого волосся - 84.0%, колір Вашого волосся - коричневий.



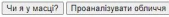
Чи я у масці? | Проаналізувати обличчя

Ваш настрій - нейтральний. Ви посміхаетесь на 0.0%. Ви виглядаєте на 25.0 років. Ви вусати на 0.0%, бородати - на 0.0% та "бакенбардати" на 0.0%. Ваша стать - жіноча, на Вас сонцезахисні окуляри, Ваші очі нафарбовані, Ваші губи нафарбовані, Ваш рот нічим не закритий, Ваші очі нічим не закриті, Ваш лоб нічим не закритий, на Вас окуляри, Ваше волосся для камери видиме, густина Вашого волосся - 96.0%, колір Вашого волосся - чорний.



Чи я у масці? | Проаналізувати обличчя

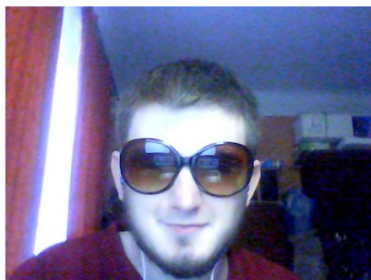
Ваш настрій - щастя. Ви посміхаетесь на 99.5%. Ви виглядаєте на 22.0 років. Ви вусати на 0.0%, бородати - на 0.0% та "бакенбардати" на 0.0%. Ваша стать - жіноча, на Вас сонцезахисні окуляри, Ваші очі нафарбовані, Ваші губи нафарбовані, Ваш рот нічим не закритий, Ваші очі нічим не закриті, Ваш лоб нічим не закритий, на Вас окуляри, Ваше волосся для камери видиме, густина Вашого волосся - 96.0%, колір Вашого волосся - чорний.



Чи я у масці? Проаналізувати обличчя

Чи я у масці? Проаналізувати обличчя

~ 58 ~



Чи я у масці? | Проаналізуйте обличчя

Ваш настрій - нейтральний. Ви посмікаєтесь на 32.0%. Ви виглядаєте на 28.0 років. Ви вусати на 10.0%, бородаті - на 10.0% та "бавенбордаті" на 10.0%. Ваша стать - чоловіча, на Вас спостережені окуляри. Ваші очі - карі. Ваші губи - фарбовані. Ваш рот нічого не закритий. Ваші очі нічого не закриті. Ваш лоб нічого не закритий, на Вас окуляри. Ваше волосся для камери відбите, густина Вашого волосся - 74.0%, колір Вашого волосся - коричневий.

Розробка системи комп'ютерного зору на основі машинного навчання для освітніх цілей

Сергій О. Семеріков^{1,2,3,4[0000—0003—0789—0272]},

Тетяна А. Вакалюк^{1,3,5[0000—0001—6825—4697]},

Ірина С. Мінтій^{1,3[0000—0003—3586—4311]},

Віта А. Гаманюк^{1[0000—0002—3522—7673]},

Володимир М. Соловйов^{1[0000—0002—4945—202X]},

Ольга В. Бондаренко^{1[0000—0003—2356—2674]},

Павло П. Нечипуренко^{1[0000—0001—5397—6523]},

Світлана В. Шокалюк^{1[0000—0003—3774—1729]},

Наталія В. Моїсеєнко^{1[0000—0002—3559—6081]},

Віталій Р. Рубан¹

¹ Криворізький державний педагогічний університет,
просп. Гагаріна, 54, Кривий Ріг, 50086, Україна

² Криворізький національний університет, вул. Віталія Матусевича, 11,
Кривий Ріг, 50027, Україна

³ Інститут цифровізації освіти Національної академії педагогічних наук
України, вул. М. Берлінського, 9, Київ, 04060, Україна

⁴ «Університет менеджменту освіти» НАПН України, вул. Січових стрільців,
52-А, Київ, 04053, Україна

⁵ Державний університет «Житомирська політехніка», вул. Чуднівська, 103,
Житомир, 10005, Україна

semerikov@gmail.com, tetianavakaliuk@gmail.com,
irina.mintiy@kdpu.edu.ua, vitana65@gmail.com,
vnsoloviev2016@gmail.com, bondarenko.olga@kdpu.edu.ua,
acinonyxleo@gmail.com, shokalyuk@kdpu.edu.ua,
n.v.moiseenko@gmail.com, vitas.uno@gmail.com

Анотація. У роботі наведено огляд витоків і сучасного стану систем комп'ютерного зору, приклади задач комп'ютерного зору. Описано використання систем комп'ютерного зору освіти як у звичайних, так і в пандемічних умовах. Пандемія COVID-19 викликала зміни в освіті, які видозмінили існуючі освітні програми систем комп'ютерного зору та породили нові, зокрема: соціальне дистанціювання, розпізнавання маски на обличчі, виявлення проникнення в університети та школи, запобігання вандалізму й ідентифікація підозрілих предметів, моніторинг відвідуваності, емоцій на обличчях у масках і без них. Системи комп'ютерного зору також можна використовувати в освіті для впровадження імерсивних освітніх ресурсів. На основі аналізу автономних бібліотек для розпізнавання динамічних об'єктів зроблено висновок, що у процесі створення систем машинного зору в освітніх цілях доцільно використовувати бібліотеки комп'ютерного зору, що ґрунтуються на глибокому навчанні (зокрема, реалізації згорткових нейронних мереж). Описано прототип системи комп'ютерного зору, розроблений на основі Microsoft Cognitive Toolkit і розгорнутий у хмарі Microsoft Azure. Система дозволяє з високим ступенем надійності виконувати головні функції: ідентифікацію емоцій і наявності маски на обличчі, а також дає можливість визначити стать, вік, колір волосся, інтенсивність посмішки, наявність макіяжу, окулярів тощо.

Ключові слова: комп'ютерний зір, COVID-19, виявлення масок, освіта.