

Центральноукраїнський національний технічний університет
Механіко-технологічний факультет
Кафедра кібербезпеки та програмного забезпечення

”Допущено до захисту”
Завідувач кафедри кібербезпеки
та програмного забезпечення
д.т.н., професор
_____ Олексій СМІРНОВ
“ ____ ” _____ 2024 р.

ВИПУСКНА КВАЛІФІКАЦІЙНА РОБОТА
за другим (магістерським) рівнем вищої освіти
на тему
**“Дослідження та програмна реалізація системи збору,
представлення, обробки, зберігання, передачі та доступу до
інформації мережевої відеоаналітики”**

Виконав здобувач вищої освіти
II курсу, групи КН-23М
ОПП «Комп’ютерні науки»
спеціальності 122 «Комп’ютерні науки»
_____ Топчій В.А.
« ____ » _____ 2024 р.

Керівник проекту
кандидат технічних наук
_____ Буравченко К.О.
« ____ » _____ 2024 р.
Рецензент _____

Центральноукраїнський національний технічний університет
Факультет Механіко-технологічний
Кафедра Кібербезпеки та програмного забезпечення
Рівень вищої освіти магістр
Галузь знань 12 "Інформаційні технології"
Спеціальність 122 "Комп'ютерні науки"
Освітньо-професійна (освітньо-наукова) програма "Комп'ютерні науки"

ЗАТВЕРДЖУЮ

Завідувач кафедри

д.т.н., проф.

Олексій СМІРНОВ

« 6 » вересня 2024 року

ЗАВДАННЯ НА ВИПУСКНУ КВАЛІФІКАЦІЙНУ РОБОТУ ЗА ДРУГИМ (МАГІСТЕРСЬКИМ) РІВНЕМ ВИЩОЇ ОСВІТИ ЗДОБУВАЧА ВИЩОЇ ОСВІТИ

Топчію Вілену Анатолійовичу

(прізвище, ім'я, по батькові)

1. Тема роботи Дослідження та програмна реалізація системи збору, представлення, обробки, зберігання, передачі та доступу до інформації мережевої відеоаналітики

2. Керівник роботи Буравченко Костянтин Олегович, канд. техн. наук

(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом вищого навчального закладу № 18-13 від 07.08.2024 року

3. Строк подання студентом роботи до захисту 2.12.2024 р.

4. Мета та завдання випускної кваліфікаційної роботи: Метою розробки є дослідження та програмна реалізація системи збору, представлення, обробки, зберігання, передачі та доступу до інформації мережевої відеоаналітики

5. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити)

1. Призначення та область використання.

6. Наукова новизна.

2. Перегляд аналогічних існуючих систем.

7. Маркетингове та економічне обґрунтування ІТ-проєкту.

3. Опис і обґрунтування проектних рішень.

8. Заходи з охорони праці та техніки безпеки.

4. Етапи програмування системи.

9. Висновки.

5. Впровадження системи в промислову експлуатацію

6. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень)

Наукова новизна 1 аркуш

Структурна схема системи 1 аркуш

Функціональна схема системи 1 аркуш

Діаграма процесів 1 аркуш

Блок-схема алгоритму роботи додатку 2 аркуша

Показники економічної ефективності 1 аркуш

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Економічний	Доренська А.О.	05.10.2024	14.11.2024
Охорона праці	Марченко К.М., к.т.н., доцент	06.10.2024	16.11.2024

7. Дата видачі завдання « 6 » вересня 2024 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти	Строк виконання етапів випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти	Примітка
1.	Аналіз існуючих систем	10.10.2024 р.	
2.	Постановка задачі, оформлення ТЗ	15.10.2024 р.	
3.	Розробка моделі компонента	20.10.2024 р.	
4.	Розробка структур даних	25.10.2024 р.	
5.	Розробка алгоритмів зв'язку та відображення	30.10.2024 р.	
6.	Програмування алгоритмів	10.11.2024 р.	
7.	Розрахунок економічної ефективності	13.11.2024 р.	
8.	Розрахунки з охорони праці та техніки безпеки	15.11.2024 р.	
9.	Оформлення ПЗ	17.11.2024 р.	
10.	Попередній захист роботи	2.12.2024 р.	

Дата видачі завдання
« 6 » вересня 2024 р.

Підпис керівника

(прізвище та ініціали)Завдання прийнято до виконання
« 6 » вересня 2024 р.

Підпис здобувача

(прізвище та ініціали)

АНОТАЦІЯ

Топчій В.А. Дослідження та програмна реалізація системи збору, представлення, обробки, зберігання, передачі та доступу до інформації мережевої відеоаналітики. 122 Комп'ютерні науки. Центральноукраїнський національний технічний університет. Кропивницький. 2024.

В даній випускній кваліфікаційній роботі за другим (магістерським) рівнем вищої освіти розроблено програмне забезпечення, яке призначено для системи збору, представлення, обробки, зберігання, передачі та доступу до інформації мережевої відеоаналітики.

Метою розробки є дослідження та програмна реалізація системи збору, представлення, обробки, зберігання, передачі та доступу до інформації мережевої відеоаналітики.

Об'єктом дослідження є процес збору, представлення, обробки, зберігання, передачі та доступу до інформації мережевої відеоаналітики.

Предметом дослідження є методи збору, представлення, обробки, зберігання, передачі та доступу до інформації мережевої відеоаналітики.

Методи дослідження базуються на методах обробки великих даних, методах математичної статистики, методах розробки програмного забезпечення.

Результат роботи – програмна реалізація системи збору, представлення, обробки, зберігання, передачі та доступу до інформації мережевої відеоаналітики.

В процесі роботи над програмною моделлю виконано аналіз існуючих апаратних та програмних засобів. В повній мірі описані всі компоненти розробленого програмного забезпечення.

Розроблено зручний інтерфейс користувача. Наведені інструкції по роботі з програмними засобами.

Програма може використовуватися на ПЕОМ з ОС Windows 10/11.

Програму розроблено в середовищі Python.

Ключові слова: Комп'ютерні науки, мережева відеоаналітика

ABSTRACT

Topchiy V.A. Research and software implementation of a system for collecting, presenting, processing, storing, transmitting and accessing network video analytics information. 122 Computer Science. Central Ukrainian National Technical University. Kropyvnytskyi. 2024.

In this graduation thesis for the second (master's) level of higher education, software is developed, which is intended for the system of collection, presentation, processing, storage, transmission and access to network video analytics information.

The purpose of the development is research and software implementation of a system for collecting, presenting, processing, storing, transmitting and accessing network video analytics information.

The object of research is the process of collecting, presenting, processing, storing, transmitting and accessing network video analytics information.

The subject of research is the methods of collecting, presenting, processing, storing, transmitting and accessing network video analytics information.

Research methods are based on big data processing methods, mathematical statistics methods, and software development methods.

The result of the work is a software implementation of a system for collecting, presenting, processing, storing, transmitting and accessing network video analytics information.

In the process of working on the software model, an analysis of existing hardware and software was performed. All components of the developed software are fully described.

A convenient user interface has been developed. Instructions for working with software tools are provided.

The program can be used on a PC with Windows OS 10/11.

The program was developed in the Python environment.

Keywords: Computer science, network video analytics

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ І ТЕРМІНІВ	3
ВСТУП.....	4
1 ПРИЗНАЧЕННЯ ТА ОБЛАСТЬ ВИКОРИСТАННЯ	7
1.1 Призначення системи.....	7
1.2 Область застосування.....	8
2 ПЕРЕГЛЯД АНАЛОГІЧНИХ ІСНУЮЧИХ СИСТЕМ	14
2.1 Огляд існуючих систем, технологій, архітектур та програмних рішень за профілем теми випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти.....	14
2.2 Обґрунтування вибору засобів для побудови системи та мови програмування.....	23
2.3 Розгорнута постановка завдання	26
3 ОПИС І ОБҐРУНТУВАННЯ ПРОЕКТНИХ РІШЕНЬ	27
3.1 Опис функціонування системи	27
3.2 Розробка структурної схеми.....	39
3.3 Розробка функціональної схеми	51
3.4 Розробка діаграми процесів.....	52
4 РЕАЛІЗАЦІЯ РОБОТИ. РОЗРАХУНКИ І ЕКСПЕРИМЕНТАЛЬНІ ДАНІ, ЩО ПІДТВЕРДЖУЮТЬ ВІРНІСТЬ ПРОЕКТНИХ ТА ПРОГРАМНИХ РІШЕНЬ.....	55
4.1 Розробка блок-схем та опис алгоритмів функціонування системи.....	55
4.2 Захист розробленого програмного забезпечення.....	68
5 ВПРОВАДЖЕННЯ СИСТЕМИ В ПРОМИСЛОВУ ЕКСПЛУАТАЦІЮ	74
6 НАУКОВА НОВИЗНА	78

					ВКРМ-122.24.0014.00.00.ПЗ			
Вим	Арк.	№ докум.	Підп.	Дата	Дослідження та програмна реалізація системи збору, представлення, обробки, зберігання, передачі та доступу до інформації мережевої відеоаналітики	Літ.	Аркуш	Аркушів
Розроб.	Топчій В.А.					М	1	105
Перев.	Буравченко К.О.					ЦНТУ КН-23М		
Н.контр.	Коваленко А.С.							
Затв.	Смірнов О.А.							

7	МАРКЕТИНГОВЕ ТА ЕКОНОМІЧНЕ ОБҐРУНТУВАННЯ ІТ-ПРОЄКТУ	79
7.1	Визначення цільової аудиторії кінцевого готового продукту	79
7.2	Оцінка привабливості шляхом застосування методів експертних оцінок ...	80
7.3	Вибір методу оцінки вартості ПЗ	82
7.4	Розрахунок економічної ефективності від впровадження реалізованого ПЗ як фактору його привабливості.....	83
7.5	Пропозиція алгоритму просування проєкту розробки ПЗ	85
7.6	Оптимізація каналів збуту та шляхів реалізації ПЗ	85
7.7	Визначення ключових факторів успіху конкретного проєкту.....	87
8	ЗАХОДИ З ОХОРОНИ ПРАЦІ ТА ТЕХНІКИ БЕЗПЕКИ	88
8.1	Вступ.....	88
8.2	Аналіз санітарно-гігієнічних умов праці на робочому місці програміста ...	90
8.3	Розробка заходів з умов поліпшення охорони праці.....	93
8.4	Розрахункова частина	94
8.5	Висновки до розділу.....	96
9	ОСНОВНІ ВИСНОВКИ.....	97
	СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	99

					ВКРМ-122.24.0014.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		2

ВСТУП

Актуальність теми. Аналітики очікують, що в найближчі роки середньорічні темпи росту світового ринку ІР-відеоспостереження перевищать 20%, причому сегмент послуг відеоспостереження й відеоаналітики буде розвиватися випереджальними темпами. В Україні основними користувачами технологій відеоспостереження й відеоаналітики є організації державного сектора, банки, а також підприємства зі сфер транспорту, роздрібно́ї торгівлі й керування нерухомістю. В інших країнах спостерігається схожа картина.

У числі основних тенденцій цього ринку – «інтелектуальне» відеоспостереження, розвиток можливостей убудованої й серверної відеоаналітики. Причому майбутнє – за інтеграцією й тих і інших засобів, за широким поширенням «розумних» камер і Інтернету речей, хмарних технологій відеоаналітики, у тому числі в реальному часі, а також за обробкою Больших Дані відеоспостереження.

Убудована відеоаналітика – оснащення камер аналітичними функціями – допомагає постійно контролювати ситуацію й швидко приймати рішення. А завдяки зниженим вимогам до каналів зв'язку «розумну» камеру можна встановлювати в таких місцях, де звичайна ІР-камера працювати не буде. Більше того, при необхідності замість відео вона просто відправляє повідомлення про інцидент, що вимагає реагування.

Попит на подібні рішення вже існує, а технічні можливості дозволяють реалізувати «на борті» камер розвинені аналітичні функції. Явних лідерів ринку поки немає, і у вендорів є гарні шанси зайняти своє місце в цьому сегменті, що розвивається.

Мета й завдання дослідження. Метою роботи є дослідження та програмна реалізація системи збору, представлення, обробки, зберігання, передачі та доступу до інформації мережевої відеоаналітики.

					ВКРМ-122.24.0014.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		4

Для досягнення поставленої мети визначена програма дослідження, що складається з наступних завдань:

– Огляд існуючих систем збору, представлення, обробки, зберігання, передачі та доступу до інформації мережевої відеоаналітики.

– Дослідження системи збору, представлення, обробки, зберігання, передачі та доступу до інформації мережевої відеоаналітики.

– Програмна реалізація системи збору, представлення, обробки, зберігання, передачі та доступу до інформації мережевої відеоаналітики.

Об'єктом дослідження є процес збору, представлення, обробки, зберігання, передачі та доступу до інформації мережевої відеоаналітики.

Предметом дослідження є методи збору, представлення, обробки, зберігання, передачі та доступу до інформації мережевої відеоаналітики.

Методи дослідження базуються на методах обробки великих даних, методах математичної статистики, методах розробки програмного забезпечення.

Наукова новизна отриманих результатів. У процесі рішення завдань, обумовлених цілями дослідження, отримані наступні результати:

– Удосконалено метод збору, представлення, обробки, зберігання, передачі та доступу до інформації мережевої відеоаналітики.

– Розроблено вітчизняний продукт збору, представлення, обробки, зберігання, передачі та доступу до інформації мережевої відеоаналітики, який має більш широкі можливості, на відміну від існуючих аналогів.

Практична цінність отриманих результатів полягає в тому, що розроблені алгоритми дозволяють успішно вирішувати задачі збору, представлення, обробки, зберігання, передачі та доступу до інформації мережевої відеоаналітики.

Достовірність наукових результатів підтверджена теоретичними викладеннями, даними комп'ютерного моделювання, коректними дослідженнями параметрів на функціонуючій обчислювальній мережі, а також відповідністю отриманих результатів окремим результатам, наведеним у науковій літературі.

					ВКРМ-122.24.0014.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		5

Робота апробована на LVII Науково-технічній конференції здобувачів вищої освіти LV науково-технічної конференції «Наука в ЦНТУ: основні досягнення та перспективи розвитку» (2024 р.), основні положення випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти надруковані у статті збірника праць молодих науковців ЦНТУ, випуск №15.

Таким чином, виходячи з вищеперерахованого, дослідження та програмна реалізація системи збору, представлення, обробки, зберігання, передачі та доступу до інформації мережевої відеоаналітики, є актуальною задачею, яка потребує вирішення у даній випускній кваліфікаційній роботі за другим (магістерським) рівнем вищої освіти.

КБПЗ_2024

					ВКРМ-122.24.0014.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		6

1 ПРИЗНАЧЕННЯ ТА ОБЛАСТЬ ВИКОРИСТАННЯ

1.1 Призначення системи

Камера стає справжнім комп'ютером і оснащується убудованим інтелектом. Підвищення продуктивності процесорів дозволяє поліпшити якість зображення й впровадити в камеру більше розвинені функції відеоаналітики. Остання обставина дає можливість здійснювати попередню обробку відеоінформації й тим самим розвантажувати мережа передачі даних. При цьому для базових функцій відеоаналітики досить процесорних потужностей окремого пристрою, а більш складна обробка повинна виконуватися на сервері й навряд чи буде реалізована «на борті» камер у доступному для огляду майбутньому.

Тим часом міняється логіка використання систем відеоспостереження, і в першу чергу камер. Якщо раніше ставилося завдання запису відео і його збереження на диску для наступного розбору, то тепер можна попереджати події. Це вимагає вдосконалювання програмного забезпечення з метою підвищення ефективності роботи оператора системи відеоспостереження.

Збільшення розрішення IP-камер, потужності процесорів і ємності убудованої пам'яті дозволяє наділяти камери усе більше зробленими аналітичними функціями. відеоспостереження, Що Відбуваються на ринку, структурні зміни характеризуються реалізацією в системах середнього й навіть початкового рівня усе більше розвинених функцій відеоаналітики, що раніше застосовувалися лише в дорогих інтеграційних проектах.

ПЗ систем відеоспостереження оптимізується під нові процесори – у ньому використовуються нові набори інструкцій. Така оптимізація дозволяє прискорити обробку відеоданих у десятки разів, що робить серверну відеоаналітику істотно більше привабливою, чим раніше, за рахунок більшої гнучкості й кращої масштабованості. Стає більше затребуваною й «відеоаналітика

					ВКРМ-122.24.0014.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		7

як сервіс». Приклади – фіксація порушень ПДР, підрахунок відвідувачів магазинів, пошук в архіві й т.д. Ця технологія поступово стає масовою, перетворюється у важливий інструмент системи відеоспостереження».

Нові можливості надає й серверна віртуалізація. У цьому випадку камери прив'язані не до фізичного сервера, а до віртуальної машини, причому навантаження балансується автоматично. Спрощується й адміністрування: замість індивідуального налаштування кожного фізичного сервера централізовано настроюється логічна група серверів («кластер» віртуальних машин). Ліцензується таке рішення по числу камер. Основні переваги даного підходу – ефективне використання встаткування й оптимальний розподіл навантаження.

Кластерна архітектура дає приблизно 30-процентну економію за рахунок більше повного завантаження встаткування, надання простих процедур налаштування, підтримки й резервування. Крім того, відкриваються можливості для впровадження відеоаналітики операторського рівня й створення сервісів хмарної відеоаналітики

1.2 Область застосування

Технології відеоаналітики часом занадто дороги для кінцевих споживачів, але вони можуть стати тою «додатковою цінністю», що дозволить операторам зв'язку й провайдерам створити вигідне для споживачів рішення.

Чим хмарна відеоаналітика цікава операторам? Налаштування алгоритмів відеоаналітики вимагає експертизи, який клієнти звичайно не володіють. Потрібно знати, які існують обмеження в застосуванні, про що багато хто не мають подання. До того ж вартість продуктів відеоаналітики занадто високий, щоб використовувати їх для рішення разових корпоративних завдань. Тому на ринку B2B и B2G оператори за допомогою відеоаналітики можуть запропонувати компаніям нові можливості, що сприяють розвитку їхнього бізнесу.

					ВКРМ-122.24.0014.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		8

Наприклад, великому підприємству роздрібної торгівлі оператор може надати наступний набір послуг, реалізованих за допомогою масштабованої відеоплатформи й модулів, що підключаються, відеоаналітики:

- відеоконтроль номера транспорту, що розвантажується в дебаркадері, і прив'язка цього відеоджерела до бази даних логістичного центра;
- відеоконтроль ваг, зважування товару в дебаркадері й зв'язок з базою даних логістичного центра;
- відеоконтроль приміщення головного касира, де запис перерахування грошей звіряється з інформацією від лічильника купюр;
- аналіз наповнення полиць товаром;
- контроль теплових зон переміщення покупців із прив'язкою до бази дані підприємства, тому що місце розташування полиць у торговельному залі і їхнє наповнення товаром постійно міняється;
- відеоконтроль кас із фіксацією типових порушень;
- підрахунок відвідувачів на вході в торговельний зал і порівняння з минаючої мимо потоком людей.

Обраний підхід до розробки інтерфейсів для різних алгоритмів відеоаналітики дозволить надати корпоративним клієнтам і замовникам з держсектора нову функціональність за розумними цінами, щоб вони змогли одержати реальну вигоду від використання хмарних додатків відеоспостереження й відеоаналітики.

Серверна відеоаналітика не вимагає обов'язкового використання дорогих апаратних рішень. Наприклад, компактна (3U) модульна серверна платформа на вісьмох процесорах Intel E3 V3 або IBM Power8 вартістю близько 15 тис. доларів здатні обробляти 480 каналів HD. Собівартість такої системи становить приблизно 30 доларів на камеру. Як правило, розробка нових аналітичних функцій на базі стандартних серверів – більше швидкий, простий і менш дорогий процес, чим розширення функціональності мережних камер, у яких нерідко застосовуються спеціалізовані апаратні платформи.

					ВКРМ-122.24.0014.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		9

Сучасна відеоаналітика радикально відрізняється від того, що пропонувалося всього три роки тому. Це продукти комерційного рівня, що дуже ефективно вирішують певний клас завдань. Для чітко формалізуємих завдань, таких як підрахунок відвідувачів або розпізнавання номерного знака автомобіля, оптимальні убудовані засоби аналітики, але завжди найдуться сегменти, де вони не потрібні. Наприклад, у масштабних системах міського відеоспостереження, на транспорті або віддалених об'єктах камери нерідко вже встановлені, і передані ними відеопотоки потрібно обробляти на сервері.

Крім того, убудована аналітика – негнучке рішення. Її важко адаптувати під потреби замовника, оскільки для цього часто необхідно реалізувати нові інтелектуальні функції, що веде до чималих витрат. До того ж найчастіше весь відеопотік, що надходить із камери, потрібно зберігати в архіві, тому має сенс аналізувати його на сервері. При використанні сучасних серверних процесорів вартість відеоаналітики розраховуючи на один канал виявляється нижче, ніж у випадку обробки даних за допомогою процесорів у камерах. Проте в убудованій відеоаналітики велике майбутнє.

Далеко не всі замовники розуміють, наскільки важлива й корисна відеоаналітика. У системах відеоспостереження звичайно обмежуються записом відео в архів, знайти в якому потрібний фрагмент найчастіше непросто. Тим часом ці технології стрімко розвиваються й уже дозволяють вирішувати найрізноманітніші завдання.

Впровадження відеоаналітики вимагає додаткових серверів, що приводить до подорожчання рішення. Убудована відеоаналітика допоможе зробити її більше доступною. У цьому випадку разом з потоком відеоданих будуть передаватися метадані – опису сцен (пропрієтарні або стандарту ONVIF). Це відкриває нові можливості – зокрема, дозволяє виконувати пошук у відеоархіві по вмісту, а не тільки за часом (наприклад, по типі, кольору об'єкта, напрямку його руху, кількості об'єктів у зоні спостереження й т.д.). Ще одна корисна функція – автозум, коли камера автоматично збільшує зображення об'єкта, що рухається.

					ВКРМ-122.24.0014.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		10

А завдяки автотрекінгу в поворотній камері пари пристроїв буде досить, щоб контролювати великий простір.

На відміну від сервера, на який надходить безліч відеопотоків, процесору камери доводиться обробляти всього один потік, причому це незжате відео будь-якого розрішення – до 4/8К, що не тільки підвищує якість аналітики, але й виключає етап декодування відео на сервері, необхідний для її застосування. Убудована відеоаналітика дозволяє також значно знизити вартість мережної інфраструктури й систем зберігання, а виходить, скоротити бюджет проекту відеоспостереження, адже можна не транслювати весь відеопотік, а обмежитися передачею тільки значимих подій, виявлених за допомогою камери.

Багато які вендори, включаючи Axis, Samsung, Panasonic, пропонують так звані відкриті платформи, що дозволяє стороннім розроблявачам доповнювати камери новими інтелектуальними функціями. Я впевнений: у підсумку цим шляхом підуть всі розроблявачі, і камери стануть оснащуватися безліччю різноманітних аналітичних функцій, спрямованих на рішення самого широкого спектра завдань. Зараз є все, щоб відеоаналітика стали масовою й доступною: потужні процесори, відкриті платформи й велика кількість розроблявачів. Сьогодні убудована відеоаналітика застосовується досить рідко, але згодом частка таких інсталяцій може вирости до 20-30% і навіть до 50%.

Деякі камери з убудованою відеоаналітикою мають функції, що дозволяють виключити сервер із системи відеоспостереження й записувати відео безпосередньо на мережні системи зберігання. Цікаві можливості відкриваються при виконанні аналізу відеопотоків і/або метаданих відразу з декількох камер, але для цього буде потрібно серверна аналітика, за допомогою якої можна реалізувати алгоритми, які поки не можна вмонтувати в програмне забезпечення камери.

Як упоратися з потоком подій від модулів відеоаналітики в реальних умовах? Журнал подій містить різноманітну інформацію, але в ній складно орієнтуватися через безліч надлишкових даних. Вихід – у фільтрації й

					ВКРМ-122.24.0014.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		11

ранжируванні інформації журналу подій. Система повинна автоматично групувати події, виключати малозначимі й виділяти найбільш важливі.

Моніторинг тривог можна зробити більше зручним: перша подія виводиться на весь екран, а якщо нові «тривоги» з'являються до завершення перегляду, то відбувається перемикання з повноекраного режиму на мультіканальну сітку й оператор бачить відразу кілька тривожних подій. Крім того, система може постачати відео метаданими, які допомагають йому краще зрозуміти що відбувається.

Варто пам'ятати, що засоби відеоаналітики розраховані на певні умови застосування. Наприклад, пошук об'єктів найбільш ефективний, коли вони переміщуються по одному, не перекриваються, повністю попадають у кадр, а фонове зображення статично. Реальна ситуація може сильно відрізнятись від ідеальної. Завдання не слід намагатися вирішити точно: у відповідь на запит користувача система, подібно інтернет-розвідувачам, повинна запропонувати свій варіант.

Як відомо, приблизно через 20 хв (а при поганій якості зображення навіть раніше) увага оператора системи відеоспостереження значно знижується. Подвоєна кількість камер погіршує цей показник у два рази. Відеоаналітика, дозволяючи зробити систему відеоспостереження проактивною, не повинна генерувати велику кількість помилкових спрацьовувань. Вона покликана виявляти події й запобігати небажані наслідки. Однак не всі аналітичні технології працюють однаково.

Відеосистема з виявленням руху «обчислює відмінності між кадрами» – в увагу приймаються всі відмінності. У камері з розширеними функціями передбачається створення «моделей фонів» і враховуються будь-які зміни в моделі тла. У відеосистемі зі створенням «шаблонів руху» виявляються відхилення від шаблону руху відомого об'єкта. Подібні системи корисні для перевіреного й нерухливого середовища, але вони не підходять для динамічних

					ВКРМ-122.24.0014.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		12

умов (дощ, хитання дерев, пересування тварин і інші не значення, що мають, дії), оскільки частка помилкових сигналів виявляється надмірною.

У визначенні шаблону руху об'єкта враховуються не тільки його розмір і рух, але й форма, що зменшує число помилкових спрацьовувань. Завдяки точній класифікації на підставі даних про рух об'єктів і застосуванню камер Full HD збільшується якість детектування об'єктів і подій, а отже – підвищується ефективність системи. Камери високої чіткості забезпечують у два рази більшу дальність спостереження й у три рази більше широку зону огляду в порівнянні з будь-яким рішенням, що включає в себе камери SD.

У відеоаналітики поки залишається чимало обмежень, але з'являється усе більше потужних і гнучких систем, що дозволяють вирішувати різні завдання. Вони орієнтовані на різні сегменти ринку й можуть не тільки забезпечувати безпека, але й швидко витягати необхідну інформацію з відеоархівів, сприяючи підвищенню ефективності роботи організацій. З удосконалюванням аналітичних алгоритмів і збільшенням обчислювальних потужностей процесорів, застосовуваних у серверах і камерах відеоспостереження, можливості таких інструментів значно розширяться, а відеоаналітика стане більше доступною й зручною в застосуванні.

Таким чином, виходячи з вищеперерахованого, дослідження та програмна реалізація системи збору, представлення, обробки, зберігання, передачі та доступу до інформації мережевої відеоаналітики, є актуальною задачею, яка потребує вирішення у даній випускній кваліфікаційній роботі за другим (магістерським) рівнем вищої освіти.

					ВКРМ-122.24.0014.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		13

при подвоєнні кількості відеовходів не відбувається. Використовувані в даній системі відеоспостереження камери є аналоговими. Даний тип камер перетворює відеосигнал у формат, що може бути отриманий телевізійним або іншим приймачем, наприклад охоронним монітором. У той же час система відеоспостереження є цифровий, тому що отримані дані зберігаються на цифровий носій. Продуктивність системи відеоспостереження практично повністю залежить від продуктивності процесора. Показником продуктивності процесора є тактова частота й архітектура. Частота процесора, застосовуваного у відеореєстраторі, звичайно коливається між 500 МГц і 1000 МГц, що обумовлено складністю відводу тепла. У даній системі всі ресурси спрямовані на обробку відеосигналу й не використовуються сторонніми програмами, тому такий, на перший погляд не високої частоти процесора, цілком достатньо.

Відеореєстратор підключається до звичайного телевізора через відеовхід VIDEO-IN, так само його називають «тюльпан» або RCA, або комп'ютерному монітору через VGA.

Програмна частина

Як операційна система у відеореєстраторах використовуються різні клони системи Linux, часто розроблені спеціально для даного пристрою.

Фахівці однозначно визначають операційну систему Linux, як найбільш стабільну при довгостроковій безперервній роботі й стійку до зовнішніх впливів – віруси, піратські зломи й т.п.

Відеореєстратори дають користувачеві обмежений доступ до програмної частини, наприклад, ви не можете встановити нові програми або драйвера, видалити старі – тільки виконання тих команд, які передбачені в меню, як при роботі з будь-яким DVD-плеєром або телевізором. Це є запорукою безперебійної роботи й більше високої надійності системи.

Керування системою

Управляється відеореєстратор за допомогою графічного меню, відображуваного на екрані телевізора, до якого підключена система. Робота в

					ВКРМ-122.24.0014.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		15

меню виробляється кнопками лицьової панелі відеореєстратора й пультом ДУ й повністю аналогічна керуванню телевизором або DVD плеєром. Багато відеореєстраторів так само мають можливість керування звичної всім комп'ютерною мишею.

Інтеграція з іншими охоронними системами

Відеореєстратор має обмежені здатності інтеграції з іншими охоронними системами: охоронною й пожежною сигналізацією, системами керування доступом і т.п. Це обумовлено «твердістю» його апаратної й програмної частини. Відеореєстратори мають стандартні тривожні входи. Кількість тривожних входів звичайно збігається з кількістю відеовходів.

До тривожного входу можна підключити кожної з охоронних датчиків, наявних на ринку, тому що принцип роботи у всіх однаковий – вони або розмикають ланцюг при спрацьовуванні й називаються нормально-замкнутими або замикають її – нормально-розімкнуті. Будь-який датчик підключається до відеореєстратора звичайним двожилиним кабелем, що називають «локшиною». Так само ви маєте можливість підключити до одного тривожного входу кілька охоронних датчиків, з'єднаних їх між собою. Така послідовно з'єднана між собою ланцюг з датчиків називається «контуром». Якщо у вас уже встановлена сигналізація, ви можете підключити тривожні входи відеореєстратора до її тривожних входів тої ж «локшиною».

При спрацьовуванні охоронного датчика відеореєстратор може виконати наступні дії: почати запис відео з обраних вами камер, подати звуковий сигнал через убудований динамік, подати сигнал тривоги на зовнішній пристрій. Зовнішнім пристроєм може бути: сирена, оповіщувач по GSM каналі, що відправить вам СМС на стільниковий телефон, пульт охорони. При цьому на один тривожний вихід можна підключити всі ці пристрої одночасно.

З однієї сторони можливості відеореєстратора по інтеграції з іншими охоронними системами обмежені, але з іншої сторони дозволяють вирішити більшість типових завдань.

Масштабованість і модернізація

Цифрова система відеоспостереження, побудована на базі відеореєстратора абсолютно не масштабована. Ні можливості збільшити число відеовходів або тривожних входів, додати ще один USB порт або встановити мережну карту. Відеореєстратор є закінченим, «жорстко зібраним» пристроєм із чітко визначеним набором функціональних можливостей.

Масштабованість системи відеоспостереження можлива тільки при підключенні декількох відеореєстраторів до локальної мережі й наявності спеціального програмного забезпечення, що дозволяє робити їхній перегляд в одному вікні на віддаленому комп'ютері.

Система відеоспостереження на базі персонального комп'ютера й плат відеозахвата

Апаратна частина

Система відеоспостереження на базі персонального комп'ютера й плат відеозахвата складається зі звичного всім персонального комп'ютера, зі спеціальною платою захвата відеозображення. Кожна така плата, найчастіше, має вузьку спеціалізацію. Тобто для можливості використання тривожних входів і виходів або PTZ камери буде потрібно покупка додаткової плати, що також буде необхідно інсталиувати в комп'ютер. Але кожна додаткова інсталяція буде знижувати як надійність, так і взагалі працездатність системи в цілому у зв'язку з можливістю виникнення конфліктів пристроїв. Чим більше додаткових плат, тим скоріше вони почнуть конфліктувати.

Плати відеозахвата, аналогічно відеореєстраторам, звичайно мають 4, 8 або 16 відеовходів. Використовувані тут камери також є аналоговими, а система відеоспостереження – цифровою.

Тактова частота сучасного комп'ютера становить порядку 3 ГГц, що перевищує тактову частоту відеореєстратора в 3 рази. Доречно буде відзначити, що сучасна архітектура процесора, що застосовує сучасну оперативну пам'ять, дає ще більш ніж дворазовий приріст продуктивності. Таким чином,

						ВКРМ-122.24.0014.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата			17

продуктивність комп'ютера вище продуктивності відеореєстратора більш ніж в 6 разів. Говорити про необхідність такого потенціалу для приватного домоволодіння й малого бізнесу складно, зате вартість такої системи буде неухильно рости разом із продуктивністю й можливістю розширення. І може відрізнятись від системи на базі відеореєстратора в 2-3 рази й більше. Оцінити цю величину точно не представляється можливим, тому що в системах відеоспостереження на базі персонального комп'ютера й плат відеозахвата істотний внесок у загальну вартість вносить ПЗ, ціна якого іноді дорівнює вартості системи.

Для перегляду зображення, як і будь-який звичайний комп'ютер, система відеоспостереження на базі ПК і плат захвата підключається до монітора або телевізора. Інтерфейс підключення залежить від відеокарти або материнської плати персонального комп'ютера.

Програмна частина

Операційна система Windows – найбільш популярне рішення для систем відеоспостереження на базі ПК. Великий набір функціональних можливостей посуває користувачів систем відеоспостереження на їхнє використання, а загальна комп'ютерна грамотність по використанню саме Windows створила армію псевдо-професіоналів, що вміють установити комп'ютерну гру, але не усвідомлюють можливі наслідки цієї дії для сучасної системи безпеки. Сучасні програми підмінюють деякі системні файли, ускладнюючи або унеможливаючи роботу системи відеоспостереження. Забираючи всі ресурси процесора, вони не дають нормально функціонувати споконвічно досить вимогливої до ресурсів системі. І навіть якщо користувачі не грають на комп'ютері, вони здатні декількома некваліфікованими діями повністю порушити роботу системи. Навіть у тому випадку, якщо система відеоспостереження підмінює Windows Explorer і завантажується до завантаження оболонки, охоронці прекрасно знають, як перемкнутися між завданнями (Ctrl-Alt-Del) і завантажити іншу програму. При цьому для надійного захисту від зовнішніх впливів (різних вірусів і атак) в

					ВКРМ-122.24.0014.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		18

операційній системі Windows потрібен комплексний підхід і покупка дорогого спеціалізованого програмного забезпечення, що не завжди доступно звичайному користувачеві.

З вищесказаного можна зробити висновок, що для стабільної й зручної роботи системи відеоспостереження на базі ПК потрібний окремий комп'ютер, що буде надійно захищений. Є думка, що використання робочого комп'ютера як знизить вартість системи, так і не вплине на стабільність роботи. Це думка помилкове. Технічне обслуговування системи відеоспостереження вимагає використання послуг інсталяторів через складність установки й налаштування програмного забезпечення. А постійні збої в роботі при використанні робочого комп'ютера, швидше за все, приведуть до покупки окремого ПК.

Керування системою

Інтерфейс користувача в комп'ютерній системі являє собою звичне для більшості користувачів рішення на базі операційної системи Windows. Після включення комп'ютера буде потрібно запустити спеціалізовану програму, у діалоговому вікні якої й буде відбуватися вся робота. Керування обмежується клавіатурою й мишею, що цілком звично й зручно звичайному користувачеві. З іншого боку, простота доступу може зіграти злий жарт як при спробі доступу до системи зловмисника, так і звичайної дитини.

Інтеграція з іншими охоронними системами

Велика кількість додаткового програмного забезпечення для комп'ютерних систем відеоспостереження дозволяє рішенням на базі ПК мати підвищену гнучкість у порівнянні з рішеннями на базі відеореєстраторів. Це викликано тим, що, використовуючи розповсюджену платформу операційної системи Windows, значно простіше створити велику кількість підсистем з різними функціональними можливостями, а також інтегрувати розробки інших виробників як програмні, так і апаратні. Комп'ютерна система відеоспостереження здатна інтегруватися з касовим робочим місцем у магазині роздрібною торгівлі, працювати разом із системою контролю доступу,

					ВКРМ-122.24.0014.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		19

розпізнавати автомобільні номери й особи людей, працювати разом із системами охоронно-пожежної сигналізації. Щоб одержати всі вищевикладені можливості необхідно витратити багато часу й сил, а, що найважливіших грошей. Необхідно відзначити, що витрати на можливого функціонала, як і на будь-які інші програмні продукти, часом, не відповідають очікуваним результатам.

Масштабованість і модернізація

Масштабованість даного рішення не викликає питань. Наприклад, для збільшення кількості камер, що підключаються, або для підключення охоронних датчиків необхідно придбати й установити в комп'ютер ще одну плату з відповідними функціональними можливостями. З іншого боку, можливо, у цьому випадку прийде міняти програмне забезпечення або, як мінімум, його переналаштувати. А, тому що самостійне налаштування практично неможливе через її складність, виникне необхідність використання послуг інсталятора, що в черговий раз приведе до додаткових грошових витрат.

IP відеоспостереження

Система відеоспостереження, побудована на базі персонального комп'ютера й IP-камер багато в чому аналогічна попередньої. Центром керування системою також є персональний комп'ютер.

Основною відмінністю від попередньої є спосіб передачі сигналу від камери до ПК і відсутність плат захвата відеозображення. Сигнал передається по мережному кабелі. Це зручно коли структура мережі добре організована, а робота стабільна. Замість плат тут використовується мережне встаткування (маршрутизатори, роутери, мости) і програмне забезпечення, що вимагає ще більш високої кваліфікації інсталяторів, із глибоким знанням мережних технологій. А, як відомо, праця професіоналів коштує дуже дорого.

Камера для системи IP відеоспостереження як мінімум в 3 рази дорожче звичайної. Якщо гнатися за дозволом 1-3 МП, то для цього вже потрібні гігабітні мережі (а не 100-мегабітні, як скрізь), що піднімає вартість у порівнянні з відеореєстраторами на порядок. И в 2-3 рази дорожче системи на базі ПК і плата

					ВКРМ-122.24.0014.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		20

відеозахвата. Одна камера на 3 МП може коштувати як повний комплект із 4 камер і відеореєстратора.

Керування системою здійснюється в діалоговому вікні програми в операційному середовищі Windows з усіма її мінусами. Необхідно відзначити, що версія операційної системи тут скоріше буде не домашня – а мережна – NT або Server.

По інтеграції й масштабованості дана система обходить всіх конкурентів, але не потрібно забувати, що й витрати на це будуть самими більшими в порівнянні з конкурентами.

Стандартні завдання, які ставляться перед системою відеоспостереження

Стандартні завдання, що коштують перед відеоспостереженням на будь-якому об'єкті, будь те великий, малий бізнес або приватний будинок, схожі. Виділимо основні:

1. Поточне спостереження.
2. Робота з архівом відеозаписів.
3. Дистанційний перегляд поточне зображення й архіву.
4. Запис відеозображення по детекторі руху, а також при спрацьовуванні охоронних датчиків або втраті сигналу.

На великому об'єкті до стандартного додаються наступні завдання:

1. Інтеграція із системою охоронної й пожежної сигналізації.
2. Інтеграція з апаратно-програмним комплексом системи контролю й управління доступом (СКУД).
3. Масштабованість і модернізація системи відеоспостереження при необхідності.
4. Поточне спостереження й керування всією системою з однієї точки, у тому числі організація відеоспостереження через Інтернет.

У великому бізнесі вкрай важливий високий рівень інтеграції й масштабованості системи із уже наявними засобами безпеки. Інтерфейс

					ВКРМ-122.24.0014.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		21

керування всіма завданнями, розв'язуваними на великому об'єкті, повинні бути інтегровані в одну оболонку, для того щоб оперативні рішення можна було реалізувати практично миттєво. Результатом такої гнучкості є необхідність постійного обслуговування системи відеоспостереження інсталяторами через неможливість установки й налаштування системи звичайним користувачем і, як наслідок, більші капіталовкладення.

Але ці гроші керівник служби безпеки підприємства змушений витратити. На свої руки сподіватися в цьому випадку не доводиться.

Таким чином, можна зробити вивід, що для великих підприємств із розвинутою інфраструктурою відеоспостереження на базі ПК і IP-відеоспостереження є оптимальним рішенням.

Основні вимоги до систем відеоспостереження для будинку, а також для малого, середнього бізнесу є надійність системи й простота експлуатації. Для даного сегмента вкрай важлива мінімізація витрат, пов'язаних з обслуговуванням системи. Поставив і забув або, інакше кажучи, відеоспостереження своїми руками – ідеальне рішення у випадку, коли обслуговування системи робить штатний системний адміністратор або електрик.

Для звичайного домовласника або невеликої фірми безумовним лідером є відеоспостереження на базі відеореєстратора, що повністю виправдує вкладені в нього засоби. Потенціал же комп'ютерних систем відеоспостереження залишається незачепленим, властиво, по тим же причинам, що й у випадку, коли додому здобувається комп'ютер останньої моделі за 1000\$, а робота на ньому відбувається тільки в офісних програмах і Косинці, де частота процесора навіть 300 МГц більш ніж достатня.

Крім того, розробка повноцінного апарату, написання або покупка ПЗ, інші працезатрати – все це збільшує вартість системи відеоспостереження на базі ПК в 2 і більше раз. А прокладка високошвидкісних мереж і їхнє обслуговування робить систему IP відеоспостереження практично недоступною звичайному користувачеві. Будь-яка зміна налаштувань – виклик фахівця, природно, за гроші. Користувачеві, відповідно, прийде чекати на них візиту.

					ВКРМ-122.24.0014.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		22

Для того щоб безпека стала ще доступніше, компанія UControl створила комплекти систем відеоспостереження, установка яких без особливої праці може бути виконана своїми руками. Для установки системи користувачеві не потрібна наявність спеціальних інструментів, а також спеціальних знань. Ці комплекти містять все необхідне для роботи й прекрасно підходять для рішення типових завдань малого й середнього бізнесу, а також для установки на дачі, у квартирі або заміському будинку.

2.2 Обґрунтування вибору засобів для побудови системи та мови програмування

Python – високорівнева мова програмування, яку називають другою за популярністю в світі. Її використовують для розробки вебзастосунків, програмного забезпечення, машинного навчання. Python застосовують для вирішення робочих завдань у компаніях Google, Instagram, Facebook, IBM, NASA, Dropbox, Netflix та інших. Розробники цінують цю мову програмування за простоту у вивченні, ефективність та мультиплатформність.

Python – скриптова мова програмування з досить простим синтаксисом. Для розуміння достатньо порівняти принципи написання найпростішої програми, яка виводить на екран текстове повідомлення. Саме тому мова програмування Python більш доступна для новачків, а професіонали встигли адаптувати її для вирішення великої кількості завдань. Це мультиплатформне рішення, тому знання Python дає можливість працювати у різних сферах: від розробки мобільних застосунків до ігрової індустрії та штучного інтелекту.

У мови програмування динамічна типізація: є можливість передавати до функцій будь-який тип даних без попереднього вказання. Інтерпретованість дозволяє знаходити помилки у коді ще до повної збірки у робочий застосунок. При цьому Python дуже чітко дає зрозуміти, де та через що виникла помилка.

Це мова об'єктноорієнтованого програмування (ООП).

					ВКРМ-122.24.0014.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		23

Програмне забезпечення на Python оформлене у вигляді моделей, які можуть бути зібраними у пакети. Тип та структуру кожного об'єкта можна запитати під час виконання програми. Для кожного з об'єктів можна отримати всю інформацію щодо його внутрішньої структури. Окрім того:

- у мови логічний синтаксис, завдяки чому вихідний код легко читати та розуміти;
- гнучкість та масштабованість Python дозволяє адаптувати високорівневу логіку та розширяти складні застосунки, як тільки виникне така необхідність;
- розробка на Python у більшості випадків проходить швидше, ніж на інших мовах програмування;
- Python – інтерпретована мова програмування. Це значить, що код можна написати у будь-якому текстовому файлі на будь-якій платформі, і потім успішно запустити;
- у Python – колосальна спільнота однодумців. Тож будь-які складнощі конкретних розробників вирішуються колективно.

Проте є декілька особливостей, які можна віднести до недоліків. Це повільність (ця мова програмування хоч і універсальна, проте повільніша за інші), велика кількість ресурсів, необхідних для роботи та «прив'язаність» до системних бібліотек.

Мова програмування Python використовується у наступних сферах:

1. Розробка програмних застосунків будь-якого напрямку.
2. Розробка серверної частини мобільних застосунків (найпопулярніший напрямок).
3. Ігри. Багато сучасних ігор для комп'ютерів (наприклад, World of Tanks) частково чи повністю написані на Python.
4. Вбудовані системи для різних пристроїв. Дуже часто Python використовують для написання внутрішніх платформ управління банкоматами.
5. Скрипти та плагіни до уже реалізованих програм для автоматизації процесів чи створення інших рішень.

					ВКРМ-122.24.0014.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		24

6. Тестування (автоматизація цього процесу).

7. Машинне навчання. – основна мова для написання алгоритмів і аналітичних застосунків у сфері Machine Learning.

Бібліотеки Python

Різні бібліотеки Python використовують для виконання конкретних завдань. Наприклад, Matplotlib підходить для відображення даних у двовимірній та тривимірній графіці. Pandas підходить для зручної роботи з даними. NumPy дозволяє створювати масиви та керувати ними. Requests використовується для веброзробки. OpenCV-Python відкриває можливості для обробки зображень з метою оптимізації систем «машинного зору».

Найвідоміші фреймворки для мови програмування Python

Фреймворки Python допомагають створити зручне та функціональне середовище для розробки. У них міститься набір інструментів, модулів та бібліотек, корисних для виконання конкретних завдань. Це значно полегшує роботу: наприклад, дає змогу не витратити час на розписування дій, які повторюються, а використати релевантний інструмент. Тож є можливість позбутися рутинних процесів та сконцентруватися на логіці проєкту.

Серед найпопулярніших фреймворків для Python:

– Django – найстаріший та найвідоміший. Створений для реалізації великих інтерактивних проєктів;

– Pyramid – зручний у налаштуваннях, і дає можливість реалізувати складні нестандартні ідеї;

– Web2py – підходить в першу чергу для вебзастосунків і може використовуватись на будь-яких архітектурах.

Популярні Python IDE

IDE або інтегровані середовища розробки – це програмне забезпечення, яке надає розробникам необхідні інструменти для написання, редагування, тестування та налаштування коду. Для розробки на Python найчастіше використовують IDE PyCharm, IDLE, Spyder та Atom.

					ВКРМ-122.24.0014.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		25

2.3 Розгорнута постановка завдання

Згідно з технічним завданням на випускню кваліфікаційну роботу за другим (магістерським) рівнем вищої освіти, реалізації підлягає програмне забезпечення, яке призначено для системи збору, представлення, обробки, зберігання, передачі та доступу до інформації мережевої відеоаналітики.

В процесі розробки випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти необхідно виконати наступний обсяг роботи:

а) провести аналіз існуючих систем-аналогів для виявлення їх позитивних і негативних якостей. Результати аналізу врахувати в подальших розробках;

б) вибрати та обґрунтувати методику побудови системи контролю роботи технологічного обладнання на виробництві в автоматизованому режимі. Розробити функціональну та структурну схеми системи;

в) розробити програмне забезпечення системи, що дозволить реалізувати поставлену технічним завданням задачу. Побудувати блок-схеми алгоритмів програми та підпрограми;

г) організувати інтерфейс користувача з метою формування та виводу на екран ЕОМ повідомлень про некоректні дії користувача та нестандартні ситуації в роботі технологічного обладнання;

д) розробити рекомендації по організаційних та методичних заходах, які забезпечать впровадження системи в промислову експлуатацію та її подальшу успішну експлуатацію;

е) провести розрахунки по визначенню економічної ефективності розробленої системи;

ж) розробити заходи по охороні праці при впровадженні та експлуатації системи, а також розробити заходи з цивільного захисту;

з) сформулювати висновки про виконаний обсяг робіт та одержані результати.

					ВКРМ-122.24.0014.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		26

3 ОПИС І ОБҐРУНТУВАННЯ ПРОЕКТНИХ РІШЕНЬ

3.1 Опис функціонування системи

За прогнозами, у найближчі роки світовий ринок відеоаналітики буде рости приблизно на 30% щорічно, що в 2,5 рази перевершує темпи росту ринку відеоспостереження в цілому. А в Україні можна чекати й перевищення цих показників. Його стимулом стануть вертикальні ринки. Так, наприклад, фінансування державою проектів по безпеці дозволяє здійснювати масштабні розробки й швидке впровадження подібних систем.

Найчастіше відеоаналітика ототожнюється із засобами розпізнавання осіб і автомобільних номерів, однак область її застосування набагато ширше. У числі найбільш затребуваних функцій – детектування руху й залишених предметів, відстеження траєкторій руху, багатоканальний трекінг, класифікація й ідентифікація об'єктів, розпізнавання ситуацій, аналіз поведінки людей і т.д. Якщо з першими із двох перерахованих завдань звичайна відеоаналітика справляється досить успішно, з рішенням третьої також більш-менш усе в порядку, то з багатоканальним трекінгом об'єктів виникають певні труднощі: ідентифікація об'єктів в одних випадках виконується задовільно, в інші – погано. А останні два завдання толком ще не вирішені.

Пошук об'єктів і виявлення людей

Попередня обробка зображень – перший етап аналітики. Це може бути вирахування одного кадру з іншого для детектування руху, виявлення змін в оптичному потоці, знаходження контурів і т.д. Другий етап – визначення місцезнаходження об'єкта, третій – його класифікація. На наступній стадії може аналізуватися поведінка об'єкта або виконуватися його ідентифікація, наприклад розпізнавання особи людини або номеру автомобіля.

					ВКРМ-122.24.0014.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		27

Завдання ідентифікації осіб і автомобільних номерів і критерії успішного досягнення бажаних результатів у цьому випадку досить чітко формулюються й добре вирішуються. Із цими завданнями, завдяки убудованим функціям відеоаналітики, цілком можуть упоратися самі камери. Так, наприклад, в ІР-камерах Sony шостого покоління з'явилася функція ідентифікації осіб. Уже найближчим часом у перспективних камерах сьомого покоління можна чекати подальшого вдосконалювання функції розпізнавання осіб (зменшення кількості помилкових спрацьовувань) для підвищення надійності й швидкості наступної ідентифікації особистості при роботі з базами даних (наприклад, у бюро пропусків).

Одними з основних цілей застосування відеоаналітики є виявлення й трекінг людей. Традиційний підхід складається в побудові моделі тла, після чого поточна картинка віднімається з моделі (або навпаки). Результати сегментуються на об'єкти для подальшої обробки – трекінга, детектування руху й т.д. Тим часом поряд з побудовою моделі тла вдосконалюються й методи простежування. Перший метод – простежування областей об'єктів, що рухаються, другий – виділення на зображенні людей за даними декількох спостережень. Зараз він модифікований і називається «фільтром часток».

На техніку нагромадження й вирахування тла заснована більшість методів виділення об'єктів, що рухаються. Найбільш популярні – Mixture Of Gaussians (MOG), Kernel Density Estimator (KDE) і Autoregressive Pseudospectrums (ARP). З їхньою допомогою можна сегментувати області, де спостерігається рух, однак їм властиві й недоліки: виявлення всіх без винятку областей, що рухаються (у маску руху попадають хвилі, вітки, відблиски й т.д.), висока чутливість до рухливого тла й шуму (тряска камери, поворот) і недостатня сприйнятливність однотонних об'єктів.

Метод фільтра часток, що відрізняється більшою інтелектуальністю, ґрунтується на пророкуванні положення об'єкта по траєкторії і яскраво-колірним характеристикам: нові області зіставляються з поточними для побудови

імовірнісної моделі, що дозволяє спрогнозувати, чи виявиться той або інший об'єкт у даній області.

Основним сучасним підходом до виявлення об'єктів у сцені є розпізнавання образів по наборі відібраних ознак. В основному використовуються методи HOG, Edgelet і група методів локальних шаблонів (Local Binary Patterns, LBP; Local Ternary Patterns, LTP; Scale Invariant LBP, SILBP; Scale Invariant LTP, SILTP). Метод, у рамках якого використовуються гістограми орієнтованих градієнтів (Histogram of Oriented Gradients, HOG), ґрунтується на контурі об'єкта й побудові градієнтів, що дозволяє відрізнити людини від інших об'єктів. Edgelet розроблений спеціально для детектування людей: за допомогою фільтрів будується контур об'єкта, із цього контуру витягає гістограмно-орієнтований градієнт, потім гістограми нормалізуються й використовуються для створення вектора ознак.

HOG – самий популярний метод виявлення людей на зображенні – у різноконтрастних сценах працює погано. У цій ситуації на виторг приходять тривимірні розмітки сцени із сегментацією, що допомагає виявляти об'єкти в конкретних областях. Поліпшити якість роботи HOG можна за рахунок навчання системи розпізнаванню людини по окремих частинах тіла (руки, ноги, тулуб, голова), що зменшує число помилкових спрацьовувань. Крім того, можлива побудова тривимірної моделі людини.

У методах, заснованих на еджлетах (Edgelet – відрізок прямій або кривій лінії), теж використовується навчання, що при розпізнаванні людей виробляється по плечовому поясі й тулубу. Еджлети розміром 4–12 пікселів – відрізки прямих, дуги окружностей, симетричні відрізки прямих і симетричні дуги окружностей – зіставляються з контурами на зображенні. При належному навчанні забезпечуються досить гарні результати розпізнавання. Для підвищення точності застосовується модифікація еджлетів на основі LBP, що дозволяє розширити вектор ознак. У результаті ймовірність розпізнавання людини по його плечовому поясі збільшується на 20-25%. Ці методи продовжують розвиватися.

					ВКРМ-122.24.0014.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		29

Підходи до виявлення людей на зображеннях методами машинного зору імітують сприйняття сцени людиною. Такі алгоритми вимагають значних обчислювальних ресурсів, тому мають потребу в істотній доробці й більше твердому відборі ознак при прийнятній імовірності розпізнавання. Їх можна застосовувати не тільки в системах безпеки, але й для підрахунку людей у групі/черги (для цього потрібний окремий детектор тулуба, плечового пояса й голови), а також при дослідженні потоків руху людей.

У масштабних системах відеоспостереження затребувані технології пошуку по відеоконтенту для спрощення формування запитів. Такі рішення створюють не тільки великі закордонні вендори, але й українські розроблювачі. У запитах можна вказувати різні параметри: місце об'єкта в кадрі, його розмір (з урахуванням перспективи), пропорції, вхід у задану зону, перетинання лінії, трекінг (ведення об'єкта в кадрі й побудова його траєкторії). Наприклад, трекінг дозволяє встановити людей, що направляються в потрібну сторону, або відокремити машини, що в'їжджають, від що виїжджають. По такому принципі можна вести підрахунок відвідувачів.

Наприклад, можна знайти всіх людей у світлих штаних і синіх куртках, що перетинають якусь лінію в заданому напрямку. У компанії працюють над тим, щоб запити можна було задавати природною мовою (голосові запити з розпізнаванням мови). Пошук по прикметах застосуємо не тільки до архівів, але й дозволяє перехоплювати об'єкти в реальному часі – наприклад, відслідковувати автомобілі червоного кольору заданого розміру. У системах масштабу міста дана функція може бути дуже корисної.

Багатокамерна відеоаналітика й міжкамерний трекінг

Багатокамерна аналітика із супроводом об'єкта декількома камерами дозволяє знизити загальне число тривожних сигналів (особливо в місці перекриття зон, контрольованих різними пристроями), а також прив'язати подія до карти місцевості, допомагаючи операторові краще орієнтуватися в що відбувається. Однак для цього потрібні висока якість калібрування й ретельна

					ВКРМ-122.24.0014.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		30

прив'язка камер до місцевості, інакше можливі більші погрішності у визначенні положення об'єктів. Стельова камера досить точно визначає позицію об'єктів, але при горизонтальному спостереженні ступінь погрішності може скласти 50%.

Багатокамерна відеоаналітика покликана визначати глобальні координати об'єкта виходячи з його екранних координат, поєднувати траєкторії від різних камер по їх просторовій і тимчасовій кореляції, згладжувати й екстраполювати отримані траєкторії, однак при інтерпретації розмірів об'єкта виникають складності (стельова й настінна камери визначають їх по-різному). Для стельових камер точність досягає 75%, для вуличних – 50%, при цьому результати залежать від розташування камер (ступеня перекриття їхніх зон видимості). Для ідентифікації людини на різних камерах, виявлення й трекінга людей можна використовувати методи машинного зору.

Стельова камера дозволяє вірогідно розпізнати людини (у тому числі нерухливого), його положення, траєкторію й швидкість руху, але розмір контрольованої зони обмежений висотою підвісу камери. Для ідентифікації використовується обмежений набір ознак, а при утворенні щільних скупчень людей можливі помилки. Ще один недолік – незручність візуального контролю з боку оператора.

На відміну від стельової, оглядова (настінна або вулична) камера забезпечує досить велику зону спостереження й велика кількість ідентифікуючих ознак, робить візуальне восприяття більше зручним, але виділення окремої людини може бути ускладнене через перекриття об'єктів, труднощів багатокамерного трекінга, помилкового визначення положення в кадрі.

У розробці Macroscop при міжкамерном трекінгу система класифікує об'єкт, виявляючи його особливості, і намагається знайти на сусідніх камерах об'єкти з аналогічними прикметами. Це ще не автоматичний, а всього лише автоматизований режим: оператор повинен підтвердити правильність виділення об'єкта, оскільки ризик помилки, особливо при великій кількості об'єктів (наприклад, людей), досить великий. Такі можливості корисні не тільки в

					ВКРМ-122.24.0014.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		31

системах безпеки. Наприклад, це дозволяє зрозуміти, як покупець переміщається по торговельному залі магазину, і виявити шаблони його поведінки. Для аналізу не потрібно повністю декодувати відео, що знижує вимоги до ресурсів. Комп'ютер із процесором Intel Core i5 може обробляти відео, передане сотнею камер з дозволом 1,3 Мпкс при 12 кадрах у секунду.

У масштабних і розподілених системах якість роботи камер відеоспостереження необхідно ретельно контролювати.

Підрахунок людей

Підрахунок числа людей необхідний не тільки з метою суспільної безпеки. Його можна використовувати для оцінки числа відвідувачів, регулювання кількості відкритих кас, планування місць установки рекламних носіїв. Системи підрахунку людей всі частіше застосовуються організаціями роздрібною торгівлі – на зміну колишнім інфрачервоним горизонтальним або термокамерам приходять відеолічильники, які у корені поміняли ринок систем підрахунку відвідувачів, витісняючи всі інші рішення.

Їхнього достоїнства – малий обсяг монтажних робіт, відсутність перешкод для персоналу й покупців, максимальний фізичний захист від навмисного перекручування даних, простота в налаштуванні, можливості використання в суміжних областях (безпека, відеоаналітика). У кожному сегменті роздрібною торгівлі (магазини, торговельні мережі та ін.) застосовуються свої системи відеопідрахунку. Вони можуть використовувати аналітичне ПЗ й програми для налаштування встаткування, передбачати інтеграцію зі стороннім програмним забезпеченням і впровадження додаткових функцій («траєкторія руху», поведінки покупців, «гарячі» і «холодні» зони, оптимальне викладення товару). Однак у багатьох випадках досить дороге аналітичне ПЗ не потрібно, а можливість самостійної установки системи дозволяє заощадити значні засоби.

Залежно від ракурсу зйомки й щільності юрби можуть використовуватися різні сценарії підрахунку людей. Перший – щільна юрба. Така ситуація характерна для заходів (мітингів, концертів), а також годин пік, коли формуються

					ВКРМ-122.24.0014.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		32

щільні людські потоки. У цьому випадку застосовуються методи підрахунку на основі займаною юрбою площі з урахуванням перспективи й «строкатості» текстури. При достатній величині об'єктів і їхньому русі можна задіяти методи на основі кластеризації траєкторій точок, що відслідковуються незалежно (синхронні траєкторії з високим ступенем імовірності ставляться до одного об'єкта).

Останні методи дозволяють визначати людей з точністю до 94%. При цьому допускаються різні ракурси й конфігурації сцени. За допомогою даного методу можна оцінювати кількість минаючих людей, але він погано працює з малорухомими людьми, до того ж для нього характерна низька швидкість – для обробки одного кадру потрібно більше 10 сек. Якщо розмір особи занадто малий, а об'єкти значно перекриваються, аналіз утрудняється, та і юрба вже не розбивається на кілька невеликих груп.

Другий сценарій – розріджений потік людей, наприклад на вулицях або у великих магазинах. Для даної ситуації характерні занадто малі зображення осіб, нещільні групи людей і великих перекриттів усередині груп. У цьому випадку застосовуються методи на основі вирахування тла, які при невисокій швидкості характеризуються можливістю роботи з нерухливими й малорухомими компонентами об'єктів (їх повинне бути не більше 10). Середня помилка становить 1,2 чоловік (у сцені від 11 до 45 чоловік).

Якщо в зображення високий дозвіл, а перекриття невеликі, підрахунок людей може вестися шляхом виявлення із супроводом між кадрами. Такий сценарій використовується в магазинах при оцінці кількості покупців і довжини черги, коли камеру можна розмістити у входу в приміщення або в каси, однак його точність (достатня для даного застосування) залежить від ракурсу і якості трекінга.

Зенітне (стельове) розташування камери сприяє більшій точності, підвищує швидкість обробки інформації, спрощує методи підрахунку. У цьому випадку можна застосовувати методи на основі оптичного потоку, а також

пошуку й супроводу голови людини. Їхня заявлена точність перевищує 95%, а помилка (коли за людей приймаються інші об'єкти) становить близько 10-20% (убік завищення). Це досить гарна точність. Проблеми можуть виникнути, якщо істотну частину минаючих у контрольованій зоні людей становить персонал або відвідувачі з візками, великими сумками або коробками, а крім того, якщо поруч із камерою стоїть охоронець, хтось курить, говорить по телефоні або чекає когось.

Виявлення асоціального поведження

Ще одне застосування відеоаналітики – виявлення асоціального поведження. Бійки, хуліганське поведження можна виявляти по сплесках активності. Для цього проводиться аналіз історії руху об'єктів (людей), виявлення загального рівня сплесків і нерівномірностей руху, перегони. На основі зібраної статистики виносяться рішення про нестандартне поведження. При цьому для зменшення числа помилкових спрацьовувань необхідна фільтрація подій. Для навчання системи використовуються відеоролики.

Аналіз поведження звичайно починається з відстеження траєкторії об'єкта, наприклад декількох осіб у кадрі. Потім аналізується допустимість цієї траєкторії для даного об'єкта. Можна спробувати ідентифікувати об'єкт. Однак «детектор бійок» практично не має успішних реалізацій, оскільки таке завдання складно сформулювати: те саме поведження людей може в одній ситуації означати бійку, в іншій – зустріну старих друзів. До того ж система повинна враховувати контекст поведження, щоб визначити його адекватність. Однак вирішити подібне завдання засобами відеоаналітики все-таки можна – необхідно її сегментувати.

Виявлення асоціального поведження засобами відеоаналітики поки що залишається проблематичним. Хоча система автоматично підбудовується до загального рівня руху в кадрі, для її якісної роботи воно повинне бути досить рівномірним (без наявності автомобілів і великих об'єктів, що рухаються), а при відсутності різких рухів виявити тривожну ситуацію складно. Проте при використанні стельових камер точність виявлення асоціального поведження

					ВКРМ-122.24.0014.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		34

становить приблизно 90% (10% – помилкові спрацьовування), а для вуличних камер – 80%.

У будь-якій системі відеоспостереження важливі не тільки точність виявлення подій, але й відсоток помилкових спрацьовувань. Наприклад, при 10-процентному показнику на 10 тис. подій буде доводитися тисяча помилкових спрацьовувань. Користуватися такою системою неможливо, особливо при великій кількості камер, тому завжди потрібно брати до уваги ймовірність помилкового спрацьовування.

Віддалене відеоспостереження

Щоб бути в курсі того, що відбувається на контрольованому об'єкті, можна застосовувати й більш прості методи. Усе більше популярним стає відеоспостереження з використанням мобільних пристроїв, однак користувачів не завжди влаштовує «живе відео». Спрацьовування тривоги при будь-якому русі в кадрі тільки дратує.

Весь запис розбивається на події, тому відео, що не представляє інтересу, можна ігнорувати. Та й обсяг відеоархіву зменшується в десятки тисяч разів. Добовий архів можна переглянути за п'ять хвилин у зручний час. Для цього компанія розробила мобільний додаток зі зручним інтерфейсом. Відео автоматично завантажується на пристрій, а тривожні події супроводжуються звуковим сигналом і/або вібрацією. При такому варіанті відеоспостереження людина може переглянути відео не тоді, коли на контрольованому об'єкті щось відбувається, а коли може це зробити.

Клієнтський додаток авторизується на віддаленому сервері, де здійснюється відбір відеофрагментів за допомогою відеосемантики – розбивки того, що відбувається в поле зору камер, на логічні сюжети з вибором ключових кадрів, що пояснюють зміст цих сюжетів. Відеосемантика, на відміну від цілого ряду функцій відеоаналітики, реально працює навіть у юрбі. Система фіксує нестандартне поведіння людей перед камерами й залишає ключові кадри на

					ВКРМ-122.24.0014.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		35

екрані до старіння інформації. Важливі події не зникають із екрана, якщо на них ще не звернули уваги.

Тим часом є цілий ряд завдань, де й складних методів відеоаналітики недостатньо – доводиться звертатися до алгоритмів (про їх уже говорилося вище) і системам машинного зору.

Машинний зір і промислові камери

Система машинного (технічного) зору – це набір програмно-апаратних засобів для захвата, обробки й зберігання зображення об'єктів або відеоряду, отриманого у видимому або невидимому (ІЧ) спектрі або іншому способі. Система ІР-відеоспостереження – окремий випадок системи машинного зору. По способах обробки інформації вони практично не відрізняються. Однак камери машинного зору дозволяють вивести відеоаналіз на новий рівень. Як правило, вони застосовуються в промисловості для візуального контролю якості й комплектації, керування машинами.

Найцікавіша область використання інтеграції з ІР-відеосистемами – ідентифікація й облік. «Одне з основних відмінностей систем машинного зору полягає в тому, що сцена, як правило, фіксована. Ми знаємо, що шукаємо, і націлені на визначення параметрів об'єкта, його геометричних і цифрових характеристик. Тому ідентифікація й облік – саме та область, де придадуться переваги машинного зору. Ці системи надають на порядок більше докладну інформацію про об'єкт, чим ІР-камери. Можна одержувати чисельні оцінки й застосовувати класичні методи ідентифікації об'єктів».

У машинному зорі будь-яка встановлена камера вимагає калібрування, а дозвіл камер може досягати 70 Мпкс. Дозвіл і швидкість (сотні кадрів у секунду з дозволом HD) – ключові параметри, по яких камери машинного зору перевершують ІР-камери. Перші не заміняють других, а надають більше даних і можливостей для аналізу. Це дозволяє використовувати камери машинного зору в краш-тестах, для аналізу швидко, що протікають процесів, у спорті та ін. У них

					ВКРМ-122.24.0014.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		36

застосовується широкий спектр матриць і інтерфейсів. Лінійні камери машинного зору дозволяють одержувати лінійну «розгорнення» об'єктів.

Для камер машинного зору є широкий вибір засобів розробки ПЗ (це не «коробковий продукт»). З їхньою допомогою можна вимірювати й ідентифікувати об'єкти самих складних форм, підраховувати виробу із заданими параметрами, збирати статистику.

Матричні камери забезпечують зйомку з дозволом 71 Мпікс зі швидкістю 3 кадри в секунду, з дозволом HD (1920×1080) – більше 340 кадрів у секунду, VGA (640×480) – 600 кадрів у секунду. Лінійні камери мають дозвіл 2-12 тис. пікселів і швидкість до 140 тис. кадрів у секунду. Інакше кажучи, по своїх можливостях камери машинного зору на 1-2 порядки перевершують IP-камери, але вони значно дорожче й використовують дорогу оптику.

Технології машинного зору дозволяють гнучко маніпулювати дозволом і швидкістю. Та сама камера дає можливість вибирати ці параметри. Вибір оптики значний, але корекція оптичних перекручувань вимагає складних алгоритмів. Камери машинного зору можна з вигодою використовувати для розпізнавання осіб, малих об'єктів або об'єктів, що перебувають у широкому полі зору, – для рішення тих завдань, з якими погано справляються камери IP-відеоспостереження. Вони передають на порядок більше інформації, що висуває особливі вимоги до інтерфейсів камер і мережам передачі даних, тому IP-камери не можуть їх замінити. Ще один цікавий напрямок, що дозволяє одержати більше інформації про спостережуваний об'єкт, – відеоаналітика 3D.

Від 2D до 3D

При спостереженні за тривимірним миром за допомогою 2D-камери губиться інформація, необхідна для відеоаналізу. «Причина невдач 2D-аналітики в тому, що мир тривимірний, а одержуване зображення двомірне. Недолік інформації заповнює гіпотезами й припущеннями, які, як підтверджують відомі оптичні ілюзії, не завжди виправдуються. Опис тривимірних об'єктів у термінах двомірних проєкцій складно й громіздко, тому двомірна відеоаналітика

					ВКРМ-122.24.0014.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		37

найчастіше із працею справляється навіть із відносно простими завданнями, наприклад з ідентифікацією об'єктів».

Природний крок – перехід до тривимірного відеоспостереження. Відомо кілька способів одержання тривимірного зображення. Один з них, лазерне сканування, займає досить тривалий час – десятки хвилин, тому для відеоспостереження за людьми не підходить. Інший метод – структуроване підсвічування у вигляді горизонтальних ліній і аналіз скривлення цих ліній (як у приставці Kinect). Недоліки – мала дальність і неможливість роботи на сонце. Третій спосіб заснований на факті кінцівки швидкості світла – вимірюється час відбиття світла від об'єктів на різній відстані. Цей перспективний метод зараз активно розвивається, але в нього є серйозні обмеження, насамперед пов'язані з невеликою відстанню до об'єкта, що повинен висвітлюватися активним підсвічуванням.

Джерела проблеми розпізнавання осіб – невірний ракурс (при ракурсі 15% жоден з існуючих алгоритмів не здатний розпізнати особа), міміка, макіяж, вуси й борода, окуляри. Тривимірний машинний зір допомагає вирішити багато хто з перерахованих проблем. Для відеоаналітики використовуються 3D-Моделі об'єктів реального миру. Знання абсолютних координат і розмірів об'єктів спрощує їхню сегментацію й виділення, дає можливість, наприклад, відокремити об'єкт від тла або розділити два об'єкти в кадрі. Ефективніше вирішується завдання ідентифікації об'єктів і підвищується вірогідність розпізнавання осіб (по 3D-Моделі особи) у різних ракурсах. За даними тривимірної траєкторії можна зробити виводи про поведінку об'єкта.

3D-система дозволяє побудувати об'ємну модель особи й розгорнути це синтетичне зображення для точного розпізнавання з використанням бази 2D-осіб. Для роботи системи потрібно ПК із процесором Intel Core i7 і двома графічними прискорювачами. Цикл обробки займає 2–3 сек. Така система може стати, наприклад, ефективним помічником у боротьбі із правопорушеннями. 2D-

					ВКРМ-122.24.0014.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		38

відеоаналіз наближається до границь своїх фізичних можливостей. Наступний великий крок у відеоаналізі – перехід до 3D

У відеоаналітики поки ще багато обмежень, не завжди зрозумілих замовникам. Звідси й помилкові очікування. Тим часом з'являється усе більше потужних і гнучких систем відеоаналітики, що дозволяють вирішувати різні завдання. Вони орієнтовані на різні сегменти ринку й можуть не тільки забезпечувати безпека, але й підвищувати ефективність роботи організацій – наприклад, підраховуючи число відвідувачів і/або віддалено контролюючи операції на об'єктах. А обробка й аналіз відеоархівів дають можливість швидко витягати необхідну інформацію й допомагають у розслідуванні інцидентів. З удосконалюванням аналітичних алгоритмів і збільшенням обчислювальних потужностей процесорів, застосовуваних у серверах і камерах відеоспостереження, можливості таких інструментів значно розширяться.

3.2 Розробка структурної схеми

Основні об'єкти відеоаналітики – люди й автомобілі. Саме їм розроблювачі відповідних рішень приділяють основну увагу. Однак сучасні технології дозволяють знайти аналізу відео більше широке застосування – як у системах охоронного відеоспостереження, так і в інших додатках.

Ринок відеоспостереження сильно фрагментований: на ньому представлені різноманітні технології й напрямки, а оскільки ця фрагментація тільки підсилюється, до остаточного його становлення ще далеко. На його формування впливають такі глобальні тенденції, як поширення хмарних технологій, Більші Дані, мобільні додатки, стандартизація (ONVIF) і, звичайно ж, що триває перехід на IP-системи. Всі ці фактори враховуються при розробці нових продуктів.

Однак, з погляду оператора системи відеоспостереження, розвиток IP-технологій не привело до значних змін його роботи, хіба що воно дозволило одержувати зображення з камер, що перебувають на великому видаленні.

					ВКРМ-122.24.0014.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		39

налаштування. Але поступово передові технології відеоспостереження проникають у масовий сегмент: камери оснащуються усе могутнішими процесорами, що дозволяє вирішувати досить складні завдання. Наприклад, камера Bosch формує потік метаданих на основі аналізу того, що відбувається в поле її зору. В остаточному підсумку це дозволяє сформувати базу даних всіх фіксуємих подій, що повинне значно спростити завдання пошуку у відеоархіві й оптимізувати зберігання даних.

Це одна з головних тенденцій. Наприклад, колись детектор руху був окремим пристроєм, а зараз ця функція убудована в більшість камер. Те ж буде відбуватися й з формуванням метаданих. Камери Bosch, Sony, Samsung і ряду інших вендорів уже мають такі можливості. Метадані пропонується збирати й використовувати відповідно до налаштовуваних користувача сценаріями.

Всім вже знайома технологія розпізнавання номерів автомобілів. Раніше для її використання було потрібно встановлювати спеціальним образом камери. Після того як дозвіл камер збільшилося, необхідність у цьому відпала. Цю технологію можна застосовувати в масових рішеннях, зокрема для пошуку автомобілів. У свою чергу, системи для розпізнавання осіб можуть мати як спеціальне, так і широке застосування – наприклад, для пошуку в архівах.

Типовим завданням відеоаналітики залишається швидкий пошук в архівах, наприклад при розслідуванні інцидентів. Ще одна цікава функція – спостереження за об'єктом за допомогою оглядової камери з одночасним керуванням поворотною камерою (трекінг). Мегапіксельні оглядові камери, що відрізняються високим дозволом, дозволяють контролювати величезні території, завдяки чому досягається істотна економія. Наприклад, камера може відслідковувати перетинання людиною віртуальної лінії й автоматично перемикається між всіма об'єктами, що рухаються, охоплюючи велику територію. І це лише один із прикладів використання відеоаналітики.

Виділяють наступні найважливіші завдання, у рішенні яких допомагає відеоаналітика:

					ВКРМ-122.24.0014.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		41

- первинна обробка відеоархіву з виділенням ключових подій, наприклад, щоб заощадити час оператора, що займається пошуком потрібної події;
- оптимізація відеоархіву шляхом запису на пристрій зберігання тільки ключових подій, що дозволяє істотно зменшити обсяг даних, що зберігаються;
- контроль за переміщенням об'єктів у кадрі, коли по відеозаписі можна визначити порушення порядку або правил руху;
- верифікація тривожних подій при активізації функцій «детектор черг», «детектор бійок», «детектор залишених предметів» і ін.;
- оперативне реагування на тривожні події, що виявляються завдяки використанню на об'єкті відеокамер.

Відеоаналітика – це виділення на відеозаписі значимих подій для подальшого реагування на них заздалегідь певним способом. Для цього застосовуються спеціальні аналітичні алгоритми. Роботу з відеозображенням можна класифікувати відповідно до трьох класів таких алгоритмів:

- Порівняння пікселів. Сюди ставляться детектори руху, детектори залишених предметів і т.д., які не є засобами відеоаналітики, але саме на цьому рівні реалізована більшість алгоритмів.
- Розпізнавання об'єктів, тобто визначення класу об'єкта по його відеозображенню й подальше порівняння з базою даних об'єктів. Сюди входить детекція осіб, розпізнавання номерних знаків і т.д. Це аналітика щодо простого рівня.
- Вивчення поведження й трекінг об'єктів у кадрі. В основу цих рішень покладені складні алгоритми. У більшості випадків важко скласти алгоритм, що надійно описує можливе поведження об'єкта в кадрі. Ще сутужніше здійснювати «передачу» об'єкта з однієї відеокамери на іншу.

Практичний приклад – спостереження за людьми в метро. Допустимо, одна з камер зафіксувала, що людина з відомими прикметами зайшов у метрополітен з небезпечним багажем. Коли це з'ясувалося, він уже сіл у поїзд і виїхав на іншу станцію. Як його знайти на відеозаписах камер, установлених на

інших станціях, і об'єднати ці відеозаписи? На допомогу приходять відеоаналітика, що вміє працювати з ідентифікацією й базами даних. Проблема в тому, що в даній області будь-яка високотехнологічна система створюється з урахуванням конкретних умов, тому її наступний перенос на інші об'єкти виявляється утруднений. Сьогодні попросту відсутній математичний апарат, що дозволяє ідентифікувати та або інша подія з належним рівнем надійності, без залучення величезних обчислювальних потужностей.

Однак, подібні завдання актуальні не для кожного клієнта – найчастіше користувач обмежується набором простих функцій, які дозволяють йому визначати якийсь набір подій, що не вимагають складного аналізу. У таких випадках доречна реалізація щодо простих аналітичних алгоритмів усередині відеокамер («убудована аналітика»).

Прості детектори руху – це ще не аналітика: для ухвалення рішення потрібно мати дані про напрямок переміщення об'єкта. Щоб витягти цю інформацію з відеопотоку, можна використовувати сервер зі спеціалізованим ПЗ або більше розвинені можливості камери.

Алгоритми відеоаналізу повинні бути уніфіковані, тим більше що завдання безпеки для багатьох об'єктів схожі. Використання ж ресурсів для рішення вузькоспеціалізованих завдань не буде ефективним через обмежений попит і більші витрати на розробку. Особливий інтерес представляють алгоритми для оповіщення про «нестандартне» поведженні об'єктів, що зменшують кількість фіктивних тривог.

Розповсюджені сьогодні алгоритми – детектори руху, перетинання віртуальної риси, вторгнення, залишених предметів і осіб – розроблялися для цілей охоронного відеоспостереження. Вони були призначені винятково для фіксації подій несанкціонованої або несвоєчасної появи/зникнення людей або предметів зі спостережуваної зони. Не можна сказати, що за минулі роки потенційні замовники повністю розібралися в можливостях відеоспостереження й відеоаналітики, але формулювати свої вимоги до систем безпеки стали точніше.

Тому сьогодні основна тенденція – адаптація прикладного програмного забезпечення до вимог замовника.

Усе більше потужних і гнучких систем відеоаналітики включається в інтегровані системи безпеки. Вони дозволяють вирішувати різні завдання – як у режимі реального часу, так і при аналізі подій у записі.

Відеоаналітика й керування підприємством

Відеоаналітикові можна використовувати не тільки для завдань безпеки й підрахунку відвідувачів магазинів. На підприємствах вона може застосовуватися для контролю за рівнем якості й строками виконання завдань, для підвищення продуктивності, навчання й мотивації персоналу. Система відеоспостереження з інтелектуальними функціями дозволяє накопичувати дані для прийняття рішень. Це можуть бути дані про покупців – їхній кількості, маршрутах і т.д. Завдання Video BI (бізнес-аналітики на основі інтелектуального відеоспостереження) – зібрати ці дані й перетворити їх у корисну інформацію для прийняття управлінських рішень. Наприклад, на будівельному майданчику Video BI допомагає заощадити засобу, фіксуючи в автоматичному режимі в'їзд техніки на об'єкт і виїзд із території. Якщо підключити дані про субпідрядників з фінансової системи, то вийде потужний інструмент для контролю витрат на будівельному об'єкті. Така система окупається вже в перший місяць будівництва й дозволяє заощадити понад 10% операційних витрат. При пошуку в архіві, увівши через Web-інтерфейс номер замовлення й атрибути виконавця, можна одержати кадри з об'єктів і переглянути відео. Це приклад того, як відео стає інструментом контролю». Рішення дозволяє стежити за ступенем готовності об'єкта, допомагає боротися зі злодійством матеріалів і встаткування, розслідувати події.

Інший приклад – логістичний центр, де система бізнес-аналітики, інтегрована зі СКУД, дозволяє вводити ідентифікатори посилок і фіксувати номери автомобілів, що під'їжджають до складу або сортувального вузла. Ці дані стають тегами для відео. За номером відправлення можна подивитися, як посилка оброблялася й куди її помістили.

					ВКРМ-122.24.0014.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		44



Рисунок 3.1 – Структурна схема системи

В автосервісі відеоаналітика дозволить чітко контролювати операції по обслуговуванню клієнтів і запобігати зловживання. На складальних виробництвах відео в сполученні з даними про технологічний процес допоможе підвищити рівень контролю якості.

Таким чином, у сучасних організаціях Video BI перетворюється в джерело інформації для керівників. Крім того, відео можна використовувати для реклами, традиційного забезпечення безпеки, показу на сайті компанії й т.д. Завдяки інтеграції з корпоративною базою дані відомості в реальному часі надходять в інформаційну систему (ERP, CRM), а при необхідності прийняття співробітниками оперативних мір можна відправляти повідомлення на їхні користувальницькі пристрої.

Тим часом ринок Video BI сильно фрагментований: розробка таких рішень вимагає детального вивчення предметної області й ситуації на стороні замовника. Ми бачимо три сегменти, суміжних з Video BI. Насамперед це ринок BI + CRM

(Corporate Performance Management), що, як очікується, до 2025 року досягне 18 млрд доларів, причому очікувані аналітиками середньорічні темпи росту складуть 7%. Інший суміжний сегмент – Video Surveillance as a Service (VSaaS). Прогноз для нього – 2,4 млрд доларів при середньорічних темпах 30%. Нарешті, ринок Video Content Analytics (VCA) виросте до 1,2 млрд доларів (на 25% у рік), а сам ринок Video BI – до 1,9 млрд доларів (30% росту).

Перспективні галузі для застосування Video BI – роздрібна торгівля, будівництво, виробництво, ПЕК, соціальні об'єкти – можуть бути цікаві постачальникам систем BI і відеоспостереження, а також стартапам. Однак, на відміну від охоронного відеоспостереження, відеоаналітика повинна враховувати відповідні бізнес-процеси. Крім того, потрібно більше чітке обґрунтування ROI.

Через різноманіття користувачів, а це практично все співробітники компанії – від інвестора/директора до виконавця, системам Video BI необхідний зручний і простий Web-інтерфейс. А для успішного розвитку даного ринку потрібні бюджетні рішення.

Як у будь-якої нової технології, ризики впровадження в Video BI високі, однак потенціал у ринку досить великий, коло потенційних користувачів широкий, а пропозицій небагато. Його ємність більше, ніж у ринку відеоспостереження в системах безпеки, а ріст може бути значніше.

Поки головним споживачем рішень в області відеоаналітики є сфера роздрібною торгівлі. Рітейл поступово починає усвідомлювати, що потенціал, закладений у камерах відеоспостереження, не обмежується тільки запобіганням втрат. За допомогою камер можна збирати безліч корисної інформації про покупців і їхнє поведіння – від підрахунку відвідувачів і виявлення найбільше часто відвідуваних зон магазину до оцінки ефективності викладення товару й визначення підлоги й віку відвідувачів.

Грамотний аналіз і використання отриманих даних дозволяють значно підвищити ефективність роботи як окремо взятої торговельної точки, так і всієї мережі в цілому. Так, наприклад, на підставі даних про «гарячі» і «холодних»

зонах магазину, отриманих за допомогою відеоаналітики, було проведено перепланування торговельного залу, що дозволило на 20% підвищити рівень продажів окремих категорій товару.

Особливий інтерес представляють відеоаналітичні рішення, убудовані в камери відеоспостереження, – наприклад, функції підрахунку відвідувачів або визначення довжини черги. Аналітичні модулі в камерах обробляють отримане відео й видають тільки потрібну користувачеві інформацію. Ніякого центрального встаткування, ніяких проблем з каналами зв'язку – все просто, ефективно й дуже економічно.

Розумне місто

Розвитку відеоаналітики сприяє також збільшення обчислювальної потужності серверів і використання апаратних транскодерів H.264. Серверне рішення з потужною відеокартою забезпечує висока швидкодія при обробці потокового відео HD. У схемах з розподіленою обробкою сервери на площадці замовника можуть обробляти основний потік даних і передавати на верхній рівень (наприклад, у хмару) дані меншого обсягу. Вендори й інтегратори поступово накопичують досвід побудови розподілених систем відеоаналітики, у тому числі в масштабах міста.

Ще десять років тому вважалося, що ефективність системи «безпечне місто» залежить від кількості камер, однак коли їх стає досить багато, оператори перестають справлятися з оперативним спостереженням, та й знайти що-небудь у відеоархівах стає неможливо. Але ж місто – це не тільки тисячі камер, але й інші пристрої фіксації подій: детектори транспорту, пожежні та інші датчики, системи « громадянин-поліція». При цьому різні служби відповідають за різні сфери діяльності й мають потребу у візуалізації подій різних типів. Нерідко потрібно одержувати агреговану інформацію про що відбувається.

У системах відеоспостереження рівня «безпечне місто» відеоаналітика – від простих детекторів руху до спеціальних систем виявлення загорянь і інших засобів моніторингу – стає незамінним помічником оператора. Вона покликана

					ВКРМ-122.24.0014.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		47

позбавити людини від перегляду безлічі екранів, допомогти йому аналізувати обстановку й приймати правильні рішення, адже, по статистиці, один оператор може ефективно спостерігати не більш ніж за 16 камерами, а за 22 хв. роботи його ефективність падає до 5%.

Щоб упоратися із цим завданням, потрібно мати можливість описати все, що відбувається в поле зору камер, за допомогою набору метаданих, переданих в аналітичні системи.

Відеоаналітика допомагає оцінити ситуацію, зрозуміти, що відбулося, швидко знайти потрібні фрагменти у відеоархіві. У ситуаційних центрах такі інструменти підказують шляхи рішення проблеми. Системи міського рівня звичайно будуються по ієрархічному принципі: рішення, розгорнуті в будинках, виконують своє коло завдань, відтинаючи помилкові спрацьовування й передаючи на рівень вище більше значимі дані.

Системи масштабу міста генерують близько 10 млн подій у день, причому мова йде аж ніяк не про мегаполіси. Це можна зіставити із завданнями, розв'язуваними із залученням аналітики Больших Даних. Необхідний єдиний інструмент для керування всією міською інфраструктурою, щоб забезпечити можливість побудови єдиної картини того, що відбувається для поліції, транспортних служб, підприємств ЖКХ. Тим часом більшість вендорів пропонує окремі системи. Єдина система дозволяє корелювати події з різних підсистем. В IBM є продукти для таких підсистем – від програм для реалізації соціальної підтримки громадян до систем забезпечення безпеки й аналізу баз даних (BI). Флагманське рішення – Intelligent Operation Center.

Інфраструктурні підсистеми є для інформаційної системи IBM джерелом даних або приймають від її команди й вказівки, а відеоаналітика відіграє ключову роль у прийнятті рішень. В інформаційну систему можна інтегрувати системи сторонніх вендорів і обмінюватися з ними інформацією через спеціальні шлюзи. Передбачені також інструменти для складання прогнозів і виводу звітів. У результаті ІС допомагає приймати рішення на основі максимально повних даних.

					ВКРМ-122.24.0014.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		48

Такі рішення накладають на систему відеоаналітики певні вимоги: до числа її необхідних властивостей ставляться незалежність від типів камер і відеосерверів, мінімальна вимогливість до формату і якості відеопотоку (далеко не всі міські камери мають високий дозвіл), максимальна відкритість функцій системи й гнучкість при інтеграції, зручний інтерфейс моніторингу подій, потужні засоби пошуку й аналізу накопичених в архіві даних про події, максимальна простота розширення й доробки функцій системи для обліку специфічних вимог замовників.

Система відеоаналітики IBM Intelligent Video Analytics (IVA) являє собою надбудову до системи відеоспостереження й складається із двох рівнів. На першому підсистема обробки відео збирає відеопотоки й обробляє їх, а на другому рівні потік, що з'являється в результаті, метаданих описує події, що відбуваються. Метадані (які характеризують колір об'єкта, його тип, швидкість, траєкторію й т.д.) надходять на сервери для постобробки й записуються в сховище. Метадані й відео зберігаються окремо, але система забезпечує однозначний зв'язок між ними.

Нерідко помилкові спрацьовування в системах відеоаналітики зводять нанівець всю користь від них. Система повинна генерувати мінімальну кількість помилкових спрацьовувань і пропусків, інакше її використання виявляється не вигідним з економічної точки зору. Оцінювати ефективність рішення потрібно в реальних умовах, тому пілотні проекти обов'язкові. Бажано відкрите порівняльне випробування систем. Корисні функції – багаторівневі карти ГІС, побудова панорам і аналітика на них. Одна поворотна камера з підтримкою аналітики дозволяє замінити кілька фіксованих камер.

Майбутнє – за великими інтегрованими й розподіленими аналітичними мережами. Адже чим ширше охоплення території, тим більше глибоко кожний користувач може поринути в контекст що відбувається. Наприклад, зараз багато роздрібних магазинів цікавляться можливістю аналізувати топологію потоків покупців на своїх торговельних площах за допомогою внутрішніх камер і відповідних засобів відеоаналітики. Зв'язавши свою систему із загальноміській,

					ВКРМ-122.24.0014.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		49

вони одержать можливість не тільки аналізувати потоки усередині магазину, але й визначати, звідки приходять покупці, на якому транспорті приїжджають. Виграш і для маркетингу, і для повсякденної операційної діяльності очевидний. Крім того, було б надзвичайно цікаво інтегрувати системи відеоаналітики з іншими джерелами подій – від різних датчиків і турнікетів до, наприклад, повідомлень у соціальних мережах, – створюючи тим самим єдиний пул подій.

Моніторинг і контроль якості

У масштабних розподілених системах відеоспостереження дуже актуальною стає завдання моніторингу й контролю якості.

Необхідно було розділити відповідальність між власниками камер і каналів зв'язку. Його можна використовувати для керування конфліктами в будь-якій великій системі відеоспостереження. Якщо кількість камер перевищує кілька сотень, то ймовірність виникнення розбіжностей різко зростає. Цьому сприяє й число учасників проекту (звичайно їх більше трьох). Моніторинг якості допомагає прийти до конструктивного діалогу між сторонами й дозволяти протиріччя.

Говорити про революцію, що зробить використання відеоаналітики в системах відеоспостереження. Є безліч стримуючих факторів: недосконалість аналітичних алгоритмів, потреба у великій обчислювальній потужності, висока вартість реалізації, наявність систем безпеки й СКУД, які найчастіше дають більше інформації для аналізу, чим відеозображення.

За останні 10–15 років продуктивність комп'ютерної техніки зросла на порядок, однак і системи відеоспостереження не стояли на місці. Більше вимогливими стали користувачі: тепер їм необхідно відео з дозволом не менш 1–2 мегапікселів із частотою 25 кадрів у секунду. Крім того, сьогодні є рішення аж до десятків мегапікселів – наприклад, 29-мегапіксельна камера Avigilon Pro дозволяє спостерігати за цілим сектором стадіону й має можливість цифрового збільшення до дозволів, придатних для розпізнавання осіб.

					ВКРМ-122.24.0014.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		50

Найбільш перспективні напрямки розвитку відеоаналітики: універсалізація аналітичних інструментів; створення комплексних систем, що поєднують безліч інструментів усередині одного програмного комплексу; перенос частини аналітичних функцій у відеокамери для розвантаження серверів; перенос частини аналітичних функцій у хмару; оптимізація алгоритмів; підвищення зручності й ергономічності керування аналітичними системами.

У першу чергу будуть розвиватися засоби відеоаналітики для розпізнавання порушень правил дорожнього руху (перевищення швидкості, проїзд на червоне світло, порушення правил паркування та ін.) – реалізація цих проектів швидко приносить значні засоби в місцеві бюджети, а їхня окупність становить кілька місяців. Підтвердженням може служити достаток камер на дорогах. Інша активно впроваджувана технологія – розпізнавання осіб для цілей безпеки й для запобігання розкрадань у банківських офісах і банкоматах. Фінансування державою проектів по безпеці дозволяє проводити масштабні розробки й швидке впровадження.

3.3 Розробка функціональної схеми

На рисунку 3.2 зображена функціональна схема системи. Найпершим блоком у схемі є кінцевий об'єкт (сектор) за яким власне і ведеться відеоспостереження. Далі за допомогою відеокамер спостереження за розробленою схемою (рисунок 3.1) і застосуванні локального обладнання сигнал надходить у кімнату операторів де обробляється на сервері за допомогою розробленого ПЗ системи збору, представлення, обробки, зберігання, передачі та доступу до інформації мережевої відеоаналітики. Спочатку проходить обробка відео-даних з послідуочим аналізом. Всі дані зберігаються у файловому архіві у форматі *.avi. Кожну добу інформація з камер спостереження формується у відповідну папку (шаблон папки ЧИСЛО_МІСЯЦЬ_РІК). Для зміни налаштувань чи перегляду документації існують відповідні вікна.

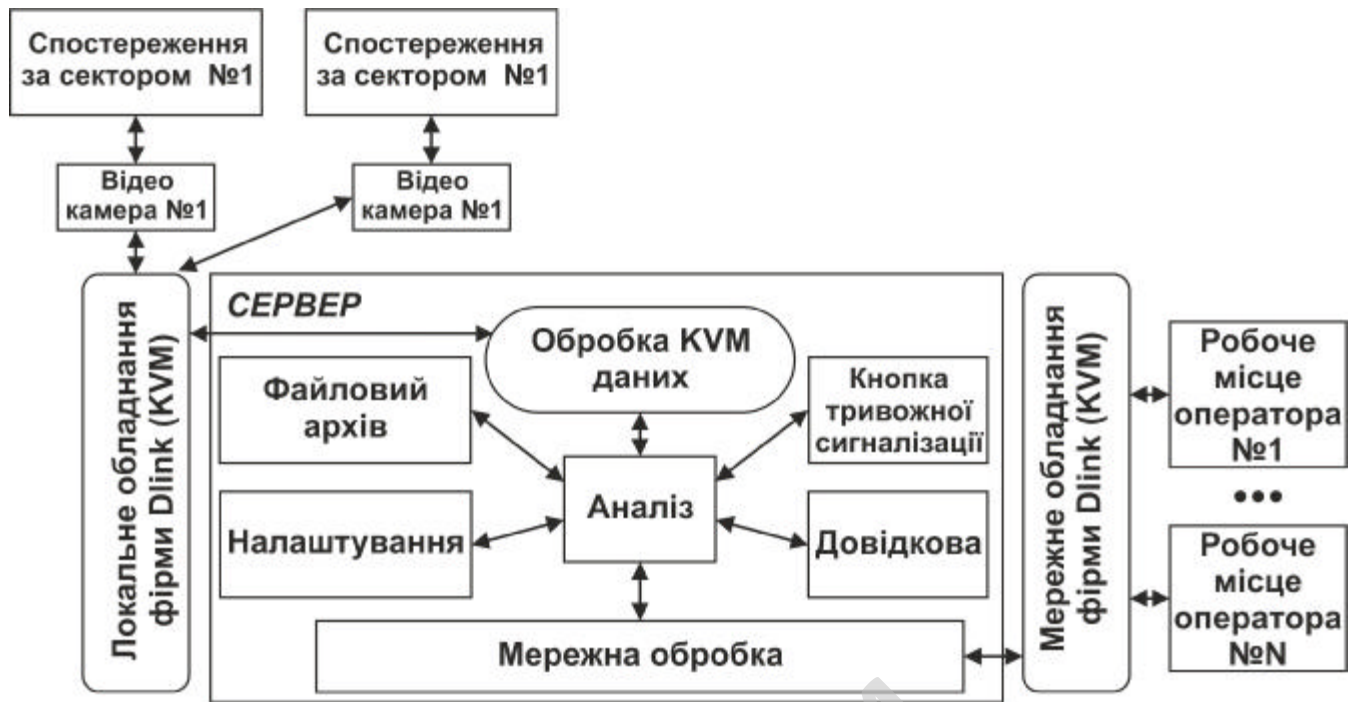


Рисунок 3.2 – Функціональна схема системи

Також для організації масштабуємі системи й контролю дій над співробітниками сигнал також через мережне обладнання надходить до керівництва. Розробляючи апаратну й програмну частину дипломного проектування враховувався ціновий фактор, а також особливості застосування камер в умовах України, а саме – застосовувався ударостійкий корпус, були обрані камери з розширеними температурними діапазонами роботи (-40 С до +55 С) і великим кутом огляду камери.

Розглянувши усі блоки функціональної схеми перейдемо до розгляду діаграми взаємодії процесів, які відбуваються у системі.

3.4 Розробка діаграми процесів

Діаграма процесів розробленої системи зображена на рисунку 3.3. При детальному її розгляді можна побачити як саме проходить взаємодія у розробленій системі. Використовується модель проектування, графічне представлення «потоків» даних в інформаційній системі.

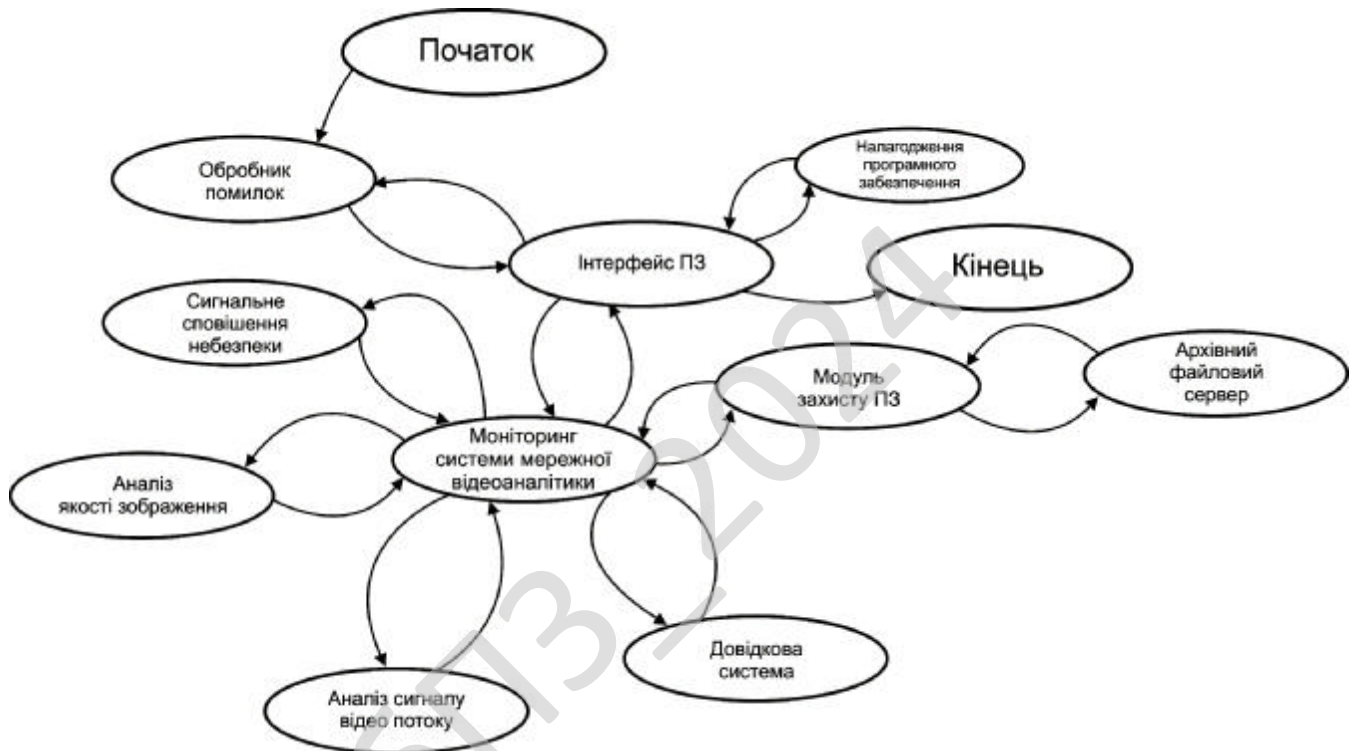


Рисунок 3.3 – Діаграма взаємодії процесів

Діаграма взаємодії процесів використовується для візуалізації процесів обробки даних (структурне проектування). Для розробника вважається звичним спочатку креслити діаграму взаємодії процесів даних рівня контексту, завдяки чому буде показано взаємодію системи.

Ця діаграма в подальшому підлягає уточненню шляхом деталізації процесів та потоків даних з метою показати систему що розробляється.

Діаграми потоків даних містять чотири типи елементів:

– Процеси які являють собою трансформацію даних в рамках описуваної системи.

– Сховища даних (репозиторії).

– Зовнішні по відношенню до системи сутності.

– Потоки даних між елементами трьох попередніх типів.

Таким чином, розглянувши опис системи, структурну, функціональну схеми системи, та діаграму взаємодії процесів перейдемо до опису блок-схем основної програми, та підпрограм, які використовуються, для реалізації системи.

КБПЗ – 2024

					ВКРМ-122.24.0014.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		54

4 РЕАЛІЗАЦІЯ ПРОЕКТУ. РОЗРАХУНКИ І ЕКСПЕРИМЕНТАЛЬНІ ДАНІ, ЩО ПІДТВЕРДЖУЮТЬ ПРАВИЛЬНІСТЬ ПРОЕКТНИХ РІШЕНЬ

4.1 Блок-схеми та опис алгоритмів функціонування системи

Блок-схеми є першоджерелами стратегії розвитку ПЗ. Тому від точності і детальної блок-схеми залежить результат всієї програми.

При виборі початкової точки відліку при побудові схем було враховано, що виходячи з вибору мови програмування і інших технічних засобів, програма буде об'єктно-орієнтована що вимагає оптимізації програми високого рівня, також те, що при розробці програми слід надати особливу увагу модулю системи збору, представлення, обробки, зберігання, передачі та доступу до інформації мережевої відеоаналітики.

При складанні блок-схем програмного забезпечення і напрацювання алгоритмів я зіткнувся з масою проблем, які вимагали напрацювання процедур і функцій над основною проблематикою. Для чого були створені додаткові класи, типи даних і константи, що забезпечило вирішення проблем.

На рисунку 4.1 наведено блок-схему основної програми, на рисунку 4.2 зображено роботу підпрограм.

Під час роботи над магістерською дипломною роботою було створено блок-схеми. Перед їх розглядом необхідно провести роз'яснення який саме тип блок-схем використовується.

Блок-схема це представлення задачі для її аналізу або розв'язування за допомогою спеціальних символів (геометричних образів), які позначають такі елементи, як операції, потік, дані тощо. Блок вхідних та вихідних даних прийнято позначати паралелограмом, блок обчислень (обробки) даних – прямокутником, блок прийняття рішень – ромбом, еліпсом – початок та кінець алгоритму.

					ВКРМ-122.24.0014.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		55

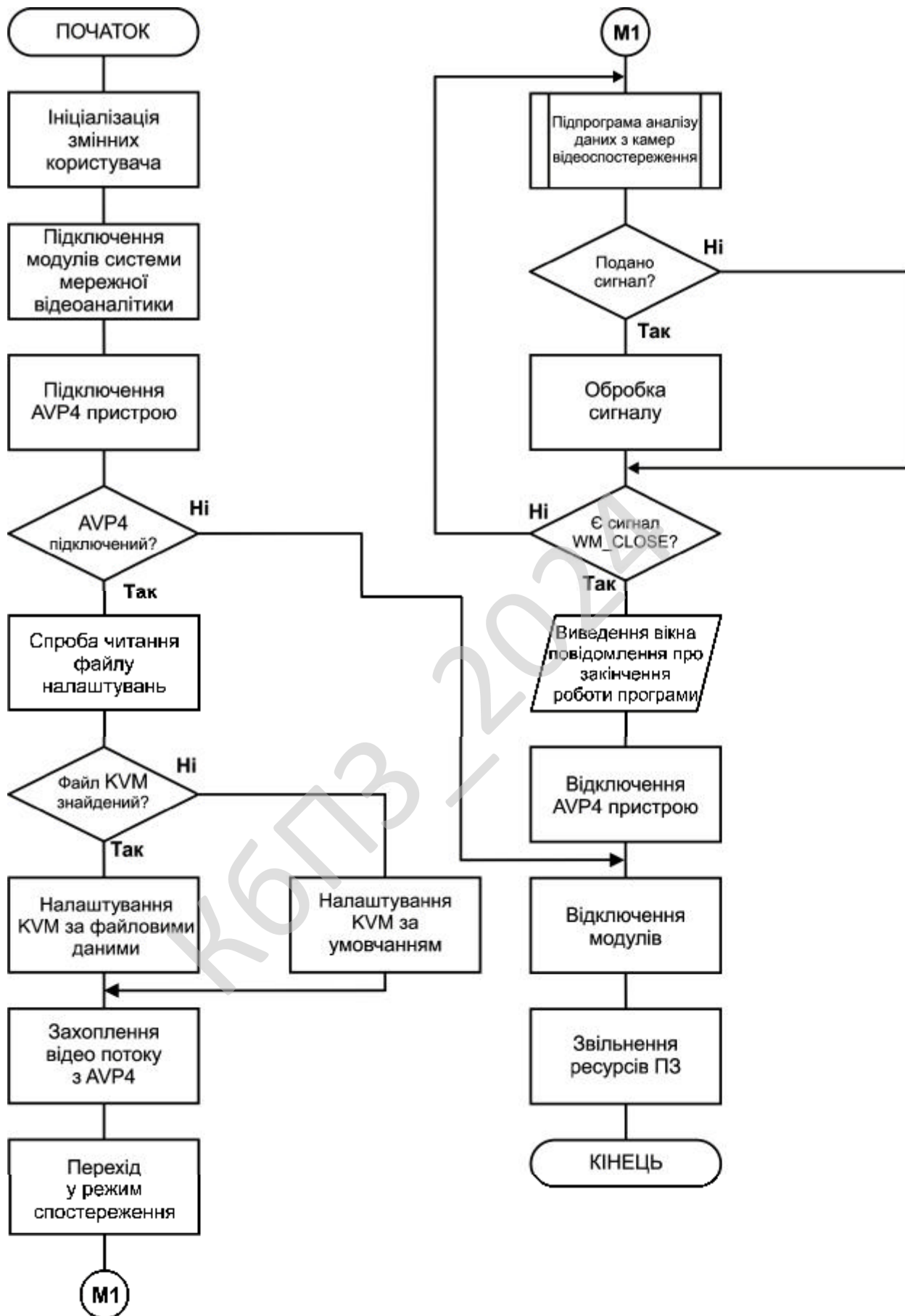


Рисунок 4.1 – Блок-схема основної програми

У інформаційних технологіях функціональна схема складається з функціональних блоків, які являють собою конструктивно відособлені частини (елементи або пристрої) автоматичних систем, які виконують певні функції. Функціональні блоки на схемі позначають прямокутниками, всередині яких надписують їх найменування відповідно до функцій, що виконуються. Зв'язки між функціональними блоками (внутрішні впливи) позначаються лініями зі стрілками, які вказують напрям впливів.

Функціональні схеми можуть виконуватися в укрупненому і розгорненому вигляді. У першому випадку на схемі зображають найважливіші блоки системи і зв'язки між ними.

У другому варіанті схема відображається більш детально, що полегшує її читання та ілюструє принцип роботи.

Основні елементи схем алгоритму це термінатор, процес, рішення, зумовлений процес (підпрограма), дані та з'єднувач.

Термінатор це елемент відображає вхід із зовнішнього середовища або вихід з неї (найчастіше застосування – початок і кінець програми). Всередині фігури записується відповідна дія.

Процес це виконання однієї або кількох операцій, обробка даних будь-якого виду (зміна значення даних, форми подання, розташування). Всередині фігури записують безпосередньо самі операції.

Рішення це показує рішення або функцію перемикального типу з одним входом і двома або більше альтернативними виходами, з яких тільки один може бути обраний після обчислення умов, визначених всередині цього елемента. Вхід в елемент позначається лінією, що входить зазвичай у верхню вершину елемента. Якщо виходів два чи три то зазвичай кожен вихід позначається лінією, що виходить з решти вершин (бічних і нижній). Якщо виходів більше трьох, то їх слід показувати однією лінією, що виходить з вершини (частіше нижній) елемента, яка потім розгалужується. Відповідні результати обчислень можуть записуватися поруч з лініями, що відображають ці шляхи.

					ВКРМ-122.24.0014.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		57

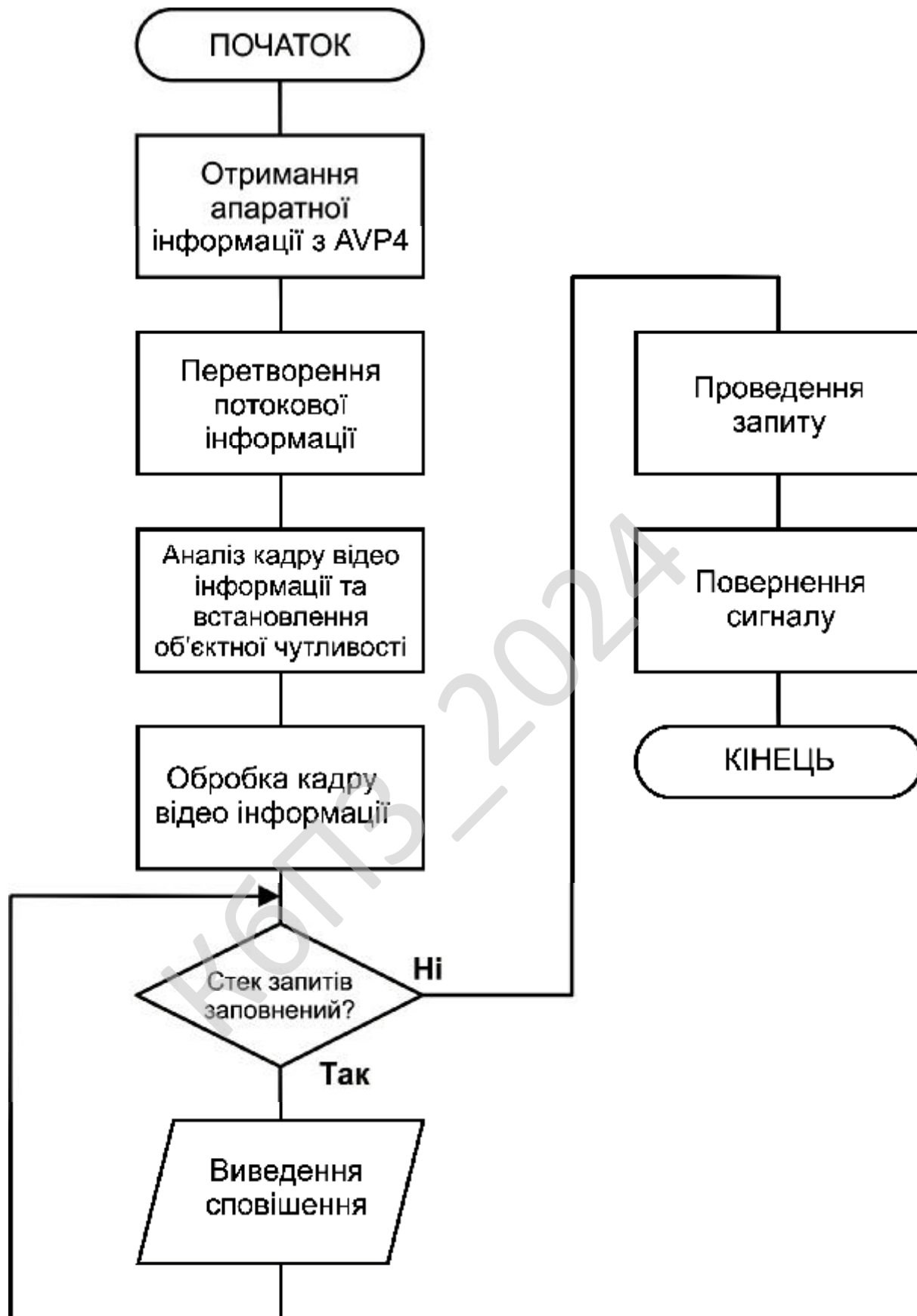


Рисунок 4.2 – Блок-схема роботи підпрограми

Зумовлений процес (підпрограма) це символ відображає виконання процесу, що складається з однієї або кількох операцій, що визначені в іншому місці програми (у підпрограмі, модулі). Всередині символу записується назва процесу і передані в нього дані.

Дані це перетворення у форму, придатну для обробки (введення) або відображення результатів обробки (виведення). Цей символ не визначає носія даних (для вказівки типу носія даних використовуються специфічні символи).

З'єднувач це символ відображає вихід в частину схеми і вхід з іншої частини цієї схеми. Використовується для обриву лінії та продовження її в іншому місці (приклад: поділ блок-схеми, що не поміщається на листі). Відповідні сполучні символи повинні мати одне (при тому унікальне) позначення.

Також при розробці магістерської дипломної роботи було використано наступні підходи UML:

- діаграма діяльності (діаграми поведінки типу);
- діаграма прецедентів (діаграми поведінки типу);
- діаграма класів;
- діаграма компонент;
- діаграма об'єктів;
- діаграма розгортання.

Діаграма діяльності. Це візуальне представлення графу діяльностей. Граф діяльностей є різновидом графу станів скінченного автомату, вершинами якого є певні дії, а переходи відбуваються по завершенню дій. Дія є фундаментальною одиницею визначення поведінки в специфікації. Дія отримує множину вхідних сигналів, та перетворює їх на множину вихідних сигналів.

Одна із цих множин, або обидві водночас, можуть бути порожніми. Виконання дії відповідає виконанню окремої дії. Подібно до цього, виконання діяльності є виконанням окремої діяльності, буквально, включно із виконанням тих дій, що містяться в діяльності. Кожна дія в діяльності може виконуватись один, два, або більше разів під час одного виконання діяльності. Щонайменше, дії

мають отримувати дані, перетворювати їх та тестувати, деякі дії можуть вимагати певної послідовності.

Специфікація діяльності (на вищих рівнях сумісності) може дозволяти виконання декількох (логічних) потоків, та існування механізмів синхронізації для гарантування виконання дій у правильному порядку.

Діаграма прецедентів це діаграма, на якій зображено відношення між акторами та прецедентами в системі. Також, перекладається як діаграма варіантів використання.

Діаграма прецедентів є графом, що складається з множини акторів, прецедентів (варіантів використання) обмежених границею системи (прямокутник), асоціацій між акторами та прецедентами, відношень серед прецедентів, та відношень узагальнення між акторами. Діаграми прецедентів відображають елементи моделі варіантів використання.

Суть даної діаграми полягає в наступному: проєктована система представляється у вигляді безлічі сутностей чи акторів, що взаємодіють із системою за допомогою так званих варіантів використання. Варіант використання (use case) використовують для описання послуг, які система надає актору. Іншими словами, кожен варіант використання визначає деякий набір дій, який виконує система при діалозі з актором.

При цьому нічого не говориться про те, яким чином буде реалізована взаємодія акторів із системою.

У мові UML є кілька стандартних видів відношень між акторами і варіантами використання:

- асоціації (association relationship);
- включення (include relationship);
- розширення (extend relationship);
- узагальнення (generalization relationship).

класів має більш високий ранг (ціле) і складається з декількох менших за рангом (частин).

Ставлення такого типу називають агрегацією; воно зараховане до відносин типу «має» (з урахуванням того, що об'єкт-ціле має кілька об'єктів-частин). Агрегація є окремим випадком асоціації і зображується у вигляді простої асоціації з незафарбованим ромбом з боку «цілого». Графічно агрегація представляється порожнім ромбом на блоці класу, і лінією, яка від цього ромба до міститься класу.

Композиція це більш суворий варіант агрегації. Відома також як агрегація за значенням.

Композиція має жорстку залежність часу існування екземплярів класу контейнера та примірників містяться класів. Якщо контейнер буде знищений, то весь його вміст буде також знищено. Графічно представляється як і агрегація, але з зафарбовани ромбиком.

Діаграма компонент в UML це діаграма, на якій відображаються компоненти, залежності та зв'язки між ними.

Діаграма компонент відображає залежності між компонентами програмного забезпечення, включаючи компоненти вихідних кодів, бінарні компоненти, та компоненти, що можуть виконуватись.

Модуль програмного забезпечення може бути представлено в якості компоненти. Деякі компоненти існують під час компіляції, деякі – під час компонування, а деякі під час роботи програми.

Діаграма компонент відображає лише структурні характеристики, для відображення окремих екземплярів компонент слід використовувати діаграму розгортання.

Компоненти об'єднуються разом використовуючи структурні зв'язки (assembly connector) щоб об'єднати інтерфейси двох компонент. Це ілюструє зв'язок типу «клієнт-сервер».

Структурна взаємодія – «зв'язок двох компонент, який передбачає, що один з них надає послуги, потрібні іншому компоненту».

При використанні діаграми компонент щоб показати внутрішню структуру компонента, клієнтські та серверні інтерфейси можуть утворювати пряме з'єднання з внутрішніми. Таке з'єднання називається з'єднанням делегації.

Діаграма об'єктів в UML це діаграма, що відображає об'єкти та їх зв'язки в певний момент часу. Діаграма об'єктів може розглядатись як окремий випадок діаграми класів, на якій можуть бути представлені як класи, так і екземпляри (об'єкти) класів. Схожою за змістом є діаграма взаємодії (collaboration diagram).

Діаграми об'єктів не мають власної нотації. Оскільки діаграми класів можуть відображати об'єкти, то діаграма класів, на якій відображено лише об'єкти, та не відображено класи, може вважатись діаграмою об'єктів.

Діаграма об'єктів відображає об'єкти та зв'язки в певний момент роботи програми. Об'єкти можуть містити інформацію про власні значення а не про описання. Для відображення загальних шаблонів об'єктів та зв'язків, що можуть багаторазово створюватись під час роботи програми, слід використовувати діаграму взаємодії, яка може відображати характеристики об'єктів та зв'язків. Екземпляр діаграми взаємодії створює діаграму об'єктів.

Діаграма об'єктів не відображає еволюцію системи під час роботи. Натомість, слід використовувати діаграми взаємодії з повідомленнями, або діаграми послідовності.

Діаграма розгортання (deployment diagram) це діаграма в UML, на якій відображаються обчислювальні вузли під час роботи програми, компоненти, та об'єкти, що виконуються на цих вузлах. Компоненти відповідають представленню робочих екземплярів одиниць коду. Компоненти, що не мають представлення під час роботи програми на таких діаграмах не відображаються; натомість, їх можна відобразити на діаграмах компонент. Діаграма розгортання відображає робочі екземпляри компонент, а діаграма компонент, натомість, відображає зв'язки між типами компонент.

Перед розглядом подробиць схеми роботи програми розглянемо виконані основні напрацювання.

```
# Імпорт бібліотек для роботи з потоковим відео та обробки зображень
import cv2
import numpy as np
import time
import os

# Клас для збору відеоданих з камер
class VideoCapture:
    def __init__(self, camera_id):
        self.camera_id = camera_id
        self.cap = cv2.VideoCapture(camera_id)

    def get_frame(self):
        ret, frame = self.cap.read()
        if ret:
            return frame
        else:
            return None

    def release(self):
        self.cap.release()

# Клас для обробки відеоданих (аналітика)
class VideoAnalytics:
    def __init__(self):
        pass

    def process_frame(self, frame):
        # Обробка кадру для виявлення об'єктів
        gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)
        # Додавання алгоритмів розпізнавання об'єктів
        return gray

# Клас для зберігання оброблених даних на сервері
class DataStorage:
    def __init__(self, storage_path):
        self.storage_path = storage_path
        if not os.path.exists(storage_path):
            os.makedirs(storage_path)
```

					ВКРМ-122.24.0014.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		65

```

def save_data(self, frame, frame_id):
    filename = os.path.join(self.storage_path, f"frame_{frame_id}.jpg")
    cv2.imwrite(filename, frame)

# Клас для передачі даних через мережу
class DataTransmission:
    def __init__(self, server_address):
        self.server_address = server_address

    def send_data(self, data):
        # Симуляція передачі даних через мережу
        print(f"Data sent to {self.server_address}")

# Клас для забезпечення доступу до даних користувачами
class DataAccess:
    def __init__(self, storage_path):
        self.storage_path = storage_path

    def get_data(self, frame_id):
        filename = os.path.join(self.storage_path, f"frame_{frame_id}.jpg")
        if os.path.exists(filename):
            return cv2.imread(filename)
        else:
            print("Frame not found")
            return None

# Основна система для управління всіма модулями
class VideoAnalyticsSystem:
    def __init__(self, camera_id, storage_path, server_address):
        self.capture = VideoCapture(camera_id)
        self.analytics = VideoAnalytics()
        self.storage = DataStorage(storage_path)
        self.transmission = DataTransmission(server_address)
        self.access = DataAccess(storage_path)
        self.frame_id = 0

    def run(self):
        while True:
            frame = self.capture.get_frame()
            if frame is None:
                break

```

```

# Обробка кадру
processed_frame = self.analytics.process_frame(frame)

# Зберігання обробленого кадру
self.storage.save_data(processed_frame, self.frame_id)

# Передача оброблених даних
self.transmission.send_data(processed_frame)

# Інкремент ідентифікатора кадру
self.frame_id += 1

# Затримка для симуляції реального часу
time.sleep(1)

def access_data(self, frame_id):
    return self.access.get_data(frame_id)

def stop(self):
    self.capture.release()

# Запуск системи відеоаналітики
if __name__ == "__main__":
    camera_id = 0 # Ідентифікатор камери
    storage_path = "./video_storage" # Шлях до місця зберігання даних
    server_address = "192.168.1.100" # Адреса сервера для передачі даних

    # Створення об'єкта системи
    system = VideoAnalyticsSystem(camera_id, storage_path, server_address)

    # Запуск обробки відео
    system.run()

# Приклад доступу до даних
frame = system.access_data(5)
if frame is not None:
    cv2.imshow("Accessed Frame", frame)
    cv2.waitKey(0)
    cv2.destroyAllWindows()

# Завершення роботи системи
system.stop()

```

					ВКРМ-122.24.0014.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		67

Система збирає дані з камери, обробляє їх для відеоаналітики, зберігає оброблені кадри на локальному сервері та передає їх на інший сервер для подальшого аналізу або використання. Вся архітектура побудована на класах, які відповідають за кожен модуль системи.

Розрахунки можуть включати:

1. Пропускна здатність мережі для передачі даних.
2. Розмір місця для зберігання залежно від роздільної здатності кадрів та частоти кадрів.
3. Час обробки одного кадру.

Проектні рішення підкріплюються правильним балансом між обробкою на місці (edge computing) та віддаленою аналітикою, забезпеченням надійної передачі даних і доступом користувачів до результатів.

При детальному розгляді рисунку 4.1 програма розбита на декілька важливих блоків, таких як:

- Блок ініціалізації динамічних бібліотек користувача.
- Блок підключення додаткових модулів – безкоштовні модулі для взаємодії з апаратурою Dlink, та обробки сигналу KVM.
- Блок читання файлів налаштування і управління ПЗ.
- Блок захоплення KVM потоку.
- Блок очікування дій користувача.
- Блок аналізу даних KVM.
- Блок обробки сигналу KVM.

4.2 Захист розробленого програмного забезпечення

Захист розробленого програмного забезпечення буде відбуватися за допомогою MARS, який є блочно-симетричним шифром з відкритим ключем. Розмір блоку при шифруванні 128 біта, розмір ключа може варіюватися від 128 до 448 біт включно (кратні 32бітам). Творці прагнули поєднати в своєму

					ВКРМ-122.24.0014.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		68

алгоритмі швидкість кодування і стійкість шифру. В результаті вийшов один з самих криптостійкий алгоритм з алгоритмів, які брали участь в конкурсі AES.

Алгоритм унікальний тим, що використовував практично всі існуючі технології, застосовувані в криптоалгоритмах, а саме:

- Найпростіші операції (додавання, віднімання, виключаюче або).
- Підстановки з використанням таблиці замін.
- Фіксований циклічний зсув.
- Залежний від даних циклічний зсув.
- Множення за модулем 2^{32} .
- Ключове забілювання.

Використання подвійного перемішування представляє складність для криптоаналізу, що деякі відносять до недоліків алгоритму. У той же час на даний момент не існує будь-яких ефективних атак на алгоритм, хоча деякі ключі можуть генерувати слабкі підключі.

Структура алгоритму

Автори шифру виходили з наступних припущень:

1. Вибір операцій. MARS був спроектований для використання на найсучасніших комп'ютерах того часу. Для досягнення найкращих захисних характеристик в нього були включені самі «сильні операції» підтримувані в них. Це дозволило добитися більшого відношення securityper-instruction для різних реалізацій шифру.

2. Структура шифру. Двадцятирічний досвід роботи в області криптографії підштовхнув творців алгоритму до думки, що кожен раунд шифрування грає свою роль в забезпеченні безпеки шифру. Зокрема, ми можемо бачити, що перший і останній раунди зазвичай сильно відрізняються від проміжних («центральных») раундів алгоритму в плані захисту від криптоаналітичних атак. Таким чином, при створенні MARSa використовувалася змішана структура, де перший і останній раунди шифрування істотно відрізняються від проміжних.

3. Аналіз. Швидше за все, алгоритм з гетерогенною структурою буде краще протистояти криптоаналітичних методам майбутнього, ніж алгоритм, всі раунди якого ідентичні. Розробники алгоритму MARS надали йому сильно гетерогенну структуру – раунди алгоритму дуже різняться між собою.

У шифрі MARS використовувалися такі методи шифрування:

1. Робота з 32-х бітними словами. Всі операції застосовуються до 32-бітовим словами. тобто вся початкова інформація розбивається на блоки по 32біта. (Якщо ж блок опинявся меншої довжини, то він доповнювався до 32біт)

2. Мережа Фейстеля. Творці шифру вважали, що це найкращий варіант поєднання швидкості шифрування і криптостійкості. В MARS використана мережа Фейстеля 3-го типу.

3. Симетричність алгоритму. Для стійкості шифру до різних атакам всі його раунди були зроблені повністю симетричними, тобто друга частина раунду є дзеркальне повторення першої його частини.

Структуру алгоритму MARS можна описати таким чином:

1. Попереднє накладення ключа: на 32-бітові субблоки A, B, C, D накладаються 4 фрагмента розширеного ключа $k_0 \dots k_3$ операцією складання за модулем 2^{32} .

2. Виконуються 8 раундів прямого перемішування (без участі ключа шифрування).

3. Виконуються 8 раундів прямого криптоперетворення.

4. Виконуються 8 раундів зворотного криптоперетворення. [2]

5. Виконуються 8 раундів зворотного перемішування, також без участі ключа шифрування.

6. Фінальне накладення фрагментів розширеного ключа $k_{36} \dots k_{39}$ операцією віднімання за модулем 2^{32} .

Пряме перемішування

У першій фазі на кожне слово даних накладається слово ключа, а потім відбувається вісім раундів змішування згідно з мережею Фейстеля третього типу

					ВКРМ-122.24.0014.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		70

спільно з деякими додатковими змішування. У кожному раунді ми використовуємо одне слово даних (зване, вихідним словом) для модифікації трьох інших слів (звані, цільовими словами). Ми розглядаємо чотири байта вихідного слова як індексів на двох S-блоків, S_0 і S_1 , кожен, що складається з 256 32-розрядних слів, а далі проводимо операції XOR або додавання даних відповідного S-блоку в три інших слова.

Якщо чотири байти вихідного слова b_0, b_1, b_2, b_3 (де b_0 є першим байтом, а b_3 є старшим байтом), то ми використовуємо b_0, b_2 , як індекси в блоку S_0 і байти b_1, b_3 , як індекси в S-блоці S_1 . Спочатку зробимо XOR S_0 до першого цільовим речі, а потім додамо S_1 до того ж слова. Ми також додаємо S_0 до другого цільовим слову і XOR блоку- S_1 до третього цільовим слову. У висновку, ми обертаємо вихідне слово на 24 біта вправо.

У наступному раунді ми обертаємо наявні у нас чотири слова: таким чином, нинішні перші цільове слово стає наступним вихідним словом, поточним другий цільове слово стає новим першим цільовим словом, третє цільове слово стає наступний другий цільовим словом, і поточне вихідне слово стає третім цільовим словом.

Більш того, після кожного з чотирьох раундів ми додаємо одне з цільових слів назад у вихідне слово. Зокрема, після першого і п'ятого раундів ми додав третю цільове слово назад у вихідне слово, а після другого і шостого раунду ми додаємо першої цільової слово назад у вихідне слово. Причиною цих додаткових операцій змішування, є ліквідація декількох простих диференціальних криптоатаки в фазі перемішування, щоб порушити симетрію у фазі змішування та отримати швидкий потік.

Криптографічне ядро

Криптографічне ядро MARS – мережа Фейстеля 3-го типу, що містить в собі 16 раундів. У кожному раунді ми використовуємо ключову E-функцію, яка є комбінацією множень, обертань, а також звернень до S-блоків. Функція приймає на вхід слово даних, а повертає три слова, з якими згодом буде здійснена операція

					ВКРМ-122.24.0014.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		71

додавання або XOR до інших трьох слів даними. У доповненні вихідне слово обертається на 13 біт вліво.

Для забезпечення, серйозного опору до криптоатаки, три вихідних значення E-функції (O_1 , O_2 , O_3) використовуються в перших восьми раундах і в останніх восьми раундах в різних порядках. У перші вісім раундів ми додаємо O_1 і O_2 до першого і другого цільовим речі, відповідно, і XOR O_3 у третьому цільовим слову. За останні вісім раундів, ми додаємо O_1 і O_2 до третього і другого цільовим речі, відповідно, і XOR O_3 до першого цільовим слову.

E-функція

E-функція приймає як вхідні дані одне слово даних і використовує ще два ключових слова, виробляючи на виході три слова. У цій функції ми використовуємо три тимчасові змінні, що позначаються L, M і R (для лівої, середньої та правої).

Спочатку ми встановлюємо в R значення вихідного слова зміщеного на 13 біт вліво, а в M – сума вихідних слів і першого ключового слова. Потім ми використовуємо перші дев'ять бітів M як індекс до однієї з 512S-блоків (яке виходить суміщенням S_0 і S_1 змішуванням фази), і зберігаємо в L значення відповідного S-блоку.

Потім помножимо друге ключове слово на R, зберігши значення в R. Потім обертаємо R на 5 позицій вліво (так, 5 старших бітів стають 5 нижніми бітами R після обертання). Тоді ми робимо XOR R в L, а також переглядаємо п'ять нижніх біт R для визначення величини зсуву (від 0 до 31), і обертаємо M вліво на цю величину. Далі ми обертаємо R ще на 5 позицій вліво і робимо XOR в L. У висновку, ми знову дивимося на 5 молодших бітів R, як на величину обертання і обертаємо L на цю величину вліво. Таким чином результат роботи E-функції – 3 слова (по порядку): L, M, R.

Зворотне перемішування

Зворотне перемішування практично збігається з прямим перемішуванням, за винятком того факту, що дані обробляються в зворотному порядку. Тобто,

					ВКРМ-122.24.0014.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		72

якби ми поєднали пряме і зворотне перемішування так, щоб їх виходи і входи були б з'єднані в зворотному порядку ($D[0]$ прямого і $D[3]$ зворотного, $D[1]$ прямого і $D[2]$ зворотного), то не побачили б результату перемішування. Як і в прямому змішування, тут ми теж використовуємо одне вихідне слово і три цільових. Розглянемо чотири перших байта вихідного слова: b_0, b_1, b_2, b_3 . Будемо використовувати b_0, b_2 як індекс до S-блоку – S_1 , а b_1, b_3 для S_0 . Зробимо XOR $S_1[b_0]$ в перше цільове слово, віднімемо $S_0[b_3]$ з другого слова, віднімемо $S_1[b_2]$ з третього цільового слів і потім зробимо XOR $S_0[b_1]$ також до третього цільового слова. Нарешті, ми повертаємо початкове слово на 24 позицій вліво. Для наступного раунду ми обертаємо наявні слова так, щоб нинішнє перше цільове слово стало наступним вихідним словом, поточне друге цільове слово стало першим цільовим словом, поточне третє цільове слово стало другим цільовим словом, і поточне вихідне слово стало третім цільовим словом. Крім того, перед одним з чотирьох «особливих» раундів ми віднімаємо одне з цільових слів з вихідного слова: перед четвертим і восьмим раундами ми віднімаємо першої цільової слово, перед третьому і сьомим раундами ми віднімаємо третя цільова слово з вихідного.

Дешифрування

Процес декодування обернений процесу кодування. Код дешифрування схожий (але не ідентичний) на код шифрування.

					ВКРМ-122.24.0014.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		73

5 МЕТОДИКА ВПРОВАДЖЕННЯ СИСТЕМИ В ПРОМИСЛОВУ ЕКСПЛУАТАЦІЮ

На рисунку 5.1 зображено інтерфейс програмного забезпечення, розробленого у результаті виконання магістерської дипломної роботи.

Розроблене програмне забезпечення системи збору, представлення, обробки, зберігання, передачі та доступу до інформації мережевої відеоаналітики складається з наступних функціональних блоків:

- Навігаційне меню: Файл; Камери; Мережна відеоаналітика; Розробник.
- Вікно виведення відео даних.
- Функціональних кнопок ПЗ.

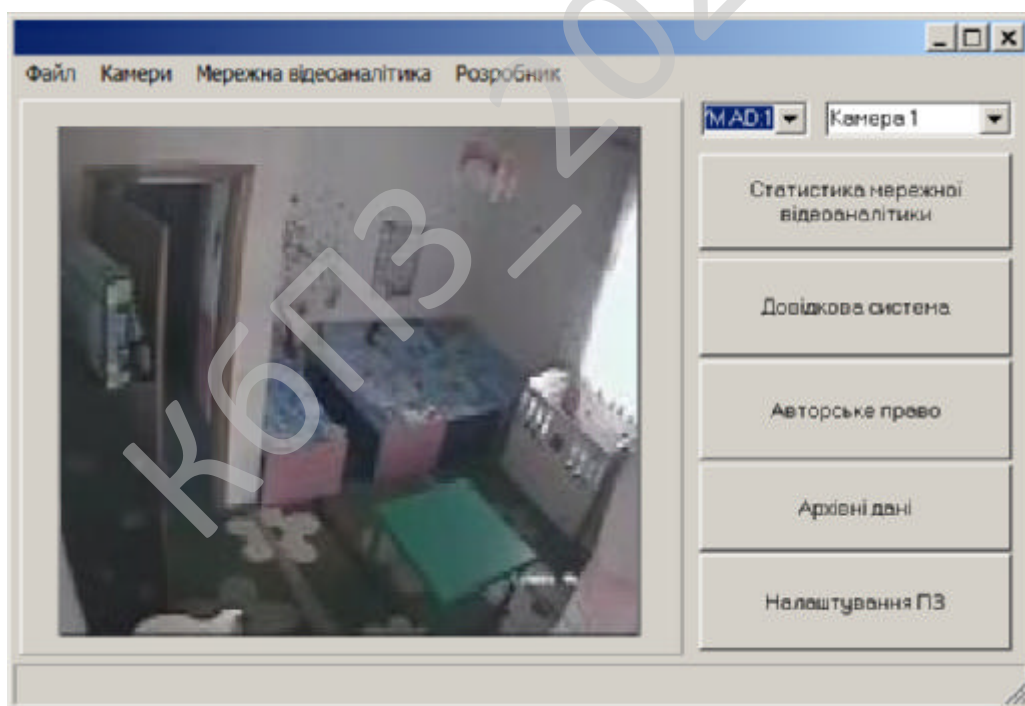


Рисунок 5.1 – Головне вікно розробленого ПЗ

Для перегляду короткої довідки про програму слід натиснути на основному вікні кнопку авторського права, після чого на екрані з'явиться вікно показане на рисунку 5.2.

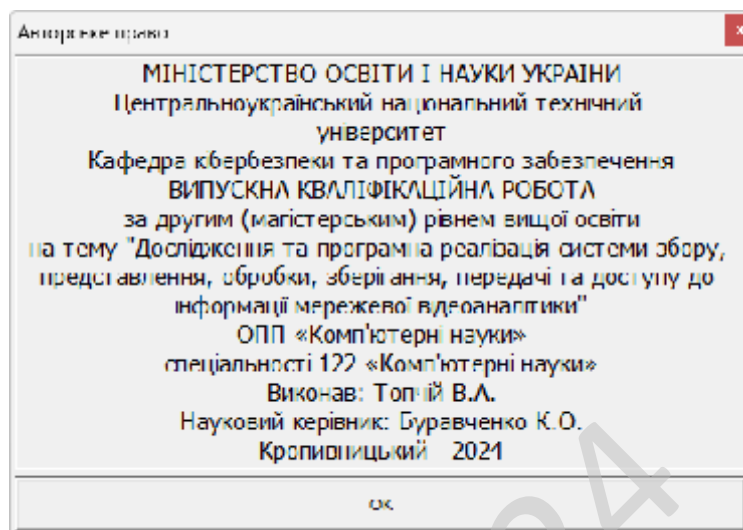


Рисунок 5.2 – Вікно розробника ПЗ

Під час роботи над програмою було проведено тестування програмного забезпечення, тобто технічне дослідження, призначене для виявлення інформації про якість продукту відносно контексту, в якому воно має використовуватись.

Тестування включає як процес пошуку помилок або інших дефектів, так і випробування програмних складових з метою їх оцінки.

Проводилась оцінка:

- відповідності поставленим вимогам;
- правильна відповідь для усіх можливих вхідних даних;
- виконання функцій за прийнятний час;
- практичність;
- сумісність з ОС та стороннім ПЗ.

Оскільки число можливих тестів для програмних компонент практично нескінченне, тому стратегія тестування полягала в тому, щоб провести всі можливі тести з урахуванням наявного часу та ресурсів.

					ВКРМ-122.24.0014.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		75

Як результат ПЗ тестувалось стандартним виконанням програми з метою виявлення помилок або інших дефектів.

Проводилось тестування форматом чорної скриньки.

Основне місце програми тестів «чорної скриньки» – інтерфейс ПЗ. Відомі: функції програми. Досліджується: робота кожної функції на всій області визначення.

Ці тести демонструють:

- Як виконуються функції програми.
- Як приймаються вихідні дані.
- Як виробляються результати.
- Як зберігається цілісність зовнішньої інформації.

При тестуванні «чорної скриньки» розглядаються системні характеристики програм, ігнорується їхня внутрішня логічна структура. Вичерпне тестування, як правило, неможливе.

Наприклад, якщо в програмі 10 вхідних величин і кожна приймає по 10 значень, то кількість тестових варіантів становитиме 10^{10} . Тестування «чорної скриньки» не реагує на багато особливостей програмних помилок.

Тестування «чорної скриньки» (функціональне тестування) дозволяє отримати комбінації вхідних даних, які забезпечують повну перевірку всіх функціональних вимог до програми.

Програмний виріб тут розглядається як «чорна скринька», чию поведінку можна визначити тільки дослідженням його входів та відповідних виходів. При такому підході бажано мати:

- Набір, утворений такими вхідними даними, які призводять до аномалій у поведінці програми (назвемо його ІТс).
- Набір, утворений такими вхідними даними, які демонструють дефекти програми (назвемо його ОТ).

Будь-який спосіб тестування «чорної скриньки» повинен:

- Виявити такі вхідні дані, які з високою ймовірністю належать набору ІТс;

					ВКРМ-122.24.0014.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		76

– Сформулювати такі очікувані результати, які з високою імовірністю є елементами набору ОТ.

Принцип «чорної скриньки» не альтернативний принципу «білої скриньки». Скоріше це доповнює підхід, який виявляє інший клас помилок.

Тестування «чорної скриньки» забезпечує пошук наступних категорій помилок:

- Некоректних чи відсутніх функцій;
- Помилки інтерфейсу;
- Помилки у зовнішніх структурах даних або в доступі до зовнішньої бази даних;
- Помилки характеристик (необхідна ємність пам'яті і т.д.);
- Помилки ініціалізації та завершення.

Обрано умови розповсюдження – Freeware.

Це власницьке програмне забезпечення, котре можна Безоплатно використовувати протягом необмеженого терміну без обмежень у функціональності, і поширюване без сирцевих кодів.

Автори такого програмного забезпечення, як правило, хочуть «дати щось спільноті», але хочуть також контролювати його подальшу розробку. Іноді, коли програмісти вирішують припинити розробку, вони передають сирцевий код іншим програмістам, або ж спільноті як вільне програмне забезпечення.

Дуже часто плутають поняття «безплатне програмне забезпечення» та «вільне програмне забезпечення», хоча вони суттєво відрізняються.

Безплатне програмне забезпечення можна безоплатно встановлювати та використовувати (іноді з певними обмеженнями, як, наприклад, «безплатне для домашнього або некомерційного вжитку»), в той час як вільне програмне забезпечення можна продавати за будь-яку суму, але при тому, у користувача, котрий його отримує, повинні бути права на вивчення, модифікацію та поширення сирцевих кодів одержаної програми.

					ВКРМ-122.24.0014.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		77

6 НАУКОВА НОВИЗНА

У випускній кваліфікаційній роботі за другим (магістерським) рівнем вищої освіти розроблено програмне забезпечення, яке призначено для системи збору, представлення, обробки, зберігання, передачі та доступу до інформації мережевої відеоаналітики.

Метою розробки є дослідження та програмна реалізація системи збору, представлення, обробки, зберігання, передачі та доступу до інформації мережевої відеоаналітики.

Об'єктом дослідження є процес збору, представлення, обробки, зберігання, передачі та доступу до інформації мережевої відеоаналітики.

Предметом дослідження є методи збору, представлення, обробки, зберігання, передачі та доступу до інформації мережевої відеоаналітики.

Методи дослідження базуються на методах обробки великих даних, методах математичної статистики, методах розробки програмного забезпечення.

Наукова новизна отриманих результатів. У процесі рішення завдань, обумовлених цілями дослідження, отримані наступні результати:

- Удосконалено метод збору, представлення, обробки, зберігання, передачі та доступу до інформації мережевої відеоаналітики.
- Розроблено вітчизняний продукт збору, представлення, обробки, зберігання, передачі та доступу до інформації мережевої відеоаналітики, який має більш широкі можливості, на відміну від існуючих аналогів.

					VKPM-122.24.0014.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		78

7 МАРКЕТИНГОВЕ ТА ЕКОНОМІЧНЕ ОБГРУНТУВАННЯ ІТ-ПРОЄКТУ

7.1 Визначення цільової аудиторії кінцевого готового продукту

Результати дослідження та програмної реалізації системи для збору, представлення, обробки, зберігання, передачі та доступу до інформації мережевої відеоаналітики (рисунок 7.1).



Рисунок 7.1 – Цільова аудиторія

Такі рішення також можуть знайти застосування у сферах охорони здоров'я, освіти, де потрібен інтелектуальний аналіз відеоданих, а також у різноманітних прикладних наукових дослідженнях.

7.2 Оцінка привабливості шляхом застосування методів експертних оцінок

Оцінка привабливості програмної реалізації системи збору, представлення, обробки, зберігання, передачі та доступу до інформації мережевої відеоаналітики за допомогою методу експертних оцінок має пройти наступні етапи.

1. Визначення критеріїв оцінки привабливості.

Критерії можуть включати:

- функціональність – наскільки повно реалізовані необхідні функції.
- продуктивність – ефективність обробки та зберігання відеоданих.
- безпека – рівень захисту даних від несанкціонованого доступу.
- сумісність – інтеграція з іншими системами та підтримка різних форматів даних.
- зручність використання – інтерфейс та зручність налаштування.
- ринковий потенціал – попит на дане рішення серед цільової аудиторії.
- інноваційність – новизна та унікальність пропозиції на ринку.

2. Вибір експертів.

Зазвичай обирають фахівців із різних сфер: розробників програмного забезпечення, аналітиків, маркетологів, експертів з кібербезпеки, представників цільових споживачів.

3. Присвоєння ваги кожному критерію.

Залежно від важливості, кожному критерію призначають вагу: функціональність – 0.25, продуктивність – 0.20, безпека – 0.20, сумісність –

					ВКРМ-122.24.0014.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		80

0.15, зручність використання – 0.10, ринковий потенціал – 0.05, інноваційність – 0.05.

4. Проведення оцінки.

Кожен експерт оцінює за 10-бальною шкалою (де 1 – мінімальна, а 10 – максимальна відповідність).

Таблиця 7.1 – Зведені результати експертних оцінок

Критерій	Експерт 1	Експерт 2	Експерт 3	Середня оцінка
Функціональність	0.25	9	8	9
Продуктивність	0.20	8	7	8
Безпека	0.20	9	9	8
Сумісність	0.15	7	8	7
Зручність	0.10	8	7	8
Ринковий потенціал	0.05	6	7	6

5. Підрахунок загальної оцінки.

Загальну оцінку отримують шляхом множення середніх значень кожного критерію на їх ваги та підсумування всіх результатів: $(0.25 \times 8.67) + (0.20 \times 7.67) + (0.20 \times 8.67) + (0.15 \times 7.33) + (0.10 \times 7.67) + (0.05 \times 6.33) + (0.05 \times 7.33)$. Загальна оцінка становить 7.99.

6. Інтерпретація результату

Якщо результат вищий за 8, проект має високу привабливість; якщо від 6 до 8 – середню; нижче 6 – низьку. Цей підхід допомагає обґрунтовано оцінити привабливість та обрати оптимальні напрями розвитку для успішної реалізації системи. За наведеним прикладом експертної оцінки можемо говорити про високу привабливість проекту.

7.3 Вибір методу оцінки вартості ПЗ

Для оцінки вартості програмної реалізації системи збору, представлення, обробки, зберігання, передачі та доступу до інформації мережевої відеоаналітики можна використати кілька методів, залежно від специфіки проекту та доступних ресурсів.

SOCOMO II або метод функціональних точок (FPA) представлені графічно на рисунку 7.2, де наведено варіації зручного їх використання.

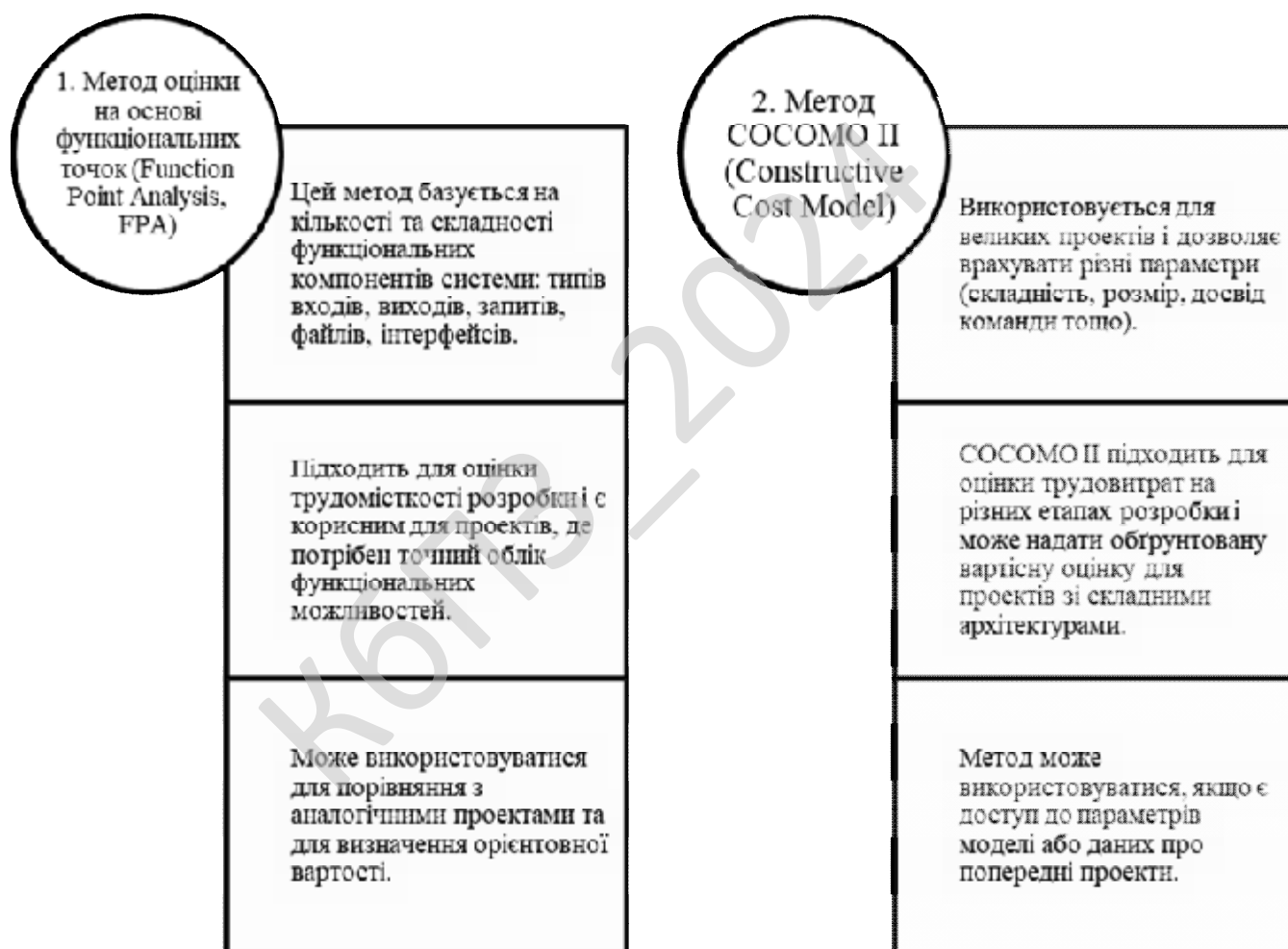


Рисунок 7.2 – Оптимальні методи оцінки вартості ПЗ

Для розробки системи збору, обробки та зберігання даних відеоаналітики, особливо з високими вимогами до безпеки і продуктивності, SOCOMO II або

метод функціональних точок (FPA) є оптимальними для отримання точнішої оцінки. Експертна оцінка також буде корисною для уточнення особливих вимог безпеки та функціональності, що допоможе уникнути можливих перевитрат.

7.4 Розрахунок економічної ефективності від впровадження реалізованого ПЗ як фактору його привабливості

Економічна ефективність від впровадження системи для проведення відеотрансляцій може бути виражена через різні аспекти економії коштів та підвищення доходів. Нижче наведено ключові фактори на рисунку 7.3.

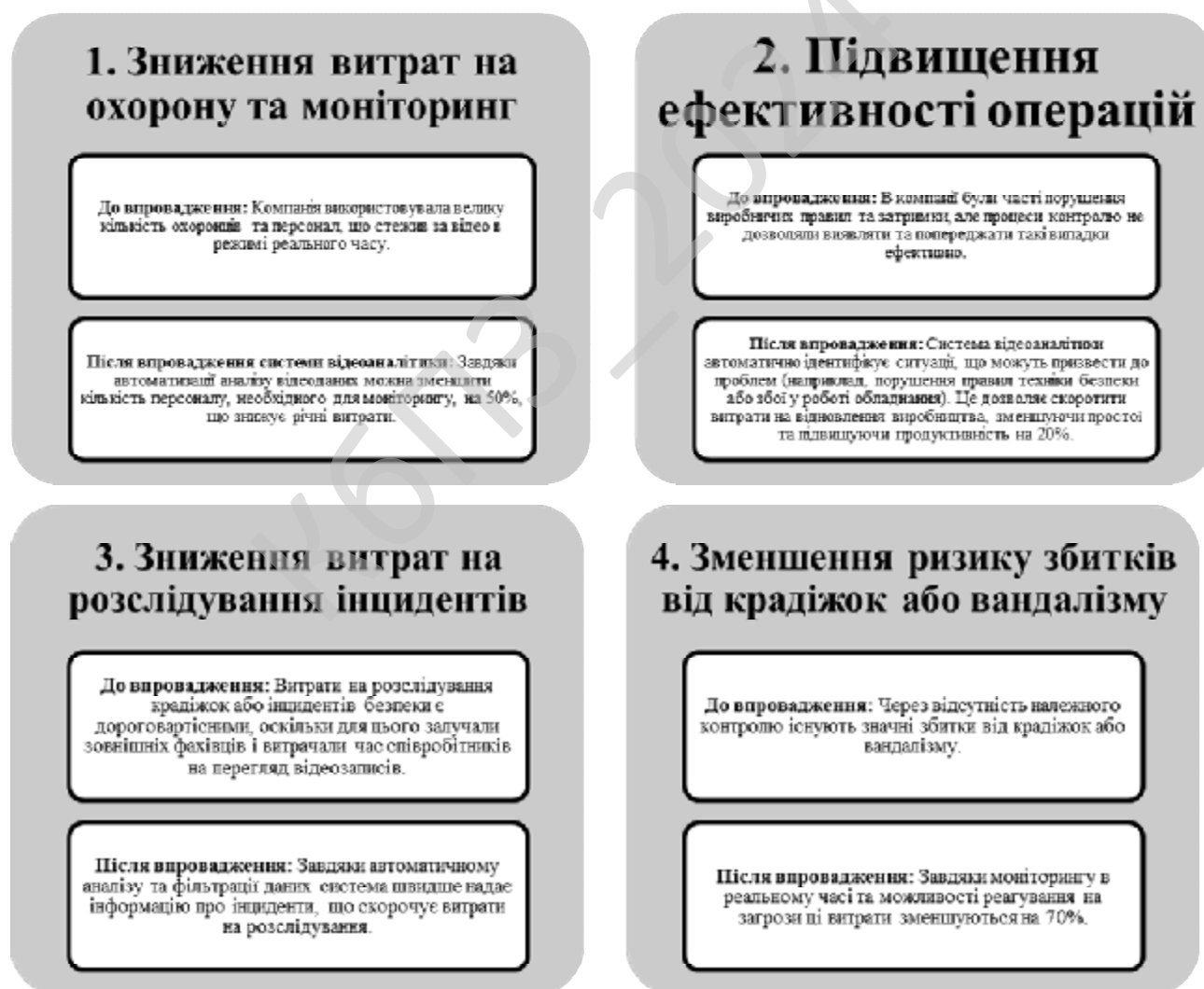


Рисунок 7.3 – Економічна ефективність від реалізації проекту для клієнта

Отже, інвестиції у впровадження системи матимуть період окупності менше року, а подальша економія забезпечить значний економічний ефект у майбутньому.

7.5 Пропозиція алгоритму просування проєкту розробки ПЗ

Для успішного просування проєкту програмної реалізації системи мережевої відеоаналітики можна використовувати поетапний алгоритм (рисунок 7.4), що охоплює визначення цільової аудиторії, створення інформаційної бази, маркетинг та залучення партнерів.

Цей алгоритм дозволяє системно підійти до просування продукту, використовуючи всі можливі канали комунікації для охоплення цільової аудиторії.

7.6 Оптимізація каналів збуту та шляхів реалізації ПЗ

Для оптимізації каналів збуту та шляхів реалізації проєкту мережевої відеоаналітики важливо зосередитися на ефективних каналах дистрибуції, співпраці з ключовими партнерами та підходах, що мінімізують витрати та підвищують охоплення. Пропозиції для оптимізації згруповано та подано на рисунку 7.5.

Цей комплексний підхід забезпечує оптимізацію каналів збуту та шляхів реалізації, що дозволяє охопити різноманітні сегменти ринку, підвищити економічну ефективність і забезпечити стійке зростання.

7.7 Визначення ключових факторів успіху конкретного проєкту

Ключовими факторами успіху для проєкту програмної реалізації системи мережевої відеоаналітики є комплексний підхід, що охоплює технічні, організаційні та ринкові аспекти (на рис. 7.6).

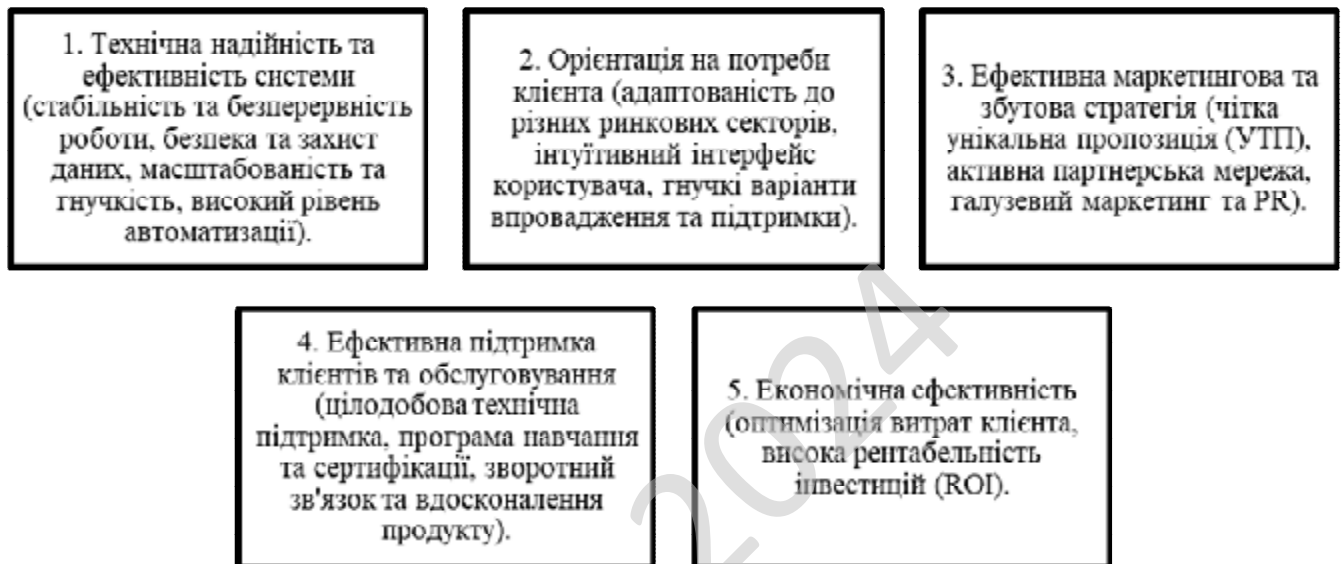


Рисунок 7.6 – Ключові фактори успіху проєкту

Ці ключові фактори успіху дозволяють проєкту стати конкурентоздатним, привабливим для клієнтів і фінансово вигідним.

					ВКРМ-122.24.0014.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		87

8 ЗАХОДИ З ОХОРОНИ ПРАЦІ ТА ТЕХНІКИ БЕЗПЕКИ

8.1 Вступ

В охорону праці включають санітарно-гігієнічні, лікувально-профілактичні та організаційно-технічні системи правових і соціально-економічних заходів.

В кожній ІТ компанії є трудові відносини з працівниками. Згідно закону України “Про охорону праці” [3] кожна компанія впроваджує заходи з охорони праці. Реалізується трудові відносини з вживанням необхідних засобів з охорони праці та розробки відповідних документів:

- Інструкцій з охорони праці по кожній професії і загальні.
- Положення про охорону праці.
- Накази з охорони праці.
- Журнали реєстрації та інструктажу.

Роботодавець створює відділ який працює відповідно до типового положення, яку затверджується центральним органом виконавчої влади і забезпечує виконання вимог державної політики у сфері охорони праці.

За недотриманням вимог, керівники ІТ компаній можуть бути притягнуті до відповідальності, яка виглядає у виді накладання штрафу. Якщо в результаті порушення умов охорони праці є постраждалі працівники то керівні особи ІТ компаній притягуються до кримінальної відповідальності.

Законом України “Про охорону праці” [3] регламентуються загальні положення державної політики в галузі охорони праці, а конкретизуються ці положення нормативно-правовими актами про охорону праці, зокрема Наказом Міністерства соціальної політики України 14.02.2018 № 207, який зареєстровано в Міністерстві юстиції України 25 квітня 2018 р. за №508/31960 «Про затвердження Вимог щодо безпеки та захисту здоров'я працівників під час роботи з екранними пристроями» [5], яким затверджено нормативно-правовий

					ВКРМ-122.24.0014.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		88

акт з охорони праці НПАОП 0.00-7.15-18, «Правила охорони праці під час експлуатації електронно-обчислювальних машин», та «Державні санітарні правила і норми роботи з візуальними дисплейними терміналами електронно-обчислювальних машин» ДСанПіН 3.3.2-007-98.

Програмісти у процесі роботи мають негативний вплив на органи зору, а також мають значну розумову напругою і нервово-емоційне навантаження. Руки (суглоби пальців та м'язи рук) при роботі з клавіатурою мають теж істотне навантаження. До шкідливих факторів, які впливають на робітників галузі інформаційних технологій (ІТ) спеціалісти відносять високочастотні електромагнітні коливання (випромінювання) роботи апаратної частини ЕОМ та виділення шкідливих газів.

Ці шкідливі фактори можуть привести до професійних захворювань.

Розглянемо шкідливі чинники роботи програмістів керуючись наступними нормативно-правовими актами: «Державні санітарні правила і норми роботи з візуальними дисплейними терміналами електронно-обчислювальних машин» ДСанПіН 3.3.2-007-98 [5], та «Вимоги щодо безпеки та захисту здоров'я працівників під час роботи з екранними пристроями» НПАОП 0.00-7.15-18.

Умови праці програміста включають наступні фактори:

- параметри повітряного середовища в приміщенні;
- вентиляція приміщення;
- освітлення приміщення;
- параметри повітряного середовища в приміщенні, тощо.

Щоб запропонувати заходи щодо зменшення негативного впливу комп'ютера на організм людини визначемо фактори, які можуть викликати професійне захворювання і впливають на працездатність програміста.

8.2 Аналіз санітарно-гігієнічних умов праці на робочому місці програміста

Розглянемо умови праці у приміщенні, в якому працюють програмісти. Геометричні розміри приміщення наведено у таблиці 8.1.

У зазначеному приміщенні працюють 9 людей. За даними, які наведено у табл. 8.1, та табл. 8.2, можна зробити висновок, що площа та об'єм приміщення у розрахунку на одно робоче місце програміста не відповідають нормативним вимогам ДСанПіН 3.3.2-007-98 «Державні санітарні правила і норми роботи з візуальними дисплейними терміналами електронно-обчислювальних машин» [2], але відповідають нормативним вимогам Наказу Міністерства соціальної політики України № 207, від 14.02.2018 «Про затвердження Вимог щодо безпеки та захисту здоров'я працівників під час роботи з екранними пристроями» та НПАОП 0.00-1.28-10 «Правила охорони праці під час експлуатації електронно-обчислювальних машин»). Тим чином можна зробити висновок, що санітарно-гігієнічні умови праці на робочому місці програміста відповідають вимогам.

Таблиця 8.1 – Розміри приміщення

Найменування	Значення, м
Ширина	7
Довжина	8
Висота	3,3

Таблиця 8.2 – Площа та обсяг приміщення, на одного працюючого*

Геометрична характеристика	Одиниця виміру	Нормативне значення*	Фактичне значення
Площа, S	м ²	не менше 6.0	6,2
Об'єм, V	м ³	не менше 20.0	20,5

* Згідно ДСанПіН 3.3.2.007-98 (Державні санітарні правила і норми роботи з візуальними дисплейними терміналами електронно-обчислювальних машин).

Проведений аналіз показує, що показники мікроклімату в приміщенні відповідають установленим нормам. Штучне опалення застосовується у холодний період року.

В літню пору застосовується кондиціонер.

Для боротьби з пилом робляться регулярні провітрювання та вологі прибирання приміщенні.

У приміщенні знаходяться наступні джерела шуму: принтер Xerox WorkCentre 3025BI (3025VBI), електродвигуни вентиляторів ЕОМ.

Одним з найважливіших факторів, які впливають на ефективність трудової діяльності людини, та попереджають травматизм і професійні захворювання програмістів є освітлення на робочому місці.

З 2019 року діють Державні будівельні норми України “Природне і штучне освітлення” – ДБН В.2.5-28:2018 [1], у яких прописані вимоги до використання всіх освітлювальних приладів, у т.ч. світлодіодних.

Працю працівника, який постійно працює за комп’ютером, згідно ДБН В.2.5-28:2018 [1], можна віднести до роботи з малою точністю (найменший розмір об’єкта розрізнення від 1 до 5 мм) V-го розряду зорової роботи, з великою контрастністю об’єкта розрізнення (символів на екрані дисплея), з темним тлом (під розряд зорової роботи В). Приміщення можна віднести до 1-ої групи приміщень, у яких проводиться розрізнення об’єктів зорової роботи при фіксованому напрямку лінії зору того, що працює на робочу поверхню. Для такого типу приміщень і розряду зорової роботи нормоване значення коефіцієнта природної освітленості (КПО) робочої поверхні (при поєднаному, спільному освітленні), повинен становити не більше 1,5%, освітленість при штучному висвітленні повинна становити 300 Лк. [1], Крім того все поле зору повинне бути освітлено достатньо рівномірно – ця основна гігієнічна вимога. Так як яскраве світло на ділянці периферійного зору значно збільшує напруженість очей і, як наслідок, призводить до їх швидкої стомлюваності, ступінь освітлення приміщення і яскравість екрану комп’ютера повинні бути приблизно однаковими.

						ВКРМ-122.24.0014.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата			92

8.3 Розробка заходів з умов поліпшення охорони праці

Згідно аналізу умов праці в розглянутому приміщенні, ми одержали наступні результати:

– розмірі приміщення, у розрахунку на одному працюючого, відповідають нормативам;

– мікроклімат відповідає нормативному значенню;

– акустичні умови роботи не перевищують нормативних значень;

Таким чином можна припустити, що основною причиною можливого зниження працездатності програміста є психофізіологічний фактор, тому основна пропозиція буде така: дотримання позитивної психологічної атмосфери в колективі та регламентованого режиму праці та відпочинку, організація робочого місця з урахуванням ергономічних вимог.

Рекомендовані заходи: регулярні періодичні наочні огляди персоналом шляхів для евакуації людей із приміщення, відповідно до плану евакуації (який повинен розташовуватись на видному місці у приміщенні), включення до колективного договору мінімально можливого вмісту аптечок з обов'язково наявністю масок-клапанів, або іншого спорядження для штучного дихання. Регулярна періодична перевірка параметрів заземлення та занулення (вимірювання опору ланцюга) [9].

Регулярна наочне знайомство персоналу із шляхами для евакуації людей із приміщення відповідно до плану евакуації, забезпечення розподільних щитів спеціальними розетками з заземлюючими контактами; організація заземлення всіх приладів і пристроїв, які працюють при нарузі вище 36 В.

Так як при ураженні електричним струмом у людини може статися фібриляція шлуночків серця, в організації бажано мати дефібрилятор і підготовлений персонал для роботи з ним.

					ВКРМ-122.24.0014.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		93

8.4 Розрахункова частина

Система освітлення робочого місця користувача ПК має відповідати наступним вимогам (рис. 8.1).

Проведемо розрахунок штучного освітлення за методом коефіцієнта використання світлового потоку для приміщення ширина якого складає 7 м, довжина – 8 м, висота – 3,3 м.

У зазначеному приміщенні працює 9 людей.



Рисунок 8.1 – Вимоги до системи освітлення робочого місця користувача ПК

Для того, щоб визначити потрібну кількість світильників, які повинні забезпечити нормований рівень освітленості, визначимо світловий потік, що падає на робочу поверхню за формулою:

$$F = E \cdot S \cdot K \cdot Z / n,$$

де: F – світловий потік, що розраховується, Лм;

E – нормована мінімальна освітленість, Лк; $E = 300$ Лк;

S – площа освітлюваного приміщення.

K – коефіцієнт запасу, що враховує зменшення світлового потоку лампи в результаті забруднення світильників в процесі експлуатації (його значення залежить від типу приміщення і характеру робіт, що проводяться в ньому, в нашому випадку $K = 1,5$);

Z – відношення середньої освітленості до мінімальної (зазвичай приймається рівним 1.1... 1.2, в нашому випадку $Z = 1,1$);

n – коефіцієнт використання світлового потоку, (відношення світлового потоку, що падає на розрахункову поверхню, до сумарного потоку всіх ламп і обчислюється в долях одиниці; залежить від характеристик світильника, розмірів приміщення, забарвлення стін і стелі, що характеризуються коефіцієнтами відбиття від стін ($\rho_{\text{стін}}$) і стелі ($\rho_{\text{стелі}}$), значення коефіцієнтів дорівнюють:

$$\rho_{\text{стін}} = 50\% \text{ і } \rho_{\text{стелі}} = 50\%.$$

Обчислимо індекс приміщення за формулою:

$$i = S / (h \cdot (A + B)),$$

де: S – площа приміщення, $S = 56$ м²;

h – розрахункова висота підвісу, $h = 3,2$ м. (співпадає з висотою стелі, т.я. лампи освітлення закріплюються на стелі);

A – ширина приміщення, $A = 7$ м;

B – довжина приміщення, $B = 8$ м.

Підставимо всі значення у формулу та визначимо індекс приміщення:
 $i = 1,1$.

Знаючи індекс приміщення, знаходимо $n = 0,46$ (з табличних даних коефіцієнтів використання світлового потоку (n) світильників з відповідним типом лампам) [8]. Підставимо всі значення у формулу, визначимо світловий потік: $F = 60260$ Лм.

					ВКРМ-122.24.0014.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		95

Для розрахунку будемо використовувати стельові *світлодіодні панелі* Призма-72 6400К, світловий потік яких $F_{л} = 7200$ Лм.

Число ламп визначається по формулі:

$$N = F / F_{л}$$

де: F – світловий потік,

$F_{л}$ – світловий потік однієї лампи.

Підставимо всі значення у формулу та визначемо індекса приміщення:

$$N = 60260 / 7200 = 8,3 \text{ шт.}$$

Приймаємо необхідну кількість *світлодіодних світильників* 9 шт.

8.5 Висновки до розділу

Дотримання всіх необхідних умов праці не лише сприяє збереженню здоров'я працівників, а також підвищує ефективність виробництва в цілому.

З цих міркувань було здійснено аналіз умов праці, призначеного для праці програмістів, проведено розгляд небезпечних та шкідливих факторів, що негативно впливають на програмістів під час роботи. Виконано розрахунок штучного освітлення, як одного з ключових факторів впливу на працездатність та здоров'я програміста. Розроблено заходи з охорони праці.

					ВКРМ-122.24.0014.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		96

9 ОСНОВНІ ВИСНОВКИ

Програмне забезпечення, створене в результаті виконання випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти, призначено для системи збору, представлення, обробки, зберігання, передачі та доступу до інформації мережевої відеоаналітики.

В межах України в недостатній мірі представлені вітчизняні розробки в цій області.

У випускній кваліфікаційній роботі за другим (магістерським) рівнем вищої освіти наведені теоретичне узагальнення й рішення наукового завдання дослідження методів збору, представлення, обробки, зберігання, передачі та доступу до інформації мережевої відеоаналітики.

Рішення даного завдання полягало у вирішенні наступних задач:

- Був проведений огляд існуючих систем збору, представлення, обробки, зберігання, передачі та доступу до інформації мережевої відеоаналітики.
- Досліджена система збору, представлення, обробки, зберігання, передачі та доступу до інформації мережевої відеоаналітики.
- На основі отриманих результатів досліджень створена програмна реалізація системи збору, представлення, обробки, зберігання, передачі та доступу до інформації мережевої відеоаналітики.

Розроблені під час виконання випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти алгоритми дозволяють успішно вирішувати завдання збору, представлення, обробки, зберігання, передачі та доступу до інформації мережевої відеоаналітики.

Проведено аналіз предметної галузі в ході якого були виявлені об'єкти, взаємодія яких носить істотний характер для функціональної діяльності предметної галузі, і їхні основні характеристики; побудована алгоритм і вибраний середовище розробки.

					ВКРМ-122.24.0014.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		97

Розроблене програмне забезпечення має простий, дружній та зручний інтерфейс користувача, що забезпечує легкість у освоєнні роботи програмного продукту, зручність у використанні, і не потребує особливих спеціальних знань.

При створенні програмного забезпечення було використано об'єктно-орієнтований підхід, що відповідає сучасним тенденціям у галузі розробки комерційних програмних систем.

Програма реалізована на мові високого рівня Python. Дана мова програмування дозволяє найбільш ефективно обробляти дані. Це дозволило мінімізувати строк розробки програмного забезпечення, і, як слід, зменшити витрати на його розробку. Запропоноване програмне забезпечення ділиться на загальне програмне забезпечення, що поставляється із засобами обчислювальної техніки й спеціальне програмне забезпечення, що спеціально розроблене для даної конкретної системи й включає програми, що реалізують її функції.

Програма призначена для виконання під управлінням багатозадачної операційної системи Windows 10/11.

Даються необхідні рекомендації з установки розробленого програмного забезпечення.

Для підвищення рівня безпеки запропоновано застосовувати алгоритм MARS.

В цілому створене програмне забезпечення підтверджує правильність використаних проектних рішень та повністю відповідає вимогам технічного завдання. Створене програмне забезпечення має потенційну можливість для подальшого вдосконалення і застосування у різних галузях.

Проведено маркетингове та економічне обґрунтування ІТ-проєкту, що дозволило визначити ключові фактори успіху даного проєкту.

					ВКРМ-122.24.0014.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		98

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Топчій В.А. Дослідження та програмна реалізація системи збору, представлення, обробки, зберігання, передачі та доступу до інформації мережевої відеоаналітики // Збірник праць молодих науковців ЦНТУ. – Вип. 14. – Кропивницький: ЦНТУ, 2024.
2. Henry Lloyd. Interactive Computer Graphics. States Academic Press. 2022. 247 p.
3. Ranjan Parekh. Fundamentals of Image, Audio, and Video Processing Using MATLAB® With Applications to Pattern Recognition. CRC Press. 2021. 406 p.
4. Alasdair McAndrew. A Computational Introduction to Digital Image Processing. Chapman & Hall. 2021. 560 p.
5. Peter Shirley, Steve Marschner. Fundamentals of Computer Graphics. 2009
6. Михайло Пічугін, Іван Канкін, Володимир Воротніков Комп'ютерна графіка. Навчальний посібник / Центр навчальної літератури 346 с. 2019р.
7. Маценко В.Г. Комп'ютерна графіка: Навчальний посібник. – Чернівці: Рута, 2009 – 343 с.
8. Інженерна комп'ютерна графіка: підручник / В.В. Проців [та ін.] / М-во освіти і науки України, Нац. гірн. унт-т. – Дніпро: НГУ, 2017. – 247 с.
9. Проців В.В. Прикладна комп'ютерна графіка [Текст]: Навч. посібник / В.В. Проців, К.А. Зіборов, К.М. Бас, Г.К. Ванжа; М-во освіти і наук, Нац. гірн. унт-т. – Д.: НГУ, 2016. – 187 с.
10. Kopf, Johannes and Lischinski, Dani. Depixelizing Pixel Art (англ.) // ACM Trans. Graph. – 2011. – Vol. 30, no. 4. – P. 99:1--99:8.
11. Giachetti, Andrea and Asuni, Nicola. Real-Time Artifact-Free Image Upscaling (англ.) // Trans. Img. Proc.. – 2011. – Vol. 20, no. 10. – P. 2760—2768.
12. Al-Mudhafar Aqeel, A.M., Smirnova, T., Buravchenko, K., Smirnov, O. «The method of assessing and improving the user experience of subscribers in

					ВКРМ-122.24.0014.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		99

software-configured networks based on the use of machine learning». *Advanced Information Systems*, 2023, 7(2), pp. 49-56.

13. Smirnov, O., Sydorenko, V., Aleksander, M., Zhyharevych, O., Yanchev, S. «Simulation of the cloud IoT-based monitoring system for critical infrastructures». *CEUR Workshop Proceedings*, Volume 3530, 2023, pp. 256-265.

14. Smirnov, O., Odarchenko, R., Smirnova, T., Bondar, S., Volosheniuk, D. «Optimal Structure Construction of Private 5G Network for the Needs of Enterprises». *Lecture Notes on Data Engineering and Communications Technologies*, 2023, 178, pp. 208–223.

15. Smirnov, O., Neskorodieva, T., Fedorov, E., Rudakov, K., Neskorodieva, A. «Method Detection Audit Data Anomalies on Basis Restricted Cauchy Machine» *CEUR Workshop Proceedings*, Volume 3187, 2022,

16. Smirnov O., Smirnova T., Anas M. Al-Oraiqat, Drieiev O., Polishchuk L., Sheroz Khan, Yassin M. Y. Hasan, Aladdein M. Amro, Hazim S. AlRawashdeh «Method for Determining Treated Metal Surface Quality Using Computer Vision Technology». *Sensors (Basel, Switzerland)* Volume 22, Issue 16, 6223, 2022.

17. Smirnov O., Kuznetsov A., Kryvinska N., Kiian A., Kuznetsova K. «Full Non-Binary Constant-Weight Codes». *SN Computer Science*, Vol 2, 337, 2021. <https://doi.org/10.1007/s42979-021-00739-w>

18. Smirnov O., Kuznetsov A., Zhora V., Onikiychuk A., Pieshkova O. «Hiding Messages in Audio Files Using Direct Spread Spectrum». 11th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications, IDAACS 2021, Cracow, Poland, 22-25 September 2021. P. 414-418.

19. Smirnov O., Kuznetsov A., Lokotkova I., Kuznetsova T., Florov S., Lebid O. «Using Orthogonal Signals to Hide Information in Images». 4 IEEE International Conference on Advanced Information and Communication Technologies (AICT) – 2021, Lviv, Ukraine, September 21-25, 2021. P. 255-260.

20. Smirnov, O., Kuznetsov, A., Potii, O., Poluyanenko, N., Stelnyk, I., Mialkovsky, D. «Combining and filtering functions in the framework of nonlinear-feedback shift register». International Journal of Computing; 2020, Volume 19, Issue 2 – Research Institute for Intelligent Computer Systems – 2020. – P. 247-256.

21. Smirnov O., Kuznetsov A., Kiian A., Kuznetsova T. «Non-binary constant weight coding technique». CEUR Workshop Proceedings. Volume 2740, 2020, Pages 102-114.

22. Smirnov O., Alimseitova Zh., Adranova A., Akhmetov B., Lakhno V., Zhilkishbayeva G. «Models and algorithms for ensuring functional stability and cybersecurity of virtual cloud resources». Journal of theoretical and applied information technology Vol.98. No 21, 2020, P. 3334-3346.

23. Smirnov O., Kuznetsov A., Kovalchuk D., Kuznetsova T. «New technique for data hiding in cover images using adaptively generated pseudorandom sequences». CEUR Workshop Proceedings Volume 2654, 2020, Pages 1-14.

24. Smirnov O., Kuznetsov A., Onikiychuk A., Makushenko T., Anisimova O., Arischenko A. «Adaptive pseudo-random sequence generation for spread spectrum image steganography». 2020 IEEE 11th International Conference on Dependable Systems, Services and Technologies (DESSERT), Ukraine, Kyiv, May 14-18. 2020. P. 161-165.

25. Smirnov O., Kuznetsov A., Kiian A., Cherep A., Kanabekova M., Chepurko I. «Testing of code-based pseudorandom number generators for post-quantum application». 2020 IEEE 11th International Conference on Dependable Systems, Services and Technologies (DESSERT), Ukraine, Kyiv, May 14-18. 2020. P. 172-177.

26. Smirnov O., Kuznetsov A., Pushkar'ov A., Serhiienko R., Babenko V., Kuznetsova T., «Representation of Cascade Codes in the Frequency Domain». In: Radivilova T., Ageyev D., Kryvinska N. (eds) Data-Centric Business and Applications. Lecture Notes on Data Engineering and Communications Technologies, vol 48. Springer, Cham. 2021. pp 557-587.

27. Smirnov, O., Drieieva, H., Drieiev, O., Polishchuk, Y., Brzhanov, R., Aleksander, M. «Method of fractal traffic generation by a model of generator on the graph». CEUR Workshop Proceedings Volume 2616, 2020, Pages 366-379.

28. Smirnov, O., Drieieva, H., Drieiev, O., Simakhin, V., Bondar, S., Odarchenko, R. «Managing multifractal properties of the binary sequence generated with the Markov chains», CEUR Workshop Proceedings Volume 2608, 2020, Pages 633-645.

29. Smirnov, O., Kuznetsov, A., Gorbacheva, L., Babenko, V., «Hiding data in images using a pseudo-random sequence», CEUR Workshop Proceedings Volume 2608, 2020, Pages 646-660.

30. Zhurakovskiy, B., Tsopa, N., Batrak, Y., Odarchenko, R., Smirnova, T «Comparative analysis of modern formats of lossy audio compression». Workshop Proceedings, 2020, 2654, стр. 315-327.

31. Smirnov O. Kuznetsov A., Zaichenko Yu., Pastukhov M., Oleshko O., Kuznetsova K., «Formation of Discrete Signals with Special Correlation Properties». International Conference on Information and Telecommunication Technologies and Radio Electronics, UkrMiCo 2019; Odessa; Ukraine; 9-13 September 2019. P.22-28.

32. Smirnov, O., Kuznetsov, A., Kolovanova, I., Kuznetsova, T., «Noise immunity of the algebraic geometric codes». International Journal of Computing; 2019, Volume 18, Issue 4 – Research Institute for Intelligent Computer Systems – 2019. – P. 393-407.

33. Smirnov, O., Kuznetsov, A., Reshetniak, O., Ivko, N., Katkova, T., Kuznetsova, T., «Generators of Pseudorandom Sequence with Multilevel Function of Correlation». 2019 IEEE International Scientific-Practical Conference Problems of Infocommunications, Science and Technology (PIC S&T), Kyiv, Ukraine, 8 – 11 October 2019 . P.517-522.

34. Smirnov, O., Krasnobayev, V., Yanko, A., Kuznetsova, T. «Methods of nulling numbers in the system of residual classes». CEUR Workshop Proceedings, Vol 2588, P. 90-106, 2019.

					БКРМ-122.24.0014.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		102

35. Smirnov, O., Kuznetsov, A., Kovalchuk, D., Averchev, A., Pastukhov, M., Kuznetsova, K., «Formation of Pseudorandom Sequences with Special Correlation Properties», 2019 3rd International Conference on Advanced Information and Communications Technologies, AICT -2019/ Lviv, Ukraine, 2-6 July, 2019, P. 395-399.

36. Smirnov, O., Kuznetsov, A., Kavun, S., Babenko, B., Nakisko, O., Kuznetsova, K., «Malware Correlation Monitoring in Computer Networks of Promising Smart Grids», 2019 IEEE 6th International Conference On Energy Smart Systems (2019 IEEE ESS), Kyiv, Ukraine April 17-19, 2019 P. 347-352.

37. Smirnov, O., Kuznetsov, A., Kovalchuk, D., Pastukhov, M., Kuznetsova, K., Prokopovych-Tkachenko, D., «Discrete Signals with Special Correlation Properties», CEUR Workshop Proceedings Volume 2353, CEUR Workshop Proceedings 2019, Pages 618-629.

38. Smirnov A.A., Kuznetsov A.A., Danilenko D.A., Berezovsky A., «The statistical analysis of a network traffic for the intrusion detection and prevention systems», Telecommunications and Radio Engineering. – Volume 74, Issue 1. – Begel House Inc. – 2015. – P. 61-78.

39. Smirnov O., Kuznetsov A., Kovalchuk D., Kuznetsova T. «New Technique for Hiding Data in Cover Images Using Adaptively Generated Pseudorandom Sequences». CEUR Workshop Proceedings Volume 2732, 2020, Pages 214-227.

40. Батрак О., Смірнова Т., Гнатюк В., Одарченко Р., Смірнов О. «Дослідження показників ефективності функціонування та перспектив розвитку систем IP-телефонії». *Підводні технології*, 2024, № 13, с. 28-35.

41. Аль-Мудхафар Акіл Абдулхуссейн М., Смірнова Т.В., Буравченко К.О., Смірнов О.А. «Метод оцінки та підвищення користувальницького досвіду абонентів в програмно-конфігурованих мережах на основі використання машинного навчання». *Сучасні інформаційні системи*, 2023, том 7, № 2, С. 49-56.

42. Т.В. Смірнова, О.М. Дреєв, О.А. Смірнов «Хмарна інформаційна система оцінювання шорсткості з використанням дискретного частотного аналізу макروفотografій». IV міжнародна науково-практична конференція «Інформаційна безпека та комп'ютерні технології», м. Кропивницький. 15-16 квітня 2021р. – Кропивницький: ЦНТУ. – 2021. – С. 30.

43. О.А. Смірнов, П.С. Усік, «Дослідження перспектив використання технологічних рішень в мережах 5G» у Кібербезпека та інформаційні технології: монографія. – Х. : ТОВ «ДІСА ПЛЮС», 2020.С. 122-135.

44. О.А.Смірнов, Т.В.Смірнова, Л.І. Поліщук, К.О. Буравченко, А.О.Макевнін, «Дослідження хмарних технологій як сервісів», Кібербезпека: освіта, наука, техніка. № 3(7). С. 43-62. 2020.

45. Смірнов О.А., Дреєва Г.М., Дреєв О.М., Смірнова Т.В. «Фрактальний аналіз генератора самоподібного трафіку на основі ланцюга Маркова». Центральнoукраїнський науковий вісник. Технічні науки. № 2(33). с. 161-172, 2019.

46. О. Смірнов, Є. Деменко, О. Онікійчук, А. Арищенко, Л. Горбачова, «Формування псевдовипадкових послідовностей для приховування даних в зображеннях» Комп'ютерні науки та кібербезпека. № 4. С. 30-37. 2019.

47. Смірнов О.А., Коноплицька-Слободенюк О.К., Смірнов С.А., Буравченко К.О., Смірнова Т.В. Поліщук Л.І. Проектування комп'ютерних систем та мереж. Навчальний посібник – Кропивницький: вид. Лисенко В.Ф. 2019. – 264 с.

48. Smirnov, O., Kuznetsov, A., Kuznetsova., K. Synthesis of Discrete Signals with Improved Correlation Properties. Монографія: In.: ISCI'2019: Information Security in Critical Infrastructures. Collective monograph. Edited by Ivan D. Gorbenko and Alexandr A. Kuznetsov, ASC Academic Publishing, USA, 2019, pp. 281-299. – ISBN: 978-0-9989826-8-7 (Hardback), ISBN: 978-0-9989826-9-4 (Ebook).

49. Смірнов О.А., Дреєва Г.М. Метод генерування фрактального трафіку за допомогою моделі генератора на графі. Монографія: Інформаційна

безпека та інформаційні технології : монографія / за заг. ред. В. С. Пономаренка. – Х. : Вид. Рожко С.Г. 2019. С. 123-139

50. Дреєва Г.М., Смірнов О.А., Дреєв О.М. Метод генерування фрактальноподібної числової послідовності на основі скінченного автомату для моделювання трафіку у мережі. Центральноукраїнський науковий вісник. Технічні науки. № 1(32). с. 173-183, 2019.

51. Смірнов О.А., Кавун С.В., Коваленко О.В., Дреєв О.М. Мережні інформаційні технології. Навчальний посібник – Кіровоград: РВЛ КНТУ, 2016. – 159 с.

52. Смірнов О.А., Смірнов С.А. Дідик А.К., Дреєв О.М. Моделі системи нейромережових експертів безпечної маршрутизації у хмарних антивірусних системах. Збірник наукових праць "Системи обробки інформації". – Випуск 3 (140). – Х.: ХУПС – 2016. – С. 36-39.

КБПЗ – 2024

					ВКРМ-122.24.0014.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		105

Додаток А
(обов'язковий)

Технічне завдання

Зміст

1 Найменування та область застосування.....	2
2 Підстава для розробки.....	2
3 Мета та призначення розробки.....	2
4 Джерела розробки.....	2
5 Технічні вимоги.....	2
5.1 Вміст проекту.....	2
5.2 Показники призначення.....	3
5.3 Вимоги до функціональних характеристик.....	3
5.4 Вимоги до архітектури.....	3
5.5 Вимоги до надійності.....	3
5.6 Умови експлуатації.....	4
5.7 Вимоги до складу та параметрів технічних засобів.....	4
5.8 Вимоги до інформаційної і програмної сумісності.....	4
5.8.1 Обладнання.....	4
5.8.2 Мова програмування.....	4
5.8.3 Вхідні дані.....	5
5.8.4 Вихідні дані.....	5
6 Вимоги до програмної документації.....	5
7 Економічні вимоги.....	5
8 Вимоги щодо охорони праці.....	5
9 Перелік документів, що розробляються.....	6
10 Етапи розробки.....	6
11 Порядок контролю та приймання.....	6

					ВКРМ-122.24.0014.00.00.ТЗ			
Вим.	Арк.	№ документа	Підпис	Дата				
Розробив	Топчій В.А.				Дослідження та програмна реалізація системи збору, представлення, обробки, зберігання, передачі та доступу до інформації мережевої відеоаналітики	Літ.	Аркуш	Аркушів
Перевірів	Буравченко К.О.					М	1	6
Н. Контр.	Коваленко А.С.					ЦНТУ КН-23М		
Затв.	Смірнов О.А.							

1 Найменування та область застосування

Це технічне завдання розповсюджується на дослідження та програмну реалізацію системи збору, представлення, обробки, зберігання, передачі та доступу до інформації мережевої відеоаналітики.

2 Підстава для розробки

Підставою для розробки служить завдання на випускню кваліфікаційну роботу за другим (магістерським) рівнем вищої освіти, видане на кафедрі кібербезпеки та програмного забезпечення (нак. № 18-13 від 07.08.2024 року).

3 Мета та призначення розробки

Метою випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти є дослідження та програмна реалізація системи збору, представлення, обробки, зберігання, передачі та доступу до інформації мережевої відеоаналітики.

4 Джерела розробки

Джерелом цієї випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти є стосовна до теми література і існуючі аналоги.

5 Технічні вимоги

5.1 Склад продукції

Складниками розробки є:

- вибір і обґрунтування методів реалізації проекту;
- розробка програмної частин системи, а також розробка взаємодії системи з ОС та з користувачем;
- техніко-економічне обґрунтування доцільності прийнятого до розробки програмного забезпечення;

					ВКРМ-122.24.0014.00.00.ТЗ	Арк.
Вим.	Арк.	№ документа	Підпис	Дата		2

- аналіз умов праці;
- розробка програми, що реалізує спроектовані алгоритми роботи системи.

5.2 Показники призначення

Система повинна забезпечувати:

- програмну реалізацію системи збору, представлення, обробки, зберігання, передачі та доступу до інформації мережевої відеоаналітики;
- цілісність даних у процесі роботи та при зберіганні;
- простий, інтуїтивно зрозумілий інтерфейс.

5.3 Вимоги до функціональних характеристик

Розроблене програмне забезпечення не повинно мати обмежень на версію драйверів та операційної системи.

5.4 Вимоги до архітектури

Компонент, що розробляється повинен використовувати системні засоби та апаратні засоби, що на даному етапі розвитку обчислювальної техніки найбільше поширені.

5.5 Вимоги до надійності

Програмні модулі написані по всім правилам, які стосуються стандартних викликів процедур, функцій, методів і форм, визначених технічною документацією на середовище розробки.

					ВКРМ-122.24.0014.00.00.ТЗ	Арк.
Вим.	Арк.	№ документа	Підпис	Дата		3

5.6 Умови експлуатації

Робочі місця користувачів ПЗ повинні задовольняти наступним умовам експлуатації:

- температура повітря: 19-20 град. по Цельсію;
- відносна вологість повітря до 80%;
- атмосферний тиск 107 кПа.

5.7 Вимоги до складу та параметрів технічних засобів

Програмне забезпечення повинно бути реалізоване на ПЕОМ, працювати в ОС Windows 10/11 і з сумісними з цією платформою пристроями і прикладним програмним забезпеченням.

5.8 Вимоги до інформаційної і програмної сумісності

Переносність програмного забезпечення повинна бути забезпечена за рахунок його реалізації стандартного інтерфейсу взаємодії з ОС, що працюють під управлінням ОС Windows 10/11.

5.8.1 Обладнання

Комп'ютер Intel® Celeron/8 Mb/1.2 Gb/SVGA 14" 1Mb або сумісні з ним.

5.8.2 Мова програмування

Середовище Python.

					ВКРМ-122.24.0014.00.00.ТЗ	Арк.
Вим.	Арк.	№ документа	Підпис	Дата		2

5.8.3 Вхідні дані

Опис алгоритму роботи запропонованої системи.

5.8.4 Вихідні дані

Робоча програма.

6 Вимоги до програмної документації

Програмна продукція повинна бути представлена у виді опису структури даних, схем та опису алгоритму, а також текстів вихідних модулів програмного забезпечення згідно ЄСПД .

7 Економічні вимоги

7.1 Для ПЗ необхідно виробити функціонально-вартісний аналіз варіантів розробки.

7.2 Виконати маркетингове та економічне обґрунтування ІТ-проєкту з урахуванням цін на 3 вересня 2024 року.

8 Вимоги щодо охорони праці

В частині охорони праці випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти повинен бути розглянутий аналіз санітарно-гігієнічних умов праці на робочому місці програміста.

					ВКРМ-122.24.0014.00.00.ТЗ	Арк.
Вим.	Арк.	№ документа	Підпис	Дата		5

9 Перелік документів, що розробляються

- Наукова новизна – 1 аркуш.
- Структурна схема системи – 1 аркуш.
- Функціональна схема системи – 1 аркуш.
- Діаграма процесів – 1 аркуш.
- Блок-схема алгоритму роботи програми – 2 аркуша.
- Показники економічної ефективності – 1 аркуш.
- Пояснювальна записка – 105 аркушів.

10 Етапи розробки

10.1 Збір і обробка інформації по темі випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти. Постановка задачі на виконання випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти (складання ТЗ).

10.2 Проведення досліджень або експериментальних робіт для уточнення основних положень випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти.

10.3 Розробка функціональних схем, блок схем алгоритмів роботи програмного забезпечення.

10.4 Побудова схем взаємодії даних.

10.5 Створення прототипу ПЗ.

10.6 Віднаходження ПЗ, аналіз отриманих результатів.

10.7 Робота над питанням охорони праці і техніки безпеки.

10.8 Маркетингове та економічне обґрунтування ІТ-проєкту.

10.9 Оформлення пояснювальної записки і виконання робіт по графічній частині.

11 Порядок контролю та приймання

11.1 Подання випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти на попередній захист 02.12.2024 р.

11.2 Подання випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти на захист 16.12.2024 р.

					ВКРМ-122.24.0014.00.00.ТЗ	Арк.
Вим.	Арк.	№ документа	Підпис	Дата		6

Додаток Б
(обов'язковий)

Міністерство освіти і науки України
Центральноукраїнський національний технічний університет

ЗАТВЕРДЖУЮ

Керівник випускної кваліфікаційної роботи за
другим (магістерським) рівнем вищої освіти
_____ Буравченко К.О.

*Дослідження та програмна реалізація
системи збору, представлення, обробки, зберігання, передачі та доступу до
інформації мережевої відеоаналітики*

Лістинг програми

Код документу 12

Носій: CD/DVD-диск / USB-флеш-накопичувач

Загальна кількість аркушів: 22

Літера: РП

Кропивницький – 2024 року

Основна програма

```
import os
import time
import requests
import json
import threading
import cv2

# Клас для управління відеоресурсами
class VideoResourceManager:
    def __init__(self, resource_id, resource_url):
        self.resource_id = resource_id
        self.resource_url = resource_url

    def get_video_metadata(self):
        # Отримання метаданих відео з сервера
        response =
requests.get(f"{self.resource_url}/metadata/{self.resource_id}")
        if response.status_code == 200:
            return response.json()
        else:
            return None

    def update_video_metadata(self, metadata):
        # Оновлення метаданих відео
        response =
requests.put(f"{self.resource_url}/metadata/{self.resource_id}", json=metadata)
        return response.status_code == 200

    def delete_video(self):
        # Видалення відео з ресурсу
        response =
requests.delete(f"{self.resource_url}/delete/{self.resource_id}")
        return response.status_code == 200

# Клас для роботи з відеопотоками
class VideoStream:
    def __init__(self, stream_url):
        self.stream_url = stream_url
        self.capture = cv2.VideoCapture(stream_url)

    def get_frame(self):
        ret, frame = self.capture.read()
        if ret:
            return frame
        else:
            return None

    def release(self):
        self.capture.release()
```

```

# Клас для управління мережею на основі EvRP
class EvRPNetworkManager:
    def __init__(self, api_url):
        self.api_url = api_url

    def get_network_status(self):
        # Отримання статусу мережі EvRP
        response = requests.get(f"{self.api_url}/network_status")
        if response.status_code == 200:
            return response.json()
        else:
            return None

    def configure_network(self, configuration):
        # Налаштування мережі EvRP
        response = requests.post(f"{self.api_url}/configure",
            json=configuration)
        return response.status_code == 200

# Клас для зберігання відеоресурсів на сервері
class VideoStorage:
    def __init__(self, storage_path):
        self.storage_path = storage_path
        if not os.path.exists(storage_path):
            os.makedirs(storage_path)

    def save_video(self, video_data, video_name):
        # Збереження відеофайлу
        video_path = os.path.join(self.storage_path, video_name)
        with open(video_path, 'wb') as video_file:
            video_file.write(video_data)

    def delete_video(self, video_name):
        # Видалення відеофайлу
        video_path = os.path.join(self.storage_path, video_name)
        if os.path.exists(video_path):
            os.remove(video_path)

# Клас для управління доступом до відеоресурсів
class AccessControl:
    def __init__(self):
        self.access_logs = []

    def log_access(self, user_id, resource_id):
        # Логування доступу до відеоресурсу
        timestamp = time.strftime("%Y-%m-%d %H:%M:%S", time.gmtime())
        log_entry = {'user_id': user_id, 'resource_id': resource_id,
            'timestamp': timestamp}
        self.access_logs.append(log_entry)

    def get_access_logs(self):
        # Отримання історії доступу
        return self.access_logs

```

```
# Клас для інтеграції з корпоративною системою аутентифікації
class AuthenticationSystem:
    def __init__(self, auth_url):
        self.auth_url = auth_url

    def authenticate_user(self, username, password):
        # Аутентифікація користувача через корпоративну систему
        credentials = {'username': username, 'password': password}
        response = requests.post(f"{self.auth_url}/login", json=credentials)
        if response.status_code == 200:
            return response.json()
        else:
            return None

# Клас для синхронізації відеоресурсів між серверами
class VideoSync:
    def __init__(self, server_url):
        self.server_url = server_url

    def sync_video(self, video_id):
        # Синхронізація відео між серверами
        response = requests.get(f"{self.server_url}/sync/{video_id}")
        if response.status_code == 200:
            return response.content
        else:
            return None

# Клас для управління ролями користувачів в корпоративній системі
class RoleManager:
    def __init__(self):
        self.roles = {}

    def assign_role(self, user_id, role):
        # Призначення ролі користувачу
        self.roles[user_id] = role

    def get_user_role(self, user_id):
        # Отримання ролі користувача
        return self.roles.get(user_id, 'guest')

# Клас для моніторингу відеоресурсів
class VideoMonitor:
    def __init__(self, resource_manager):
        self.resource_manager = resource_manager

    def monitor_resource(self, resource_id):
        # Моніторинг статусу відеоресурсу
        metadata = self.resource_manager.get_video_metadata()
        if metadata:
            print(f"Monitoring resource {resource_id}:")
            print(json.dumps(metadata, indent=4))
        else:
```

```

        print(f"Resource {resource_id} not found")

# Клас для управління потоковою трансляцією
class StreamManager:
    def __init__(self, stream_url):
        self.stream_url = stream_url
        self.stream = VideoStream(stream_url)

    def start_streaming(self):
        # Запуск трансляції відео
        while True:
            frame = self.stream.get_frame()
            if frame is not None:
                cv2.imshow("Streaming", frame)
                if cv2.waitKey(1) & 0xFF == ord('q'):
                    break

    def stop_streaming(self):
        # Зупинка трансляції
        self.stream.release()
        cv2.destroyAllWindows()

# Клас для роботи з обмеженнями трафіку
class TrafficLimiter:
    def __init__(self, limit):
        self.limit = limit

    def apply_limit(self, user_id):
        # Застосування обмеження трафіку для користувача
        print(f"Applying traffic limit of {self.limit} MB for user {user_id}")

# Клас для створення звітів по відеоресурсам
class ReportGenerator:
    def __init__(self, access_control):
        self.access_control = access_control

    def generate_access_report(self):
        # Генерація звіту по доступам до відеоресурсів
        logs = self.access_control.get_access_logs()
        report = "Access Report\n"
        for log in logs:
            report += f"User {log['user_id']} accessed resource {log['resource_id']} at {log['timestamp']}\n"
        return report

# Основна система управління відеоресурсами
class CorporateVideoManagementSystem:
    def __init__(self, auth_url, api_url, storage_path):
        self.auth_system = AuthenticationSystem(auth_url)
        self.network_manager = EvRPNetworkManager(api_url)
        self.storage = VideoStorage(storage_path)
        self.access_control = AccessControl()
        self.role_manager = RoleManager()

```

```
self.report_generator = ReportGenerator(self.access_control)

def authenticate_user(self, username, password):
    return self.auth_system.authenticate_user(username, password)

def add_video_resource(self, resource_id, resource_url):
    return VideoResourceManager(resource_id, resource_url)

def sync_videos(self, server_url, video_id):
    sync_manager = VideoSync(server_url)
    return sync_manager.sync_video(video_id)

def monitor_video(self, resource_id, resource_url):
    resource_manager = VideoResourceManager(resource_id, resource_url)
    monitor = VideoMonitor(resource_manager)
    monitor.monitor_resource(resource_id)

def manage_stream(self, stream_url):
    stream_manager = StreamManager(stream_url)
    stream_manager.start_streaming()

def generate_report(self):
    return self.report_generator.generate_access_report()

# Приклад запуску системи
if __name__ == "__main__":
    auth_url = "https://auth.corp.com"
    api_url = "https://network.corp.com"
    storage_path = "./videos"

    system = CorporateVideoManagementSystem(auth_url, api_url, storage_path)

    # Аутентифікація користувача
    user = system.authenticate_user("admin", "password123")
    if user:
        print("User authenticated:", user)

    # Додавання відеоресурсу
    video_manager = system.add_video_resource("resource_01",
    "https://videosever.com")

    # Синхронізація відео між серверами
    video_data = system.sync_videos("https://backupserver.com", "resource_01")
    if video_data:
        system.storage.save_video(video_data, "synced_video.mp4")

    # Моніторинг ресурсу
    system.monitor_video("resource_01", "https://videosever.com")

    # Запуск потокової трансляції
    system.manage_stream("https://streamingserver.com/stream")

    # Генерація звіту
```

```
report = system.generate_report()  
print(report)
```

КБПЗ_2024

Файл multiple_video_resources.py

```
import requests
import json

# Клас для управління відеоресурсами
class MultiVideoResourceManager:
    def __init__(self, resource_list):
        # Ініціалізація списку відеоресурсів
        self.resource_list = resource_list

    def get_all_metadata(self):
        # Отримання метаданих для всіх відеоресурсів
        metadata = {}
        for resource_id, resource_url in self.resource_list.items():
            response = requests.get(f"{resource_url}/metadata/{resource_id}")
            if response.status_code == 200:
                metadata[resource_id] = response.json()
            else:
                metadata[resource_id] = None
        return metadata

    def update_resource_metadata(self, resource_id, metadata):
        # Оновлення метаданих для певного ресурсу
        resource_url = self.resource_list.get(resource_id)
        if resource_url:
            response = requests.put(f"{resource_url}/metadata/{resource_id}",
                json=metadata)
            return response.status_code == 200
        return False

    def delete_resource(self, resource_id):
        # Видалення певного ресурсу
        resource_url = self.resource_list.get(resource_id)
        if resource_url:
            response = requests.delete(f"{resource_url}/delete/{resource_id}")
            return response.status_code == 200
        return False
```

Файл cloud_storage_integration.py

```
import boto3
from botocore.exceptions import NoCredentialsError

# Клас для роботи з хмарним сховищем AWS S3
class CloudStorageManager:
    def __init__(self, bucket_name, aws_access_key, aws_secret_key):
        # Ініціалізація з'єднання з S3
        self.s3 = boto3.client('s3', aws_access_key_id=aws_access_key,
aws_secret_access_key=aws_secret_key)
        self.bucket_name = bucket_name

    def upload_video(self, video_file_path, video_name):
        # Завантаження відеофайлу в хмару
        try:
            self.s3.upload_file(video_file_path, self.bucket_name, video_name)
            print(f"Upload of {video_name} successful")
        except FileNotFoundError:
            print("The file was not found")
        except NoCredentialsError:
            print("Credentials not available")

    def download_video(self, video_name, download_path):
        # Завантаження відео з хмари на локальний сервер
        try:
            self.s3.download_file(self.bucket_name, video_name, download_path)
            print(f"Download of {video_name} successful")
        except Exception as e:
            print(f"Failed to download {video_name}: {e}")
```

Файл ai_video_analytics.py

```
import cv2
import numpy as np
import tensorflow as tf

# Клас для аналітики відео на основі AI
class AIVideoAnalytics:
    def __init__(self, model_path):
        # Завантаження моделі AI для аналізу відео
        self.model = tf.keras.models.load_model(model_path)

    def analyze_frame(self, frame):
        # Попередня обробка кадру
        preprocessed_frame = cv2.resize(frame, (224, 224))
        preprocessed_frame = np.expand_dims(preprocessed_frame, axis=0)

        # Визначення об'єктів на основі моделі
        predictions = self.model.predict(preprocessed_frame)

        # Обробка результатів
        return predictions

    def detect_objects(self, frame):
        # Виявлення об'єктів на основі прогнозу
        predictions = self.analyze_frame(frame)
        if np.max(predictions) > 0.8: # Приклад: поріг для виявлення об'єкта
            return True
        return False
```

```
from cryptography.fernet import Fernet

# Клас для шифрування та дешифрування відеоданих
class VideoDataEncryption:
    def __init__(self, key):
        # Ініціалізація ключа для шифрування
        self.cipher = Fernet(key)

    def encrypt_data(self, data):
        # Шифрування даних
        return self.cipher.encrypt(data)

    def decrypt_data(self, encrypted_data):
        # Дешифрування даних
        return self.cipher.decrypt(encrypted_data)

    @staticmethod
    def generate_key():
        # Генерація нового ключа для шифрування
        return Fernet.generate_key()
```

КБПЗ_2024

Файл real_time_notifications.py

```
import smtplib
from email.mime.text import MIMEText

# Клас для сповіщень в реальному часі
class RealTimeNotifier:
    def __init__(self, smtp_server, port, sender_email, password):
        # Ініціалізація SMTP серверу для відправки email сповіщень
        self.server = smtplib.SMTP(smtp_server, port)
        self.sender_email = sender_email
        self.password = password

    def send_notification(self, recipient_email, message):
        # Відправка повідомлення
        self.server.starttls()
        self.server.login(self.sender_email, self.password)
        msg = MIMEText(message)
        msg['Subject'] = "Real-time Notification"
        msg['From'] = self.sender_email
        msg['To'] = recipient_email
        self.server.sendmail(self.sender_email, recipient_email,
msg.as_string())
        self.server.quit()
```

Файл geolocation_resources.py

```
import geopy
from geopy.geocoders import Nominatim

# Клас для роботи з геолокацією відеоресурсів
class GeolocationManager:
    def __init__(self):
        # Ініціалізація геокодера
        self.geolocator = Nominatim(user_agent="video_resource_locator")

    def get_location_by_address(self, address):
        # Отримання координат за адресою
        location = self.geolocator.geocode(address)
        if location:
            return (location.latitude, location.longitude)
        else:
            return None

    def get_address_by_coordinates(self, latitude, longitude):
        # Отримання адреси за координатами
        location = self.geolocator.reverse((latitude, longitude),
            exactly_one=True)
        return location.address if location else None
```

```
import cv2

# Клас для роботи з різними форматами відео
class VideoFormatManager:
    def __init__(self, video_file):
        self.video_file = video_file

    def convert_format(self, output_format):
        # Конвертація відео у вказаний формат
        output_file = self.video_file.split('.')[0] + f".{output_format}"
        cap = cv2.VideoCapture(self.video_file)
        fourcc = cv2.VideoWriter_fourcc(*'XVID') if output_format == 'avi' else
cv2.VideoWriter_fourcc(*'mp4v')
        out = cv2.VideoWriter(output_file, fourcc, 20.0, (640, 480))

        while cap.isOpened():
            ret, frame = cap.read()
            if ret:
                out.write(frame)
            else:
                break

        cap.release()
        out.release()
        return output_file
```

Файл `scheduled_streaming.py`

```
from apscheduler.schedulers.background import BackgroundScheduler
import time

# Клас для роботи з розкладом потокових трансляцій
class StreamingScheduler:
    def __init__(self):
        self.scheduler = BackgroundScheduler()

    def schedule_stream(self, stream_function, start_time):
        # Додавання запланованого завдання
        self.scheduler.add_job(stream_function, 'date', run_date=start_time)

    def start_scheduler(self):
        # Запуск планувальника
        self.scheduler.start()

    def stop_scheduler(self):
        # Зупинка планувальника
        self.scheduler.shutdown()

# Приклад використання
def start_stream():
    print("Streaming started")

scheduler = StreamingScheduler()
scheduler.schedule_stream(start_stream, '2024-10-30 10:00:00')
scheduler.start_scheduler()
time.sleep(10)
scheduler.stop_scheduler()
```

Файл backup_restore.py

```
import shutil
import os

# Клас для резервного копіювання та відновлення даних
class BackupRestoreManager:
    def __init__(self, backup_path):
        self.backup_path = backup_path

    def create_backup(self, source_folder):
        # Створення резервної копії
        backup_folder = os.path.join(self.backup_path, "backup")
        if not os.path.exists(backup_folder):
            os.makedirs(backup_folder)
        shutil.copytree(source_folder, backup_folder)

    def restore_backup(self, destination_folder):
        # Відновлення з резервної копії
        backup_folder = os.path.join(self.backup_path, "backup")
        if os.path.exists(backup_folder):
            shutil.copytree(backup_folder, destination_folder)
```

Файл traffic_analysis.py

```
import psutil
import time

# Клас для аналізу мережевого трафіку
class NetworkTrafficAnalyzer:
    def __init__(self):
        self.initial_data = psutil.net_io_counters()

    def get_current_traffic(self):
        # Отримання поточного мережевого трафіку
        current_data = psutil.net_io_counters()
        sent = current_data.bytes_sent - self.initial_data.bytes_sent
        recv = current_data.bytes_recv - self.initial_data.bytes_recv
        return sent, recv

    def monitor_traffic(self, interval=1):
        # Моніторинг трафіку протягом певного часу
        while True:
            sent, recv = self.get_current_traffic()
            print(f"Bytes sent: {sent}, Bytes received: {recv}")
            time.sleep(interval)
```

Файл `role_based_access_control.py`

```
# Клас для управління ролями доступу до відеоресурсів
class RoleBasedAccessControl:
    def __init__(self):
        self.user_roles = {}
        self.role_permissions = {
            "admin": ["read", "write", "delete"],
            "editor": ["read", "write"],
            "viewer": ["read"],
        }

    def assign_role(self, user_id, role):
        # Призначення ролі користувачу
        if role in self.role_permissions:
            self.user_roles[user_id] = role
        else:
            raise ValueError(f"Role {role} does not exist")

    def check_permission(self, user_id, action):
        # Перевірка прав доступу для певної дії
        role = self.user_roles.get(user_id)
        if role and action in self.role_permissions.get(role, []):
            return True
        return False

# Приклад використання
rbac = RoleBasedAccessControl()
rbac.assign_role("user1", "admin")
print(rbac.check_permission("user1", "delete")) # True
```

Файл `extended_logging.py`

```
import logging

# Клас для розширеного логування
class ExtendedLogger:
    def __init__(self, log_file):
        logging.basicConfig(filename=log_file, level=logging.DEBUG,
                            format='%(asctime)s %(levelname)s: %(message)s')

    def log_info(self, message):
        # Логування інформаційних повідомлень
        logging.info(message)

    def log_warning(self, message):
        # Логування попереджувальних повідомлень
        logging.warning(message)

    def log_error(self, message):
        # Логування помилок
        logging.error(message)

    def log_access(self, user_id, resource_id, action):
        # Логування доступу до відеоресурсів
        message = f"User {user_id} performed {action} on resource {resource_id}"
        logging.info(message)

# Приклад використання
logger = ExtendedLogger("system.log")
logger.log_info("System started")
logger.log_access("user1", "resource_01", "read")
```

Файл `microservice_architecture.py`

```
import flask
from flask import request, jsonify

app = flask.Flask(__name__)

# Приклад мікросервісу для керування відеоресурсами
@app.route('/metadata/<resource_id>', methods=['GET'])
def get_metadata(resource_id):
    # Обробка запиту на отримання метаданих
    return jsonify({"resource_id": resource_id, "status": "active"})

@app.route('/metadata/<resource_id>', methods=['PUT'])
def update_metadata(resource_id):
    # Обробка запиту на оновлення метаданих
    new_metadata = request.json
    return jsonify({"resource_id": resource_id, "new_metadata": new_metadata})

@app.route('/delete/<resource_id>', methods=['DELETE'])
def delete_resource(resource_id):
    # Обробка запиту на видалення ресурсу
    return jsonify({"resource_id": resource_id, "status": "deleted"})

# Запуск сервера
if __name__ == '__main__':
    app.run(host='0.0.0.0', port=5000)
```

Файл auto_delete_old_videos.py

```
import os
import time

# Клас для автоматичного видалення старих відеофайлів
class AutoDeleteOldVideos:
    def __init__(self, video_directory, max_age_days):
        self.video_directory = video_directory
        self.max_age_days = max_age_days

    def delete_old_videos(self):
        # Видалення файлів старіших за max_age_days
        current_time = time.time()
        for filename in os.listdir(self.video_directory):
            file_path = os.path.join(self.video_directory, filename)
            if os.path.isfile(file_path):
                file_age = current_time - os.path.getmtime(file_path)
                if file_age > self.max_age_days * 86400: # Дні переведені в
секунди
                    os.remove(file_path)
                    print(f"Deleted: {file_path}")

# Приклад використання
auto_delete = AutoDeleteOldVideos("./video_storage", 30) # Видаляти файли
старші 30 днів
auto_delete.delete_old_videos()
```

```
import cv2

# Клас для моніторингу якості відеопотоку
class VideoQualityMonitor:
    def __init__(self, stream_url):
        self.stream_url = stream_url
        self.capture = cv2.VideoCapture(stream_url)

    def monitor_quality(self):
        # Моніторинг основних показників якості
        while True:
            ret, frame = self.capture.read()
            if ret:
                fps = self.capture.get(cv2.CAP_PROP_FPS)
                resolution = (self.capture.get(cv2.CAP_PROP_FRAME_WIDTH),
self.capture.get(cv2.CAP_PROP_FRAME_HEIGHT))
                print(f"FPS: {fps}, Resolution: {resolution}")
            else:
                print("Stream ended or error occurred")
                break

    def release(self):
        self.capture.release()

# Приклад використання
quality_monitor = VideoQualityMonitor("stream_url")
quality_monitor.monitor_quality()
```