

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**  
**КРИВОРІЗЬКИЙ ДЕРЖАВНИЙ ПЕДАГОГІЧНИЙ УНІВЕРСИТЕТ**  
**Фізико-математичний факультет**  
**Кафедра інформатики та прикладної математики**

«Допущено до захисту»

Завідувач кафедри

\_\_\_\_\_ Соловйов В. М.

Реєстраційний № \_\_\_\_\_

« \_\_\_ » \_\_\_\_\_ 2021 р.

« \_\_\_ » \_\_\_\_\_ 2021 р.

**ІДЕНТИФІКАЦІЯ ДИНАМІЧНИХ ОБ'ЄКТІВ**  
**У СИСТЕМАХ МАШИННОГО ЗОРУ**

Кваліфікаційна робота студента групи Ім-16  
ступінь вищої освіти «магістр»  
спеціальності 014.09 Середня освіта (інформатика)  
**Рубана Віталія Романовича**

Керівник доктор педагогічних наук, професор,  
старший дослідник  
**Семеріков Сергій Олексійович**

Оцінка:

Національна шкала \_\_\_\_\_

Шкала ECTS \_\_\_\_\_ Кількість балів \_\_\_\_\_

Голова ЕК \_\_\_\_\_

(підпис)

(прізвище, ініціали)

Члени ЕК \_\_\_\_\_

(підпис)

(прізвище, ініціали)

(підпис)

(прізвище, ініціали)

(підпис)

(прізвище, ініціали)

(підпис)

(прізвище, ініціали)

## ЗАПЕВНЕННЯ

Я, Рубан Віталій Романович, розумію і підтримую політику Криворізького державного педагогічного університету з академічної доброчесності. Запевняю, що ця кваліфікаційна робота виконана самостійно, не містить академічного плагіату, фабрикації, фальсифікації. Я не надавав і не одержував недозволену допомогу під час підготовки цієї роботи. Використання ідей, результатів і текстів інших авторів мають покликання на відповідне джерело.

Із чинним Положенням про запобігання та виявлення академічного плагіату в роботах здобувачів вищої освіти Криворізького державного педагогічного університету ознайомлений. Чітко усвідомлюю, що в разі виявлення у кваліфікаційній роботі порушення академічної доброчесності робота не допускається до захисту або оцінюється незадовільно.



(підпис)

## ЗМІСТ

ВСТУП.....	4
РОЗДІЛ 1 ОСВІТНІ ЗАСТОСУВАННЯ СИСТЕМ МАШИННОГО ЗОРУ .....	6
1.1 Витоки та сучасний стан систем машинного зору .....	6
1.2 Зміни освітніх застосувань систем машинного зору, спричинені поширенням COVID-19.....	13
Висновки до розділу 1.....	22
РОЗДІЛ 2 ПРОЄКТУВАННЯ ПРОТОТИПУ СИСТЕМИ МАШИННОГО ЗОРУ ОСВІТНЬОГО ПРИЗНАЧЕННЯ .....	23
2.1 Бібліотеки машинного зору для ідентифікації динамічних об’єктів...	23
2.2 Розробка прототипу системи машинного зору освітнього призначення .....	25
2.2.1 Розпізнавання статичних обличь у Microsoft Azure .....	25
2.2.2 Розпізнавання динамічних обличь у Microsoft Azure .....	32
2.2.3 Тестування розробленого прототипу.....	41
Висновки до розділу 2.....	42
ВИСНОВКИ .....	43
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ .....	45
ДОДАТКИ .....	48
А Клієнтська частина прототипу (/templates/home.html) .....	48
Б Серверна частина прототипу (app.py).....	50
В Тестування прототипу .....	55

## ВСТУП

**Актуальність теми.** Машинний зір – галузь інформаційних технологій, що займається ідентифікацією, відслідковуванням та класифікацією об'єктів. Класичні застосування: наука, технологія, розваги – аналіз зображень, відеоспостереження, обчислювальна фотографія та ін. Основні типи об'єктів, що ідентифікуються – статичні, динамічні, а провідні методи ідентифікації – методи машинного навчання. Теорія ідентифікації статичних об'єктів є достатньо розвиненою, тому їх застосування до послідовності зображень об'єкту, що відрізняється позицією, кутом огляду, освітленням тощо, надає можливість ідентифікації динамічних об'єктів.

Класичні застосування машинного зору – системи доповненої реальності, широко поширені в освіті. *Проблема дослідження* – нові освітні застосування систем машинного зору в умовах пандемії COVID-19: дотримання соціального дистанціювання, виявлення порушень маскового режиму, виявлення вторгнень, підозрілих об'єктів та запобігання вандалізму, аналіз емоцій здобувачів освіти, моніторинг відвідуваності.

**Мета дослідження** – розробити прототип системи машинного зору освітнього призначення.

Для досягнення мети дослідження були поставлені такі **завдання**:

1. Дослідити еволюцію систем машинного зору.
2. Виявити освітні застосування систем машинного зору.
3. Проаналізувати програмні засоби та бібліотеки машинного зору для ідентифікації динамічних об'єктів.
4. Розробити прототип системи машинного зору освітнього призначення.

**Об'єкт дослідження** – системи машинного зору.

**Предмет дослідження** – ідентифікація динамічних об'єктів у системах машинного зору.

**Методи дослідження:** аналіз джерел із проблеми дослідження, методи інженерії програмного забезпечення для прототипування системи машинного



зору освітнього призначення.

**Структура роботи.** Робота складається із вступу, двох розділів, висновків, списку використаних джерел (25 найменувань), 3 додатків.

## РОЗДІЛ 1

### ОСВІТНІ ЗАСТОСУВАННЯ СИСТЕМ МАШИННОГО ЗОРУ

#### 1.1 Витоки та сучасний стан систем машинного зору

Перші загадки про машинний зір (machine vision) або комп'ютерний зір (computer vision) з'являються наприкінці 1950-х рр., й на сьогодні в англomовних джерелах друге поняття є суттєво частіше згадуваним (рис. 1.1).

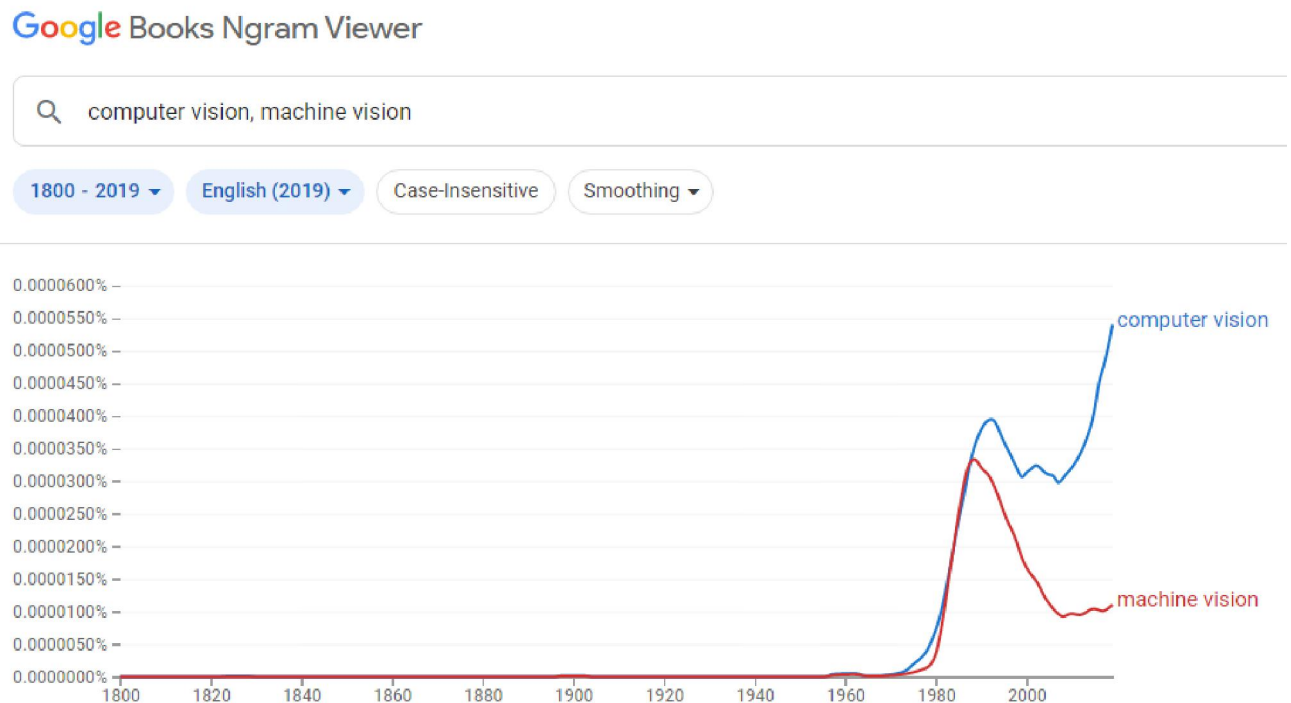


Рис. 1.1. Згадування про машинний зір у англomовних джерелах (за [8])

Г. Р. Грейп (Gunnar Rutger Grape) в дисертації [9] згадує роботу «розробника Інтернет» Л. Г. Робертса (Lawrence Gilman Roberts) 1963 року “Machine perception of three-dimensional solids” [19] як піонерську. Виконана при сприянні К. Шеннона у М.І.Т. Lincoln Laboratory, ця робота, так само, як робота А. Е. Сазерленда (Ivan Edward Sutherland) [23], заклала основи комп'ютерної графіки.

Л. Г. Робертс вказує, що в основу машинного розпізнавання простих образів вже у 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) перцептрон [23, с. 11]. Проте існуючі розробки для розпізнавання літер були непридатними для розпізнавання зображень, побудованих за допомогою відрізків прямих: проблема стосувалась як плоских (двовимірних), так й просторових (тривимірних) зображень – саме останнім й була присвячена дисертація [23].

Л. Г. Робертс спирався на психологічні дослідження, присвячені сприйняттю людиною глибини та розпізнавання фігур, насамперед – Дж. Дж. Гібсона (James Jerome Gibson) [7]. Результатом роботи Л. Г. Робертса стала система для перетворення фотографічних зображень на тривимірні моделі (рис. 1.2).

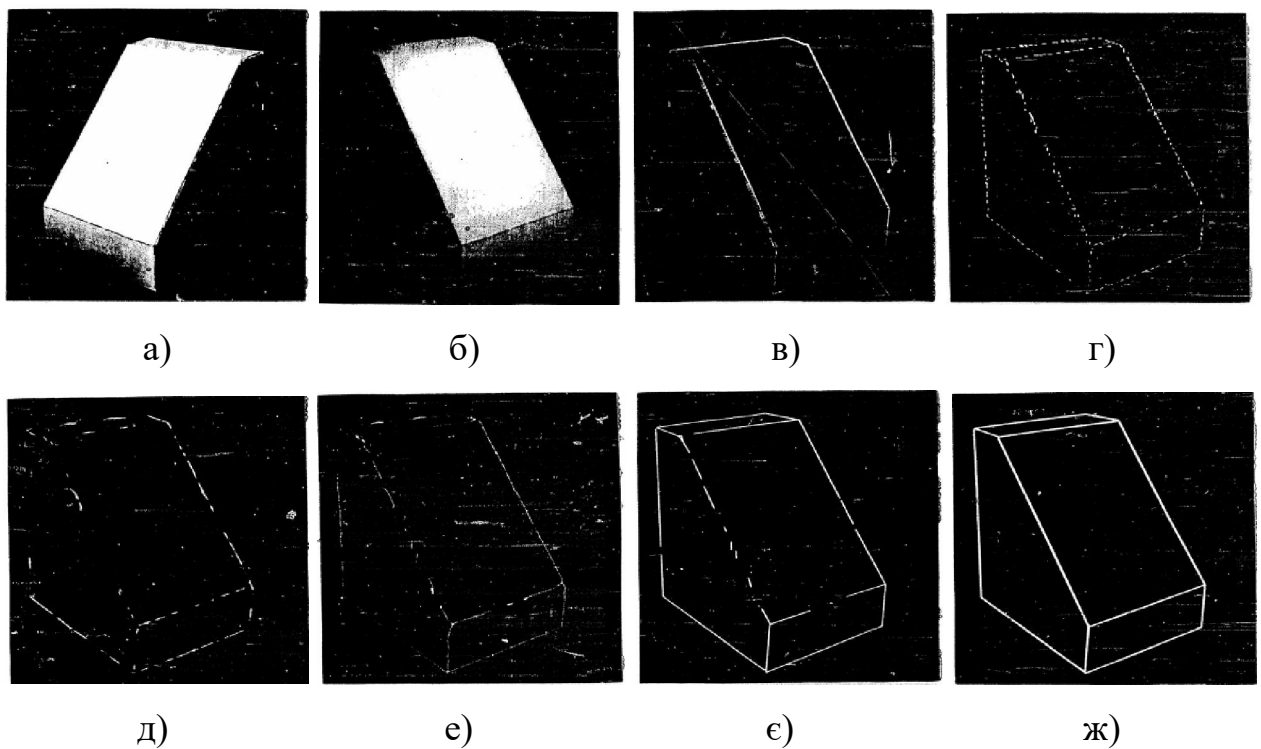
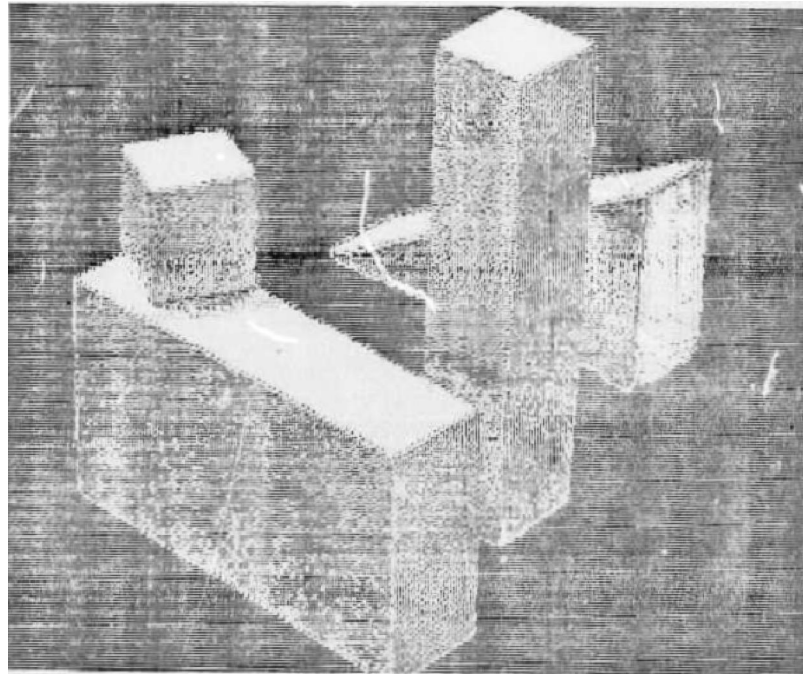


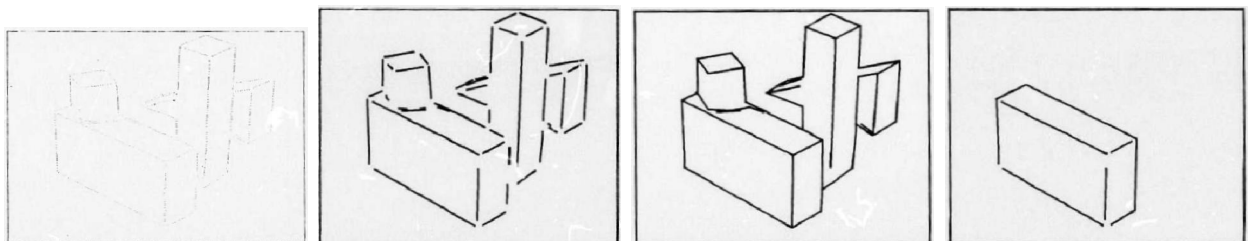
Рис. 1.2. Перетворення зображення за Л. Г. Робертсом: а) оригінальне зображення; б) оцифроване зображення; в-є) перетворення; ж) результуюча модель [23, с. 35-36]

За 10 років після роботи Л. Г. Робертса стан справ у галузі машинного зору суттєво змінився. Г. Р. Грейп у [9] представлено 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 проекції. Оцифроване зображення сканувалось для виявлення градієнтів високої інтенсивності (граней), які, ймовірно, з'являться на стиках регіонів і об'єктів. Г. Р. Грейп продемонстровані перспективно узгоджені двовимірні 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. 1.3).



а)

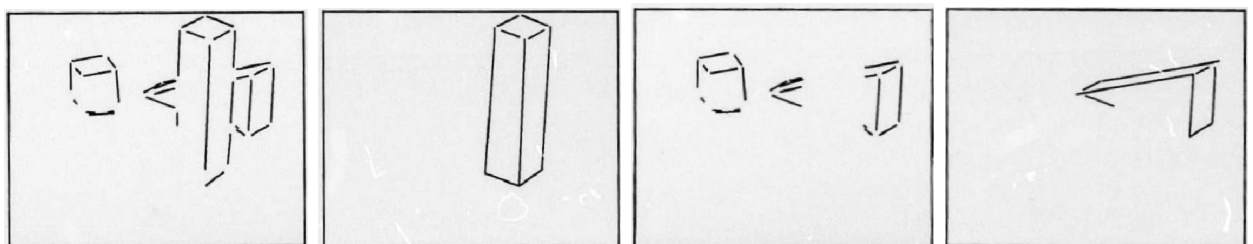


б)

в)

г)

д)



е)

ж)

з)

и)

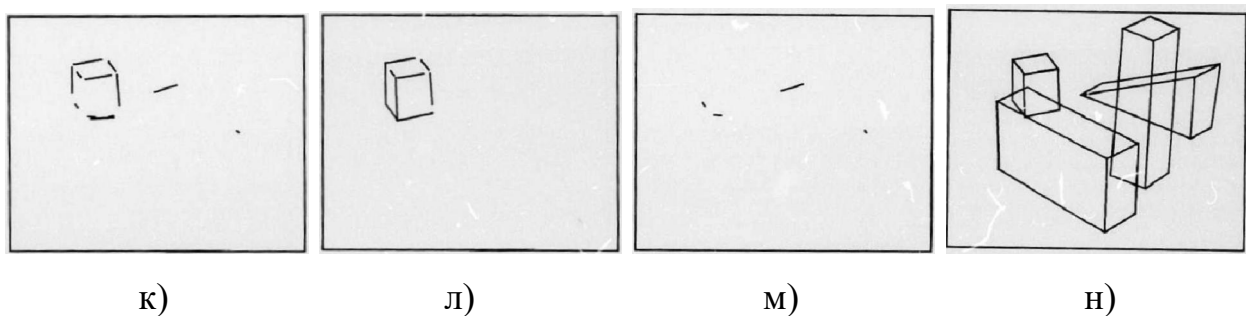


Рис. 1.3. Перетворення зображення за Г. Р. Грейпом: а) зображення з ТВ-камери; б, в) виділення граней; г, д) виділення першого об'єкту; е, ж) виділення другого об'єкту; з, и) виділення третього об'єкту; к, л) виділення четвертого об'єкту; м, н) фінальна інтерпретація [9, с. 174-181]

Визначаючи майбутні напрямки розвитку систем машинного зору, Г. Р. Грейп запропонував концептуальну модель опрацювання потоків даних у тривимірній системі (рис. 1.4).

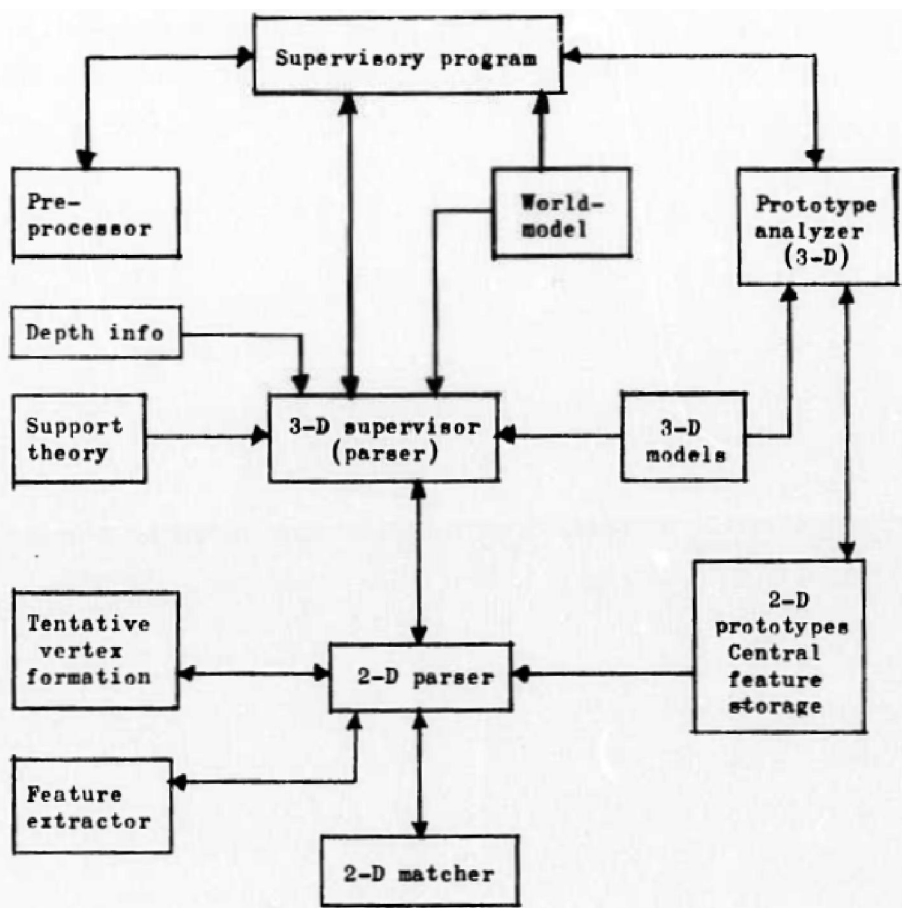


Рис. 1.4. Опрацювання потоків даних у 3D-системі [9, с. 242]

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 [3].

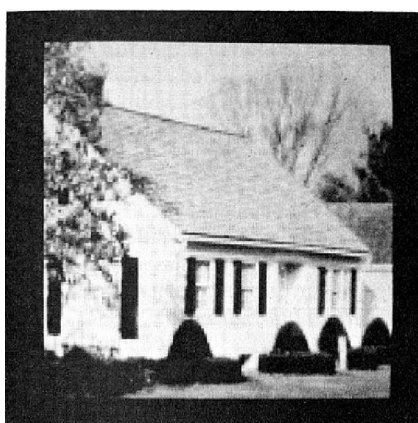
Д. Г. Баллард та К. М. Браун запропонували 4 категорії подання зображень у системах машинного зору (рис. 1.5). У табл. 1.1 наведено основні галузі застосування систем машинного зору так, як їх бачили автори у 1982 році.

Таблиця 1.1

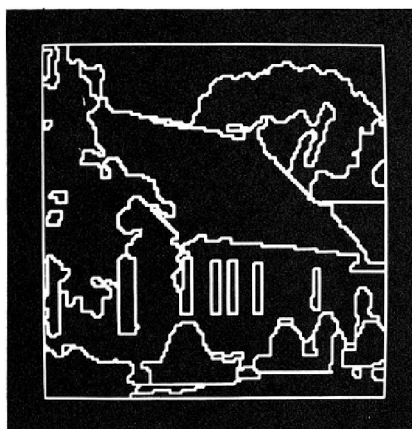
### Приклади задач машинного зору [3, с. 11]

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	Геометричні моделі форм

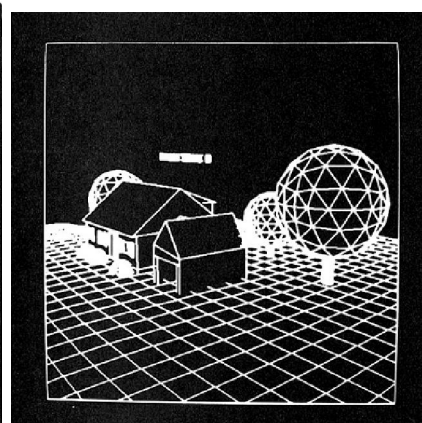
Галузь	Об'єкти	Модальність	Задачі	Джерела інформації
			зображень	
Медицина (макро)	Частини тіла	X-промені Ультразвук Ізотопи Тепло	Діагностика відхилень Планування операцій та лікування	Анатомічні моделі Моделі формування зображень
Медицина (мікро)	Клітини Ланцюги протеїнів Хромосоми	Електронна мікроскопія 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
Фізика	Треки частинок	Світло	Знаходження нових частинок Ідентифікація треків	Атомна фізика



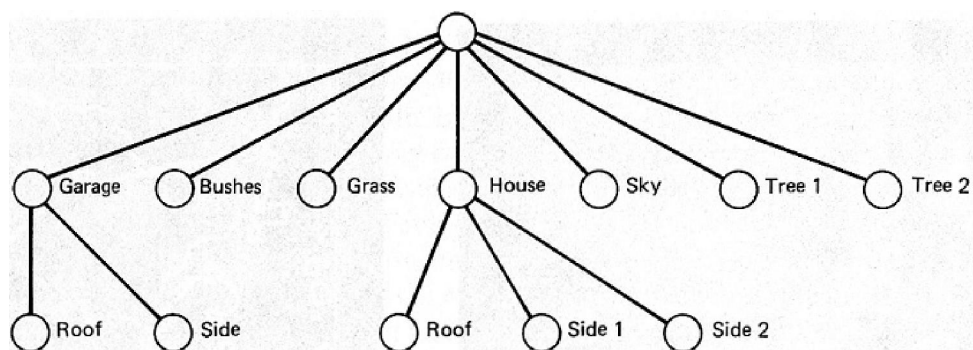
a)

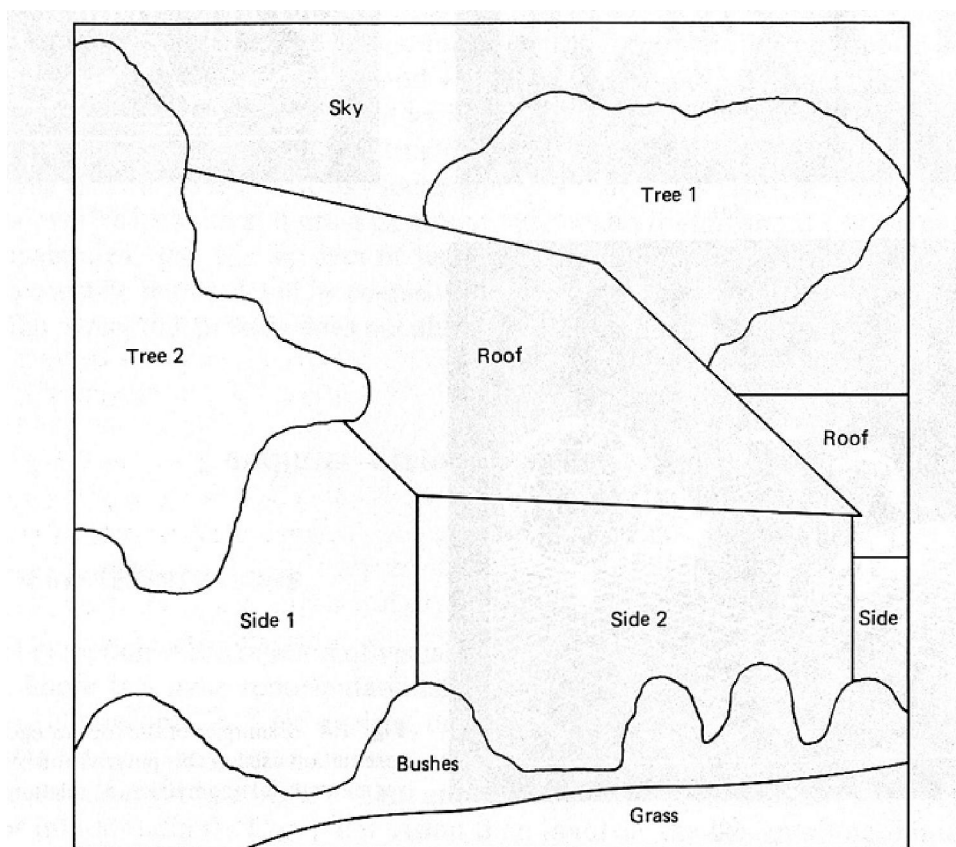


б)



в)





г)

Рис. 1.5. Категорії подання зображень у системах комп'ютерного зору: узагальнене (а), сегментоване (б), геометричне (в) та реляційне (г) [3, с. 7-8]

«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. Принса (Simon J. D. Prince) «Computer Vision: Models, Learning, and Inference» [15], яка дає основи машинного навчання, необхідні для підтримки поточних застосувань і досліджень у галузі машинного зору.

Таким чином, за 60 років розвитку системи машинного зору проникли у значну кількість галузей, і не в останню чергу – в освітню, де вони застосовуються для реалізації імерсивних освітніх ресурсів. Проте зміни в освітній галузі, спричинені пандемією COVID-19 [24], не могли не відобразитись на класичних освітніх застосування систем машинного зору, модифікувавши існуючі 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, в 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, таких як школи та університети. Системи глибокого навчання можна використовувати для моніторингу груп в аудиторії, натовпів на поверхах, щоб аналізувати соціальне дистанціювання, виявляти вузькі місця та запускати сповіщення у разі постійних порушень (рис. 1.6).



Рис. 1.6. Аналіз соціального дистанціювання за допомогою <https://viso.ai/application/social-distancing-monitoring/>

Засоби, необхідні для соціального моніторингу – недорогі камери спостереження, вже наявні у окремих школах.

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

У [22] визначені наступні можливості моніторингу соціального дистанціювання за допомогою систем машинного зору:

а) автоматичне виявлення та відстеження кількох динамічних об'єктів (осіб) у режимі реального часу;

б) аналіз траєкторії руху людей та рівня порушення соціального дистанціювання;

в) визначення зон підвищеного ризику з найбільшою небезпекою можливого поширення коронавірусу.

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

*Розпізнавання обличчя в масці* – спосіб контролювати дотримання та дотримання масок у людних місцях, таких як університети чи школи. 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 у реалізації, оскільки можна використовувати відеозапис будь-якої камери (рис. 1.7).

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

а) автоматичний моніторинг дотримання маскового режиму є більш послідовним і точним, ніж перевірка людиною;

б) виявлення людей без масок підвищує безпеку, знижуючи ризик передачі COVID-19 та інших інфекційних захворювань;

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

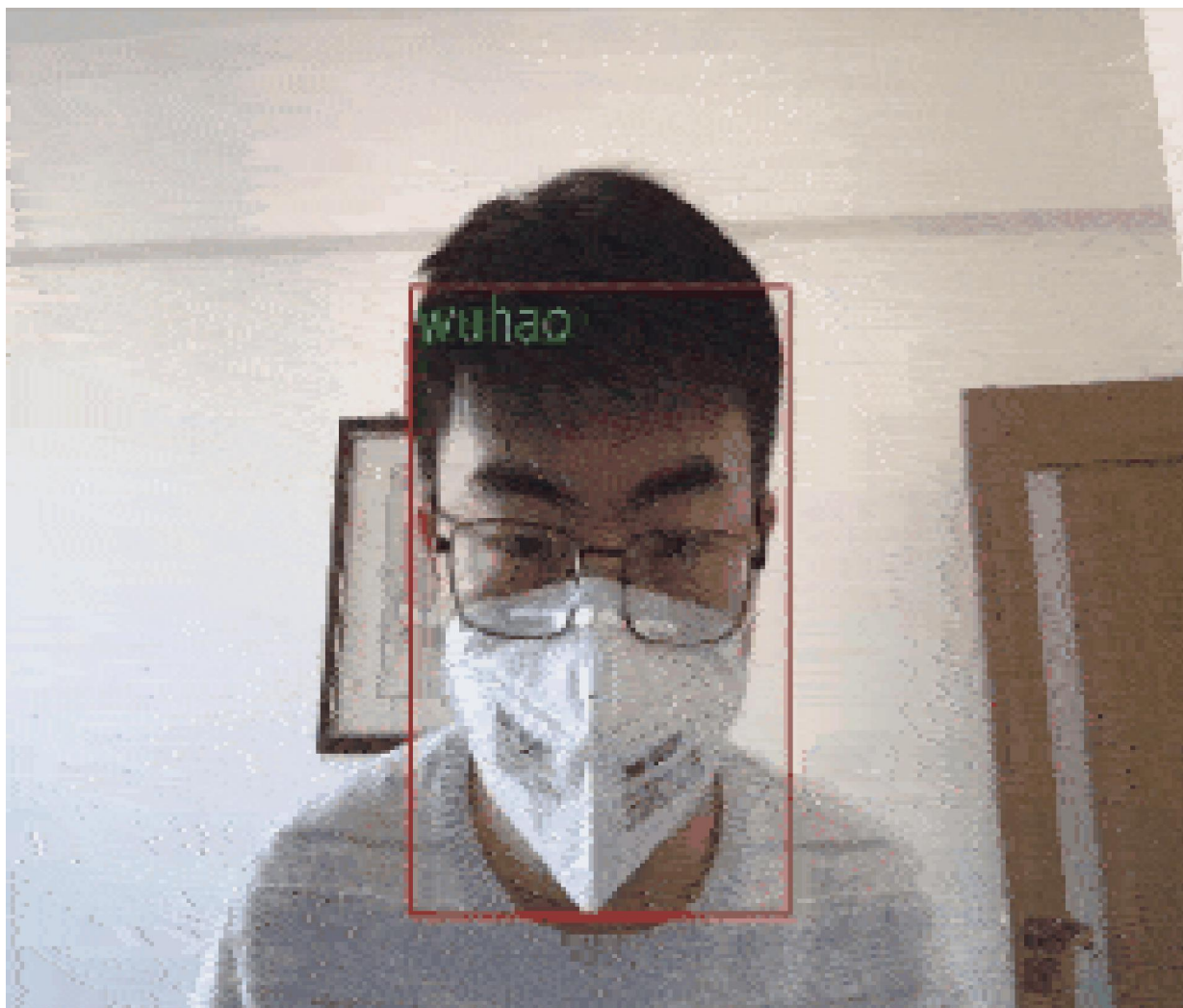
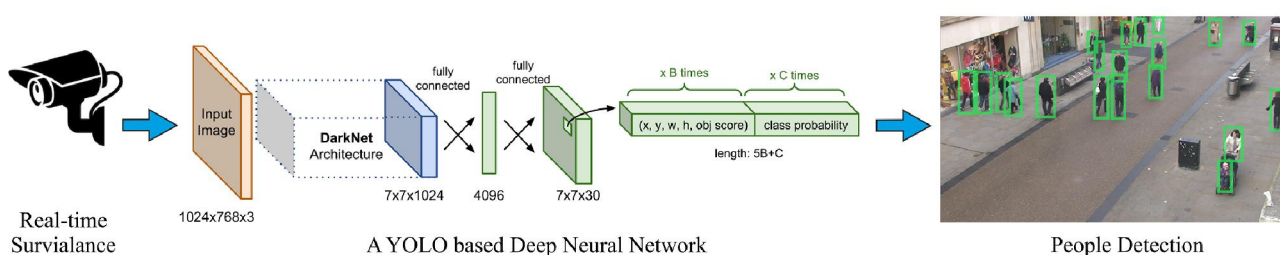


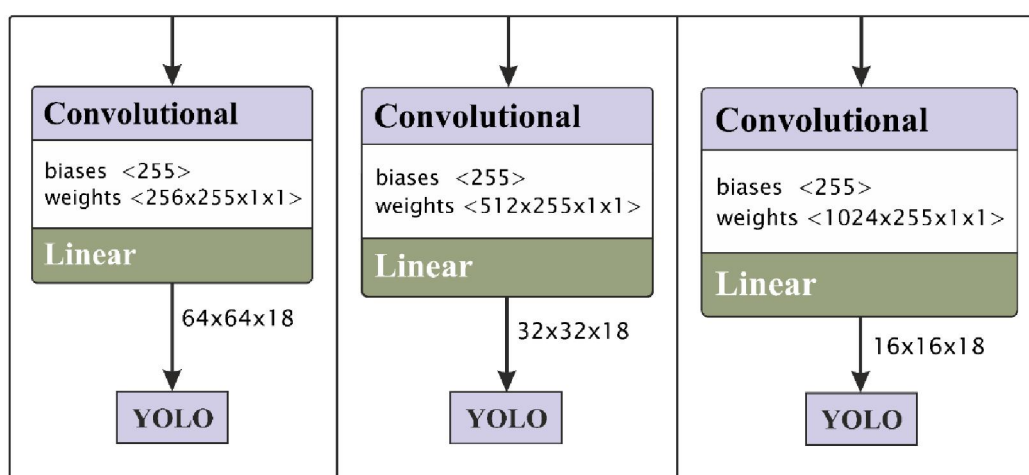
Рис. 1.7. Розпізнавання обличчя в масці за допомогою <https://viso.ai/application/mask-detection/>

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. 1.8). Запропонована авторами модель у поєднанні з адаптованою технікою відображення зворотної перспективи та алгоритмом відстеження SORT веде до надійного виявлення людей та моніторингу соціального дистанціювання. Модель була навчена на основі двох наборів даних – Microsoft Common Objects in Context (MS COCO) і Google Open Image. Оцінку моделі було проведено в складних умовах, включаючи часткову видимість та при змінах освітлення із

середньою точністю 99.8% і 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 та 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, таких як автономні транспортні засоби, розпізнавання людських дій, виявлення аномалій.



a)



б)

Рис. 1.8. 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) у огляді [20] наводять граф застосувань глибокого навчання, серед яких чільне місце займають застосування машинного зору (рис. 1.9).

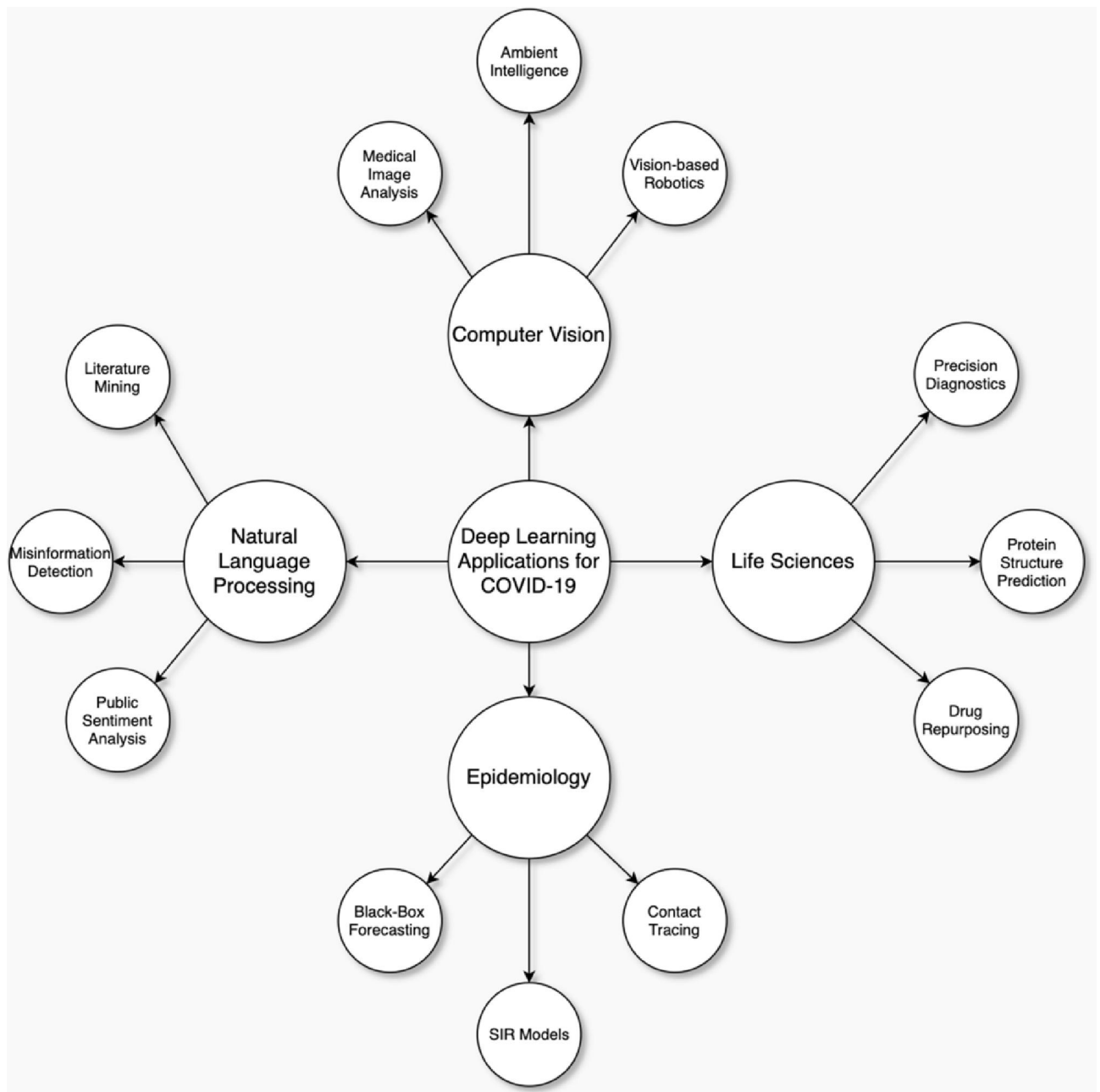


Рис. 1.9. Застосування глибокого навчання для подолання наслідків COVID-19  
[20, с. 18]

*Системи управління паркуванням для виявлення заповненості автостоянок при школах або університетах є ще одним класичним застосуванням машинного зору, корисним й у пандемічний період. Камери, які також використовуються для 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 [14] визначені наступні переваги системи управління паркуванням для виявлення заповненості автостоянок при школах або університетах:

а) виявлення зайнятості автостоянки на основі зору з глибоким навчанням економить витрати в порівнянні з дорогими сенсорними методами;

б) оптимізація паркування допомагає зменшити затори в години пік і транспортні потоки в містах, викликані людьми, які шукають місця для паркування (близько 30%);

в) для рішень на основі камер не потрібна додаткова інфраструктура, за умови, що об'єкт оснащений камерами відеоспостереження, які охоплюють місця для паркування;

г) рішення на основі машинного зору забезпечують точне розташування вільного паркувального місця.

*Виявлення вторгнень в університетах і школах* – застосування глибокого навчання із звичайними камерами спостереження для здійснення моніторингу периметра та автоматичного виявлення злоумисників (рис. 1.10).



Рис. 1.10. Виявлення вторгнень за допомогою <https://viso.ai/application/intrusion-detection/>

Переваги застосування систем виявлення вторгнень [10]:

- а) наявні алгоритми виявлення об'єктів у режимі реального часу для виявлення людей та об'єктів на відео з кількох камер;
- б) підозрілі області можна візуально визначити, намалювавши та назвавши певні області в кадрах камери;
- в) 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);
- г) 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. 1.11).

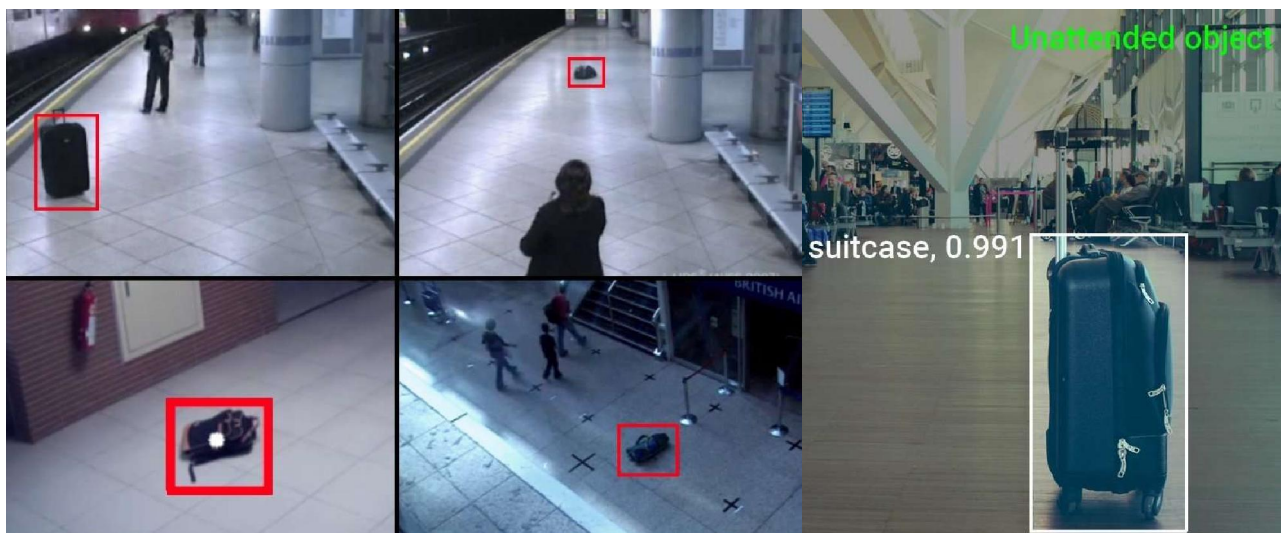


Рис. 1.11. Виявлення підозрілих об'єктів без нагляду за допомогою <https://viso.ai/application/abandoned-luggage-detection/>

Переваги застосування систем запобігання вандалізму та виявлення підозрілих об'єктів без нагляду [1]:

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

поверхні, ліфти тощо);

в) масштабований та об'єктивний моніторинг громадських місць, громадського транспорту, шкіл чи лікарень;

г) раннє виявлення в реальному часі, щоб допомогти людям виявити підозрілі предмети та організувати евакуацію.

Відеопотік можна використовувати для розпізнавання емоційних станів однієї людини або кількох людей, застосовуючи алгоритми глибокого навчання, у тому числі для *розпізнавання емоцій на обличчях у масках та без масок* (рис. 1.12).

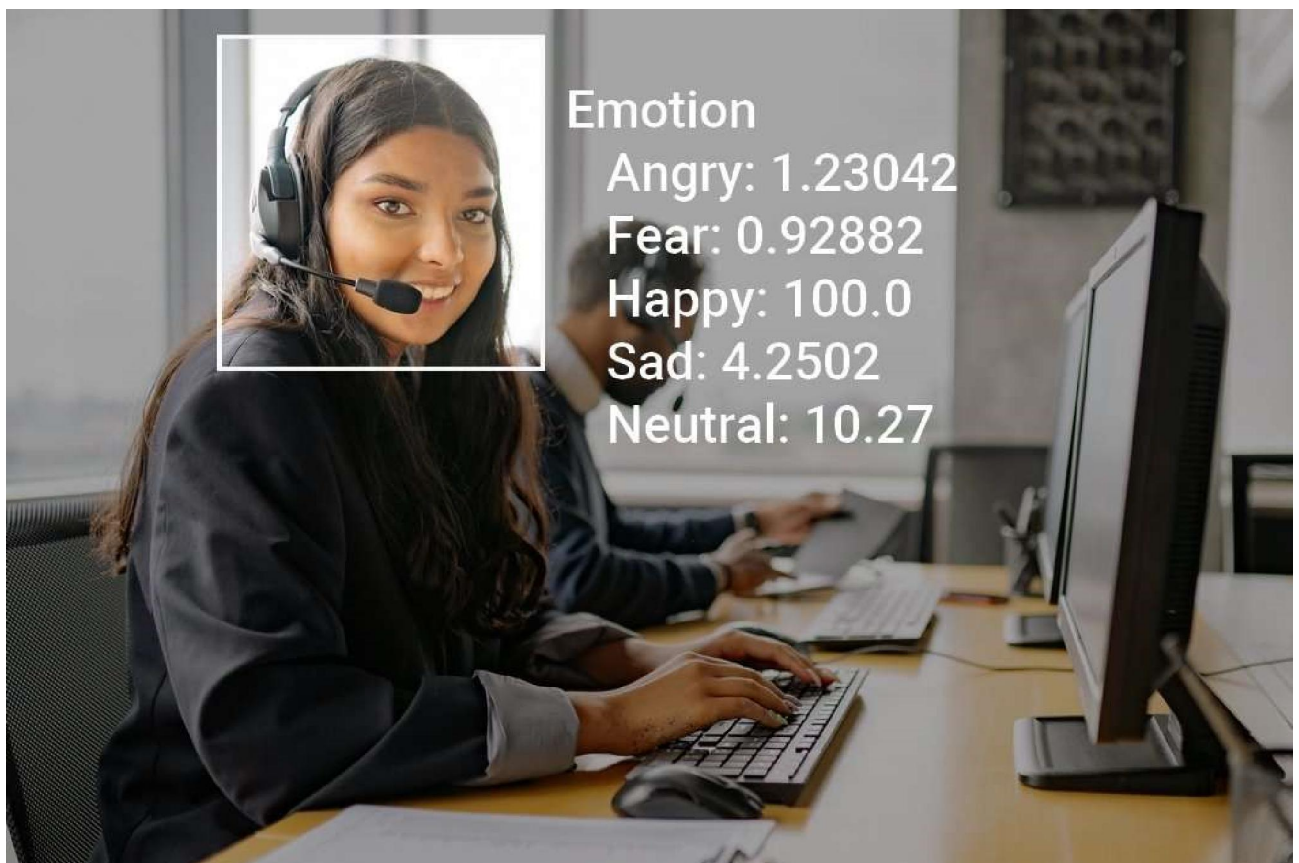


Рис. 1.12. Розпізнавання емоцій за допомогою <https://viso.ai/application/emotion-analysis/>

Системи розпізнавання емоцій надають можливості [6]:

а) виявлення емоційних станів: смуток, гнів, щастя, страх, здивування, нейтральний стан;

б) визначення зміни емоційних станів залежно від конкретних умов



навчання;

в) визначення оцінки впевненості для розпізнаних емоцій.

*Моніторинг відвідуваності* – розпізнавання та пошук обличчя в базі даних зображень для ідентифікації студентів (рис. 1.13).



Рис. 1.13. Розпізнавання обличчя за допомогою <https://viso.ai/application/face-recognition>

Переваги систем моніторингу відвідуваності [5]:

а) автоматизована та неінвазивна ідентифікація однієї людини або кількох людей;

б) розумне відеоспостереження для захисту інфраструктури установи;

в) наявні моделі глибокого навчання від Google, Facebook, стійкі до виразу обличчя, освітлення та пози.

Такі системи можуть бути реалізовані апаратно на основі Raspberry-pi [21] із застосуванням згорткових нейронних мереж – класу глибоких нейронних мереж, найбільш часто застосовуваних для аналізу візуальних зображень [2; 17; 21] та реалізованих у таких бібліотеках, як Caffe, Deeplearning4j, Dlib, Microsoft Cognitive Toolkit, TensorFlow, Theano, Torch.

## **Висновки до розділу 1**

1. Системи машинного (комп'ютерного) зору з початку 1960-х рр. пройшли тривалу еволюцію та широко застосовуються у 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.

2. Зміни в освітній галузі, спричинені пандемією COVID-19, не могли не відобразитись на класичних освітніх застосування систем машинного зору, модифікувавши існуючі та породивши нові напрями, серед яких – 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.

## РОЗДІЛ 2

### ПРОЄКТУВАННЯ 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 та рекурентних мереж для розпізнавання зображень (Python);

`OpenCV` – бібліотека комп'ютерного зору, орієнтована на програми реального часу та аналіз відео (C++ та ін.);

`PyTorchCV` – фреймворк на основі `PyTorch` для задач машинного зору: класифікації зображень, сегментації, виявлення та 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, так само як опрацювання даних у хмарі розробника.

Попри широкий спектр можливостей, що їх надає `viso.ai` ([1], [6] та ін.), цей сервіс для конструювання систем машинного зору із машинним навчанням має

високу вартість, через що є малопродатним 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) [12], 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 обладнанням, двовимірна графіка та машинне навчання.

При застосуванні хмарних фреймворків для розпізнавання, таких як Google Cloud [25], вартість визначається за одиницю (одне зображення або сторінка багатосторінкового зображення). Так, 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, таких як документи (PDF/TIFF), та зображень із рукописним текстом;

FACE\_DETECTION – визначення обличчя на зображенні;

IMAGE\_PROPERTIES – визначення набору властивостей зображення, таких як домінуючі кольори;

LABEL\_DETECTION – додавання міток на основі вмісту зображення;

LANDMARK\_DETECTION – визначення географічних орієнтирів на зображенні;

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, з новин, подій чи знаменитостей, і пошук подібних зображення 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 в якості технологічної основи для побудови систем машинного зору освітнього призначення.

## 2.2 Розробка прототипу системи машинного зору освітнього призначення

### 2.2.1 Розпізнавання статичних обличь у Microsoft Azure

Практичне заняття із розпізнавання емоцій «Happy, Sad, Angry Workshop», розроблене Дж. Беннетт (Jim Bennett) [4], надає можливість розробки прототипу відповідної системи машинного зору з використанням Azure Face API – програмного інтерфейсу, що надає можливість визначення положення обличчя на зображенні, визначення 27 точок на зображенні включно із позицією очей, а також визначення 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 у домені другого рівня (зокрема, Криворізького державного педагогічного університету) не розпізнаються як такі, що належать академічним установам – у цьому випадку рекомендується звичайна реєстрація (рис. 2.1).

Після отримання ключа його необхідно перевірити за допомогою тестового коду (рис. 2.2) або у веб-формі (рис. 2.3). У випадку неправильного ключа результатом буде повідомлення про помилку 400, 401, 403, 408, 415 або 429:

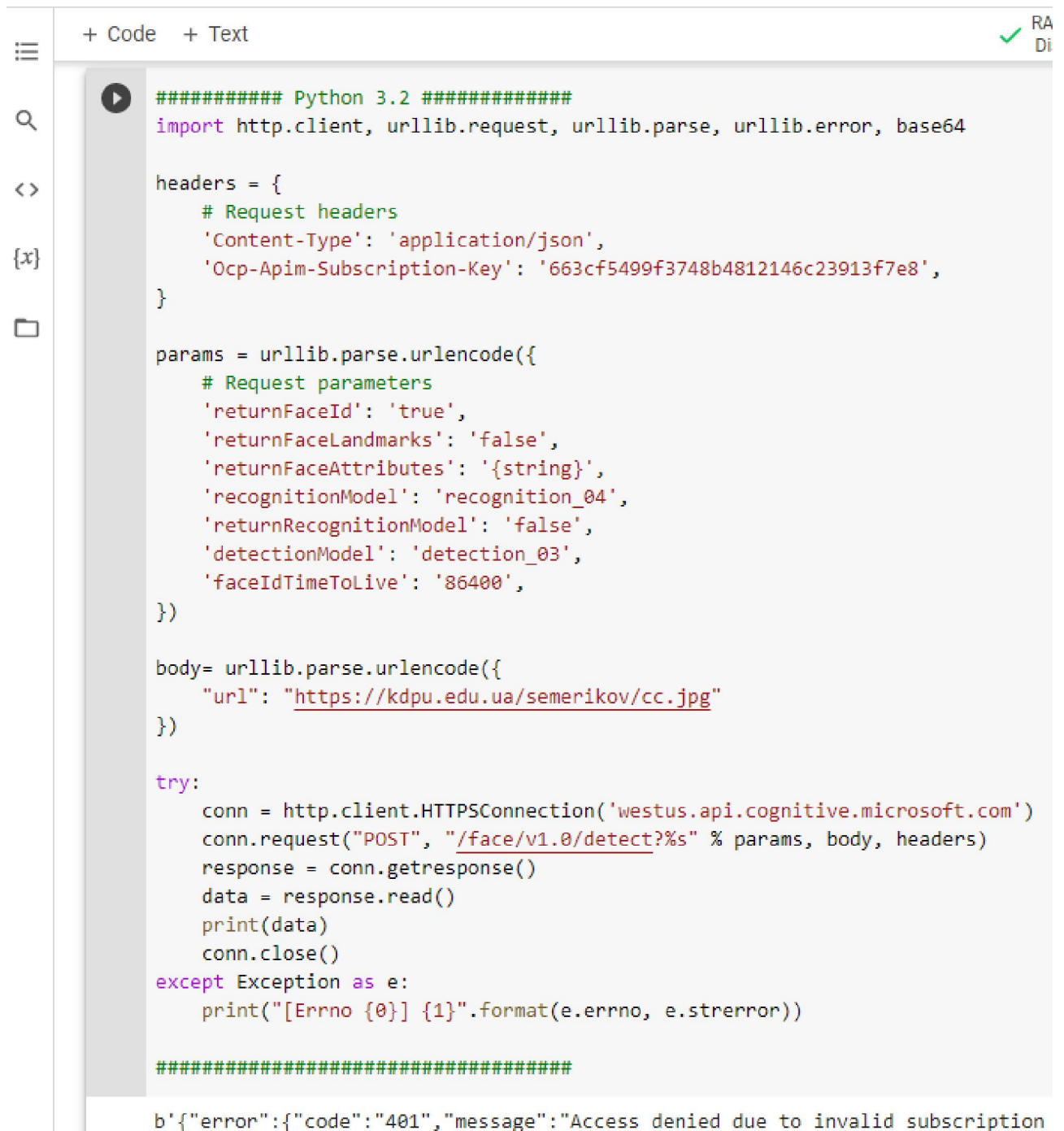
```
b'{"error":{"code":"401","message":"Access denied due to invalid subscription key or wrong API endpoint. Make sure to provide a valid key for an active subscription and use a correct regional API endpoint for your resource."}}'
```

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

The screenshot shows the Microsoft Azure portal interface. At the top, there's a navigation bar with the Microsoft Azure logo and a search bar. Below that, the user's account information is displayed: 'ruban' (Cognitive services multi-service account). A search bar is present with the text 'Search (Ctrl+ /)'. The left sidebar contains a navigation menu with categories: Overview, Activity log, Access control (IAM), Tags, Diagnose and solve problems, Resource Management (Keys and Endpoint, Pricing tier, Networking, Identity, Cost analysis, Properties, Locks), Monitoring (Alerts, Metrics, Diagnostic settings, Logs), and Automation (Tasks (preview), Export template). The main content area shows a 'Help us improve All Cognitive Services. Take our survey!' banner. Below it, the 'Essentials' section for a resource group named 'diploma' is displayed, including a 'JSON View' link. The details listed are: Resource group (diploma), Status (Active), Location (West Europe), Subscription (Azure subscription 1), Subscription ID (f3f31c92-3e7d-40d5-8915-e51900b1f383), API type (All Cognitive Services), Pricing tier (Standard), Endpoint (https://ruban.cognitiveservices.azure.com/), and a link to 'Manage keys'. At the bottom, there's a 'Get Started' section with a heading 'Build intelligent apps using a comprehensive family of AI services and cognitive APIs' and sub-sections for 'Decision', 'Language', and 'Speech'.

Рис. 2.1. Реєстрація у Cognitive Services

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.



```

+ Code + Text
RA Di
##### 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({
    "url": "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. 2.2. Перевірка ключа за допомогою тестового коду

Рекомендована модель розпізнавання (recognitionModel) – recognition\_04: саме за цією моделлю, уведеною в лютому 2021 року, відбувається розпізнавання масок.

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

**Face verification**

Check the likelihood that two faces belong to the same person and receive a confidence score.


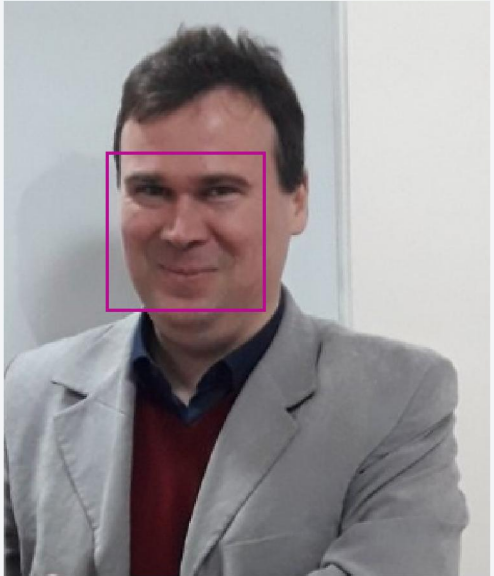



Image URL

Submit

Browse

Image URL

Submit

Browse

Verification result: The two faces belong to the same person. **Confidence is 0.93301.**

Рис. 2.3. Результати ідентифікації статичного зображення

Розглянемо код для отримання властивостей правого зображення з рис. 2.3:

```
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%80%D0%B0%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'
```

```
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,headpo
se,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 скористаємось наведеним кодом. Результат запиту для вказаного зображення буде наступним:

"faceId": "1b98e2a9-0c6c-4864-91d7-b8281e9377cc" – унікальний ідентифікатор обличчя 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, в якому на зображенні розташовано обличчя;

"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":

{ "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 } } – набір орієнтирів із 27 точок, що вказують на важливі позиції компонентів обличчя;

"faceAttributes" – атрибути обличчя:

{ "headPose": { "pitch": -14.2, "roll": 1.3, "yaw": -9.3 } } – 3-D кути тангажу / нахилу / повороту обличчя;

"mask": { "type": "noMask", "noseAndMouthCovered": 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 – на середньої або вище. Даний атрибут доступний лише при використанні будь-яких комбінацій моделей виявлення detection\_01 або detection\_03 і моделей розпізнавання recognition\_03 або recognition\_04.

Встановивши параметр returnFaceAttributes у 'accessories, age, blur, emotion, exposure, facialhair, gender, glasses, hair, headpose, makeup, noise, occlusion, qualityforrecognition, smile' та detectionModel у 'detection\_01', отримаємо наступні атрибути обговорюваного обличчя:

"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. Довжина – це число між [0, 1]: 0 для відсутності волосся на обличчі в цій області, 1 для довгих або дуже густих волосся на обличчі в цій області;

"glasses": "NoGlasses" – наявність окулярів: 'NoGlasses', '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": false, "lipMakeup": false} – макіяж: нафарбовані області очей та губ чи ні;

"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 обличчя у Microsoft Azure

Адаптація прикладу [4] 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 використано будь-яке середовище програмування, у тому числі – й спеціалізований текстовий редактор Visual Studio Code, в якому необхідно встановити модуль розширення Python Extension for Visual Studio Code та Azure App Service Extension for Visual Studio Code (рис. 2.4).

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 можливість застосування різних версій Python (наразі – до 3.9 включно).

Налаштування Visual Studio Code для використання Flask виконаємо за процедурою, описаною Дж. Беннеттом (рис. 2.5) – це надає можливість налагодження серверної частини системи. У процесі налаштування виявилось, що для правильної роботи Flask у терміналі необхідно налаштувати змінну оточення `FLASK_APP`, встановивши її у повний шлях до серверної частини системи – файлу `app.py` (рис. 2.6, додаток Б), а також увести зміни до файлу конфігурації Flask у Visual Studio Code (рис. 2.7).

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

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

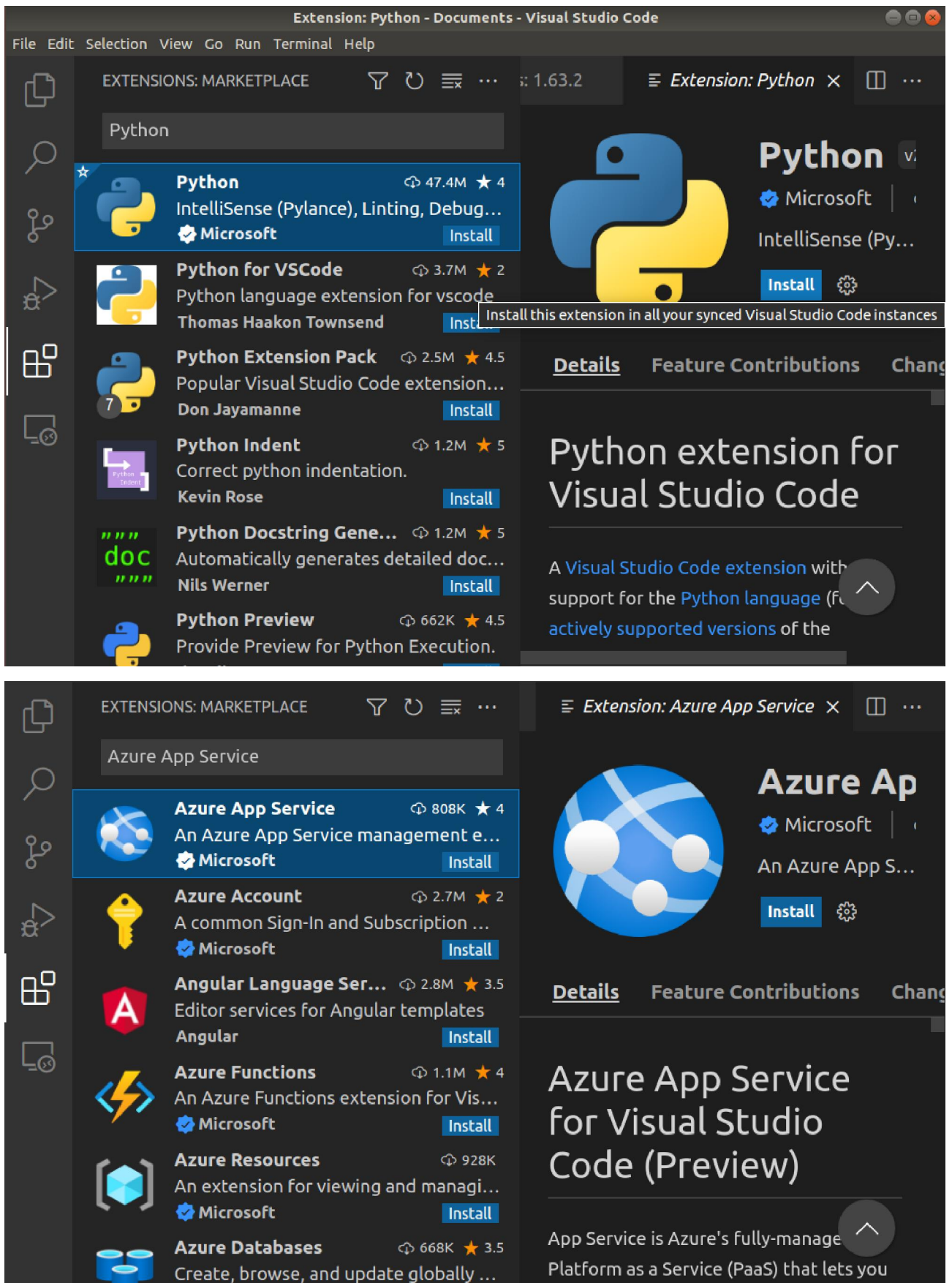


Рис. 2.4. Розширення Visual Studio Code, необхідні для розробки

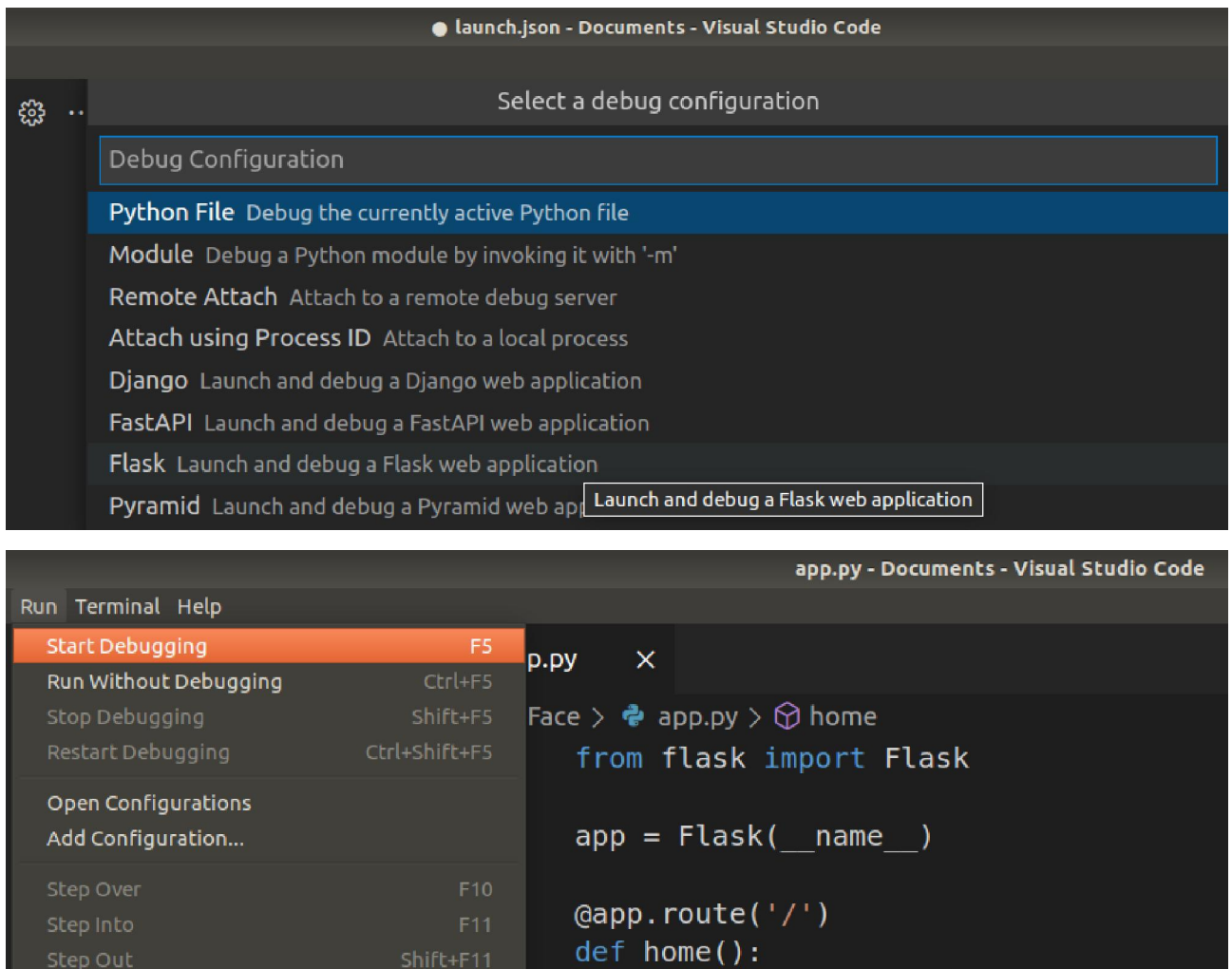


Рис. 2.5. Налаштування Visual Studio Code для запуску серверної частини з використанням Flask

```

/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. 2.6. Запуск Flask у терміналі Linux

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. 2.8). 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. 2.9).

На рис. 2.10 подано протокол розгортання системи у хмарі.

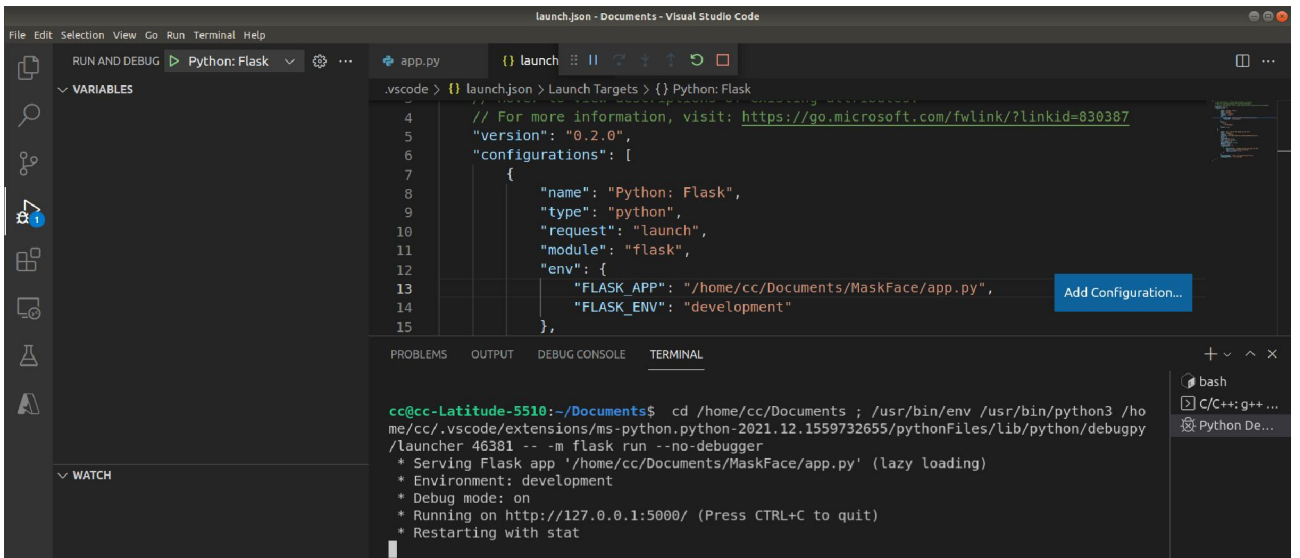


Рис. 2.7. Запуск Flask у терміналі Visual Studio Code

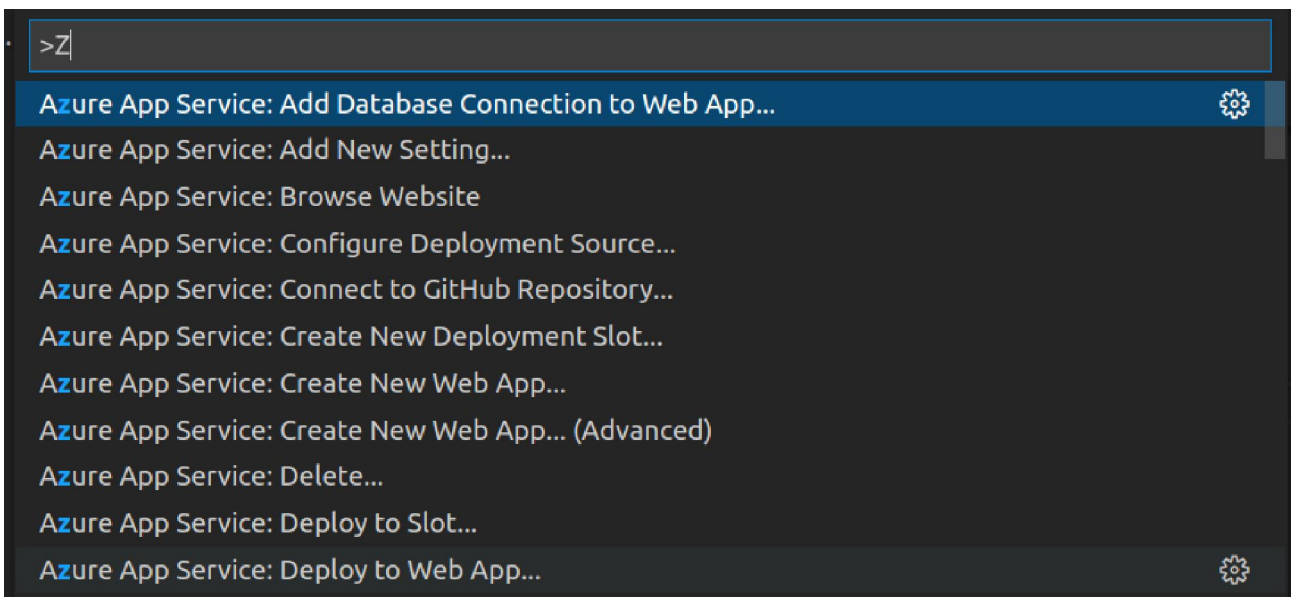


Рис. 2.8. Налаштування програмного забезпечення у хмарі Azure

Моделі машинного навчання можуть бути створені як повністю самостійно, так й на основі моделей, створених іншими. Microsoft пропонує ряд попередньо навчених моделей, які й називаються Cognitive Services. Ці моделі включають розпізнавання зображень, розпізнавання мовлення та переклад між різними мовами. Face API реалізує моделі машинного навчання для пошуку обличчя на зображенні. Для доступу для нього є необхідним ключ підписки, який можна отримати на порталі Azure (рис. 2.11).



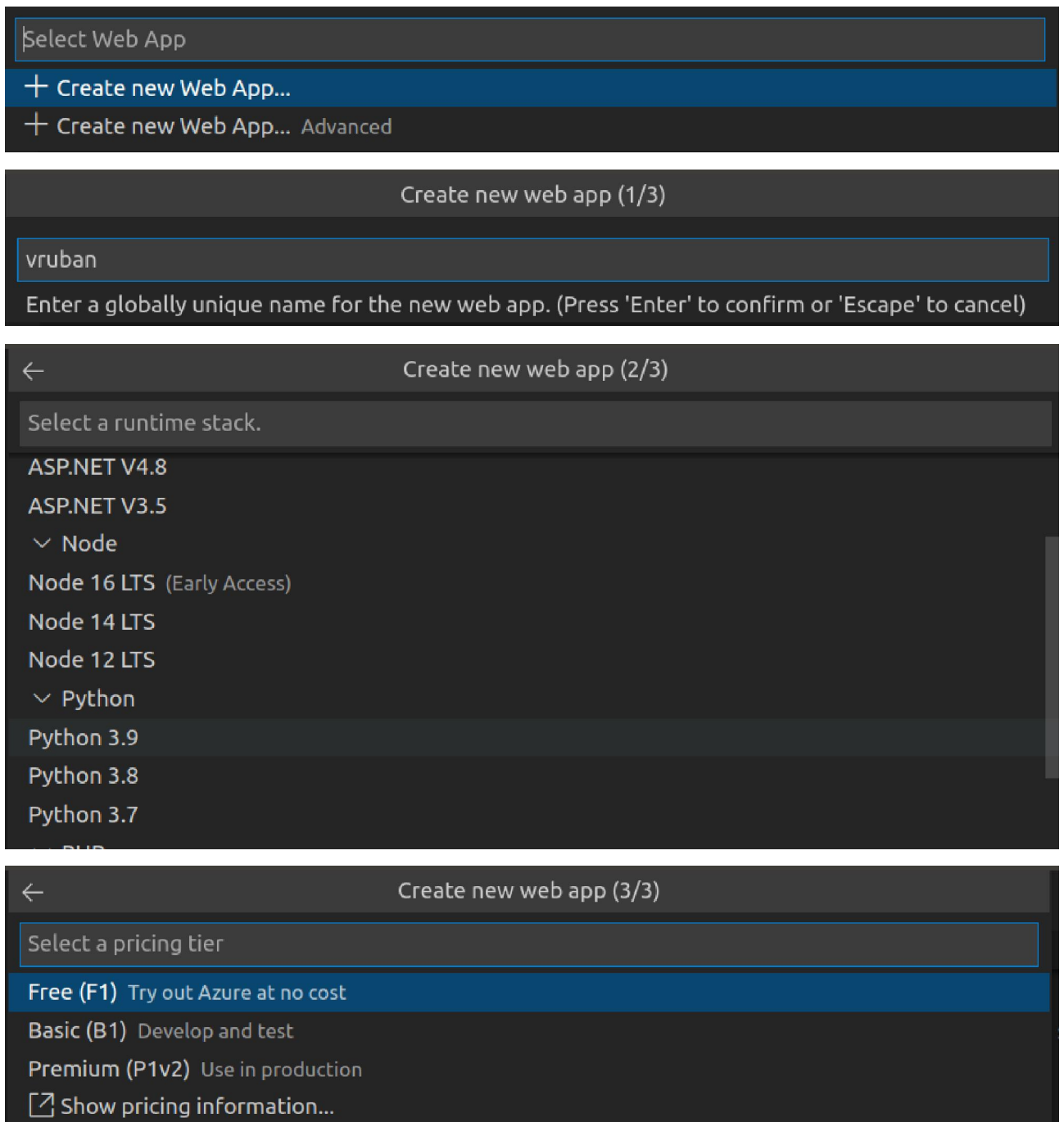


Рис. 2.9. Кроки розгортання системи у хмарі

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 20



викликів API на хвилину та менше 30000 викликів на місяць, можна обрати безкоштовний рівень F0 – лише один для кожної служби Azure.

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
Azure App Service
1:14:15 AM: Creating resource group "appsvc_linux_centralus" in location "Central US"...
1:14:18 AM: Successfully created resource group "appsvc_linux_centralus".
1:14:18 AM: Ensuring App Service plan "appsvc_linux_centralus" exists...
1:14:19 AM: Creating App Service plan "appsvc_linux_centralus"...
1:14:34 AM: Successfully created App Service plan "appsvc_linux_centralus".
1:14:34 AM: Verifying that Application Insights is available for this location...
1:14:35 AM: Creating Application Insights resource "vruban"...
1:14:46 AM: Successfully created Application Insights resource "vruban".
1:14:46 AM: Creating new web app "vruban"...
1:15:10 AM: Created new web app "vruban": https://vruban.azurewebsites.net
1:15:18 AM vruban: Starting deployment...
Deploying to "vruban"... Check output window for status.
Always deploy the workspace
"/home/cc/Documents/MaskFace" to "vruban"?
Source: Azure App Service (...) Yes Don't show again Skip for now

1:16:14 AM vruban: Activating virtual environment...
1:16:14 AM vruban: Could not find setup.py or requirements.txt; Not running pip install
1:16:14 AM vruban: Not a vso image, so not writing build commands
1:16:14 AM vruban: Preparing output...
1:16:14 AM vruban: Copying files to destination directory '/tmp/_preCompressedDestinationDir'...
1:16:14 AM vruban: Done in 0 sec(s).
1:16:14 AM vruban: Compressing content of directory '/tmp/_preCompressedDestinationDir'...
1:16:15 AM vruban: Copied the compressed output to '/home/site/wwwroot'
1:16:15 AM vruban: Removing existing manifest file
1:16:15 AM vruban: Creating a manifest file...
1:16:15 AM vruban: Manifest file created.
1:16:15 AM vruban: Done in 22 sec(s).
1:16:17 AM vruban: Running post deployment command(s)
1:16:17 AM vruban: Triggering recycle (preview mode)
1:16:17 AM vruban: Deployment successful.
1:16:32 AM: Deployment to "vruban" completed.
Deployment to "vruban" completed.
Source: Azure A... Browse Website Stream Logs Upload Settings


```

Рис. 2.10. Протокол розгортання системи у хмарі

Microsoft Azure Search resources, services, and docs (G+)

Home > Create a resource >

**Face** ✨ ...  
Microsoft

 **Face** ❤️ Add to Favorites  
Microsoft  
★ 4.4 (199 Azure ratings)

Create

Overview Plans Usage Information + Support Reviews

Embed facial recognition into your apps for a seamless and highly secured user experience. No machine learning expertise is required. Features include: face detection that perceives faces and attributes in an image; person identification that matches an individual in your private repository of up to 1 million people; emotion recognition that perceives a range of reactions like happiness, contempt, neutrality, and fear; and recognition and grouping of similar faces in images.

Рис. 2.11. Створення ресурсу для використання Face API

**Basics** Network Identity Tags Review + create

Embed facial recognition into your apps for a seamless and highly secured user experience. No machine learning expertise is required. Features include: face detection that perceives faces and attributes in an image; person identification that matches an individual in your private repository of up to 1 million people; emotion recognition that perceives a range of reactions like happiness, contempt, neutrality, and fear; and recognition and grouping of similar faces in images. [Learn more.](#)

### Project Details

Subscription \* ⓘ Azure subscription 1

Resource group \* ⓘ appsvc\_linux\_centralus  
[Create new](#)

### Instance Details

Region ⓘ Central US

Name \* ⓘ ruban1

Basics **Network** Identity Tags Review + create

**i** Configure network security for your cognitive service resource. ↗

- 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)

Рис. 2.12. Налаштування ресурсу для використання Face API

За умови успішного налаштування можна буде отримати секретні ключи та кінцеву точку для доступу до Face API (рис. 2.13).

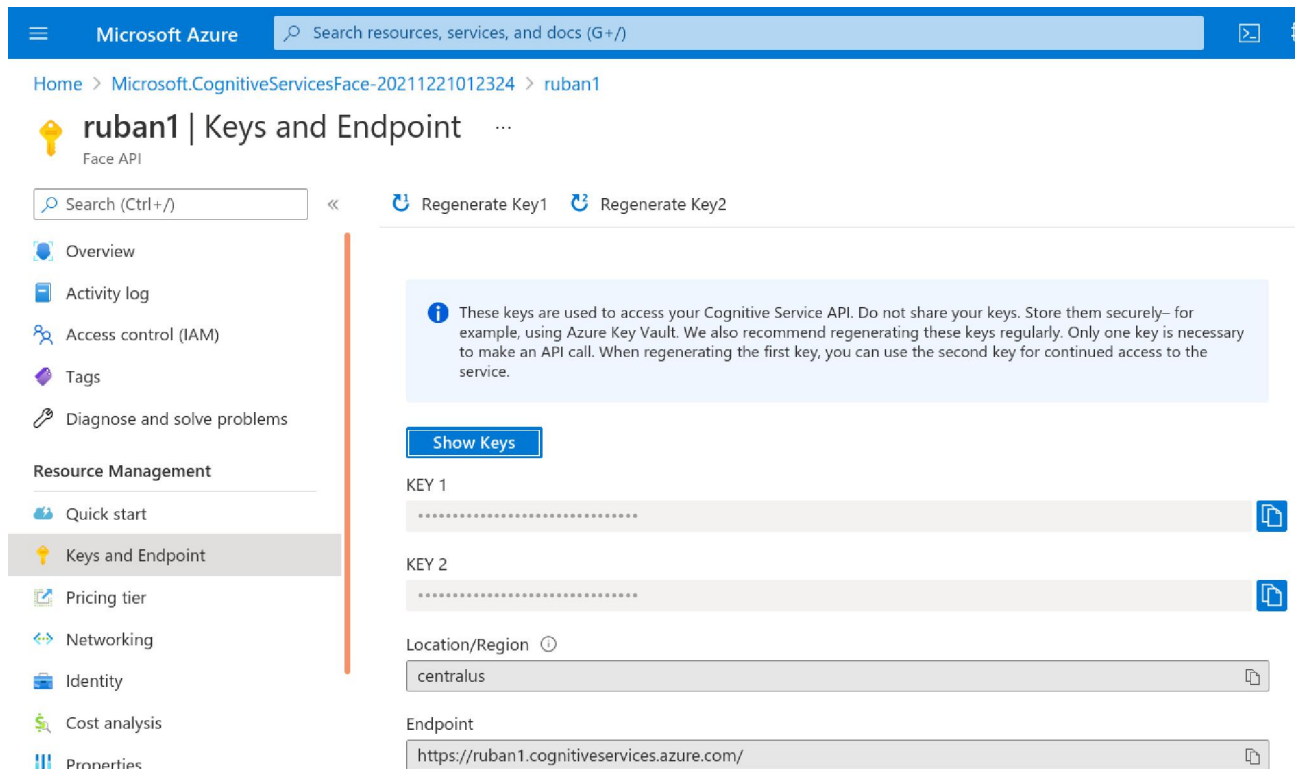


Рис. 2.13. Отримання секретних ключів та адреси кінцевої точки для доступу

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

У процесі доопрацювання коду [4] були виконані наступні основні зміни у клієнтській частині:

1) підвищено роздільну здатність до 1024x768 для кращого розпізнавання обличчя – 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. 2.14) 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.



Чи я у масці? Проаналізувати обличчя  
На Вашому обличчі немає маски, Ваші рід та ніс не закриті

а)



Чи я у масці? Проаналізувати обличчя  
На Вашому обличчі є маска, Ваші рід та ніс закриті

б)



Чи я у масці? Проаналізувати обличчя  
На Вашому обличчі є маска, Ваші рід та ніс не закриті

в)



Чи я у масці? Проаналізувати обличчя  
На Вашому обличчі є маска, Ваші рід та ніс не закриті

г)

Рис. 2.14. Тестування системи у режимі визначення маски

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 кнопкою «Проаналізувати обличчя» відбувається отримання таких атрибутів обличчя, яка 'emotion', 'accessories', 'age', 'facialhair', 'gender', 'glasses', 'hair', 'makeup' та 'smile'.

### 2.2.3 Тестування розробленого прототипу

На рис. 2.14 подано результати тестування системи у режимі визначення маски: із 4 тестів 3 виконано правильно, а один (рис. 2.14 г) – частково правильно (визначено, що рот маскою не закритий).

У додатку В наведено результати тестування системи на мобільних та стаціонарних пристроях. Узагальнення результатів тестування надало можливість зробити наступні висновки щодо ідентифікації обличч за допомогою навчених моделей Face API:

1) «видимий вік» обличч суттєво залежить від настрою їх власників – чим він краще, тим вік ближче до біологічного: це пов'язано із тим, що навчання відбувалось на множині переважно усміхнених обличч;

2) найгірше ідентифікується світле (русяве) волосся – у залежності від освітлення воно може бути «будь-якого кольору» (хибно ідентифікуватись як сіре, блондинисте, чорне);

3) нерівномірне освітлення та тіні породжують хибні ефекти закривання 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.

## **Висновки до розділу 2**

1. Серед великої кількості бібліотек машинного зору загального призначення найбільш доцільними в освітньому процесі є застосування автономних бібліотек типу 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 надає можливість конструювання систем машинного зору непрограмуючим користувачам. Розгортання програмного забезпечення можлива 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, так само як опрацювання даних у хмарі розробника, проте компенсується доступом до великої кількості навчених моделей машинного навчання, таких як Cognitive Services.

2. Розроблений на основі практикуму Дж. Беннетта із розпізнавання емоцій «Happy, Sad, Angry Workshop» прототип системи машинного зору інтегрує у собі розпізнавання емоцій учасників освітнього процесу та виявлення порушень маскового режиму, додатково надаючи можливість з високим ступенем надійності визначати стать, інтенсивність посмішки і наявність окулярів та з середнім – вік, наявність головного убору, макіяжу, кольору волосся тощо.

## 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 у 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 й за пандемічних. Зміни в освітній галузі, спричинені пандемією COVID-19, відобразились на класичних освітніх застосування систем машинного зору, модифікувавши існуючі та породивши нові напрями, серед яких – 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 типу 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 надає можливість конструювання систем машинного зору непрограмуємим користувачам. Розгортання програмного забезпечення можлива 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, так само як опрацювання даних у хмарі розробника, проте компенсується доступом до великої кількості навчених моделей машинного навчання, таких як Cognitive Services.

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

## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Abandoned Luggage [Electronic resource] / viso.ai. – [2021?]. – Access mode : <https://viso.ai/application/abandoned-luggage-detection/>
2. Agarwal L. Face Recognition Based Smart and Robust Attendance Monitoring using Deep CNN / L. Agarwal, M. Mukim, H. Sharma, A. Bhandari, A. Mishra // 2021 8th International Conference on Computing for Sustainable Global Development (INDIACom). – 2021. – P. 699-704. – DOI : 10.1109/INDIACom51348.2021.00124.
3. Ballard D. H. Computer Vision / Dana H. Ballard, Christopher M. Brown. – Englewood Cliffs : Prentice Hall, 1982. – Access mode : <https://archive.org/details/computervision0000ball>
4. Bennett J. Happy, Sad, Angry Workshop [Electronic resource] / Jim Bennett. – 2020. – Access mode : <https://github.com/jimbobbennett/HappySadAngryWorkshop>
5. Face Recognition [Electronic resource] / viso.ai. – [2021?]. – Access mode : <https://viso.ai/application/face-recognition/>
6. Facial Emotion Analysis [Electronic resource] / viso.ai. – [2021?]. – Access mode : <https://viso.ai/application/emotion-analysis/>
7. Gibson J. J. The Perception of the Visual World / J. J. Gibson. – Boston : Houghton Mifflin, 1950.
8. Google Ngram Viewer [Electronic resource]. – 2021. – Access mode : [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](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)
9. Grape G. R. Model Based (Intermediate-Level) Computer Vision : PhD Dissertation / Gunnar Rutger Grape. – Stanford University, 1973. – Access mode : <https://apps.dtic.mil/sti/pdfs/AD0763673.pdf>

10. Intrusion Detection [Electronic resource] / viso.ai. – [2021?]. – Access mode : <https://viso.ai/application/intrusion-detection/>
11. Klingler N. Top 8 Applications of Computer Vision in the Education Sector [Electronic resource] / Nico Klingler. – September 7, 2021. – Access mode : <https://viso.ai/applications/computer-vision-in-education/>
12. Libraries comparison [Electronic resource] / Adaptive Vision. – 2021. – Access mode : [https://docs.adaptive-vision.com/avl/technical\\_issues/LibrariesComparison.html](https://docs.adaptive-vision.com/avl/technical_issues/LibrariesComparison.html)
13. Mask Detection: Automatically detect unmasked people in public spaces or indoors [Electronic resource] / viso.ai. – [2021?]. – Access mode : <https://viso.ai/application/mask-detection/>
14. Parking Lot Occupancy [Electronic resource] / viso.ai. – [2021?]. – Access mode : <https://viso.ai/application/parking-lot-occupancy-detection/>
15. Prince S. J. D. Computer Vision: Models, Learning, and Inference / Simon J. D. Prince. – Cambridge University Press, 2012.
16. Pulliam J. R. C. Increased risk of SARS-CoV-2 reinfection associated with emergence of the Omicron variant in South Africa / Juliet R. C. Pulliam, Cari van Schalkwyk, Nevashan Govender, Anne von Gottberg, Cheryl Cohen, Michelle J. Groome, Jonathan Dushoff, Koleka Mlisana, Harry Moultrie // medRxiv. – 2021. – 2021.11.11.21266068. – DOI : 10.1101/2021.11.11.21266068
17. Raj A. Smart Attendance Monitoring System with Computer Vision Using IOT / Ashwin Raj, Aparna Raj, Imteyaz Ahmad // Journal of Mobile Multimedia. – 2021. – Vol. 17(1-3). – P. 115-125. – DOI : 10.13052/jmm1550-4646.17135
18. Rezaei M. DeepSOCIAL: Social Distancing Monitoring and Infection Risk Assessment in COVID-19 Pandemic / Mahdi Rezaei, Mohsen Azarmi // Applied Sciences. – 2020. – Vol. 10, no. 21. – Article 7514. – DOI : 10.3390/app10217514
19. Roberts L. G. Machine perception of three-dimensional solids : Thesis (Ph. D.) [Electronic resource] / Lawrence Gilman Roberts. – Massachusetts Institute of Technology, Dept. of Electrical Engineering, 1963. – Access mode : <https://dspace.mit.edu/bitstream/handle/1721.1/11589/>

20. Shorten C. Deep Learning applications for COVID-19 / Connor Shorten, Taghi M. Khoshgoftaar, Borko Furht // *Journal of Big Data*. – 2021. – Vol. 8. – Article 18. – DOI : 10.1186/s40537-020-00392-9
21. Sivakumar S. A. IoT based Intelligent Attendance Monitoring with Face Recognition Scheme / S. A. Sivakumar, T. J. John, G. T. Selvi, B. Madhu, C. U. Shankar, K. P. Arjun // 2021 5th International Conference on Computing Methodologies and Communication (ICCMC). – 2021. – P. 349-353. – DOI : 10.1109/ICCMC51019.2021.9418264
22. Social Distancing Monitoring [Electronic resource] / viso.ai. – [2021?]. – Access mode : <https://viso.ai/application/social-distancing-monitoring/>
23. Sutherland I. E. Sketchpad, a man-machine graphical communication system [Electronic resource] : submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy / Ivan Edward Sutherland ; Massachusetts Institute of Technology, Department of Electrical Engineering. – January, 1963. – 176 p. – Access mode : [http://images.designworldonline.com.s3.amazonaws.com/CADhistory/Sketchpad\\_A\\_Man-Machine\\_Graphical\\_Communication\\_System\\_Jan63.pdf](http://images.designworldonline.com.s3.amazonaws.com/CADhistory/Sketchpad_A_Man-Machine_Graphical_Communication_System_Jan63.pdf)
24. Tkachuk V. Using Mobile ICT for Online Learning During COVID-19 Lockdown / Viktoriia Tkachuk, Yuliia Yechkalo, Serhiy Semerikov, Maria Kislova, Yana Hladyr // *Information and Communication Technologies in Education, Research, and Industrial Applications. 16th International Conference, ICTERI 2020, Kharkiv, Ukraine, October 6–10, 2020, Revised Selected Papers* / Editors : Andreas Bollin, Vadim Ermolayev, Heinrich C. Mayr, Mykola Nikitchenko, Aleksander Spivakovsky, Mykola Tkachuk, Vitaliy Yakovyna, Grygoriy Zholtkevych // *Communications in Computer and Information Science*. – Cham : Springer, 2021. – Vol. 1308. – P. 46–67. – DOI : 10.1007/978-3-030-77592-6\_3.
25. Vision API Product Search pricing [Electronic resource] / Google Cloud. – 2021. – Access mode : <https://cloud.google.com/vision/product-search/pricing>

## ДОДАТКИ

### 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"),
        }

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

            var jsonResponse = await result.json()
            message.textContent = jsonResponse.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>

```

## Б 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['огида'] = 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 на Вашому обличчі маска'
        result += ', Ваші ріт та ніс '
        if faces[0].face_attributes.mask.nose_and_mouth_covered:
            result += 'закриті'
        else:
            result += 'не закриті'
        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 вусаті на ' +
str(faces[0].face_attributes.facial_hair.moustache*100) + '%, '
        result += 'бородаті - на ' +
str(faces[0].face_attributes.facial_hair.beard*100) + '% '
        result += 'та \"бакенбардаті\" 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 убор, "
        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'
        })

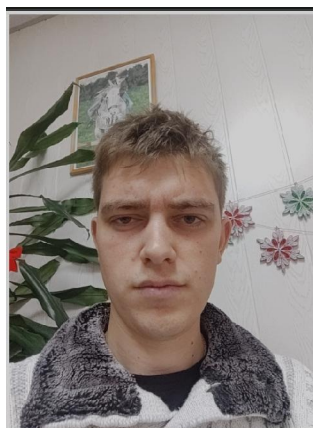
```

## В Тестування прототипу



Чи я у масці? | Проналізувати обличчя

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 убор, Ваше волосся для камери невидиме.



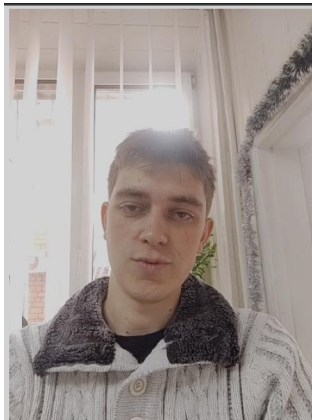
Чи я у масці? | Проналізувати обличчя

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.



Чи я у масці? Прозналізувати обличчя

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.



Чи я у масці? Прозналізувати обличчя

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.



Чи я у масці? Прозналізувати 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.



Чи я у масці? Прозналізувати 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.



Чи я у масці? | Проаналізувати 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.



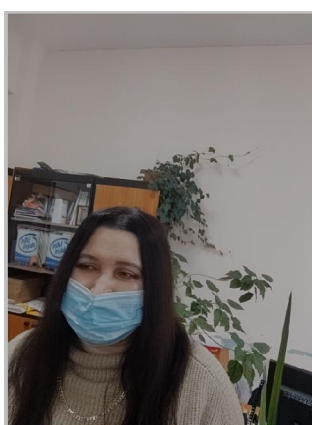
Чи я у масці? | Проаналізувати обличчя

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 убор, маска, Ваше волосся для камери невидиме.



Чи я у масці? | Проаналізувати обличчя

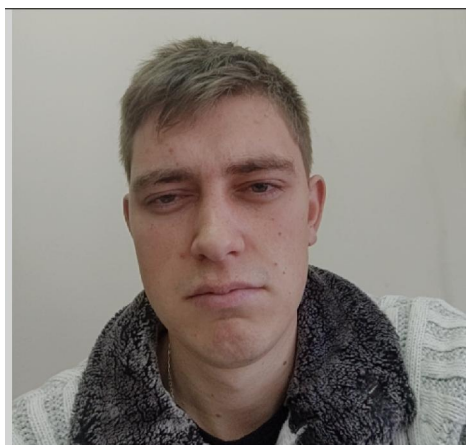
Ваш настрій - нейтральний, Ви посміхаєтесь на 0.3%, Ви виглядаєте на 28.0 років, Ви вусаті на 0.0%, бородаті - на 0.0% та "бакенбардаті" на 0.0%, Ваша стать - жіноча, на Вас немає окулярів, Ваші очі нафарбовані, Ваші губи не нафарбовані, Ваш рот чимось закритий, Ваші очі нічим не закриті, Ваш лоб чимось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



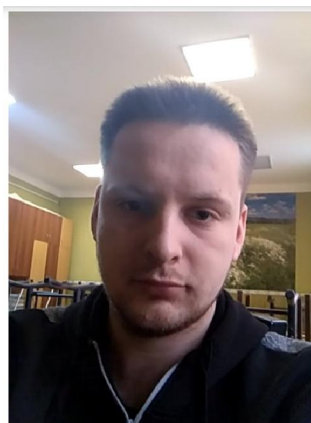
Чи я у масці? Проаналізувати обличчя

Ваш настрій - смуток, Ви посміхаєтесь на 0.0%, Ви виглядаєте на 30.0 років, Ви вусаті на 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, Ваше волосся для камери невидиме, густина Вашого волосся - 40.0%, колір Вашого волосся - сірий.



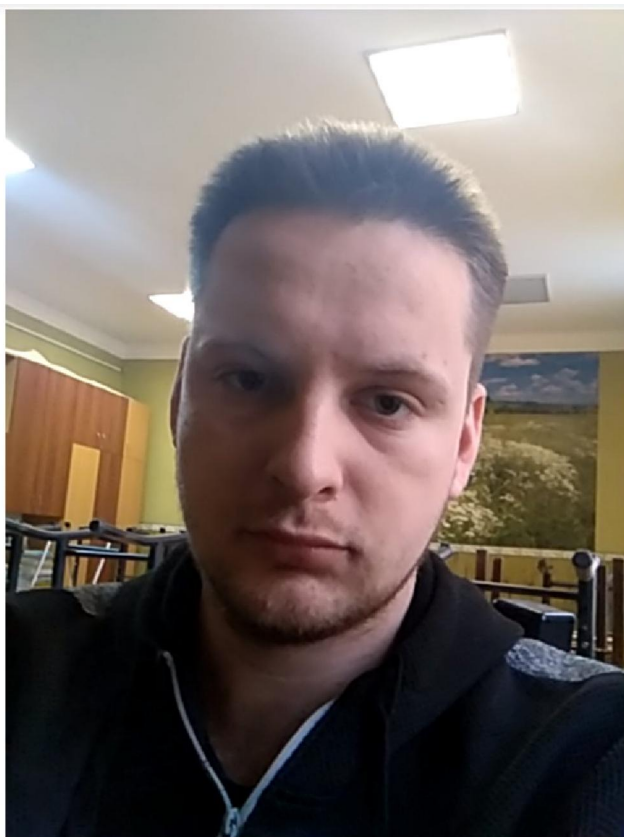
Чи я у масці? Проаналізувати обличчя

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 невидиме, густина Вашого волосся - 40.0%, колір Вашого волосся - сірий.



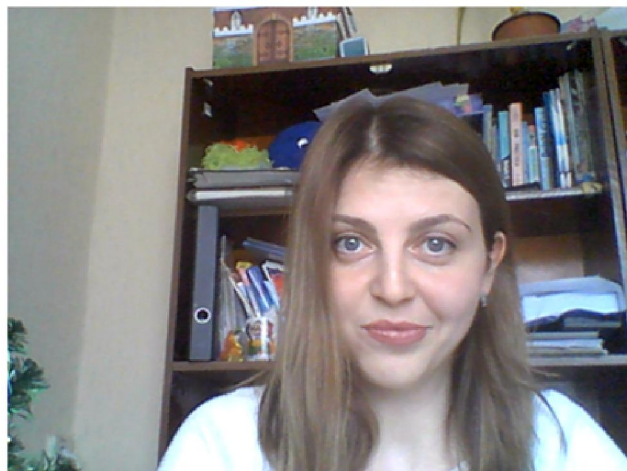
Чи я у масці? Проаналізувати обличчя

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 невидиме, густина Вашого волосся - 49.0%, колір Вашого волосся - сірий.



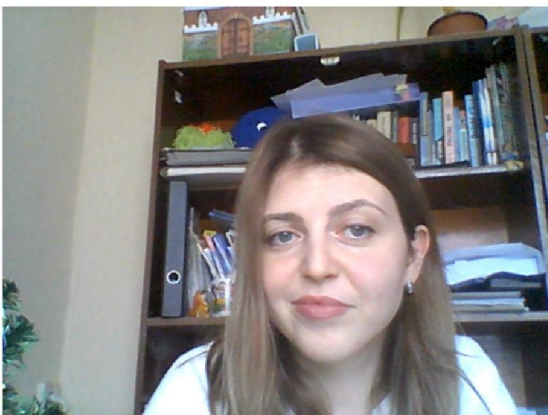
Чи я у масці? Проаналізувати обличчя

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



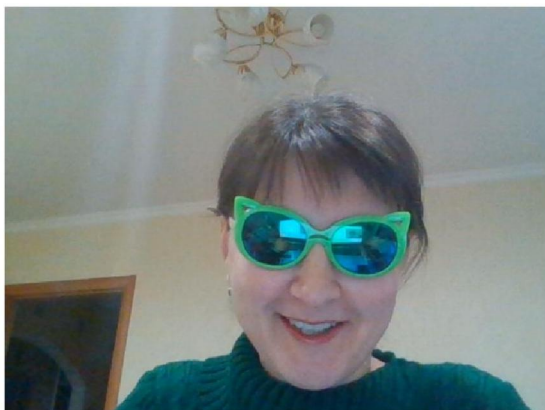
Чи я у масці? Проаналізувати обличчя

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



Чи я у масці? Проаналізувати обличчя

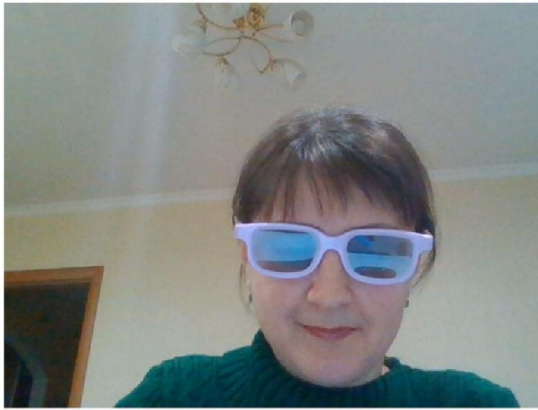
Ваш настрій - презрето, Ви посміхаєтесь на 5.4%, Ви виглядаєте на 32.0 років, Ви вусаті на 0.0%, бородаті - на 0.0% та "бакенбардаті" на 0.0%, Ваша стать - жіноча, на Вас немає окулярів, Ваші очі нафарбовані, Ваші губи нафарбовані, Ваш рот нічим не закритий, Ваші очі нічим не закриті, Ваш лоб нічим не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



Чи я у масці? Проаналізувати обличчя

Ваш настрій - щастя, Ви посміхаєтесь на 100.0%, Ви виглядаєте на 41.0 років, Ви вусаті на 0.0%, бородаті - на 0.0% та "бакенбардаті" на 0.0%, Ваша стать - жіноча, на Вас окуляри для плавання, Ваші очі не нафарбовані, Ваші губи не нафарбовані, Ваш рот нічим не закритий, Ваші очі нічим не закриті, Ваш лоб чимось закритий, на Вас окуляри, головний убір, Ваше волосся для камери невидиме.





Чи я у масці? Проаналізувати обличчя

Ваш настрій - щастя, Ви посміхаєтесь на 100.0%, Ви виглядаєте на 35.0 років, Ви вусаті на 0.0%, бородаті - на 0.0% та "бакенбардаті" на 0.0%, Ваша стать - жіноча, на Вас сонцезхисні окуляри, Ваші очі нафарбовані, Ваші губи не 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 - чорний.



Чи я у масці? Проаналізувати обличчя

Ваш настрій - нейтральний, Ви посміхаєтесь на 5.3%, 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 сонцезхисні 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 - коричневатий.



Чи я у масці? Проаналізувати обличчя

Ваш настрій - нейтральний, Ви посміхаєтесь на 0.0%, Ви виглядаєте на 25.0 років, Ви вусаті на 0.0%, бородаті - на 0.0% та "бакенбардаті" на 0.0%, Ваша стать - жіноча, на Вас сонцезхисні 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.



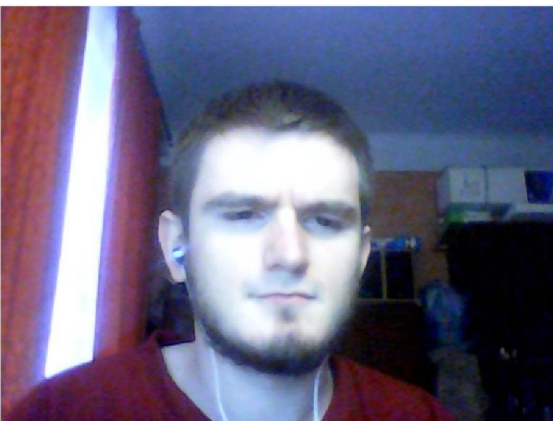
Чи я у масці? Проаналізувати обличчя

Ваш настрій - щастя, Ви посміхаєтесь на 99.5%, 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 сонцезхисні 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.



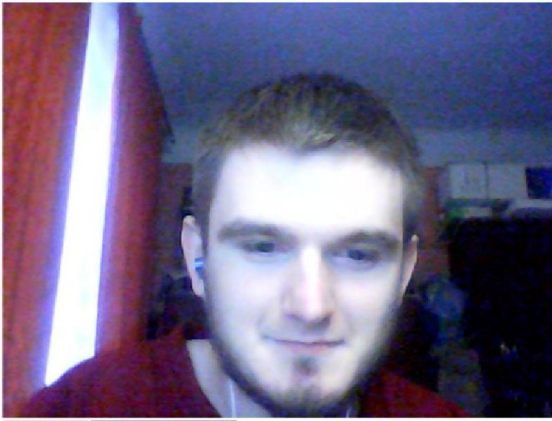
Чи я у масці? Проаналізувати обличчя

Ваш настрій - щастя, Ви посміхаєтесь на 99.8%, Ви виглядаєте на 21.0 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 - 100.0%, колір Вашого волосся - чорний.



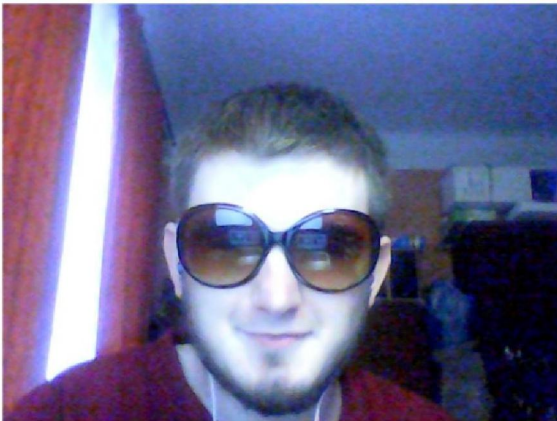
Чи я у масці? Проаналізувати обличчя

Ваш настрій - щастя, Ви посміхаєтесь на 78.60000000000001%, Ви виглядаєте на 30.0 років, Ви вусаті на 10.0%, бородаті - на 10.0% та "бакенбардаті" на 40.0%, 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 - 98.0%, колір Вашого волосся - рудий.



Чи я у масці?

Ваш настрій - щастя, Ви посміхаєтесь на 78.60000000000001%, Ви виглядаєте на 30.0 років, Ви вусаті на 10.0%, бородаті - на 10.0% та "бакенбардаті" на 40.0%, 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 - 98.0%, колір Вашого волосся - рудий.



Чи я у масці?

Ваш настрій - нейтральний, Ви посміхаєтесь на 32.0%, Ви виглядаєте на 28.0 років, Ви вусаті на 10.0%, бородаті - на 10.0% та "бакенбардаті" на 10.0%, Ваша стать - чоловіча, на Вас сонцезхисні 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 - 74.0%, колір Вашого волосся - коричневий.