

Міністерство освіти і науки України
Національний технічний університет
«Дніпровська політехніка»

Навчально-науковий
інститут електроенергетики
(навчально-науковий інститут)
Факультет інформаційних технологій
(факультет)
Кафедра інформаційних технологій та комп'ютерної інженерії
(повна назва)

ПОЯСНЮВАЛЬНА ЗАПИСКА
кваліфікаційної роботи ступеня магістра

Здобувача вищої освіти _____ Воеводи Максима Віталійовича _____
(ПІБ)
академічної групи _____ 123М-23-1 _____
(шифр)
спеціальності _____ 123 Комп'ютерна інженерія _____
(код і назва спеціальності)
за освітньо-професійною програмою _____ «Комп'ютерна інженерія» _____
(офіційна назва)

на тему «Обґрунтування структури комп'ютерної системи Управління патрульної служби міста Дніпро з функцією ідентифікації людей, що розшукуються, по фото»
(назва за наказом ректора)

Керівники	Прізвище, ініціали	Оцінка за шкалою		Підпис
		рейтинговою	інституційною	
кваліфікаційної роботи	проф. Цвіркун Л.І.			
розділів:				
синтез системи	доц. Бешта Д.О.			
розроблення програмного забезпечення	ас. Панферова Я.В.			
Рецензент				
Нормоконтролер	проф. Цвіркун Л.І.			

Дніпро
2024

ЗАТВЕРДЖЕНО:

завідувач кафедри
інформаційних технологій
та комп'ютерної інженерії
(повна назва)

_____ В.В. Гнатушенко
(підпис) (ініціали, прізвище)

« _____ » _____ 2024 року

ЗАВДАННЯ
на кваліфікаційну роботу
ступеня магістра
(бакалавра, магістра)

здобувача вищої освіти Воєводи М.В. академічної групи 123М-23-1
(прізвище та ініціали) (шифр)

спеціальності 123 Комп'ютерна інженерія

за освітньою-професійною програмою «Комп'ютерна інженерія»
(офіційна назва)

на тему «Обґрунтування структури комп'ютерної системи Управління патрульної служби міста Дніпро з функцією ідентифікації людей, що розшукуються, по фото»,
затверджену наказом ректора НТУ «Дніпровська політехніка» від 17 жовтня 2024 р. №1388-с

Розділ	Зміст	Термін виконання
Стан питання та постановка завдання	На основі матеріалів практик, інших науково-технічних джерел сформулювати наукове завдання, конкретизувати предмет та мету досліджень	11.10.2024
Теоретичний	Розробити математичні моделі для розв'язання наукового завдання кваліфікаційної роботи	25.10.2024
Синтез системи	Розробка комп'ютерної системи Управління патрульної служби міста Дніпро	15.11.2024
Розроблення програмного забезпечення	Розробка програмного забезпечення з функцією ідентифікації людей, що розшукуються, по фото	29.11.2024
Експериментальний розділ	Проведення і обробка результатів експериментів з перевіркою роботи системи ідентифікації людей, що розшукуються, по фото	06.12.2024

Завдання видано _____
(підпис керівника)

Дата видачі 06 вересня 2024 р.

Дата подання до екзаменаційної комісії

Прийнято до виконання _____
(підпис здобувача вищої освіти)

проф. Л. І. Цвіркун
(ініціали, прізвище)

10.12.2024 р.

М.В. Воєвода
(ініціали, прізвище)

РЕФЕРАТ

Пояснювальна записка: 89 с., 29 рис., 11 табл., 1 дод., 14 джерел.

КОМП'ЮТЕРНА СИСТЕМА, ПАТРУЛЬНА СЛУЖБА, КОРПОРАТИВНА МЕРЕЖА, ПРОГРАМУВАННЯ, ВІДЕО, КОНТРОЛЬ ДОСТУПУ

Об'єкт дослідження: комп'ютерна система Управління патрульної служби міста Дніпро.

Мета: обґрунтування структури комп'ютерної системи створення комп'ютерної системи Управління патрульної служби міста Дніпро з ідентифікацією людей, що розшукуються на вході до приміщень.

У вступі показано актуальність, мету, завдання, об'єкт, предмет, ідею дослідження.

У розділі «Стан питання та постановка завдання» розглянуті умови застосування комп'ютерної системи Управління патрульної служби, проаналізований типовий процес ідентифікації людей по фотозображенням, сформульовано мету і завдання роботи.

У теоретичному розділі описані методи отримання фотозображення людини, особливості ідентифікації осіб на зображенні, розроблені моделі зображень людей для перевірки роботи комп'ютерної системи.

У розділі «Синтез комп'ютерної системи» проаналізовані складові компоненти системи, вибрано обладнання та розроблена структурна схема системи.

У розділі «Розроблення програмного забезпечення» обґрунтовані технічні характеристики та структура і методи реалізації програмного забезпечення, з функцією ідентифікації людей, що розшукуються, по фото.

В експериментальному розділі поставлена задача експерименту і обґрунтована методика його проведення, проведено тестування на підібраних фотозображеннях груп людей і на реальному відео та перевірено функції розпізнавання обличчя і ідентифікації людей.

ЗМІСТ

Перелік умовних позначень, символів, скорочень і термінів	6
Вступ	7
1 Стан питання та постановка завдання	10
1.1 Огляд умов застосування комп'ютерної системи Управління патрульної служби міста Дніпро	10
1.2 Типовий процес ідентифікації людей по фотозображенням	14
1.3 Особливості вибору відеокамер для ідентифікації людей	15
1.4 Задача та мета роботи	21
1.5 Визначення напрямку вирішення поставленої задачі	21
2 Теоретичний розділ	22
2.1 Особливості отримання фотозображення людини в комп'ютерних системах	22
2.2 Методи знаходження обличчя людини на фотозображенні	26
2.3 Особливості ідентифікації осіб на зображенні	31
2.4 Розробка моделей зображень людей для перевірки комп'ютерної системи Управління патрульної служби	33
3 Синтез комп'ютерної системи Управління патрульної служби	40
3.1 Вимоги до системи в цілому	40
3.2 Вимоги до функцій, що виконуються системою	42
3.3 Вимоги до видів забезпечення	42
3.3.1 Вимоги до інформаційного забезпечення	42
3.3.2 Вимоги до лінгвістичного забезпечення	42
3.3.3 Вимоги до технічного забезпечення	42
3.4 Вибір обладнання для побудови комп'ютерної системи Управління патрульної служби	45
3.5 Модернізація структурної схеми комп'ютерної системи Управління патрульної служби	45

		5
4	Розроблення програмного забезпечення	52
	4.1 Призначення і сфера застосування програми	52
	4.2 Обґрунтування технічних характеристик програми	52
	4.2.1 Постановка завдання на розробку програми	52
	4.2.2 Структура і алгоритм функціонування програми	52
	4.2.3 Вибір складу програмних засобів	52
	4.3 Опис розробленої програми	53
	4.3.1 Загальні відомості	53
	4.3.2 Функціональне призначення	53
	4.3.2.1 Типи розв'язуваних задач	53
	4.3.2.2 Функціональні обмеження	53
	4.3.3 Опис логічної структури програми	53
	4.3.4 Використовувані технічні засоби	54
	4.3.5 Цикл роботи програми	54
	4.3.6 Вхідні та вихідні дані	54
5	Експериментальний розділ	56
	5.1 Постановка завдання експерименту і обґрунтування методики	56
	5.2 Проведення експерименту для тестування розробленої системи на підібраних фотозображеннях груп людей	56
	5.2.1 Перевірка роботи на підібраних зображеннях груп людей	56
	5.2.2 Перевірка роботи при різній відстані до камери	62
	5.3 Тестування розробленої комп'ютерної системи при отриманні реального відео	69
	5.3.1 Перевірка функції розпізнавання обличь	69
	5.3.2 Перевірка функції ідентифікації людей по зображенню обличь	71
	5.4 Оцінка результатів експерименту	76
	Висновки	77

	6
Перелік посилань	78
Додаток А Текст програми комп'ютерної системи Управління патрульної служби м. Дніпро	80

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, СКОРОЧЕНЬ І ТЕРМІНІВ

БД – база даних;

МВС – міністерство внутрішніх справ;

ПК – персональний комп'ютер;

ПЗ – програмне забезпечення;

ID – identity document;

OIS – Optical Image Stabilizer;

IBIS – In-Body Image Stabilization;

EIS – Electronic Image Stabilization;

AIIS – Artificial Intelligence Image Stabilization;

HDR – High Dynamic Range;

JPEG – Joint Photographic Experts Group;

MJPEG – Motion JPEG;

AVI – Audio Video Interleave;

MKV – Matroska Video;

WMV – Windows Media Video;

AVCHD – Advanced Video Coding High Definition;

FPS – frames per second;

SD (Standard Definition;

ED – Extra Low Dispersion;

DNN – Deep Neural Network;

CNN – Convolutional Neural Networks;

SVM – Support Vector Machine;

YOLO – You Only Look Once;

SSD – Single Shot MultiBox Detector;

MTCNN – Multi-task Cascaded Convolutional Networks;

ВСТУП

У сучасному світі діджиталізація та комп'ютерні системи є вирішальними для ефективності роботи адміністративних структур. Такі системи з корпоративними мережами стали незмінною складовою щоденної діяльності всіх організацій, зокрема і Управління патрульної служби м. Дніпро. Вони сприяють автоматизації процесів, підвищенню продуктивності, забезпеченню безпечного обміну інформацією та захисту даних від несанкціонованого доступу.

Для поліпшення управлінських процесів, підвищення якості обслуговування і захисту громадян необхідно модернізувати сучасну та надійну комп'ютерну систему Управління патрульної служби, додавши функцію ідентифікації розшукуваних осіб за фотографіями.

Необхідність ідентифікації розшукуваних осіб за фотографіями виникає в наступних випадках:

- в правоохоронних органах для розслідування злочинів, впізнавання підозрюваних та свідків, а також у справах пропажі людей;
- для безпеки в аеропортах, на прикордонних пропускних пунктах або на інших об'єктах, що потребують підвищеної безпеки, для контролю доступу;
- в соціальних мережах для автоматичного розпізнавання обличь для маркування друзів на фотографіях;
- в бізнесі можуть використовуватися технології для аналізу поведінки споживачів або для персоналізації реклами;
- в медичних дослідженнях для ідентифікації пацієнтів або у випадках, коли пацієнти потребують термінової медичної допомоги;
- в криміналістиці для порівняння збережених фотозображень з відбитками або іншими доказами;
- в культурі та мистецтві для документування та архівування культурних подій чи особистостей.

Мета і завдання дослідження: *Метою роботи є визначення структури комп'ютерної системи Управління патрульної служби міста Дніпро з опрацюванням функції ідентифікації людей, що розшукуються по фото.*

Для досягнення поставленої мети необхідно вирішити такі завдання:

- провести аналіз умов застосування комп'ютерної системи Управління патрульної служби м. Дніпро.
- виконати аналіз особливостей типового процесу ідентифікації людей по фотозображенням;
- вивчити особливості підготовки зображень для розпізнавання людей мовою Python;
- розробити структуру комп'ютерної системи Управління патрульної служби з ідентифікацією людей, які входять в Управління;
- розробити програми знаходження обличч людей на зображенні та їх порівняння;
- обґрунтувати застосування камер для відеоконтролю людей, які входять до приміщення Управління;
- провести експериментальні дослідження на підібраних фотозображеннях людей і на реальному відео та обґрунтувати параметри комп'ютерної системи Управління патрульної служби м. Дніпро.
- обґрунтувати параметри комп'ютерної системи контролю Управління патрульної поліції м. Дніпро.

Об'єкт дослідження: комп'ютерна система Управління патрульної поліції м. Дніпро.

Предмет дослідження: обробка зображень для ідентифікації осіб.

Методи дослідження: теорія нейромереж, теорія моделювання.

Ідея роботи: додаткове обладнання комп'ютерної системи підсистемою відеоконтролю з ідентифікацією людей, які входять в Управління патрульної служби м. Дніпро.

Наукові положення:

1. Встановлено, що для виявлення обличь на зображеннях людей з достовірністю не менше 0.85 на відеокамеру необхідно подавати сигнал початку роботи при наближенні людини на відстань не більше 2.0 м.

Наукові результати:

1. Обґрунтовано використання бібліотеки Face Recognition Library мови Python з достовірністю понад 0.9865 при ідентифікації людей по зображенню обличчю на вході в приміщення Управління патрульної служби.

Обґрунтованість і достовірність наукових положень, висновків і рекомендацій підтверджуються результатами експериментальних досліджень.

Практичне значення отриманих результатів полягає в розробці програмного забезпечення, що дозволяє ідентифікувати людей по зображенню обличчя і збирати ці зображення в окрему базу для подальшого застосування.

1 СТАН ПИТАННЯ ТА ПОСТАНОВКА ЗАВДАННЯ

1.1 Огляд умов застосування комп'ютерної системи Управління патрульної служби міста Дніпро

В структуру Міністерства внутрішніх справ (МВС) України входять наступні підрозділи:

- Національна Поліція України;
- Державна прикордонна служба України;
- Державна служба України з надзвичайних ситуацій;
- Державна міграційна служба України;
- Національна гвардія України
- Експертна служба МВС
- Сервісні центри МВС [2].

В систему Національної поліції України включені центральний орган управління поліцією та територіальні органи поліції. Національна поліція підпорядкована Міністерству внутрішніх справ України та очолюється Генералом поліції 3-го рангу.

Наказом Національної поліції України від 06.11.2015 № 1 затверджені структури центрального органу управління поліцією та територіальних органів поліції.

Територіальні органи поліції України включають кримінальну поліцію, патрульну поліцію, органи досудового розслідування, поліцію охорони, спеціальну поліцію та поліцію особливого призначення.

Поліція призначена для забезпечення особистої безпеки громадян, захисті їх власності, прав та свобод від злочинних посягань, а також для підтримки правопорядку і запобіганню злочинності.

Управління Патрульної поліції України у м. Дніпро розташовано на площі Троїцька, 2а, а Call-Center – в іншій. Він працює як складова Національної Поліції України.

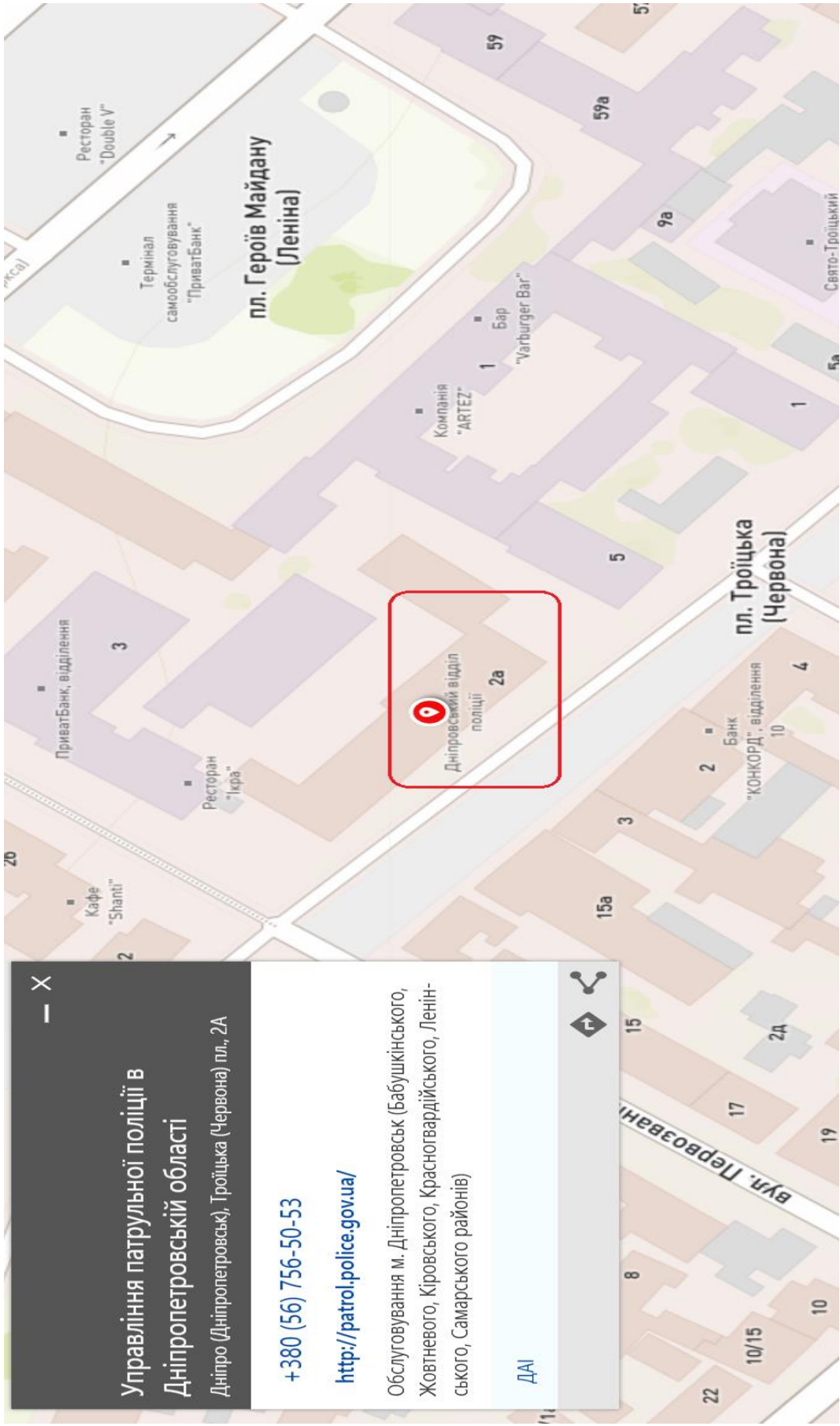


Рисунок 1.1 – Розташування Управління управління патрульної поліції в м. Дніпро

Структура організації Управління управління патрульної поліції м. Дніпро має наступну організаційну структуру (рисунок 1.2) і складається з: керівництва, оперативних груп, чергової групи, бухгалтерії, відділів превенції (1), моніторингу (2) та кадрового забезпечення (3).

Керівництво організовує виконання задач по Управління управління патрульної поліції з підключенням до виконання інших структурних підрозділів Фінансове забезпечення діяльності Управління та їх співробітників забезпечує бухгалтерія.

Сектор превенції поліції займається профілактичною діяльністю, яка спрямована на запобігання злочинності через діалог та співпрацю з громадою. Він виконує контроль за дотриманням законодавства, зокрема у сфері зберігання та використання зброї. Основна мета сектора – підвищення довіри до поліції та поліпшення безпеки в суспільстві.

Прийом від населення заяв та викликів здійснюється завдяки Call-Center. Усі приміщення Управління оснащені системою ідентифікації за допомогою ID-карток. Для запобігання несанкціонованому доступу та підвищення загального рівня безпеки Управління буде додатково обладнано системою з функцією ідентифікації людей, що розшукуються по фотозображенню.

Окрім контролю доступу, система також забезпечуватиме облік робочого часу, що сприятиме підвищенню трудової дисципліни співробітників.

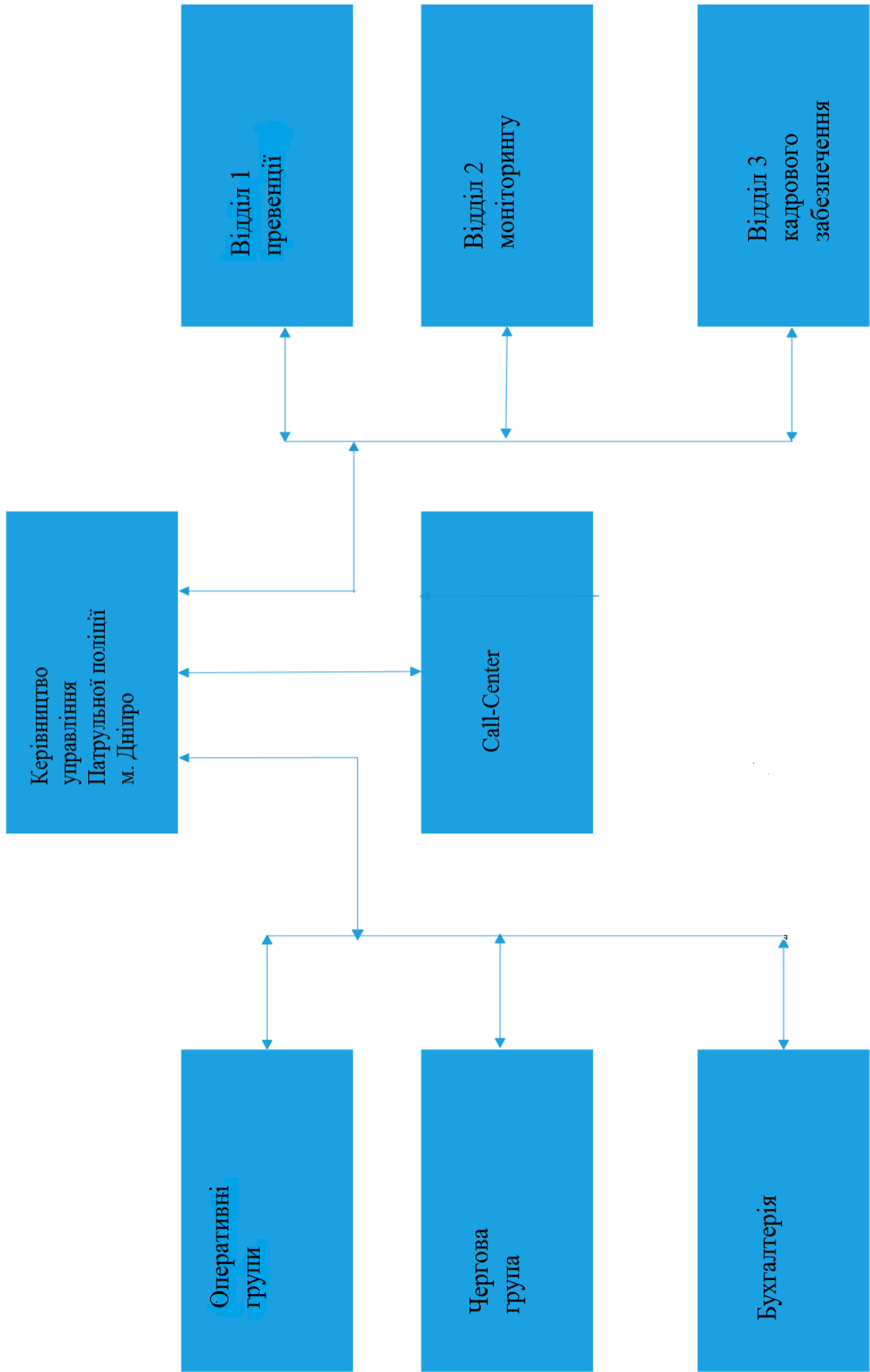


Рисунок 1.2 – Структура організації Управління управління патрульної поліції м. Дніпро

1.2 Типовий процес ідентифікації людей по фотозображенням

Необхідність ідентифікації людей по фотозображенню виникає в різних сферах і застосовується:

- правоохоронними органами;
- в аеропортах, на прикордонних пропускних пунктах або на інших об'єктах, що потребують підвищеної безпеки;
- в системах контролю доступу;
- в медичних дослідженнях;
- для дистанційної верифікації клієнтів фінансових компаній та банків, онлайн страхування.

При цьому зображенням людей може завантажуватися з веб-камер, відеокамер, систем відеоспостереження та інших фото/відео джерел.

Процес ідентифікації людей включає наступні кроки:

- отримання фотозображення людини. Це може бути статичні зображення так і відео;
- якщо це було відео, то потрібно виділити його окремі фрейми для обробки;
- попередня обробка з метою покращення якості зображення перед аналізом. Це може бути корекція освітлення, вирівнювання, усунення шумів тощо;
- виділення обличчя людини на зображенні (якщо воно присутнє);
- ідентифікація по зображенню обличчя людини, тобто порівняння з базою даних зображень, наприклад, людей, які розшукуються або співробітників Управління управління патрульної поліції м. Дніпро. Під час порівняння виділяються ключові метрики зображення обличчя людини з метриками обличчя людей, які зберігаються в базі даних;
- по результату ідентифікації приймається відповідне рішення, а дані зберігаються;
- на усіх кроках необхідно дотримуватися етичних норм і законодавства України стосовно обробки та зберігання особистих даних.

1.3 Особливості вибору відеокамер для ідентифікації людей

Під час вибору камери важливо враховувати, для яких завдань вона буде використовуватися. Тому для завдань ідентифікації людей важливо звертати на наступні характеристики відеокамер, які поділені на загальні характеристики відеокамери і характеристики об'єктиву.

До загальних характеристик відеокамери відносять:

- оптична стабілізація;
- автофокус;
- алгоритми обробки зображень;
- формат зйомки відео;
- швидкість зйомки відео.

Оптична стабілізація зображення використовується для зменшення ефектів тремтіння камери та покращення якості зображення. Різні виробники можуть мати свої специфічні технології та варіації цієї характеристики. Розрізняють систему стабілізації на об'єктиві (OIS – Optical Image Stabilizer), на сенсорі (IBIS – In-Body Image Stabilization), гібридну, електронну (EIS – Electronic Image Stabilization) та з використанням штучного інтелекту (AIIS – Artificial Intelligence Image Stabilization).

OIS передбачає використання рухомих елементів всередині самого об'єктива і реалізується з рахунок вільного переміщення лінз, які коригують зображення в момент тремтіння.

IBIS стабілізує зображення переміщуючи сам сенсор зображення, тобто навіть якщо камера рухається, зображення залишається стабільним.

Гібридні системи поєднують елементи OIS і IBIS для досягнення максимальної стабільності і використання переваги обох технологій.

EIS обробляє зображення цифровим способом, видаляючи зображення з країв кадру, щоб компенсувати рухи камери.

AIS можна відносити і до EIS, але вона по-особливому ефективна, коли справа йде про фотографії з тривалою витримкою та зйомку при недостатньому

освітленні. Завдяки цим технологіям забезпечується висока деталізація та різкість зображень.

Автофокус – це технологія, яка дозволяє автоматично налаштувати фокус камери, наприклад, на людину без потреби вручну змінювати фокусування. Це значно спрощує процес зйомки, особливо в умовах зрушення або при швидкому русі людини. Автофокус став невід'ємною частиною сучасних камер, дозволяючи автоматично проводити зйомки без участі оператора.

Розрізняють контрастний автофокус, фазовий, гібридний, автофокус по очах та лазерний.

Контрастний автофокус забезпечує фокусування за рахунок аналізу контрастності в зображенні. Камера вибирає найкращі налаштування, щоб отримати максимальний контраст.

Фазовий автофокус використовує датчики, які розміщуються в камері для паралельного порівняння двох світлових променів, і це дає змогу швидше і точніше фокусуватися, особливо в умовах недостатнього освітлення.

Гібридний автофокус поєднує обидва методи (контрастний і фазовий) і це забезпечує кращу продуктивність у різних умовах зйомки.

Деякі сучасні камери можуть автоматично фокусуватися по очам людини, що особливо корисно при ідентифікації людей.

При лазерному автофокусі застосовується лазерний промінь для визначення відстані до об'єкта. Цей метод частіше можна побачити в смартфонах.

У відеокамерах використовується дуже багато **алгоритмів обробки зображень**, які допомагають отримувати якісні зображення, що відповідають вимогам користувачів та умовам зйомки.

Розрізняють алгоритми шумозаглушення, корекції кольору, контрастності та яскравості, детекції країв, сегментації, фільтрації, геометричних трансформацій, адаптивної обробки яскравості HDR (High Dynamic Range) та стиснення зображень.

Алгоритми шумозаглушення зменшують цифровий шум у зображеннях, особливо при зйомці в умовах низького освітлення.

Алгоритми корекції кольору допомагають відновити природні кольори, компенсуючи освітлення або умови зйомки.

Алгоритми Регулювання контрастності та яскравості дозволяють покращити видимість деталей у зображенні.

Алгоритми фільтрації (наприклад, гаусовий фільтру) дозволяє покращити якість зображень, зменшити шуми або виділення певних характеристик.

Алгоритми детекції країв (наприклад Sobel або Canny) використовуються для виявлення країв об'єктів у зображеннях, що допомагає у подальшій обробці.

Алгоритми сегментації дозволяють Відокремлювати різні об'єкти або регіони зображення для подальшого аналізу.

Алгоритми геометричних трансформацій дозволяють змінювати масштаб, обертати або змінювати перспективу зображення.

Алгоритми адаптивної обробки яскравості HDR підвищують якість зображення шляхом адаптивного налаштування яскравості в певних ділянках зображення.

Алгоритми стиснення зображень, такі як JPEG зменшують розміри файлів без значної втрати якості.

Формати і швидкості зйомки відео залежать від типу відеокамери та призначення відео. Відомі наступні **формати зйомки відео**:

- MP4 (MPEG-4 Part 14), один із найпопулярніших форматів, який забезпечує хороший баланс між якістю та розміром файлу і використовується для стрімінгу та зберігання.

- AVI (Audio Video Interleave), формат від Microsoft, який підтримує різні кодеки, завантажує великі файли, але забезпечує високу якість.

- MOV, формат Apple, який часто використовується в продуктах Mac. Забезпечує високу якість і підтримує багато різних кодеків.

- MKV (Matroska Video), формат, який підтримує кілька аудіо- та субтитрових доріжок і використовується для зберігання відео високої якості;

- WMV (Windows Media Video), формат Microsoft, який часто використовується для потокового відео і має високий ступінь стиснення;
- AVCHD (Advanced Video Coding High Definition), формат, який часто використовується у камерах для запису відео в HD та 4K;
- HEVC (H.265), найновіший формат, який забезпечує високу якість зображення при зменшеному обсязі файлу в порівнянні з H.264.

Швидкість зйомки визначається в кадрах на секунду (FPS – frames per second). Відомі наступні **формати швидкості зйомки відео**:

- 24 FPS, стандартна швидкість для кінофільмів, яка створює "кінематографічний" вигляд.
- 30 FPS, часто використовується в телевізійних передачах та відео для інтернету.
- 60 FPS, використовується для зйомок, які потребують більшої плавності, наприклад, в іграх або спортивних подіях.
- 120 FPS і вище, використовується для уповільненої зйомки для спортивної та природної зйомки.
- 240 FPS, дозволяє створювати дуже плавні ефекти уповільненої зйомки.

До характеристик об'єктиву відносять:

- роздільну здатність;
- якість оптики;
- розмір сенсору.

Роздільну здатність об'єктиву може варіюватися і зазвичай ділиться на кілька основних категорій:

а) SD (Standard Definition);

1) 480p (640x480), найменша якісна роздільна здатність. Використовується у старих пристроях і для базових задач відеоспостереження.

б) HD (High Definition);

1) 720p (1280x720), покращена якість у порівнянні з SD, підходить для більшості завдань відеоспостереження;

2) 1080p (Full HD, 1920x1080), висока якість зображення; підходить для більшості сучасних застосунків, включаючи стрімінг та відеоконференції.

в) 2K;

1) 1440p (2560x1440), надає ще більше деталей, ніж 1080p, але вимагає більшого обсягу пам'яті і пропускну здатності.

г) 4K (Ultra HD);

1) 2160p (3840x2160), дуже висока якість. Ідеально для професійних завдань, таких як відеоспостереження, стрімінг або кінематографія.

д) 8K;

1) 4320p (7680x4320), найвища доступна роздільна здатність на сьогодні, застосовується у спеціалізованих сферах та потребує значних ресурсів для зберігання й обробки.

Якість оптики об'єктива значною мірою впливає на загальну якість отриманого зображення і важлива для досягнення високоякісних фотографій. Якість оптики об'єктива визначають оптичні характеристики, покриття лінз, конструкція об'єктива та вид діафрагми.

Відомі наступні оптичні характеристики:

– різкість (sharpness), високоякісні об'єктиви забезпечують більшу різкість як у центрі, так і по краях зображення. Це можна оцінити за допомогою тестування на різницю деталей при різних діафрагмах;

– хроматичні аберації, оптичний дефект, що призводить до кольорових розсіяних країв на контрастних крайках. Високоякісні об'єктиви мають поліпшену оптику для мінімізації цього ефекту;

– викривлення (distortion), високоякісні об'єктиви повинні мати мінімальний рівень викривлення, яке може спотворити форму об'єктів (наприклад, "риб'яче око" ефект).

Стосовно **покриття лінз**. Відомі, так звані, антиблікові покриття лінз. Таке покриття на лінзах може суттєво зменшити відблиски і підвищити пропускну здатність лінз, що, в свою чергу, покращує яскравість і контраст зображення.

Конструкції об'єктиву характеризуються:

– кількістю елементів і груп, більша кількість елементів може покращити корекцію аберацій, але кожен елемент також споживає частину світла. Баланс між кількістю елементів і якістю скла є важливим;

– формою та матеріалом лінз, використання асферичних лінз і спеціальних видів скла (такі як ED – Extra Low Dispersion) дозволяє поліпшити якість зображення.

Відомі наступні види діафрагм з:

– максимальною апертурою (f-число), великі значення (наприклад, $f/1.4$ або $f/2.8$) дозволяють камері пропускати більше світла і забезпечують кращу якість зображення в умовах низького освітлення;

– розмиттям фону, якість розмиття фону залежить від конструкції діафрагми (круглі секції створюють більше приємний ефект).

1.4 Задача та мета роботи

Основним завданням та метою кваліфікаційної роботи є розробка комп'ютерної системи Управління патрульної служби міста Дніпро.

Для додаткового забезпечення безпеки Управління патрульної служби в цілому, а також для уникнення несанкційованого входу на територію, планується обладнати ще й системою відео контролю.

Метою роботи є розробка комп'ютерної системи Управління патрульної служби міста Дніпро з функцією ідентифікації людей, що розшукуються, на вході на територію.

1.5 Визначення напрямку вирішення поставленої задачі

Встановлення IP-відеокамер на вході в Управління патрульної служби і підключення їх до комп'ютерної системи дозволить реалізувати функцію додаткового відеоконтролю входу на територію осіб шляхом розпізнавання обличчя із застосуванням нейронної мережі OpenCV за допомогою модуля dnn OpenCV в області Deep Learning. При цьому модель створюється в фреймворці глибокого навчання Caffe.

2 ТЕОРЕТИЧНА ЧАСТИНА

2.1 Особливості отримання фотозображення людини в комп'ютерних системах

Для ідентифікації людей по фотозображенню, яке потрібне в різних сферах, необхідно спочатку отримати саме зображення людини. Це може бути статичне зображення так і відео.

Для отримання статичного зображення людини відеокамера повинна мати функцію збереження окремих кадрів.

Професійні відеокамери, а також деякі побутові моделі, зазвичай мають можливість збереження покадрових знімків.

Спеціалізовані камери для відеонагляду також можуть мати цю функцію, але не завжди. Зазвичай це залежить від програмного забезпечення, за допомогою якого вони працюють.

Тому розглянемо більш складний варіант по отриманню окремих кадрів з відео яке надсилається з камери відеонагляду.

Для виділення окремих кадрів з потокового відео від камери можна використовувати різні мови програмування та бібліотеки.

Це може бути мова Python з бібліотекою OpenCV та ffmpeg. Потужна бібліотека для комп'ютерного зору OpenCV дозволяє легко захоплювати відеопотоки, зберігати кадри та обробляти їх. Інструмент командного рядку ffmpeg з бібліотекою subprocess можна застосовувати через Python для обробки відео.

Мова C++ може як і Python застосовувати потужну бібліотеку для комп'ютерного зору OpenCV.

Мова Java з бібліотекою Juggler теж застосовується для обробки media.

Мова JavaScript з бібліотекою WebRTC дозволяє працювати з потоками відео в реальному часі.

Розглянемо приклад програми мовою Python, яка використовує бібліотеку OpenCV для захоплення відеопотоку з камери і запису окремих кадрів.

Середовище для роботи програми включатиме каталог video-img із зображеннями, які отримані з відеокамери.

В програмі виконується захоплення відеопотоку у змінну cap і створюється цикл працюючий продовж виводу відео while (cap.isOpened()).

В циклі визначаються параметри (fps, divider) зчитування кадрів і запису файлів із зображеннями та номера фрейму (frame_id). Коли залишок від поділу номера фрейму (frame_id) на дільник (divider) дорівнює нулю черговий кадр записується у файл як зображення. Для захоплення відеопотоку з камери video_path вказує на підключену камеру, тобто, наприклад video_path = 0. Для перевірки роботи програми можливо підключити раніше записаний файл з відео, в цьому разі, наприклад, video_path = video_example1.mp4

Example 2.1

```
def take_img_from_video(video_path):
    cap = cv.VideoCapture(video_path)
    wave = Path(video_path)
    count = 0

# Перевірка наявності каталогу frame_img
if not os.path.exists("frame_img"):
    os.mkdir("frame_img")

# Організація циклу
while (cap.isOpened()):
    ret, frame = cap.read()
    fps = cap.get(cv.CAP_PROP_FPS)
    divider = int(fps * 5)
```



```

if ret:
    frame_id = int(round(cap.get(1)))
    cv.imshow("Frame", frame)
    k = cv.waitKey(2)

# Запис чергового кадру
if int(frame_id % divider) == 0:
    cv.imwrite(f"frame_img/{count}_{wave.name}.jpg", frame)
    print(f"Take a frame_img {count}_{wave.name}")
    count += 1

```

При застосуванні програми в подальшому знадобиться контрольний запис і надсилання зображення для обробки на сервер при спрацюванні датчика. Виконаємо імітацію цієї події для налагодження програми натисненням на клавішу «d». Для цього організуємо спеціальну перевірку.

Для налагодження програми і виходу з нескінченного циклу зробимо можливість завершення роботи програми при натисканні клавіші «q» та вивід відповідних повідомлень на дисплей.

Example 2.2

```

# Перевірка імітації спрацювання датчика
if k == ord("d"):
    cv.imwrite(f"frame_img/{count}_br_{wave.name}.jpg", frame)
    print(f"Send a frame_img br_{count}_{wave.name}")
    count += 1

# Перевірка натискання клавіші «q»
if k == ord("q"):
    print(f"Q pressed, end of the app")
    break

```

```
else:
```

```
    print("It's end!")
```

```
    break
```

```
# Звільнення пам'яті від змінною cap
```

```
#Закриття всіх відкритих вікон Windows
```

```
cap.release()
```

```
cv.destroyAllWindows()
```

2.2 Методи знаходження обличчя людини на фотозображенні

Відомо кілька методів і технологій для виявлення обличчя на фотографіях. Розрізняють Гаусасову модель та її варіанти, методи на основі машинного навчання, глибокі нейронні мережі (DNN, Deep Neural Network) та детектори ключових точок.

Метод Гаусасової моделі (Haar Cascades) заснований на характеристиках Хаара (Haar-like features), який використовує каскади класифікаторів для виявлення обличчя. Він швидкий і має можливість працювати в реальному часі, але може бути менш точним у складних умовах.

Метод на основі машинного навчання SVM (Support Vector Machine) використовує вектори ознак для класифікації зображень. Цей метод ще не так популярний як нейронні мережі, але все ще застосовується в деяких системах.

Ансамблевий метод машинного навчання Random Forests об'єднує результати кількох дерев рішень для покращення точності виявлення.

Нейронні мережі Convolutional Neural Networks (CNN) використовують багатоваріантні нейронні мережі для автоматичного видобування ознак і зазвичай забезпечують найкращі результати в задачах вимірювання обличчя.

Нейронні моделі на зразок YOLO (You Only Look Once) та SSD (Single Shot MultiBox Detector), які використовують для об'єктного розпізнавання, здатні швидко знаходити обличчя в зображеннях.

MTCNN (Multi-task Cascaded Convolutional Networks) призначені для одночасного виявлення обличчя і визначення ключових точок на обличчі.

Детектори ключових точок використовують виявлення специфічних точок на обличчі (наприклад, очі, ніс, рот) для визначення наявності обличчя. Це може бути реалізовано за допомогою алгоритмів, таких як Dlib.

Бібліотека OpenCV містить готові реалізації виявлення обличь за допомогою Haar Cascades і DNN, які добре перевірені і знаходяться в основному репозиторію.

При завантаженні DNN відбувається конвертація моделей у внутрішнє уявлення. Підтримуються всі основні шари: починаючи від базових (Convolution і Fully connected) і до більш спеціалізованих – всього понад 30.

Розпізнаються обличчя за наступним алгоритмом.

Вхідні зображення беруться з каталогу `frame_img`, куди вони були записані на попередньому кроці. Далі на зображеннях знаходять обличчя (якщо вони є) і помічаються їх рамкою, а файли розміщуються в каталог `img_updated`.

Потім ці зображення обличь вилучаються і поміщаються в окремий каталог `faces`.

Моделі `prototxt` та `caffemodel` розміщуємо в каталог `model_data`. Файл `prototxt` містить текстовий опис мережі, а `caffemodel` – вага. Визначаємо шляхи та завантажуюмо модель.

Example 2.3

```
import os
import cv2 as cv
import numpy as np

base_dir = os.path.dirname(__file__)

# base_dir
prototxt_path = os.path.join(base_dir + '/model_data/deploy.prototxt')

# prototxt_path
caffemodel_path = os.path.join(base_dir + '/model_data/weights.caffemodel')

# caffemodel_path
```

```

model = cv.dnn.readNetFromCaffe(prototxt_path, caffemodel_path)
#Перевіряємо наявність папок img_update та faces
#Якщо папок немає, створюємо їх

if not os.path.exists('img_updated'):
    print("New directory img_updated created")
    os.makedirs('img_updated')

if not os.path.exists('faces'):
    print("New directory faces created")
    os.makedirs('faces')

```

Далі зчитуємо послідовно всі зображення у папці frame_img і перевіряємо чи всі файли мають розширення png та jpg.

За допомогою cv.imread зчитуємо зображення та створюємо blob за допомогою cv.dnn.blobFromImage, відправляємо blob модель і отримуємо виявлення за допомогою model.forward().

Example 2.4

```

# Переглядаємо всі зображення та визначаємо обличчя
count = 0
for file in os.listdir(base_dir + '/frame_img'):
    file_name, file_extension = os.path.splitext(file)
    if (file_extension in ['.png', '.jpg']):
        print("Image path: {}".format(base_dir + '/frame_img /' + file))

        image = cv.imread(base_dir + '/ frame_img /' + file)

        (h, w) = image.shape[:2]
        blob = cv.dnn.blobFromImage(cv.resize(image, (300, 300)), 1.0, (300, 300),
(104.0, 177.0, 123.0))

```

```
model.setInput(blob)
detections = model.forward()
```

Далі проведемо невелику передобробку і отримуємо зображення на виході.

Побудуємо рамки навколо обличь і збережемо зображення в каталозі `img_updated`, а зображення з отриманими обличчями зберігаємо в каталозі `faces`. Далі отримуємо значення достовірності і якщо значення більше ніж на 50%, то отримане зображення є обличчям і навколо нього рисуємо прямокутник.

Командою `cv.imwrite`, зберігаємо зображення в папку `imgs_updated_` з тим самим ім'ям файлу.

Example 2.5

```
# Рисуємо прямокутник навколо обличчя
for i in range(0, detections.shape[2]):
    box = detectionsa[0, 0, 1, 3:7] * np.array([w, h, w, h])
    (startX, startY, endX, endY) = box.astype('int')
    Confidence = detections[0, 0, i, 2]
# If confidence > 0.5, show box around face
    if (confidence > 0.5):
        cv.rectangle(image, (startX, startY), (endX, endY), (255, 0, 255), 10)
        cv.imwrite(work_dir + '/img_updated/' + file, image)
        print("Image " + 'up_' + file + ' converted successfully')
# Identify each face
for i in range(0, detections.shape[2]):
    box = detections[0, 0, i, 3:7] * np.array([w, h, w, h])
    (startX, startY, endX, endY) = box.astype("int")
    confidence = detections[0, 0, i, 2]
# If confidence > 0.5, save it as a separate file
    if (confidence > 0.5):
        count += 1
```

```
frame = image[startY:endY, startX:endX]
cv.imwrite(work_dir + '/faces/' + str(i) + '_' + file, frame)
print(file, 'confidence',str(i), '=', confidence, 'w,i,' =',(endX - startX))
```

Нове зображення зберігається в каталозі faces. Імена файлів надаються таким чином: якщо ім'я вихідного зображення було sampleImage.png, ім'я файлу після обробки буде 0_sampleImage.png. З кожною особою збільшуємо лічильник і після повного виконання виводимо його на консоль.

Таким чином можна сформувати набір зображень обличч людей, які розшуковуються або перелік співробітників Управління патрульної поліції з яким можливо виконувати порівняння і контролювати доступ в приміщення.

2.3 Особливості ідентифікації осіб на зображенні

Сформовану баз зображень обличч в каталозі faces можна застосовувати для порівняння та контролю доступу в приміщення Управління патрульної служби міста Дніпро.

При отриманні нового зображення особи, яка відвідує Управління патрульної служби, необхідно виконати ідентифікацію, тобто порівняти з наявними в базі зображеннями обличч людей, що розшукуються.

Розробимо відповідну програму для порівняння двох зображень з метою ідентифікації людей, що розшукуються, або визначення співробітників Управління для вільного допуску у приміщення.

В програмі застосуємо наступні змінні: `img1_path` – шлях до зображення обличчя ідентифікуємої людини, `img2_path` – шлях до зображення обличчя розшукуємих людей, `img3_path` – шлях до зображень співробітників Управління патрульної служби.

Для порівняння зображень застосуємо бібліотеку Face Recognition Library мови Python, робота якої базується на нейромережі dlib.

Для ідентифікації особи по координатам характерних точок та їх співвідношенням складається карта обличчя з 128 параметрами.

Example 2.6

```
def compare_faces(img1_path, img2_path, img2_path):
    img1 = face_recognition.load_image_file(img1_path)
    img1_encodings = face_recognition.face_encodings(img1)[0]
    print(img1_encodings)

    img2 = face_recognition.load_image_file(img2_path)
    img2_encodings = face_recognition.face_encodings(img2)[0]
    print(img2_encodings)

#Порівняння img1 з img2
    result = face_recognition.compare_faces([img1_encodings],
    img2_encodings)
```



```
print(result)
```

Якщо це розшукувана людина, то передається тривожне повідомлення в чергову групу, якщо ні – перевірка продовжується і виконується порівняння з базою обличь Управління патрульної служби.

Example 2.6

```
if result[0]:
    print("Attention! This person is wanted.")
else:
```

```
    img3 = face_recognition.load_image_file(img3_path)
    img3_encodings = face_recognition.face_encodings(img3)[0]
    print(img3_encodings)
```

#Порівняння img1 з img3

```
    result = face_recognition.compare_faces([img1_encodings],
img3_encodings)
    print(result)
    if result[0]:
        print("This person is an employee of the Patrol Service Department!")
    else:
        print("This person is not an employee of the Patrol Service Department!")
```

Якщо це співробітник Управління патрульної служби, то запрошується на вхід, якщо ні – пропонується звернутися до чергового.

2.4 Розробка моделей зображень людей для перевірки комп'ютерної системи Управління патрульної поліції

Розробимо перелік моделей зображень людей для перевірки комп'ютерної системи Управління патрульної поліції.

Ці моделі розіб'ємо по групам. Перша група призначена для початкової перевірки роботи системи із статичними фотозображеннями. В якості похідних візьмемо фотографії викладачів факультету інформаційних технологій НТУ «Дніпровська політехніка» з сайту <https://www.nmu.org.ua/ua/> (рисунок 2.1).

Наступна група зображень призначена для перевірки роботи системи із зображенням людей обличчя яких знаходиться в різних положеннях, затінено сторонніми предметами і в різному емоціональному стані (рисунок 2.2).

В попередньому розділі вказувалося, що на результати впливає відстань, різний рівень освітлення. Наступна групи зображень призначена для перевірки роботи системи з такими параметрами, де є чітко фіксована відстань (рисунок 2.3) і з хаотичним набором цих параметрів (рисунок 2.4).

Крім того для перевірки роботи підготовлена модель відеозапису переміщення людини, кадр з якого показано на рисунку 2.5.

Характеристики цих зображень приведені в таблиці 2.1.

Таблиці 2.1

Характеристика підібраних моделей зображень

№ групи	Рисунок	Фрагмент	Формат файлу	Розмір, пкс	Роздільна здатність, точок на дюйм	Глибина кольору
1	2.1	1-1	JPG	900 x 650	144	24
1	2.1	1-2	JPG	900 x 650	144	24
2	2.2	2	JPG	650 x 700	144	24
3	2.3 а)	3-1	JPG	500 x 650	144	24
3	2.3 б)	3-2	JPG	500 x 650	144	24
3	2.3 в)	3-3	JPG	500 x 650	144	24
3	2.3 г)	3-4	JPG	500 x 650	144	24

Продовження табл. 2.1

№ групи	Рисунок	Фрагмент	Формат файлу	Розмір, пкс	Роздільна здатність, точок на дюйм	Глибина кольору
3	2.4 а)	3-5	JPG	500 x 650	144	24
3	2.4 б)	3-6	JPG	500 x 650	144	24
3	2.4 в)	3-7	JPG	500 x 650	144	24
3	2.4 г)	3-8	JPG	500 x 650	144	24
4	2.5	4	JPG	2050 x 1350	144	24
5	2.6	5	MOV	720 x 1280	–	–

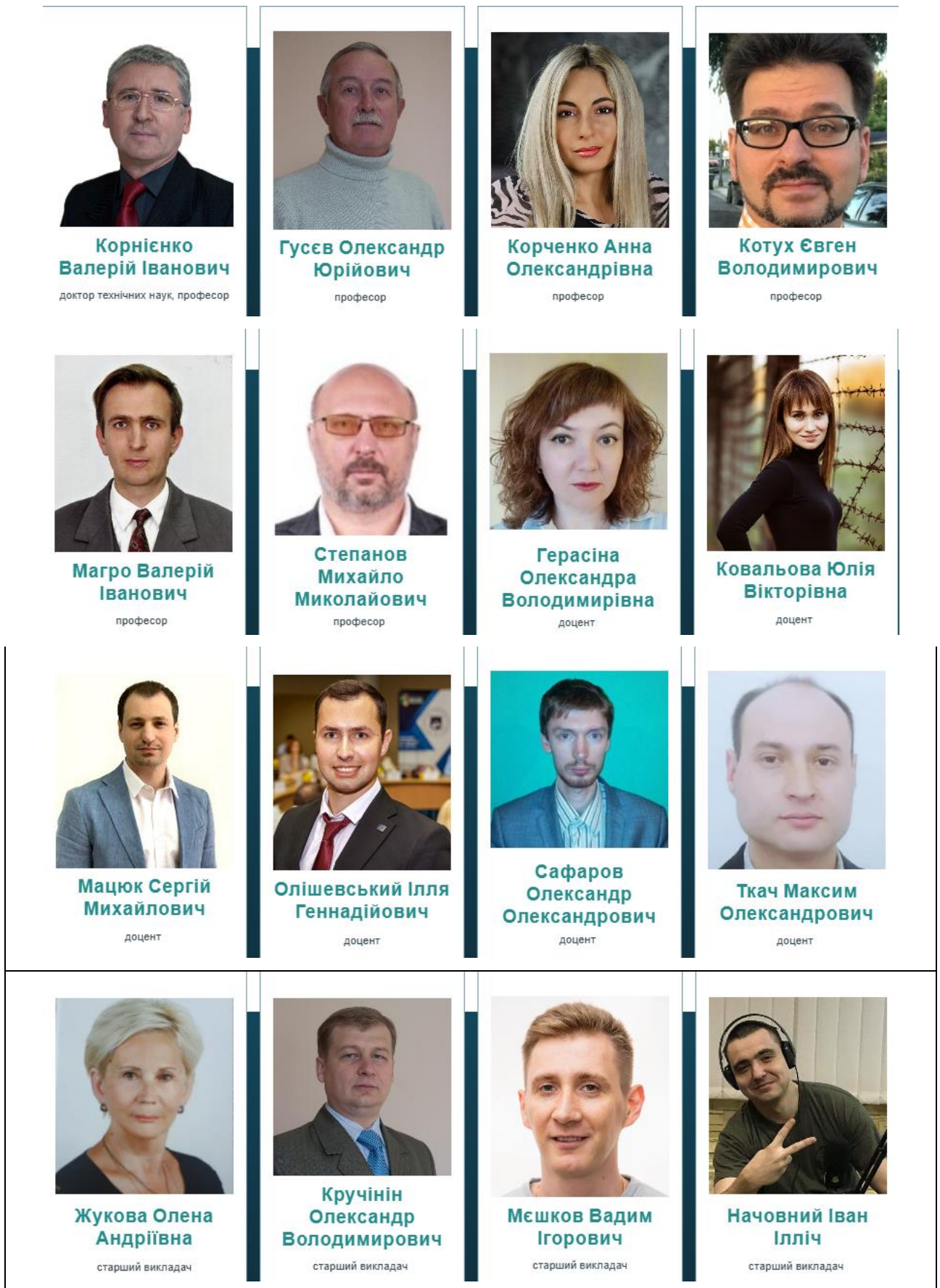
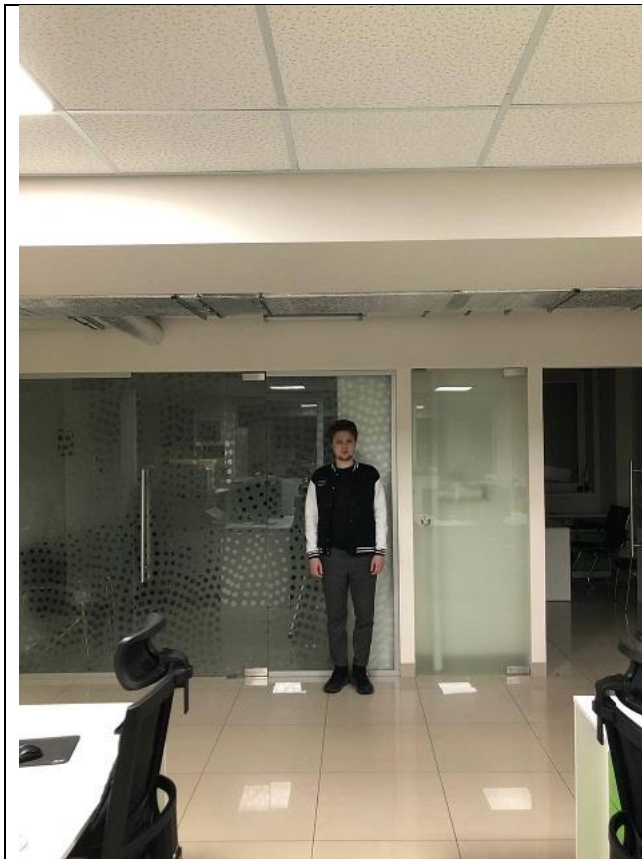


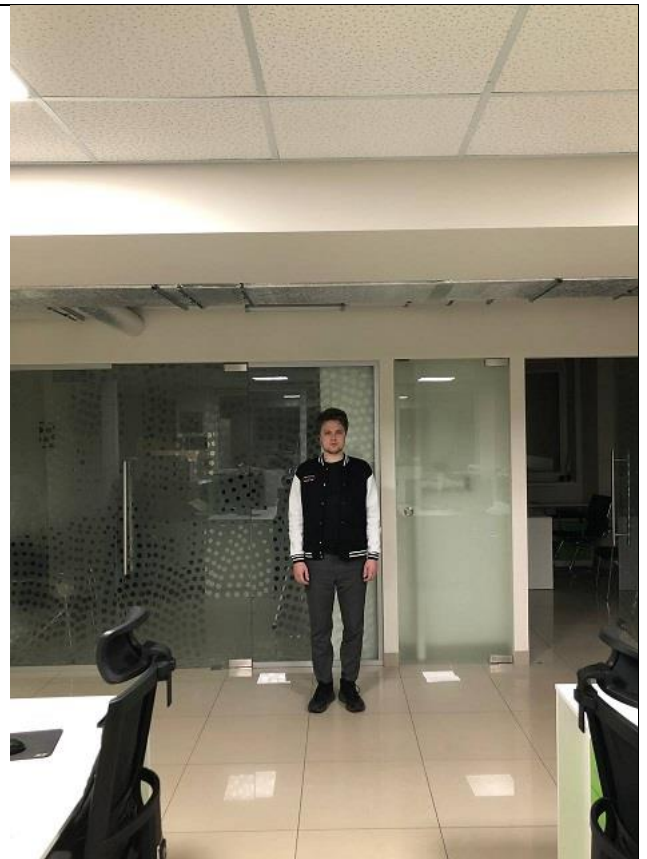
Рисунок 2.1 – Група №1 із зображенням для перевірки роботи системи



Рисунок 2.2 – Група №2 із зображеннями для перевірки роботи системи



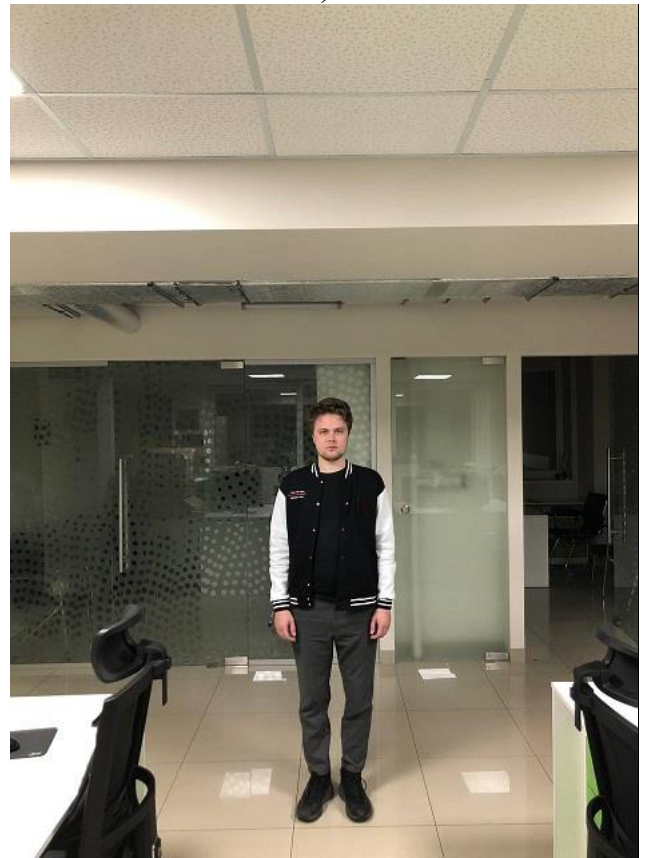
а)



б)



в)



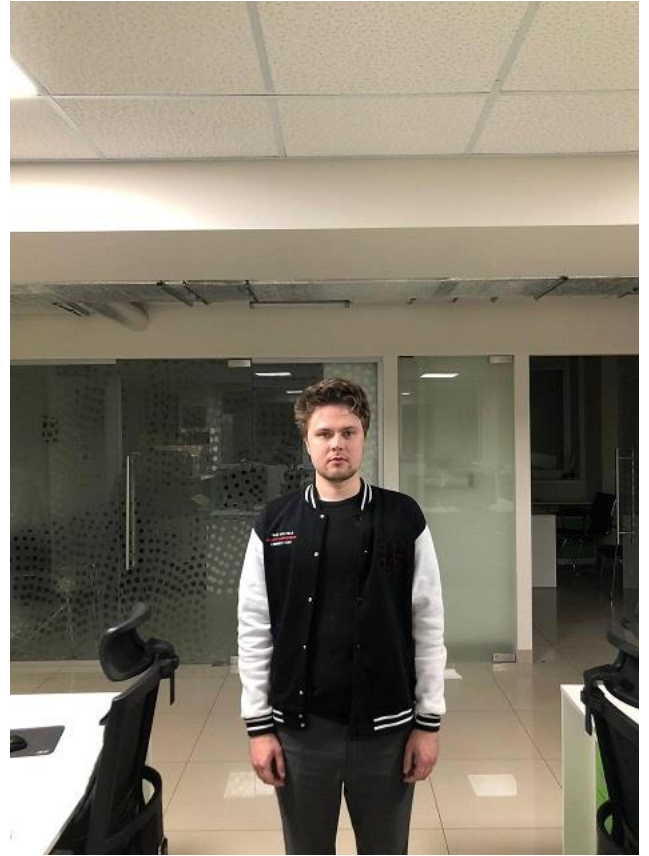
г)

Рисунок 2.3 – Група №3 із зображеннями на відстані 2.5 – 4.0 м,

а) 4.0 м; б) 3.5 м; в) 3.0 м; г) 2.5 м



а)



б)



в)



г)

Рисунок 2.4 – Група №3 із зображеннями на відстані 0.5 – 2.0 м,

а) 2.0 м; б) 1.5 м; в) 1.0 м; г) 0.5 м



Рисунок 2.5 – Група №4 із зображеннями для перевірки роботи системи



Рисунок 2.6 – Фрагменти відео-файлу, група №5 для перевірки роботи системи

3 СИНТЕЗ КОМП'ЮТЕРНОЇ СИСТЕМИ УПРАВЛІННЯ ПАТРУЛЬНОЇ СЛУЖБИ

3.1 Вимоги до системи в цілому

Комп'ютерна система Управління патрульної служби міста Дніпро повинна:

- мати модулі систем «Керівництво управління патрульної служби», «Оперативні групи», «Чергова група», Call-Center, «Бухгалтерії», та відділів Превенції 1, моніторингу 2 та кадрового забезпечення 3 та підсистему відеоконтролю;
- об'єднання складових здійснювати за рахунок корпоративної мережі, з кабельною системою, комутаторами, маршрутизаторами та серверами (активне обладнання відповідно до стандарту ISO / IEC 11801 Class D, категорія 5E) на 90 робочих місць;
- відповідати компонентам за стандартом ISO 9001 (ГОСТ 40.9001-88) при проектуванні корпоративної мережі;
- бути стійкою до збоїв і забезпечувати безперервну роботу 24/7 без урахування часу для виконання регламентних робіт відповідно до рекомендацій виробника;
- інтегруватися з існуючими інформаційними системами та технічними засобами, враховуючи стандарти України;
- мати систему резервного копіювання і забезпечувати надійне зберігання даних з можливістю швидкого відновлення у разі збоїв;
- реалізувати заходи для захисту персональних даних та конфіденційної інформації;
- впроваджувати систему аутентифікації та авторизації для запобігання несанкціонованому доступу;
- використовувати шифрування для передачі даних між компонентами системи;

- мати інтуїтивно зрозумілий інтерфейс, простий у користуванні для патрульних і адміністративного персоналу;
- мати можливість доступу до системи через мобільні засоби, що дозволить патрульним отримувати інформацію по місцю патрулювання.

3.2 Вимоги до функцій, що виконуються системою

Комп'ютерна система Управління патрульної служби міста Дніпро повинна мати такі функції:

- забезпечувати можливість відстеження патрулів, транспортних засобів та їх місцезнаходження тобто моніторинг в реальному часі;
- реалізовувати автоматичне приймання, обробку та розподіл викликів;
- мати інструменти для збору та виконувати аналіз статистичних даних, що буде дозволяти приймати обґрунтовані рішення;
- забезпечувати інтеграцію і комунікацію з іншими службами (пожежна, медична, аварійна) для швидкої координації дій;
- через підсистему відеоконтролю завантажувати зображення людей, відокремлювати з них зображення обличь, формувати бази даних з обличчями людей, які розшукуються, та співробітників Управління, ідентифікувати людей, які заходять до Управління, та передавати отриману інформацію відповідальному співробітнику.

3.3 Вимоги до видів забезпечення

3.3.1 Вимоги до інформаційного забезпечення

У співробітників Управління повинен бути доступ до:

- особистого службового електронного кабінету;
- службових баз даних;
- до документації з проектами рішень керівництва відповідно напряму їх діяльності;
- інших підрозділів для взаємодії з співробітниками з питань своєї діяльності;
- дошки новин та оголошень Управління патрульної служби.

3.3.2 Вимоги до лінгвістичного забезпечення

Розробка програмного забезпечення виконується мовою Python із застосуванням бібліотек OpenCv, Face Recognition, NumPy та інших.

Інтерфейс програм – українською або англійською мовами.

3.3.3 Вимоги до технічного забезпечення

Вимоги до комп'ютерів на робочих місцях комп'ютерної системи Управління патрульної служби міста Дніпро:

- процесор: мінімум рівень моделі Intel Core i5 або аналог AMD з частотою 2.5 ГГц та 4 ядра та інтегрованим графічним чіпом рівня Intel UHD Graphics 630 або вище;
- оперативна пам'ять: мінімум 8 ГБ ОЗУ DDR4;
- жорсткий диск: HDD, SSD ємністю мінімум 256 ГБ;
- мережева карта: Gigabit Ethernet;
- операційна система: Windows 10 Pro або новіше / Ubuntu 18.04 LTS або новіше.

Вимоги до серверів:

- процесор: Intel Xeon або аналог AMD з мінімальною кількістю 16 ядер;

- оперативна пам'ять: мінімум 64 ГБ ОЗУ DDR4;
- жорсткий диск: RAID 1 конфігурація з SSD, ємністю мінімум 1 ТБ;
- мережева карта: Gigabit Ethernet;
- операційна система: Windows Server 2019, Linux серверні дистрибутиви або інші аналогічні системи.

Вимоги до комутаторів та маршрутизаторів представлені в таблицях 3.1 та 3.2.

Таблиця 3.1 – Технічні вимоги до комутаторів

Тип	Характеристика
Портів Gigabit Ethernet 10/100/1000 (кількість)	8-24 порти
Кількість портів SFP	Від 4 слот
Пропускна здатність	Від 48 Гбіт/сек; 35.7 Mpps
Об'єм буфера пакетів	до 0.75 Мбайт
Вбудована флеш-пам'ять	Від 32 Мбайт
VLAN (кількість)	Від 1024
Розміри бази даних адрес	Від 8000 MAC-адрес
Маршрутизованих VLAN (кількість)	Від 32
Системна пам'ять memory	Від 128 Мбайт
Транків (кількість)	Від 64
Черг (кількість)	Від 8

Таблиця 3.2 – Технічні вимоги до маршрутизаторів

Тип	Характеристика
Процесор	ARM, не менш ніж, 680MHz
Memory	Не менш ніж: 512MB DDR
Hard Disk	Не менш ніж: 512MB на чипі пам'яті NAND, microSD слот
Ethernet порти	Не менш ніж: П'яти 10/100/1000 Mbit/s Ethernet портів з підтримкою Auto MDI/X
Підтримка PPPoE тунелів	Не менш ніж 500
Підтримка VLAN інтерфейсів	Необмежено
Підтримка EoIP тунелів	Необмежено
Підтримка OVPN тунелів	Необмежено
Підтримка PPTP тунелів	Не менш ніж 500
Працездатність у режимі брандмауера	Не менш ніж 1 Гбіт/с
Підтримка L2TP тунелів	Не менш ніж 500
Підтримка протоколів маршрутизації RIP, OSPF, BGP	Так
Активних користувачів Хот-Спот	500
NAT правила	Необмежено
Правила брандмауера P2P	Необмежено

3.4 Вибір обладнання для побудови комп'ютерної системи

Управління патрульної служби

Будівлі Управління патрульної служби міста Дніпро та Call-Center розміщені на відстані 150 м. Для забезпечення безперервної роботи 24/7 Управління патрульної служби підключені до двох незалежних провайдерів Інтернет.

Логічна схема комп'ютерної системи Управління патрульної служби представлена на рисунку 3.1.

Відоме пасивне і активне мережне обладнання. Для схеми системи було підібрано наступне обладнання (таблиця 3.3).

Для підсистеми відеонагляду вибрано IP роботизована камера 4Мп IP-камера серії WizSense. Оснащена 1/2.8" CMOS-матрицею і моторизованим об'єктивом з 4-кратним оптичним збільшенням. Фокусна відстань 2.8-12 мм дозволяє регулювати кут огляду від 101 до 31.36. Камера підтримує карти пам'яті об'ємом до 512 ГБ і має ІК-підсвітку до 50 метрів. Вона також забезпечує функції відеоспостереження, такі як виявлення вторгнень, відмітку зниклих речей та інші можливості.

3.5 Модернізація структурної схеми комп'ютерної системи

Управління патрульної служби

Відповідно до вибраного обладнання та логічної схеми комп'ютерної системи (рисунок 3.1) була розроблена в бакалаврській роботі структурну схему комп'ютерної системи Управління патрульної служби (рисунок 3.2).

Для модернізації системи Управління патрульної служби і реалізації додаткових функцій по завантаженню зображення людей, відокремлення з них зображення обличчя, формування бази даних з обличчями людей, які розшукуються, та співробітників Управління, ідентифікування людей, які

заходять до Управління, та передавання отриманої інформації відповідальному співробітнику розроблена IoT підсистема відеонагляду, яка підключена до Wi-Fi маршрутизатора відділу Leader ship.

Таблиця 3.3 – Специфікація засобів обладнання

Позиція	Найменування і технічна хар-ка	Тип, марка, позначення	Одиниці виміру	Кількість
1	Комутатор	Cisco RV345P Dual WAN Gigabit VPN Router (RV345P-K9-G5)	шт	4
2	Маршрутизатор	D-Link (DES-1210-28P)	шт	3
3	Маршрутизатор	D-Link DGS-1008P PoE	шт	1
4	Маршрутизатор	D-Link DGS-1210-52	шт	2
5	Сервер	Dell PowerEdge T30 (T30v01)	шт	2
6	Бездротовий маршрутизатор	D-Link DIR-615	шт	2
7	Системний блок	Everest Home&Office 1036	шт	90
8	Монітор	QUBE H24F75	шт	90
9	Клавіатура	2E KS 101 USB Black	шт	90
10	Миша	HP X1500 USB Black (H4K66AA)	шт	90
11	Принтер	CANON i-SENSYS MF641Cw	шт	5
12	Гарнітура	Sennheiser PC 5 CHAT	шт	90
12	ОС сервера	DELL Windows Server 2019 Standard ROK (634-BSFX)	шт	2
13	ОС робочих станцій	Windows 10 Professional	шт	90
14	IP-відеокамера	IP камера Dahua DH-SD29204DB-GNY-W	шт	2

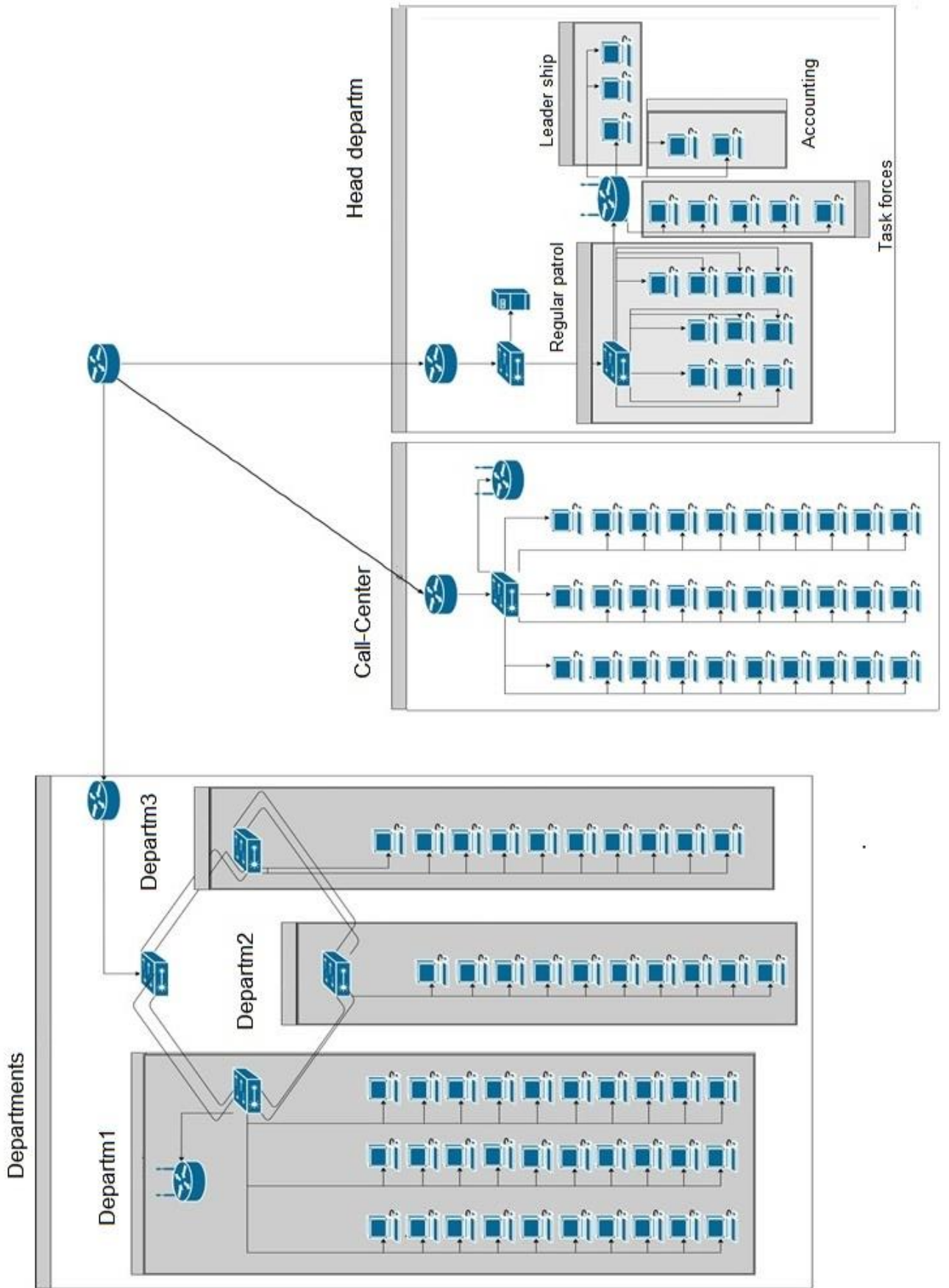


Рисунок 3.1 – Логічна схема комп'ютерної системи Управління

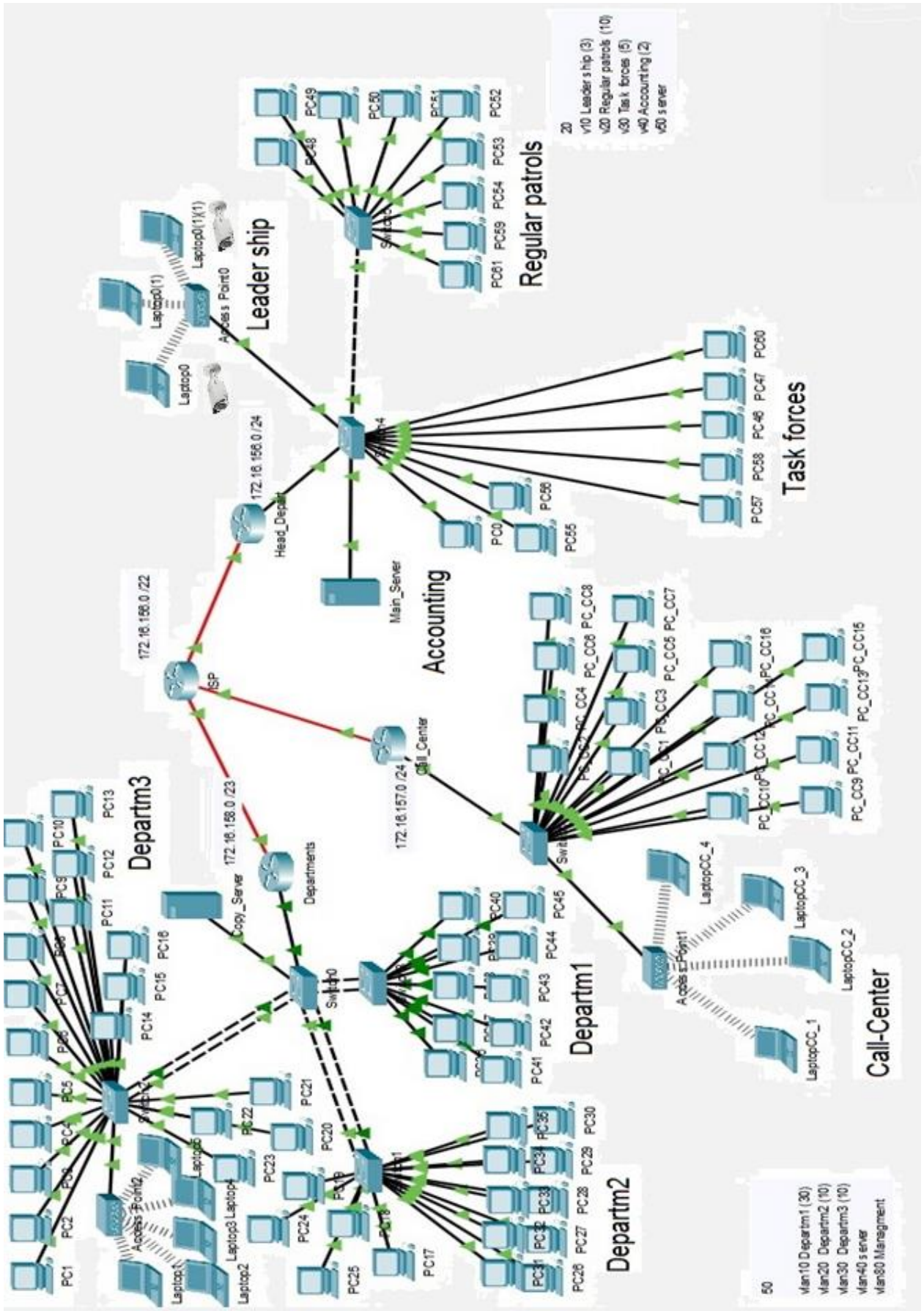


Рисунок 3.2 – Структурна схем комп'ютерної системи Управління

В таблиці 3.4 представлена схема адресації мережі.

Таблиця 3.4 – Схема адресації мережі

Назва мережі	Кількість вузлів	Номер мережі	Маска мережі	Діапазон можливих адрес вузлів у підмережі
Head_depart	254	172.16.156.0	255.255.255.0	172.16.156.1-254
Call_Center	254	172.16.157.0	255.255.255.0	172.16.157.1-254
Departments	510	172.16.158.0	255.255.254.0	172.16.158.1-159.254

В таблиці 3.5 показано схему адресації підмереж VLAN .

Таблиця 3.5 - Схема адресації VLAN

Назва мережі	Кількість вузлів	Номер мережі	Маска мережі	VLAN	Діапазон можливих адрес вузлів у підмережі
Head_depart					
Leader_ship	14	172.16.156.0	255.255.255.240	10	172.16.156.1-14
Managment_depart	14	172.16.156.16	255.255.255.240	20	172.16.156.17-30
Task_forces	14	172.16.156.32	255.255.255.240	30	172.16.156.33-46
Accounting	14	172.16.156.48	255.255.255.240	40	172.16.156.49-62
Regular_patr	14	172.16.156.64	255.255.255.240	80	172.16.156.65-78
Server	2	172.16.156.80	255.255.255.252	50	172.16.156.81-82
Departments					
Departments	64	172.16.158.0	255.255.255.192	10	172.16.156.1-62
Departm1	14	172.16.158.64	255.255.255.240	20	172.16.156.65-78
Departm2	14	172.16.158.80	255.255.255.240	30	172.16.156.81-94
Departm3	14	172.16.158.112	255.255.255.240	80	172.16.156.113-126
Server	14	172.16.158.96	255.255.255.240	40	172.16.156.97-110

Структурна схема IoT підсистеми відеонагляду представлена на рисунку 3.3.

Вона складається з:

- SBC-PT – мікрокомп'ютера Raspberry Pi;
- датчик руху (MotionDet), який сигналізує про появу людини на підході до відеокамери;

- датчик перешкод (TripSen), що визначає наявність людини в зоні дії камери на відстані 2 м;
- світловий пристрій (Light), яким можна керувати для додаткового освітлення;
- камера (Cam) для ідентифікації людини на вході в Управління патрульної служби;
- перемикач (SwitchR), який дозволяє вручну вмикати або вимикати освітлення.

Схема алгоритму роботи IoT підсистеми відеонагляду представлена на рисунку 3.4.

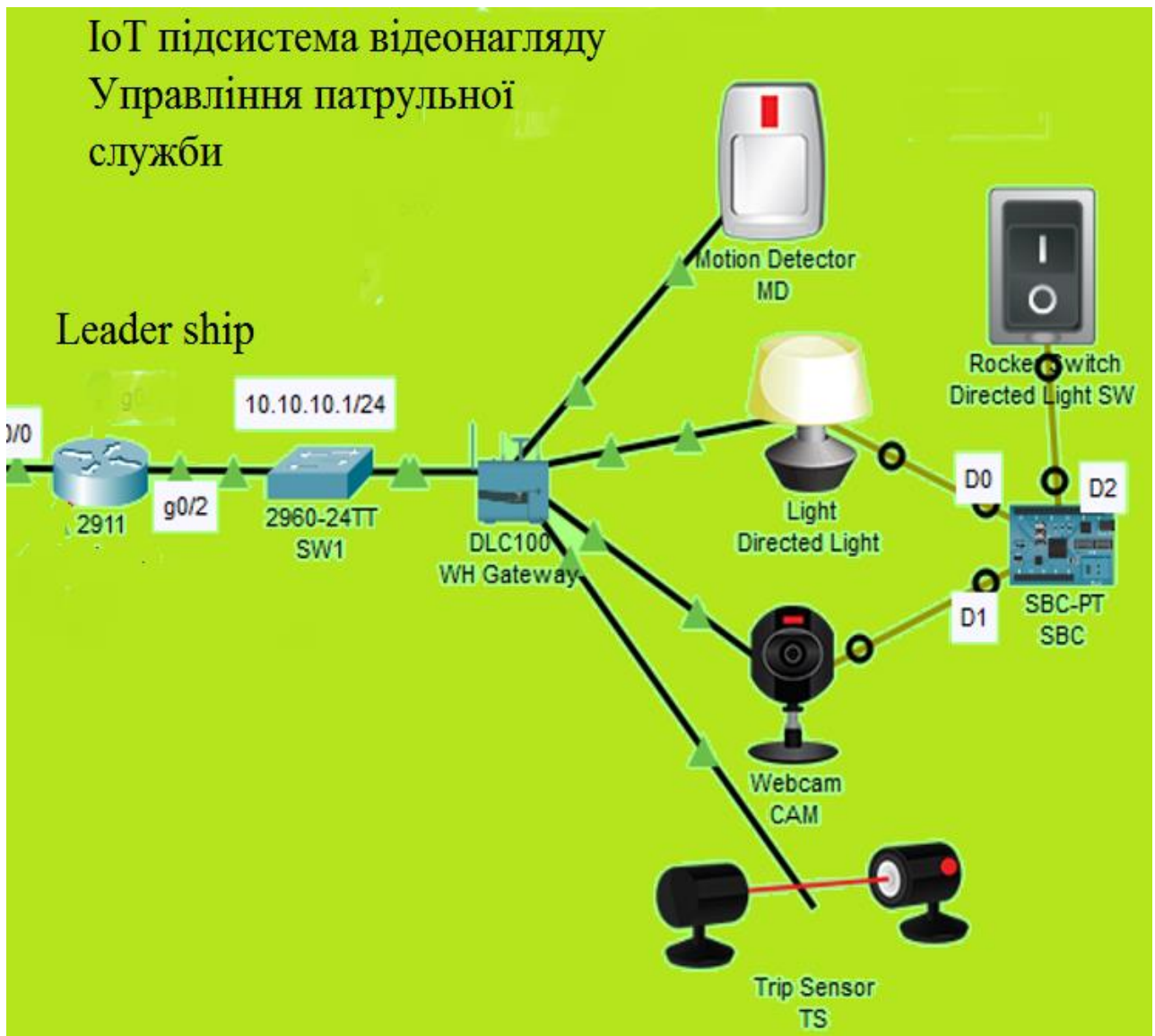


Рисунок 3.3 – Структурна схема IoT підсистеми відеонагляду

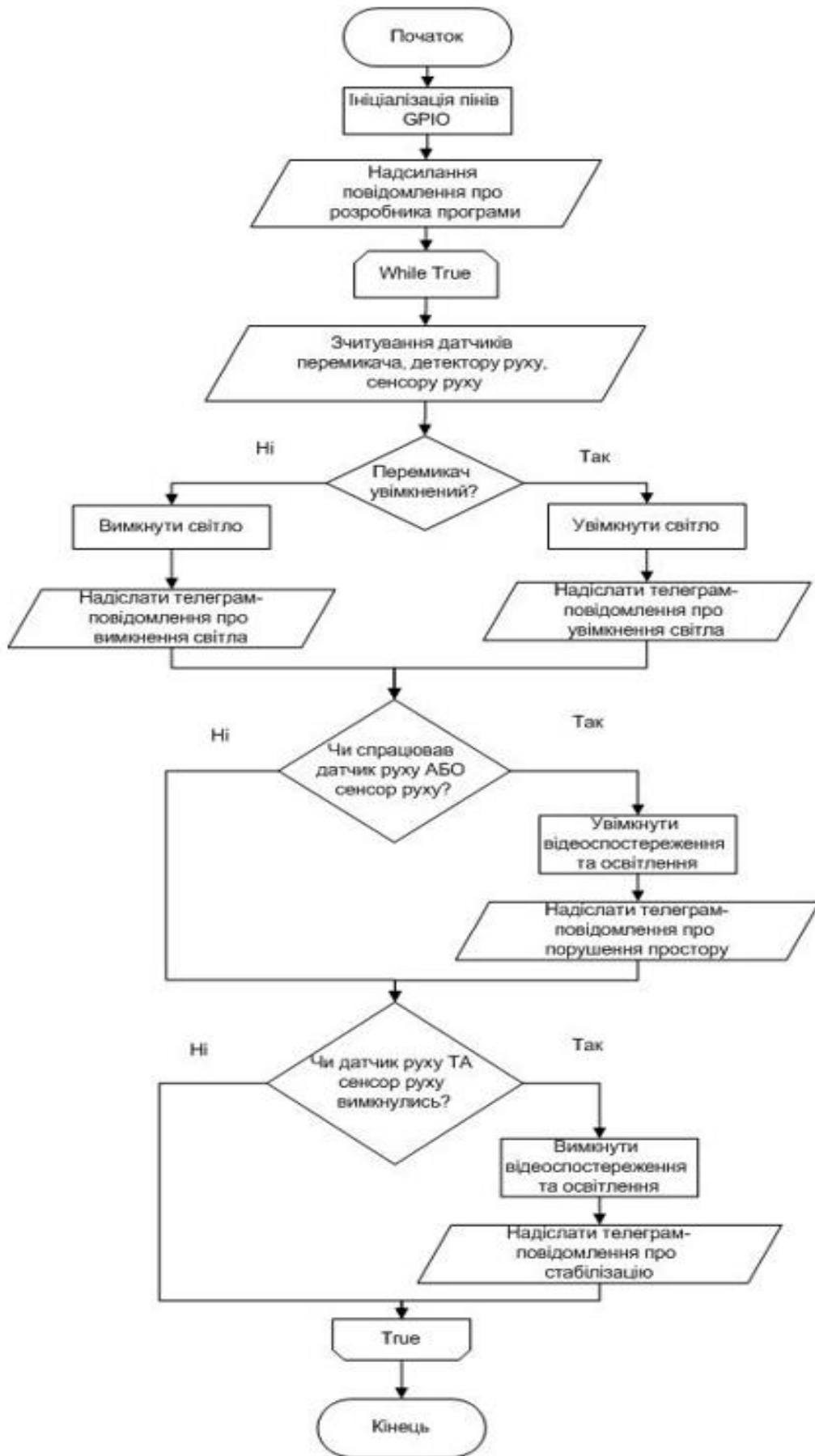


Рисунок 3.4 – Схема алгоритму роботи IoT підсистеми відеонагляду

4 РАЗРОБЛЕННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

4.1 Призначення та сфера застосування програми

Програмне забезпечення комп'ютерної системи Управління патрульної служби м. Дніпро призначене для отримання фотозображення людини, якщо це було відео, то виділення з його окремих кадрів для обробки, попередньої обробки, виділення обличчя людини на зображенні, обробки зображень, які надсилаються з відеокамер, ідентифікації по зображенню обличчя людини, тобто порівняння з базою даних зображень, наприклад, людей, які розшукуються або співробітників Управління патрульної поліції м. Дніпро.

4.2 Обґрунтування технічних характеристик програми

4.2.1 Постановка завдання на розробку програми

Для розробки програмного забезпечення комп'ютерної системи Управління патрульної служби м. Дніпро була вибрана мова програмування Python з бібліотеками роботи із зображеннями OpenCV та NumPy в середовищі IDLE.

4.2.2 Структура і алгоритм функціонування програми

Комп'ютерна система повинна працювати 24/7 і тому структура програми повинна мати циклічний характер.

4.2.3 Вибір складу програмних засобів

З метою скорочення часу для розробки програми пропонуються застосувати наступні бібліотеки:

- OpenCV;
- NumPy.

4.3 Опис розробленої програми

4.3.1 Загальні відомості

Розроблена програма (Додаток А) складається з модулів для обробки зображень системи Управління, які застосовувалися під час обробки та підготовки зображень в підсистемі відеонагляду та експериментальних досліджень.

Текст програми реалізований мовою Python версії 3.12 в середовищі IDLE.

4.3.2 Функціональне призначення

4.3.2.1 Типи розв'язуваних завдань

Програма призначена:

- для отримання фотозображення людини, якщо це було відео, то виділення з його окремих кадрів для обробки;
- попередньої обробки;
- виділення обличчя людини на зображенні;
- обробки зображень, які надсилаються з відеокамер;
- ідентифікації по зображенню обличчя людини, тобто порівняння з базою даних зображень, наприклад, людей, які розшуковуються або співробітників Управління патрульної поліції м. Дніпро.

4.3.2.2 Функціональні обмеження

Дозволені формати для зображень – jpg, png

4.3.3 Опис логічної структури програми

Якщо проводиться обробка відео, то виділення з його окремих кадрів для обробки зображень виконується за схемою алгоритму рисунок 4.1 а).

Схема алгоритму роботи програми представлена на рисунку 4.1 б). Вона складається з ініціалізації бібліотек та основного класу, циклу підготовки та обробки зображень та збереження конфігурації основного класу.

4.3.4 Використовувані технічні засоби

Для роботи комп'ютерної системи застосовується цифрова камера.

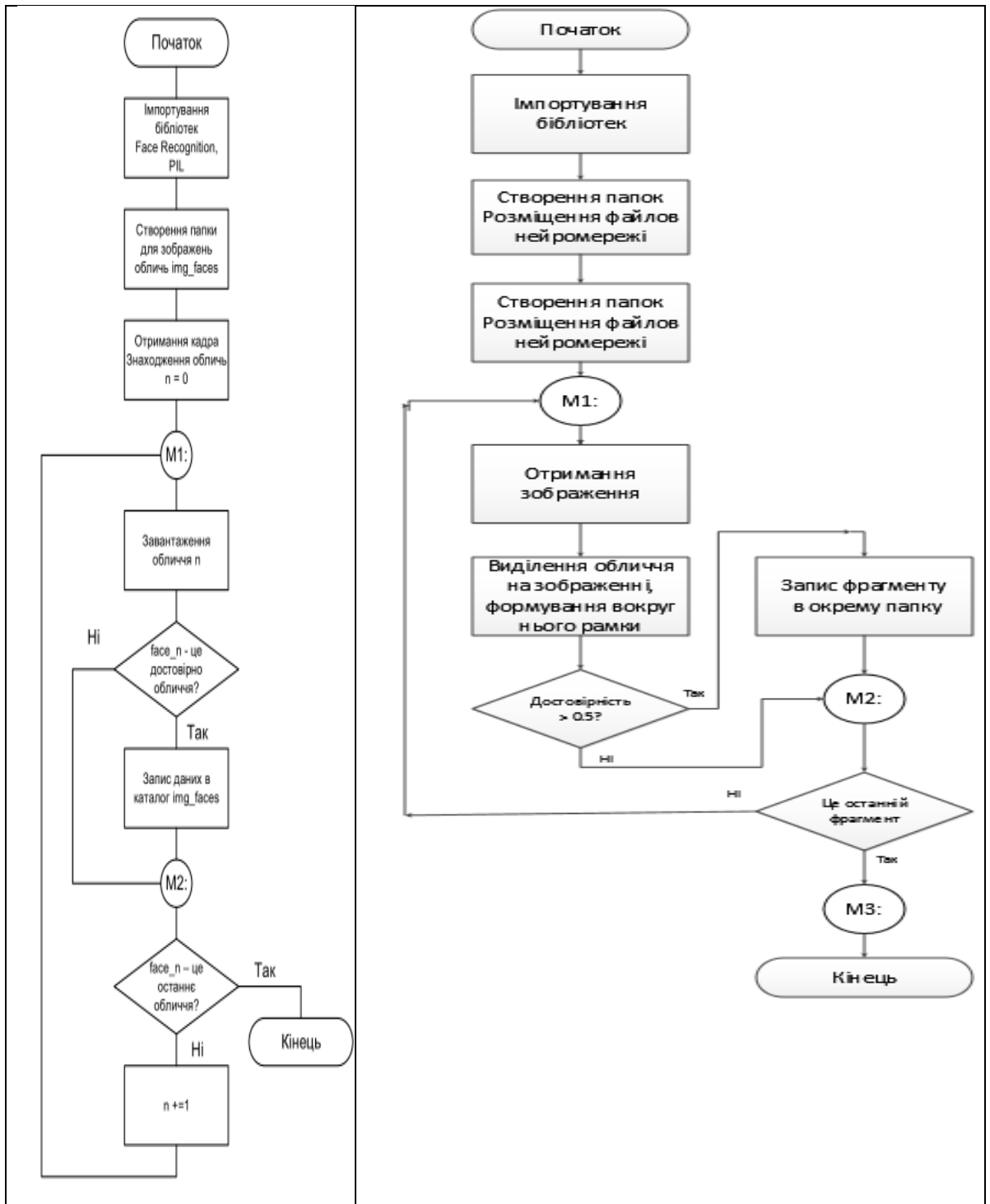
4.3.5 Цикл роботи програми

Цикл обробки зображень включає: імпортування бібліотек, створення папок для вхідних зображень та вилучених зображень осіб, розміщення файлів нейромережі prototxt та caffemodel в папці model_data, отримання зображення, читання зображень, виділення обличчя на зображенні і формування вокруг нього прямокутної рамки та запис виділеного фрагменту в окремий файл в окрему папку, якщо програма більш ніж на 50% упевнена, що виявлення є обличчям, рисунок 4.1 б).

4.3.6 Вхідні та вихідні дані

Зображення, які отримуються, мають вид масиву $X * Y * Z$ цілих чисел розміром від 0 до 255 пк (пікселей). Де: X – ширина, Y – висота.

При роботі з зображенням, отриманим з відеокамери, у програмі, що розробляється, воно представляється у вигляді масиву X на Y на Z цілих величин від 0 до 255, де X та Y – висота та ширина зображення у пікселях, а Z – колірна компонента, яка дорівнює 3 для RGB.



а)

б)

Рисунок 4.1 – Схема алгоритму програми отримання кадрів з відео і обробки зображень

5 ЕКСПЕРИМЕНТАЛЬНИЙ РОЗДІЛ

5.1 Постановка завдання експерименту і обґрунтування методики

Виконаємо тестування роботи комп'ютерної системи Управління патрульної служби з функцією ідентифікації людей. В процесі експерименту необхідно оцінити роботи системи та можливість застосування на реальних зображеннях с точки зору можливості вилучення зображення з відеофайлів, знаходження обличь людей на зображенні, екстракції зображень обличь для організації баз даних людей що розшукуються, так і співробітників Управління патрульної служби, ідентифікації приналежності людей, що входять в приміщення в перераховані бази даних. Крім того в процесі експерименту потрібно виявити характеристики, які можуть впливати на структуру комп'ютерної системи Управління патрульної служби.

5.2 Проведення експерименту для тестування розробленої системи на підібраних фотозображеннях груп людей

5.2.1 Перевірка роботи на підібраних зображеннях груп людей

На цьому кроці застосуємо зображення людей підібраних для експерименту в розділі 2, групи 1 та 2 (рисунки 2.1, 2.2).

З використанням розробленого ПО виконаємо знаходження обличь людей на зображенні, екстракції зображень обличь і підготовку до порівняння.

В результаті отримаємо зображення з розпізнаними обличчями, які помічені рамками світло-бірюзового кольору (рисунки 5.1, 5.2) і окремий каталог з вилученими зображеннями обличь (рисунки 5.3, 5.4).

В наступних таблицях показані значення достовірності з якими розпізнані обличчя на зображеннях людей групи 1 та 2 (таблиця 5.1, 5.2)

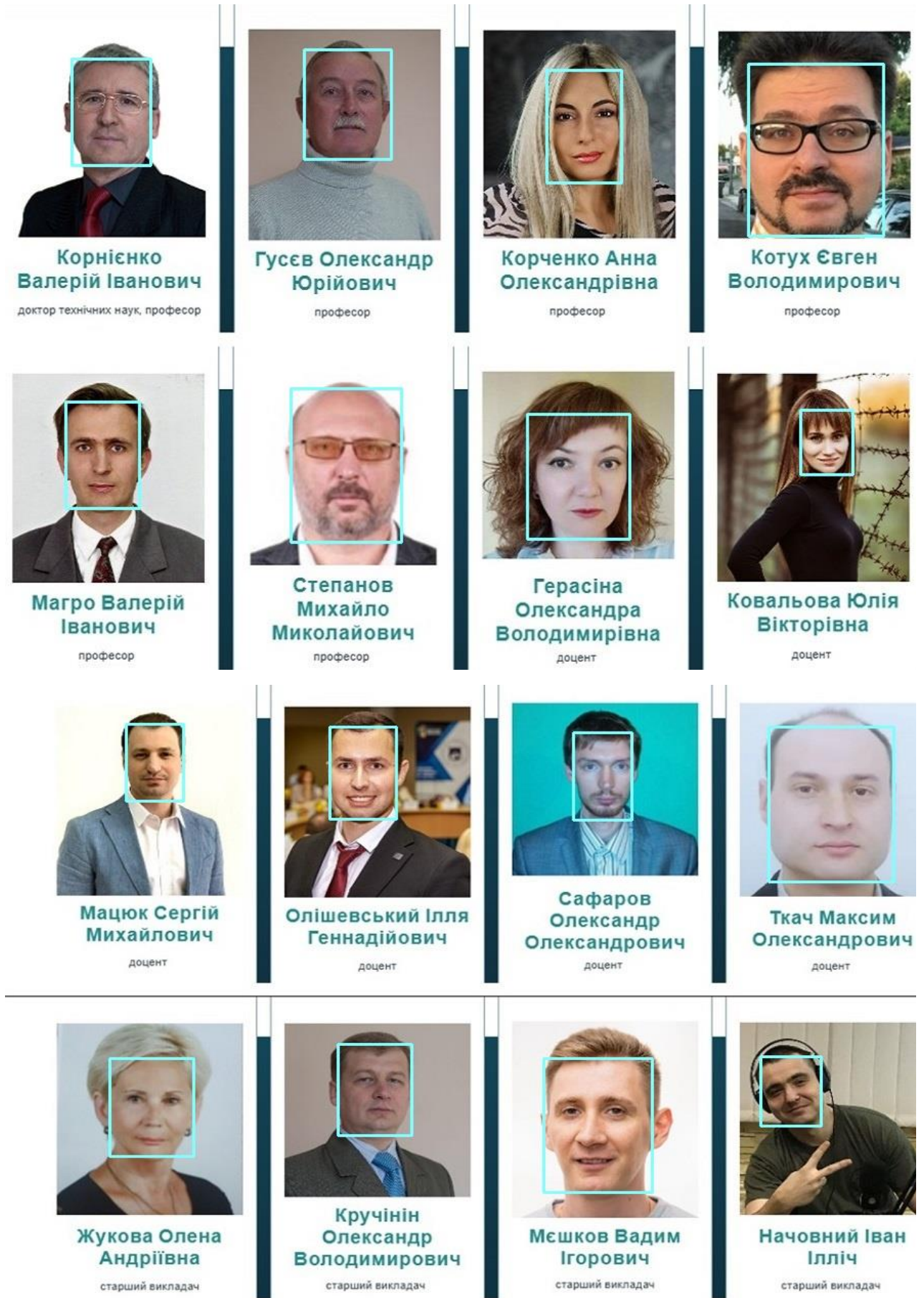


Рисунок 5.1 – Зображень людей групи №1 з розпізнаними обличчями

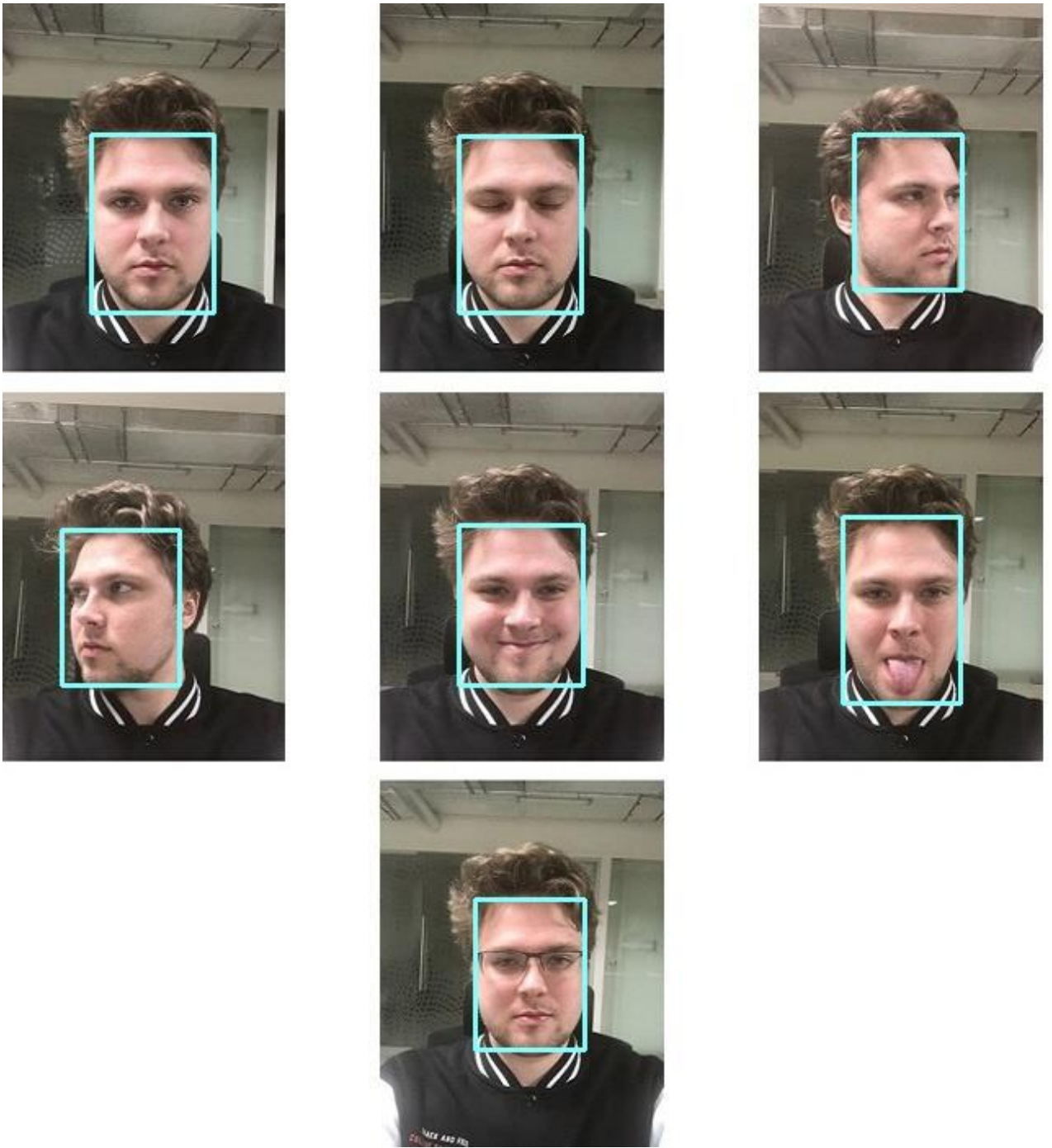


Рисунок 5.2 – Зображень людей групи №2 з розпізнаними обличчями

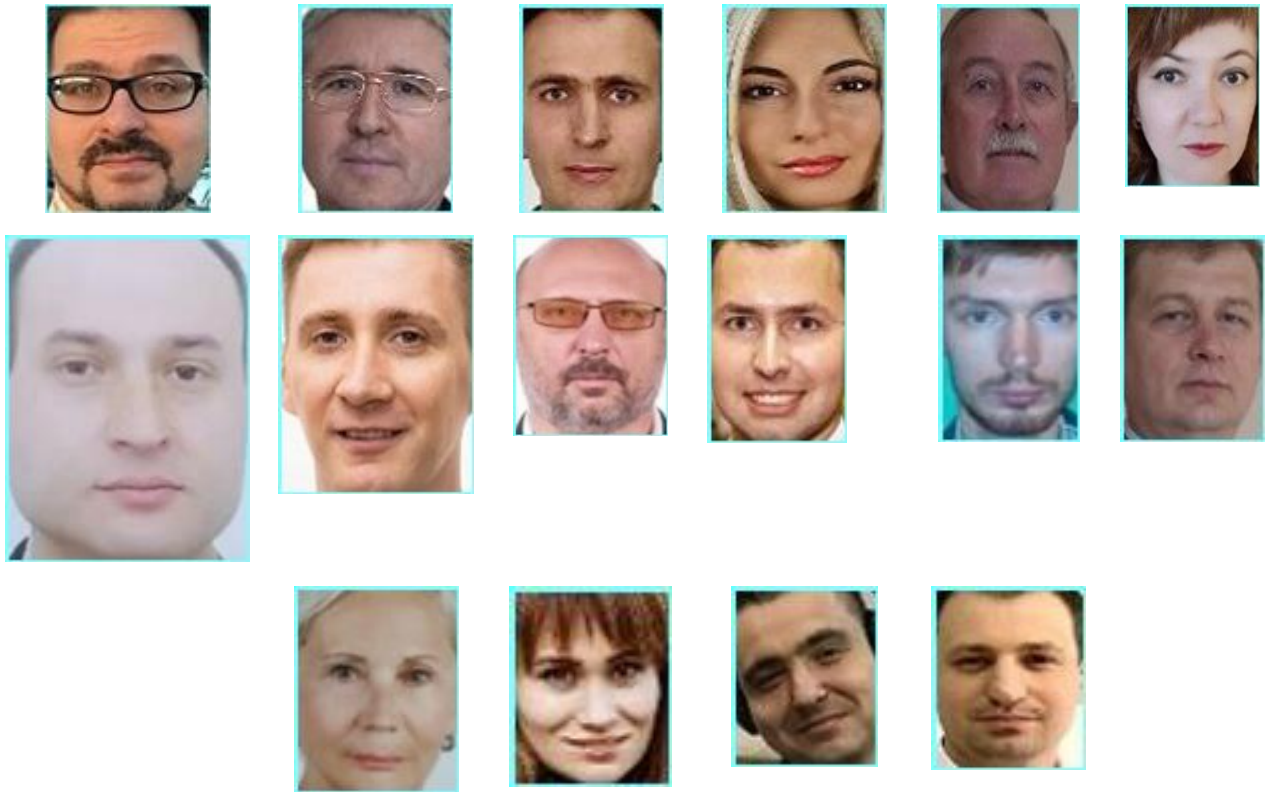


Рисунок 5.3 – Вилучені зображення обличь людей групи 1



Рисунок 5.4 – Вилучені зображення обличь людей групи 2

Таблиця 5.1 – Значення достовірності при розпізнанні обличчя людей гр. 1

№ зображення	Достовірність	№ зображення	Достовірність	№ зображення	Достовірність
1	0.99997103	7	0.792419	13	0.96072006
2	0.9920512	8	0.6986949	14	0.8847965
3	0.98829925	9	0.9991202	15	0.86229026
4	0.97826546	10	0.99890065	16	0.82840514
5	0.94980454	11	0.9910712	–	–
6	0.90420926	12	0.98904747	–	–

Таблиця 5.2 – Значення достовірності при розпізнанні обличчя людей гр. 2

№ зображення	Достовірність	№ зображення	Достовірність	№ зображення	Достовірність
1	0.9985858	4	0.9961402	7	0.9737248
2	0.9983779	5	0.98623997	–	–
3	0.997414	6	0.9764557	–	–

5.2.2 Перевірка роботи при різній відстані до камери

На цьому кроці застосуємо статичне зображення людей підібраних для експерименту в розділі 2, групи 3 та 4 (рисунки 2.3 – 2.5). Особливістю цього кроку є те, люди знаходяться на різній відстані від камери. В групі 3 ця відстань фіксована і складає від 4.0 м до 0.5. Людину тут знято з інтервалом 0.5 м.

В групі 4 люди знаходяться умовно на 6-ти рядах столів і можна визначити, що відстань до камери змінюється орієнтовно від 1 м. до 5 м.

З використанням розробленого ПО виконаємо знаходження обличчя людей на зображенні, екстракції зображень обличчя і підготовку до порівняння.

В результаті роботи програми по обробці зображень групи людей №3, які зняти на відстані від 4.0 м до 2.5 м (рисунок 2.3) обличчя не було знайдено (рисунок 5.5).

А на відстані від 0.5 м до 2.0 м (рисунок 2.4), отримаємо зображення з розпізнаними обличчями людей, які помічені рамками світло-бірюзового кольору (рисунок 5.6).

При першій спробі обробки зображень групи людей №4 був встановлений поріг достовірності 0.5, як і при роботі з групою №3, і розпізнано тільки чотири обличчя (рисунок 5.7).

З метою визначення можливостей розробленої програми була спроба зменшити поріг достовірності до 0.3. Робота з даними зображеннями показала, що це дає можливість розширити кількість розпізнаних обличчя людей частково і за третім рядком столів (рисунок 5.8).

Отримано окремий каталог з вилученими зображеннями обличчя групи №3 (рисунок 5.9), групи №4 при першій спробі (рисунок 5.10) та групи №4 при другій спробі (рисунок 5.11)

В таблицях покажемо значення достовірності з якими розпізнані обличчя на зображеннях людей (рисунки 5.6 – 5.8) групи 3 та 4 (таблиці 5.3 – 5.5).

Залежності значення достовірності з якими розпізнані обличчя в таблиці 5.3 представимо і у вигляді графіку (рисунок 5.12).

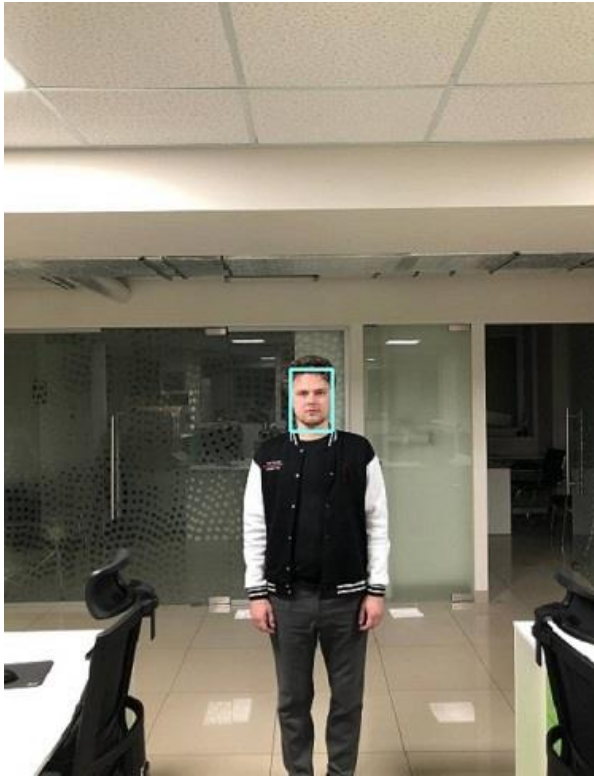


а)

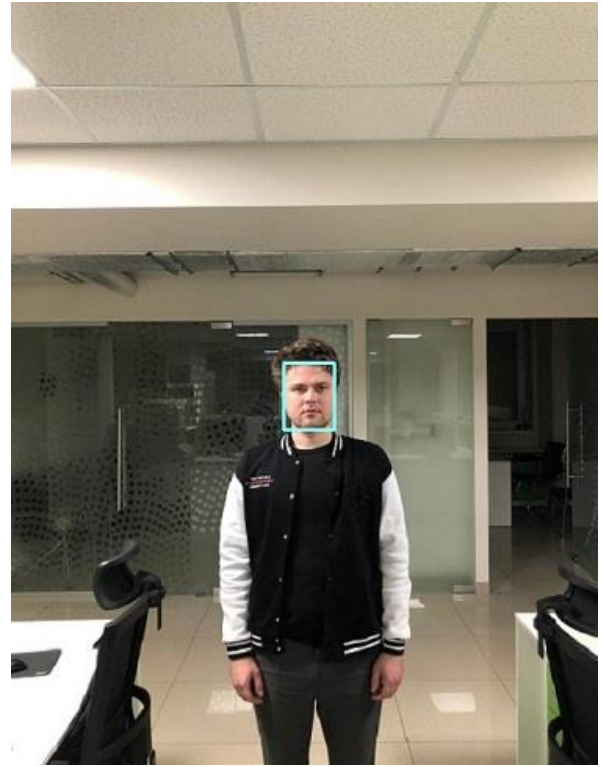


б)

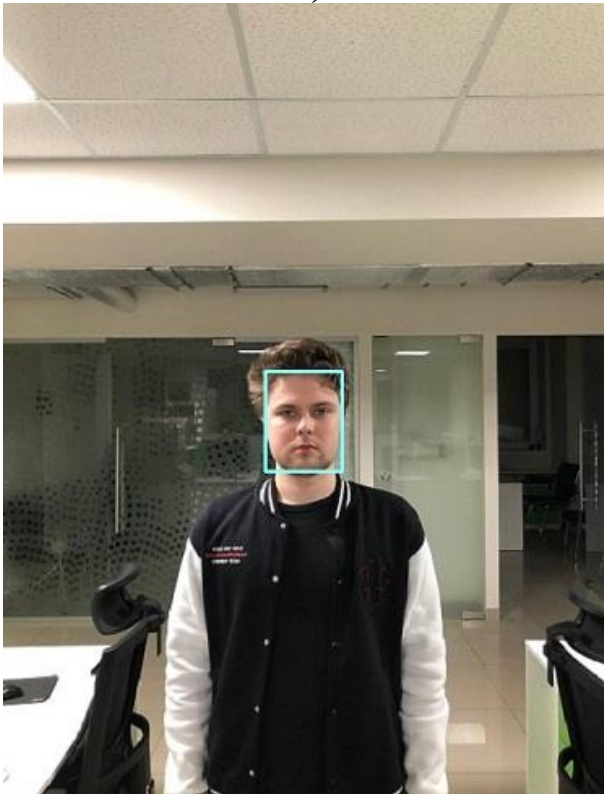
Рисунок 5.5 – Група №3 із зображеннями на відстані 4.0 і 2.5 м (обличчя не розпінано), а) 4.0 м; б) 2.5 м



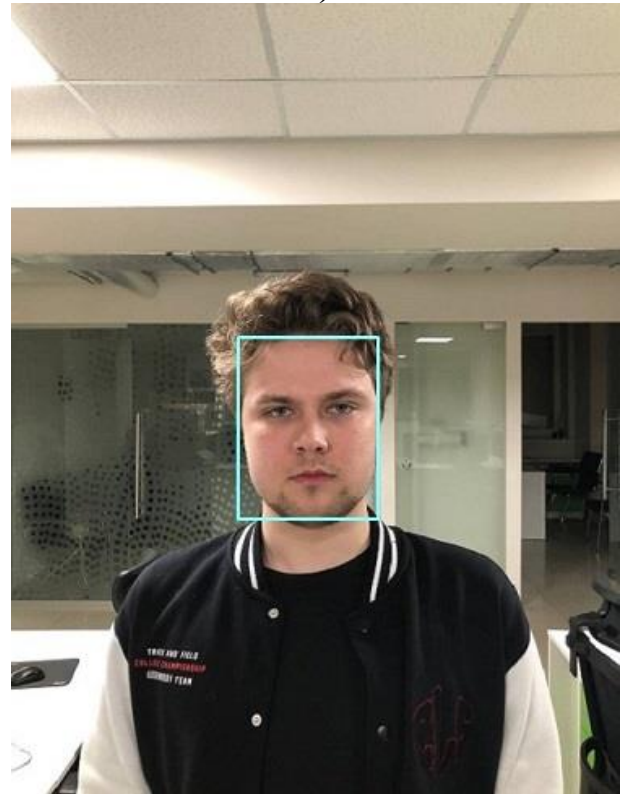
а)



б)



в)



г)

Рисунок 5.6 – Група №3 із зображеннями на відстані 2.0 – 0.5 м,

а) 2.0 м; б) 1.5 м; в) 1.0 м; г) 0.5 м



Рисунок 5.7 – Група №4, поріг достовірності 0.5



Рисунок 5.8 – Група №4, поріг достовірності 0.3



Рисунок 5.9 – Вилучені зображення обличч людей групи №3



Рисунок 5.10 – Вилучені зображення обличч людей групи №4



Рисунок 5.11 – Вилучені зображення обличч людей групи №4 при другій спробі

Таблиця 5.3 – Значення достовірності при розпізнанні обличчя людей групи №3

№ рисунка	Відстань, м	Достовірність
5.6 а)	2	0.8531617
5.6 б)	1.5	0.9783261
5.6 в)	1.0	0.99827695
5.6 г)	0.5	0.99982363

Таблиця 5.4 – Значення достовірності при розпізнанні обличчя людей групи №4, поріг достовірності 0.5

№ зображення	№ ряду столів	Достовірність
1	1	0.8963308
2	1	0.7859434
3	2	0.59525317
4	2	0.5689191

Таблиця 5.5 – Значення достовірності при розпізнанні обличчя людей групи №4, поріг достовірності 0.3

№ зображення	№ ряду столів	Достовірність	№ зображення	№ ряду столів	Достовірність
1	1	0.8963308	5	3	0.42675638
2	1	0.7859434	6	3	0.37268013
3	2	0.59525317	7	3	0.3395013
4	2	0.5689191	8	2	0.32496005

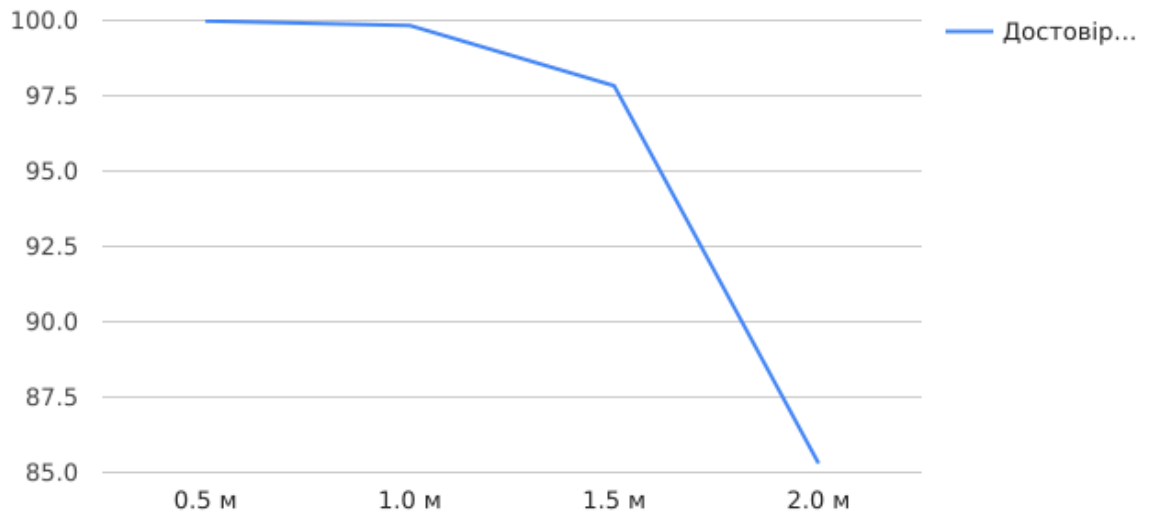


Рисунок 5.12 – Залежність значень достовірності при розпізнанні обличчя людей групи №3 від відстані

5.3 Тестування розробленої комп'ютерної системи при отриманні реального відео

5.3.1 Перевірка функції розпізнавання обличч

Розглянемо роботу розробленої системи при захоплені відеопотоку з камери відеонагляду, виділення окремих кадрів і розпізнавання на зображеннях цих кадрів обличч людей.

В зв'язку з тим, що буде проходити імітація входу до приміщення Управління на відео буде присутня тільки одна людина яку потрібно в подальшому по зображенню обличчя ідентифікувати по базі людей, які розшуковуються, та базі співробітників Управління.

На першому етапі спробуємо перевірити функцію розпізнавання обличч на зображеннях людей. Для цього застосуємо підготовлений і описаний в підрозділі 2.4 відеозапис на якому людина переміщається на вхід приміщення Управління з відстані 4.0 м. Програма для виділення окремих кадрів з відеопотоку і розпізнавання на зображеннях цих кадрів обличч людей розроблені в підрозділах 2.1, 2.2.

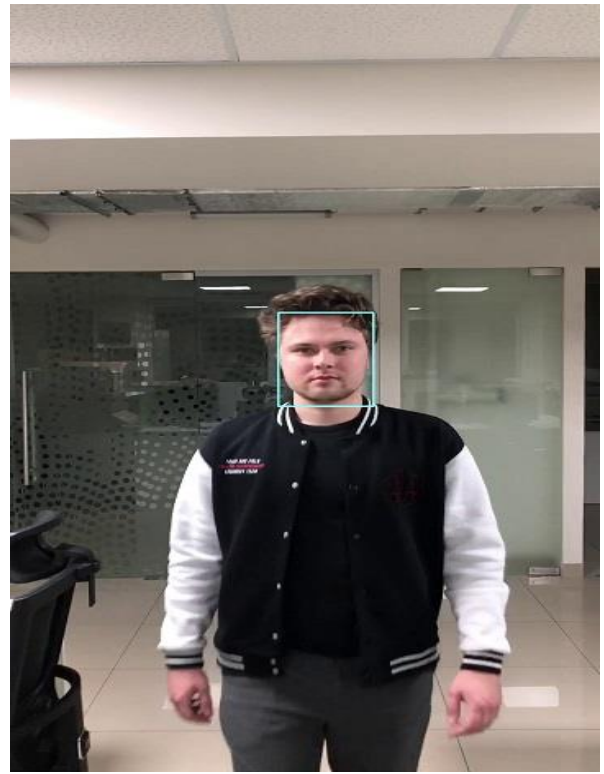
При обробці відео-файлу було виявлено, що людина проходить на вхід Управління 4 м за 6 секунд, тобто її швидкість складає 0.67 м/С або 2.4 км/год. Ці результати відповідають дійсності тому, що проведені дослідження (Outdoor Walking Speeds of Apparently Healthy Adults: A Systematic Review and Meta-analysis <https://pmc.ncbi.nlm.nih.gov/articles/PMC7806575/>) показали, що середня швидкість повільної ходьби на відкритому просторі становила $0,82 \pm 0,02$ м/с (9 досліджуваних груп, $n = 201$). Що практично збігається з отриманим значенням 0.67 м/С, тим паче запис переміщення проводився в обмеженому просторі.

За налаштуваннями програми виділявся з відеопотоку і записувався для обробки кожен 30-й кадр. При довжині відео в 180 кадрів було записано шість кадрів, але обличчя було розпізнано тільки на чотирьох (рисунок 5.13).

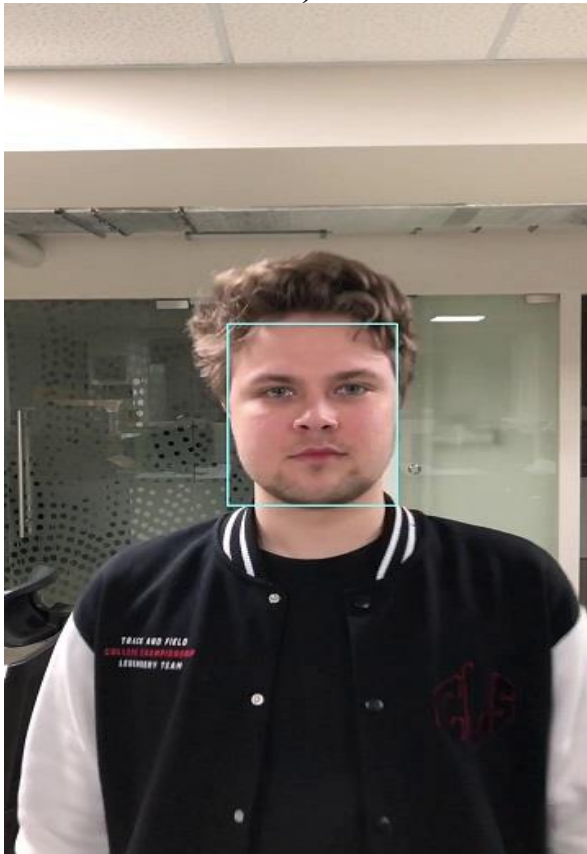
Як і в попередньому експерименті (рисунок 5.6) розпізнання обличчя на зображенні виконано при відстані до камери від 2.0 м і ближче. Якщо відстань більше 2.0 м розпізнання при наявній відеокамері і застосованого ПЗ неможливо.



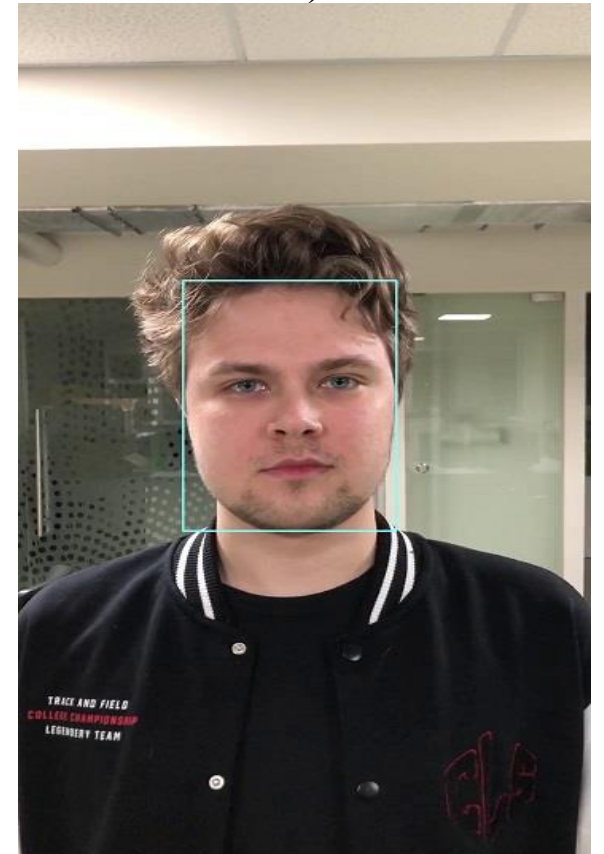
а)



б)



в)



г)

Рисунок 5.13 – Група №6 із зображеннями з відеопотоку на відстані 2.0 – 0.0 м,

а) 2.0 м; б) 1.33 м; в) 0.67 м; г) 0.0 м

5.3.2 Перевірка функції ідентифікації людей по зображенню обличь

Перевіримо розроблене програмне забезпечення в розділі 2.3 при обробці реальних зображень осіб отриманих з відео на вході в Управління для порівняння двох зображень з метою ідентифікації людей, що розшукуються, або визначення співробітників Управління для вільного допуску у приміщення.

При порівнянні застосовується бібліотека Face Recognition Library мови Python, робота якої базується на нейромережі dlib.

Програма при порівнянні зображень обличь формує матриці з 128 параметрами кожного обличчя.

В якості людини, що розшукується представимо зображення №2 рисунка 2.3 групи людей №1, а в якості бази обличь людей, що розшукаються, завантажимо зображення обличь рисунок 5.3.

Програма формує матрицю обличчя людини, що розшукується, і матриці обличь людей з бази і виконує їх порівняння. При співпаданні параметрів з вірогідністю > 0.8 видається повідомлення: «Attention! This person is wanted!» (рисунок 5.14). Якщо перевірка не дає позитивного результату, то видається повідомлення: «This person is not wanted!», а перевірка продовжується (рисунок 5.15).

Розглянемо такий варіант, коли виконується перевірка з базою обличь співробітників Управління.

При співпаданні параметрів з вірогідністю > 0.8 видається повідомлення: «This person is an employee of the Patrol Service Department!» (рисунок 5.16).

При негативному результаті перевірки видається повідомлення: «This person is not an employee of the Patrol Service Department!» (рисунок 5.17).

=====RESTART:E:\Diplom24.12\Ident\main1.py=====

img1/1_1.jpg

```
[-0.08309721 0.13508761 0.04695711 -0.0303251 -0.11299005 -0.00831402
-0.13330069 -0.08423181 0.01290667 -0.04902124 0.22730255 -0.05002961
-0.26350614 0.05895156 -0.05491035 0.15215828 -0.10079271 -0.10885511
-0.17822301 -0.05937494 0.03103823 0.00864732 -0.07510658 0.03645425
-0.07173587 -0.30925333 -0.02588677 -0.07802886 0.03903908 -0.14316942
-0.01452429 0.00697493 -0.0973518 -0.03665408 0.08563334 0.00971138
-0.02258581 -0.09886289 0.21765362 0.02559109 -0.13598426 0.0370877
0.08466195 0.27044007 0.23111735 0.08047093 0.05328194 -0.07512479
0.14900628 -0.27684951 0.04094721 0.12351754 0.15673159 0.12058617
0.08051699 -0.17722739 0.04170166 0.11862428 -0.15894955 0.08429052
0.02210581 -0.11650215 -0.09011411 -0.07416122 0.22238208 0.08819635
-0.09959257 -0.08606474 0.21603337 -0.08106166 -0.11628384 -0.04332479
-0.1266284 -0.11737472 -0.24157429 0.04847138 0.33929217 0.15111613
-0.27824438 0.02222247 -0.00917822 -0.00155878 -0.009915 0.00720793
-0.11570834 -0.13140647 -0.03244095 -0.02056564 0.23833658 -0.02545743
0.02343878 0.19053808 -0.00083461 -0.05968615 0.08536716 0.06993084
-0.14742085 -0.0942644 -0.10691888 -0.11872785 0.12560691 -0.12499915
0.03573279 0.15493751 -0.26417264 0.14696859 -0.0743521 -0.00043932
0.0800707 0.06483318 -0.05300179 -0.04439183 0.19225958 -0.14804026
0.21271621 0.21841258 0.05767109 0.06724489 0.08666692 0.06578683
0.02313012 0.05796478 -0.15156271 -0.18970217 0.04519731 0.03561738
-0.04201681 0.03872568]
```

img2/1.jpg

```
[-0.08309721 0.13508761 0.04695711 -0.0303251 -0.11299005 -0.00831402
-0.13330069 -0.08423181 0.01290667 -0.04902124 0.22730255 -0.05002961
-0.26350614 0.05895156 -0.05491035 0.15215828 -0.10079271 -0.10885511
-0.17822301 -0.05937494 0.03103823 0.00864732 -0.07510658 0.03645425
-0.07173587 -0.30925333 -0.02588677 -0.07802886 0.03903908 -0.14316942
-0.01452429 0.00697493 -0.0973518 -0.03665408 0.08563334 0.00971138
-0.02258581 -0.09886289 0.21765362 0.02559109 -0.13598426 0.0370877
0.08466195 0.27044007 0.23111735 0.08047093 0.05328194 -0.07512479
0.14900628 -0.27684951 0.04094721 0.12351754 0.15673159 0.12058617
0.08051699 -0.17722739 0.04170166 0.11862428 -0.15894955 0.08429052
0.02210581 -0.11650215 -0.09011411 -0.07416122 0.22238208 0.08819635
-0.09959257 -0.08606474 0.21603337 -0.08106166 -0.11628384 -0.04332479
-0.1266284 -0.11737472 -0.24157429 0.04847138 0.33929217 0.15111613
-0.27824438 0.02222247 -0.00917822 -0.00155878 -0.009915 0.00720793
-0.11570834 -0.13140647 -0.03244095 -0.02056564 0.23833658 -0.02545743
0.02343878 0.19053808 -0.00083461 -0.05968615 0.08536716 0.06993084
-0.14742085 -0.0942644 -0.10691888 -0.11872785 0.12560691 -0.12499915
0.03573279 0.15493751 -0.26417264 0.14696859 -0.0743521 -0.00043932
0.0800707 0.06483318 -0.05300179 -0.04439183 0.19225958 -0.14804026
0.21271621 0.21841258 0.05767109 0.06724489 0.08666692 0.06578683
0.02313012 0.05796478 -0.15156271 -0.18970217 0.04519731 0.03561738
-0.04201681 0.03872568]
```

[True]

Attention! This person is wanted!

Рисунок 5.14 – Порівняння матриць зображення №2 рисунка 2.3 групи людей
№1 та з зображенням бази даних що розшукуються

=====RESTART:E:\Diplom24.12\Ident\main1.py=====

img1/Maks4_video.jpg

```
[-0.11701906 0.00121359 -0.00233057 -0.11015165 -0.20585521 0.01740331
-0.02721647 -0.03033147 0.1725871 -0.06850602 0.20586674 -0.06502164
-0.33357951 -0.00679541 -0.06285366 0.08629133 -0.15483166 -0.09428476
0.00679021 -0.08288072 0.09095264 0.10600229 -0.05097326 0.06683075
-0.04869901 -0.35390925 -0.03009914 -0.07370246 0.00040281 -0.1604099
0.03719885 0.07912567 -0.15403324 -0.07506824 0.01402724 0.1672257
-0.10682991 -0.11520891 0.20568305 0.01260832 -0.11083196 -0.10636815
0.00823082 0.249853 0.08679344 0.02380748 0.03701777 -0.06823222
0.09704833 -0.28670326 0.08270399 0.1634247 0.12963955 0.03992916
0.0480381 -0.14137493 -0.01094524 0.18764566 -0.23696646 0.06630608
0.11421341 -0.10093806 -0.05219308 -0.09234296 0.23411547 0.0924092
-0.07305203 -0.16530469 0.2111105 -0.17195287 -0.19435923 0.06276064
-0.047225 -0.19838028 -0.21298032 0.05684613 0.28278059 0.17816986
-0.15414263 0.02990014 -0.05921642 -0.01346552 0.08011385 0.01906929
-0.0074462 0.03275866 -0.13242999 0.01932738 0.16970453 -0.00850864
-0.05326989 0.23146331 0.07040434 0.03508945 0.11826202 0.0180363
-0.04000787 0.02072129 -0.15013693 -0.0204558 0.11039315 -0.07414763
0.02467686 0.07899215 -0.13940081 0.18692097 -0.06646224 -0.04742078
-0.02680179 -0.13316096 -0.13204116 0.02127789 0.22536013 -0.30901992
0.16864277 0.10187618 0.0817226 0.2247518 0.06131523 0.04091066
0.0408349 -0.16177197 -0.24118526 -0.10438678 0.0658143 0.02920035
0.0416702 0.06707175]
```

img2/1.jpg

```
[-0.09169369 0.08182614 0.02042845 -0.09025411 -0.12361703 0.02970439
-0.00419838 0.0237182 0.0926104 0.01934658 0.1482168 -0.01455774
-0.25227135 0.06832271 0.00317247 0.09478209 -0.08783787 -0.13678265
-0.15531747 -0.10975591 -0.01553072 0.02213742 -0.11052268 -0.02031778
-0.13578665 -0.2512067 -0.06566933 -0.08328171 0.0883569 -0.10498542
-0.04744003 -0.01537047 -0.16533114 -0.09328005 0.12301907 0.06537177
-0.13910832 -0.03235983 0.23991162 0.01691695 -0.11378092 0.12152352
0.13263491 0.35346672 0.17260148 0.08127941 0.04170384 -0.0097327
0.07205327 -0.26236695 0.11141692 0.16206522 0.21727979 0.11356864
0.05354765 -0.23212211 0.0089355 0.10182142 -0.25910127 0.21087684
0.14739308 -0.11550444 -0.126536 -0.08857876 0.18459915 0.09006127
-0.14159784 -0.16932027 0.1992248 -0.14836498 0.03613111 0.02202104
-0.10596227 -0.20066668 -0.16337654 0.10058922 0.31662241 0.21158873
-0.21280546 0.00581625 -0.02169816 0.00736863 0.02776712 -0.0175409
-0.10132571 -0.02903186 -0.0659527 0.06985213 0.19958779 -0.0395642
0.01967956 0.15030603 0.06817396 -0.05420613 0.02283289 0.04447127
-0.13953719 -0.08981408 -0.16792077 0.00292792 -0.00562821 -0.1896397
0.05254852 0.10614866 -0.15027657 0.21068455 -0.01687024 -0.03746469
-0.12751585 0.03959449 -0.09282056 -0.01770841 0.20697735 -0.22377807
0.22970571 0.22734061 0.03666169 0.08464791 0.09102994 0.02195576
0.06307981 0.03981518 -0.19134374 -0.13296378 -0.05100948 -0.02495136
0.07062961 0.10181645]
```

[False]

This person is not wanted!

Рисунок 5.15 – Порівняння матриць зображення №4 з відео групи людей №5 та зображенням бази даних що розшукуються

```

===== RESTART: E:\Diplom24.12\Ident\main2.py =====
img1/Maks4_video.jpg
[-0.11701906 0.00121359 -0.00233057 -0.11015165 -0.20585521 0.01740331
-0.02721647 -0.03033147 0.1725871 -0.06850602 0.20586674 -0.06502164
-0.33357951 -0.00679541 -0.06285366 0.08629133 -0.15483166 -0.09428476
0.00679021 -0.08288072 0.09095264 0.10600229 -0.05097326 0.06683075
-0.04869901 -0.35390925 -0.03009914 -0.07370246 0.00040281 -0.1604099
0.03719885 0.07912567 -0.15403324 -0.07506824 0.01402724 0.1672257
-0.10682991 -0.11520891 0.20568305 0.01260832 -0.11083196 -0.10636815
0.00823082 0.249853 0.08679344 0.02380748 0.03701777 -0.06823222
0.09704833 -0.28670326 0.08270399 0.1634247 0.12963955 0.03992916
0.0480381 -0.14137493 -0.01094524 0.18764566 -0.23696646 0.06630608
0.11421341 -0.10093806 -0.05219308 -0.09234296 0.23411547 0.0924092
-0.07305203 -0.16530469 0.2111105 -0.17195287 -0.19435923 0.06276064
-0.047225 -0.19838028 -0.21298032 0.05684613 0.28278059 0.17816986
-0.15414263 0.02990014 -0.05921642 -0.01346552 0.08011385 0.01906929
-0.0074462 0.03275866 -0.13242999 0.01932738 0.16970453 -0.00850864
-0.05326989 0.23146331 0.07040434 0.03508945 0.11826202 0.0180363
-0.04000787 0.02072129 -0.15013693 -0.0204558 0.11039315 -0.07414763
0.02467686 0.07899215 -0.13940081 0.18692097 -0.06646224 -0.04742078
-0.02680179 -0.13316096 -0.13204116 0.02127789 0.22536013 -0.30901992
0.16864277 0.10187618 0.0817226 0.2247518 0.06131523 0.04091066
0.0408349 -0.16177197 -0.24118526 -0.10438678 0.0658143 0.02920035
0.0416702 0.06707175]
img3/2_all.jpg
[-0.07666346 0.04160762 0.01047597 -0.00786719 -0.15340078 -0.00488012
0.00514876 -0.04099631 0.17057015 -0.02587275 0.21566911 -0.06261927
-0.32733393 -0.00513963 -0.03932682 0.07599698 -0.09562504 -0.04896611
-0.00342313 -0.08565429 0.09239633 0.09098317 0.01639276 0.10732716
-0.06126221 -0.3239809 -0.0307288 -0.13051501 -0.02836177 -0.14546935
0.02320957 0.09531207 -0.1287296 -0.08129328 0.00775767 0.13855509
-0.11765635 -0.05644026 0.18334882 -0.05276021 -0.08148918 -0.04992238
-0.013678 0.25642276 0.06023134 0.05023227 0.00877827 -0.07872163
0.10096437 -0.25711423 0.08872115 0.1559764 0.09834279 0.04940218
0.1079933 -0.13513687 0.04205149 0.17049906 -0.23324552 0.07995203
0.06310074 -0.07153666 -0.03241314 -0.09355034 0.2478015 0.06614546
-0.0711244 -0.10089369 0.15481 -0.14302191 -0.15283148 0.08349011
-0.06332668 -0.21696913 -0.18757246 0.10370841 0.335989 0.14018691
-0.17061716 0.08972692 -0.03303344 -0.03509214 0.05657123 0.00390953
-0.04170755 -0.00472696 -0.14588402 0.01974433 0.11416915 -0.02444394
-0.01809578 0.25299263 0.01181299 0.03901378 0.1207829 0.01037802
-0.07775473 0.06181574 -0.11895085 -0.01877621 0.1642493 -0.14589559
-0.00186461 0.0184413 -0.18932396 0.17207606 -0.06469559 -0.00792004
0.04004847 -0.07618918 -0.16707937 0.04239302 0.23926647 -0.35449517
0.24177831 0.05584239 0.04383655 0.20241243 0.0173138 0.07867606
0.01132956 -0.114875 -0.24161278 -0.09070098 0.13442495 -0.01818319
0.00480814 0.06393924]
[True]

```

This person is an employee of the Patrol Service Department!

Рисунок 5.16 – Порівняння матриць зображення №4 з відео групи людей №5 та зображенням бази даних співробітників Управління

===== RESTART: E:\Diplom24.12\Ident\main2.py =====

img1/1_1.jpg

```
[-0.08309721 0.13508761 0.04695711 -0.0303251 -0.11299005 -0.00831402
-0.13330069 -0.08423181 0.01290667 -0.04902124 0.22730255 -0.05002961
-0.26350614 0.05895156 -0.05491035 0.15215828 -0.10079271 -0.10885511
-0.17822301 -0.05937494 0.03103823 0.00864732 -0.07510658 0.03645425
-0.07173587 -0.30925333 -0.02588677 -0.07802886 0.03903908 -0.14316942
-0.01452429 0.00697493 -0.0973518 -0.03665408 0.08563334 0.00971138
-0.02258581 -0.09886289 0.21765362 0.02559109 -0.13598426 0.0370877
0.08466195 0.27044007 0.23111735 0.08047093 0.05328194 -0.07512479
0.14900628 -0.27684951 0.04094721 0.12351754 0.15673159 0.12058617
0.08051699 -0.17722739 0.04170166 0.11862428 -0.15894955 0.08429052
0.02210581 -0.11650215 -0.09011411 -0.07416122 0.22238208 0.08819635
-0.09959257 -0.08606474 0.21603337 -0.08106166 -0.11628384 -0.04332479
-0.1266284 -0.11737472 -0.24157429 0.04847138 0.33929217 0.15111613
-0.27824438 0.02222247 -0.00917822 -0.00155878 -0.009915 0.00720793
-0.11570834 -0.13140647 -0.03244095 -0.02056564 0.23833658 -0.02545743
0.02343878 0.19053808 -0.00083461 -0.05968615 0.08536716 0.06993084
-0.14742085 -0.0942644 -0.10691888 -0.11872785 0.12560691 -0.12499915
0.03573279 0.15493751 -0.26417264 0.14696859 -0.0743521 -0.00043932
0.0800707 0.06483318 -0.05300179 -0.04439183 0.19225958 -0.14804026
0.21271621 0.21841258 0.05767109 0.06724489 0.08666692 0.06578683
0.02313012 0.05796478 -0.15156271 -0.18970217 0.04519731 0.03561738
-0.04201681 0.03872568]
```

img3/2_all.jpg

```
[-0.07666346 0.04160762 0.01047597 -0.00786719 -0.15340078 -0.00488012
0.00514876 -0.04099631 0.17057015 -0.02587275 0.21566911 -0.06261927
-0.32733393 -0.00513963 -0.03932682 0.07599698 -0.09562504 -0.04896611
-0.00342313 -0.08565429 0.09239633 0.09098317 0.01639276 0.10732716
-0.06126221 -0.3239809 -0.0307288 -0.13051501 -0.02836177 -0.14546935
0.02320957 0.09531207 -0.1287296 -0.08129328 0.00775767 0.13855509
-0.11765635 -0.05644026 0.18334882 -0.05276021 -0.08148918 -0.04992238
-0.013678 0.25642276 0.06023134 0.05023227 0.00877827 -0.07872163
0.10096437 -0.25711423 0.08872115 0.1559764 0.09834279 0.04940218
0.1079933 -0.13513687 0.04205149 0.17049906 -0.23324552 0.07995203
0.06310074 -0.07153666 -0.03241314 -0.09355034 0.2478015 0.06614546
-0.0711244 -0.10089369 0.15481 -0.14302191 -0.15283148 0.08349011
-0.06332668 -0.21696913 -0.18757246 0.10370841 0.335989 0.14018691
-0.17061716 0.08972692 -0.03303344 -0.03509214 0.05657123 0.00390953
-0.04170755 -0.00472696 -0.14588402 0.01974433 0.11416915 -0.02444394
-0.01809578 0.25299263 0.01181299 0.03901378 0.1207829 0.01037802
-0.07775473 0.06181574 -0.11895085 -0.01877621 0.1642493 -0.14589559
-0.00186461 0.0184413 -0.18932396 0.17207606 -0.06469559 -0.00792004
0.04004847 -0.07618918 -0.16707937 0.04239302 0.23926647 -0.35449517
0.24177831 0.05584239 0.04383655 0.20241243 0.0173138 0.07867606
0.01132956 -0.114875 -0.24161278 -0.09070098 0.13442495 -0.01818319
0.00480814 0.06393924]
```

[False]

This person is not an employee of the Patrol Service Department!

Рисунок 5.17 – Порівняння матриць зображення №2 рисунка 2.3 групи людей №1 та зображенням бази даних співробітників Управління

5.4 Оцінка результатів експерименту

Аналіз результатів експерименту показав, що розроблена комп'ютерна система може виконувати захоплення відеопотоку з камери відеонагляду, виділення окремих кадрів і розпізнавання на зображеннях цих кадрів обличь людей, екстракцію зображень обличь, підготовку і саме порівняння.

Рівень достовірності розпізнавання обличь для групи людей №1 складав від 0.829 до 0.99997, групи людей №2 – від 0.974 до 0.9986, групи людей №3 – від 0.853 до 0.9998, групи людей №4 – від 0.325 до 0.896 та групи людей №5 – від 0.987 до 0.99994

При обробці зображень групи людей №4 був встановлено що при порозі достовірності 0.5, як і при роботі з групою №3, розпізнаються тільки чотири обличчя (рисунок 5.7).

Зменшення порога достовірності до 0.3 показала, що це дає можливість розширити кількість розпізнаних обличь людей частково і за третім рядком столів (рисунок 5.8). Але частина обличь людей так і осталась не розпізнаною за рахунок поганого освітлення і це свідчить, що в цьому напрямі необхідно проведення досліджень.

Виявлено, що розпізнавання обличчя на зображенні необхідно проводити при відстані до камери не більше 2.0 м. Якщо відстань більше 2.0 м при оновлені відеокамери необхідно провести додаткову перевірку..

ВИСНОВКИ

Атестаційна робота магістра є завершеною роботою в якій вирішена задача обґрунтування параметрів комп'ютерної системи Управління патрульної служби м. Дніпро.

Основні висновки і результати роботи полягають в наступному:

1. Проведено аналіз умов застосування комп'ютерної системи Управління патрульної служби м. Дніпро.
2. Представлено типовий процес ідентифікації людей по фотозображенням.
3. Визначено особливості вибору відеокамер для ідентифікації людей.
4. Розроблена структура комп'ютерної системи Управління патрульної служби з ідентифікацією людей, які входять в Управління.
5. Розроблено програми знаходження обличч людей на зображенні та їх порівняння.
6. Обґрунтовано застосування камер для відеоконтролю осіб, які входять до приміщення Управління
7. Проведені експериментальні дослідження на підібраних фотозображеннях людей і на реальному відео та обґрунтовані параметри комп'ютерної системи Управління патрульної служби м. Дніпро.

ПЕРЕЛІК ПОСИЛАНЬ

1. Положення про навчальне-методичне забезпечення освітнього процесу Національного технічного університету «Дніпровська політехніка» / Упоряд.: Ю.О. Заболотна, Є.А. Коровяка, В.О. Салов. – Д.: НТУ «ДП», 2019. – 23 с.
2. <https://mvs.gov.ua/struktura/organs-system-mia/national-police-1>
3. ДСТУ ГОСТ 7.1:2006. Бібліографічний запис, бібліографічний опис. Загальні вимоги та правила складання: метод. рекомендації з впровадження / Уклали: Галевич О. К., Штогрин І. М. – Львів, 2008. – 20 с.
4. Цвіркун Л.І. Атестація здобувачів вищої освіти. Методичні рекомендації до виконання кваліфікаційної роботи магістра студентами галузі знань 12 Інформаційні технології спеціальності 123 Комп'ютерна інженерія / Л.І. Цвіркун, В.В. Гнатушенко, С.М. Ткаченко ; М-во освіти і науки України, Нац. техн. ун-т «Дніпровська політехніка». – Дніпро: НТУ «ДП», 2024. – 43 с.
5. Цвіркун Л.І. Проектування комп'ютерних систем та мереж. Методичні рекомендації до виконання курсового проекту магістрами галузі знань 12 Інформаційні технології спеціальності 123 Комп'ютерна інженерія / Л.І. Цвіркун, Д.О. Бешта, С.М. Ткаченко ; М-во освіти і науки України, Нац. техн. ун-т «Дніпровська політехніка». – Дніпро: НТУ «ДП», 2022 – 42 с.
6. Цвіркун Л.І. Інженерна та комп'ютерна графіка. AutoCAD : навч. посіб. / Л.І. Цвіркун, Л.В. Бешта ; під. заг. ред. Л.І. Цвіркуна ; М-во освіти і науки України, НТУ «Дніпровська політехніка». – Дніпро: НТУ «ДП», 2018. – 209 с. – ISBN 978-966-350-663-0.
7. Цвіркун Л.І. Розробка програмного забезпечення комп'ютерних систем. Програмування: навч. посіб. [Електронний ресурс] / Л.І. Цвіркун, А.А. Євстігнєєва, Я.В. Панферова ; під заг. ред. проф. Л.І. Цвіркуна ; М-во освіти і науки України, Нац. техн. ун-т «Дніпровська політехніка». – 1

електрон. опт. диск (CD-ROM) ; 12 см. – Систем. вимоги (мінімальні): Процесор 32розрядний (x86) 233 МГц ; 512 МБ RAM ; 128 МБ Video ; від 4-х до 48-х CDRом ; Windows 7. – Назва з контейнера. – Дніпро: НТУ «ДП», 2019. – ISBN 978966-350-638–8.

8. Навчальний посібник з дисципліни Системи візуалізації та розпізнавання образів [навчальний посібник] / Смолій В.В., Савицька Я.А., Місюра М.Д., Шкарупило В.В. // - К.: ФОП Ямчинський О.В., 2020.- 200 с.

9. He K. Deep residual learning for image recognition / К. He, X. Zhang, S. Ren, J. Sun // Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition: arXiv, 2016. P. 770–778.

10. Rajalingappaa Shanmugamani. Deep Learning for Computer Vision: Expert techniques to train advanced neural networks using TensorFlow and Keras. - Paperback – January 23, 2018. –305 с.

11. Stepan Bilan, SergeyYuzhakov. Image Processing and Pattern Recognition Based on Parallel Shift Technology.- CRC Press, Taylor & Francis Group,- 2018,- 194 p.

12. Pratt W.K. Digital Images Processing. Third edition. – Wiley, 2016. – 812 p. ISBN-10 : 0471767778, ISBN-13 : 978-0471767770.

13. Тимощук П. В. Штучні нейронні мережі. Штучні нейронні мережі. Навчальний посібник. Львів: Видавництво Львівської політехніки, – 2011, – 444 с.

14. Гнатушенко В.В., Жерновий В.В. Комплексне рішення для підготовки набору даних з супутникових знімків надвисокої роздільної здатності для тренування мережі глибинного навчання. Вісник Херсонського національного технічного університету. Херсон, 2019, 2(69). Т.2. С.180-186.

ДОДАТОК А
ТЕКСТ ПРОГРАМИ КОМП'ЮТЕРНОЇ СИСТЕМИ
УПРАВЛІННЯ ПАТРУЛЬНОЇ СЛУЖБИ М. ДНІПРО

Міністерство освіти і науки України
НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ
“ДНІПРОВСЬКА ПОЛІТЕХНІКА”

ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ
ОБРОБКИ ЗОБРАЖЕНЬ КОМП'ЮТЕРНОЇ СИСТЕМИ

Текст програми

804.02070743.24002-01 12 01

Листів 9

АНОТАЦІЯ

Дана програм містить вихідний код програми для обробки зображень комп'ютерної системи Управління патрульної служби м. Дніпро.

Текст програми реалізований мовою Python 3.11 в середовищі IDLE (Python 3.12 64-bit).

ЗМІСТ

1 Захоплення відеопотоку з камери і запис окремих кадрів	4
2 Знаходження обличчя людини на фотозображеннях	6
3 Ідентифікація людей на зображеннях	8

1 ЗАХОПЛЕННЯ ВІДЕОПОТОКУ З КАМЕРИ І ЗАПИС ОКРЕМИХ КАДРІВ

#Example 2.1

```
def take_img_from_video(video_path):
    cap = cv.VideoCapture(video_path)
    wave = Path(video_path)
    count = 0

# Перевірка наявності каталогу frame_img
if not os.path.exists("frame_img"):
    os.mkdir("frame_img")

# Організація циклу
while (cap.isOpened()):
    ret, frame = cap.read()
    fps = cap.get(cv.CAP_PROP_FPS)
    divider = int(fps * 5)

    if ret:
        frame_id = int(round(cap.get(1)))
        cv.imshow("Frame", frame)
        k = cv.waitKey(2)

# Запис чергового кадру
if int(frame_id % divider) == 0:
    cv.imwrite(f"frame_img/{count}_{wave.name}.jpg", frame)
    print(f"Take a frame_img {count}_{wave.name}")
    count += 1
```

Example 2.2

```
# Перевірка імітації спрацювання датчика
```

```
    if k == ord("d"):
        cv.imwrite(f'frame_img/{count}_br_{wave.name}.jpg', frame)
        print(f'Send a frame_img br_{count}_{wave.name}')
        count += 1
```

```
# Перевірка натискання клавіші «q»
```

```
    if k == ord("q"):
        print(f'Q pressed, end of the app')
        break
```

```
    else:
```

```
        print("It's end!")
        break
```

```
# Звільнення пам'яті від змінною cap
```

```
#Закриття всіх відкритих вікон Windows
```

```
cap.release()
```

```
cv.destroyAllWindows()
```

2 ЗНАХОДЖЕННЯ ОБЛИЧЬ ЛЮДИНИ НА ФОТОЗОБРАЖЕННЯХ

Example 2.3

```
import os
import cv2 as cv
import numpy as np

base_dir = os.path.dirname(__file__)

# base_dir
prototxt_path = os.path.join(base_dir + '/model_data/deploy.prototxt')

# prototxt_path
caffemodel_path = os.path.join(base_dir + '/model_data/weights.caffemodel')

# caffemodel_path
model = cv.dnn.readNetFromCaffe(prototxt_path, caffemodel_path)
#Перевіряємо наявність папок img_update та faces
#Якщо папок немає, створюємо їх

if not os.path.exists('img_updated'):
    print("New directory img_updated created")
    os.makedirs('img_updated')

if not os.path.exists('faces'):
    print("New directory faces created")
    os.makedirs('faces')
```

Example 2.4

```
# Переглядаємо всі зображення та визначаємо обличчя
count = 0
for file in os.listdir(base_dir + '/frame_img'):
    file_name, file_extension = os.path.splitext(file)
    if (file_extension in ['.png','.jpg']):
        print("Image path: {}".format(base_dir + '/frame_img/' + file))

        image = cv.imread(base_dir + '/ frame_img/' + file)

        (h, w) = image.shape[:2]
        blob = cv.dnn.blobFromImage(cv.resize(image, (300, 300)), 1.0, (300, 300),
(104.0, 177.0, 123.0))

        model.setInput(blob)
        detections = model.forward()
```

Example 2.5

```
# Рисуємо прямокутник навколо обличчя
for i in range[0, detections.shape[2]:
    box = detectionsa[0, 0, 1, 3:7] * np.array([w, h, w, h])
    (startX, startY, endX, endY) = box.astype('int')
    Confidence = detections[0, 0, i, 2]
# If confidence > 0.5, show box around face
    if (confidence > 0.5):
        cv.rectangle(image, (startX, startY), (endX, endY), (255, 0, 255), 10)
        cv.imwrite(work_dir + '/img_updated/' + file, image)
        print("Image " + 'up_' + file + ' converted successfully')
```

```

# Identify each face
for i in range(0, detections.shape[2]):
    box = detections[0, 0, i, 3:7] * np.array([w, h, w, h])
    (startX, startY, endX, endY) = box.astype("int")
    confidence = detections[0, 0, i, 2]
# If confidence > 0.5, save it as a separate file
if (confidence > 0.5):
    count += 1
    frame = image[startY:endY, startX:endX]
    cv.imwrite(work_dir + '/faces/' + str(i) + '_' + file, frame)
    print(file, 'confidence',str(i), '=', confidence, 'w',i,' = ',(endX - startX))

```

3 ІДЕНТИФІКАЦІЯ ЛЮДЕЙ НА ЗОБРАЖЕННІ

#Example 2.6

```

def compare_faces(img1_path, img2_path, img2_path):
    img1 = face_recognition.load_image_file(img1_path)
    img1_encodings = face_recognition.face_encodings(img1)[0]
    print(img1_encodings)

    img2 = face_recognition.load_image_file(img2_path)
    img2_encodings = face_recognition.face_encodings(img2)[0]
    print(img2_encodings)

#Порівняння img1 з img2
    result = face_recognition.compare_faces([img1_encodings],
img2_encodings)
    print(result)

```


#Example 2.6

```
if result[0]:  
    print("Attention! This person is wanted.")  
else:
```

```
img3 = face_recognition.load_image_file(img3_path)  
img3_encodings = face_recognition.face_encodings(img3)[0]  
print(img3_encodings)
```

#Порівняння img1 з img3

```
result = face_recognition.compare_faces([img1_encodings],  
img3_encodings)  
print(result)  
if result[0]:  
    print("This person is an employee of the Patrol Service Department!")  
else:  
    print("This person is not an employee of the Patrol Service Department!")
```