МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
СХІДНОУКРАЇНСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ІМ. В. ДАЛЯ
ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ ТА ЕЛЕКТРОНІКИ
КАФЕДРА КОМП'ЮТЕРНИХ НАУК ТА ІНЖЕНЕРІЇ

До захисту допускається
Т.в.о.завідувача кафедри
_____ Сафонова С.О.
«____»_____ 2020 р.

**МАГІСТЕРСЬКА РОБОТА**

НА ТЕМУ:

Методи і моделі прогнозування дій об'єкту з використанням відео

Освітньо-кваліфікаційний рівень "Магістр"
Спеціальність 122 – "Комп'ютерні науки"

Науковий керівник роботи: _____ (підпис) Білобородова Т.О. (ініціали, прізвище)

Консультант з охорони праці: _____ (підпис) Критська Я.О. (ініціали, прізвище)

Студент: _____ (підпис) Сіроштан І.В. (ініціали, прізвище)

Група: КН-18дм

Сєвєродонецьк 2020

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
СХІДНОУКРАЇНСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ВОЛОДИМИРА ДАЛЯ

Факультет Інформаційних технологій та електроніки
Кафедра Комп'ютерних наук та інженерії
Освітньо-кваліфікаційний рівень Магістр
Напрям підготовки _____
(шифр і назва)
Спеціальність 122 – "Комп'ютерні науки"
(шифр і назва)

**ЗАТВЕРДЖУЮ:**

Т.в.о. завідувача кафедри
_____С.О. Сафонова
«_____» _____2020 р.

**З А В Д А Н Н Я**
**НА МАГІСТЕРСЬКУ РОБОТУ СТУДЕНТУ**

Сіроштану Івану Володимировичу
(прізвище, ім'я, по батькові)

1. Тема роботи Методи і моделі прогнозування дій об'єкту з використанням відео
керівник проекту (роботи) к.т.н. Білобородова Тетяна Олександрівна
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)
затверджені наказом вищого навчального закладу від " 11" 10 2019 р. № 35/15.15
2. Строк подання студентом роботи _____
3. Вихідні дані до роботи Матеріали науково-дослідної практики

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити)
1. Аналіз галузі застосування та технології розпізнавання образів, сегментації зображень, та подальшого прогнозування дій об'єктів у відеопотоці;
2. Реалізація моделей нейронних мереж та визначення їх конфігурації для роботи з відеопотоком;
3. Тестування розпізнавання образів та сегментації зображення, проведення експерименту з розпізнавання об'єктів на даних ендоскопічних відеозображень
4. Охорона праці в галузі;
5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень)

електронні плакати

6. Консультанти розділів проекту (роботи)

| Розділ | Прізвище, ініціали та посада консультанта | Підпис, дата | |
|---|---|---|---|
| | | завдання видав | завдання прийняв |
| Охорона праці | ст. викл. Критська Я.О. | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |

7. Дата видачі завдання _____

Керівник _____
(підпис)

Завдання прийняв до виконання _____
(підпис)

**КАЛЕНДАРНИЙ ПЛАН**

| № з/п | Назва етапів дипломного проекту (роботи) | Строк виконання етапів проекту ( роботи ) | Примітка |
|---|---|---|---|
| 1 | Огляд та аналіз вимог до роботи. | 02.09.19-19.09.19 | |
| 2 | Аналіз досліджень технології розпізнавання та сегментації об'єктів у лапроскопічних, та ендоскопічних відео | 22.09.19-10.10.19 | |
| 3 | Розробка програмної реалізації моделей для розпізнавання та сегментації зображень | 11.10.19-15.11.19 | |
| 4 | Тестування запропонованої технології шляхом проведення експерименту з розпізнавання та сегментації на даних з ендоскопії | 16.11.18-09.12.19 | |
| 5 | Дослідження та аналіз отриманих результатів | 10.12.19-15.12.19 | |
| 6 | Розробка заходів з охороні праці. | 16.12.19-20.12.19 | |
| 7 | Оформлення пояснювальної записки. | 21.12.19-31.12.19 | |
| 8 | Підготовка та подання магістерської роботи до захисту. | 02.01.20-08.01.20 | |

**Студент** _____ Сіроштан І.В.
( підпис ) (прізвище та ініціали)

**Науковий керівник** _____ Білобородова Т. О.
( підпис ) (прізвище та ініціали)

## АНОТАЦІЯ

Сіроштан І.В. Методи і моделі прогнозування дій об'єкту з використанням відео.

В роботі представлено вирішення задачі підвищення ефективності інтелектуальних медичних систем з використанням відео зйомки, що використовуються у закладах надання медичної допомоги за рахунок розробки методу прогнозування на підставі даних моделей розпізнавання, розроблених з використанням обмежуючої коробки і сегментації, що дозволить виявляти аномальні дії у відео хірургічних втручань в режимі реального часу та надати підтримку прийняття рішень лікарю під час проведення діагностичних та оперативних втручань. Проведено аналіз методів та моделей розпізнавання, сегментації відео, прогнозування розвитку ситуацій. Визначена систематизована сукупність етапів прогнозування дій об’єктів у відеопотоці. Досліджено використання глибокого навчання, нейронних мереж для технології розпізнавання відео. Розроблена модель розпізнавання образів на відео з ендоскопічного дослідження з використанням методу обмежуючої коробки, також реалізована модель для виконання семантичної сегментації ендоскопічних графічних даних. Проведено тестування та визначення точності розроблених моделей.

Рис.: 28. Табл.: 10. Бібліогр.: 35.

**Ключові слова:** розпізнавання, медичне відеозображення, прогнозування, сегментація, нейронна мережа

## АННОТАЦИЯ

Сероштан И.В. Методы и модели прогнозирования действий объекта с использованием видео.

В работе представлено решение задачи повышения эффективности интеллектуальных медицинских систем с использованием видеосъемки, используемых в учреждениях оказания медицинской помощи, за счет разработки метода прогнозирования на основании данных моделей распознавания, разработанных с использованием методов ограничивающей коробки и сегментации, что позволит выявлять аномальные действия в видео хирургических вмешательств в режиме реального времени и оказывать поддержку принятия решений врачу при проведении диагностических и оперативных вмешательств. Проведен анализ методов и моделей распознавания, сегментации видео, прогнозирования развития ситуаций. Определена систематизированная совокупность этапов прогнозирования действий объектов в видеопотоке. Исследовано использование глубокого обучения, нейронных сетей для технологии

распознавания видео. Разработанная модель распознавания образов на видео эндоскопического исследования с использованием метода ограничивающей коробки, также реализована модель для выполнения семантической сегментации эндоскопических графических данных. Проведено тестирование и определение точности разработанных моделей

Рис.: 28. Табл.: 10. Библиогр..: 35.

**Ключевые слова:** распознавание, медицинское видеоизображения, прогнозирования, сегментация, нейронная сеть

## ABSTRACT

Siroshtan Ivan. Methods and models of object action prediction using video.

The paper presents a solution to the problem of increasing the efficiency of intelligent medical systems using video recordings used in medical care institutions by developing a forecasting method based on data from recognition models developed using restrictive box and segmentation methods, which will allow detecting abnormal actions in video of surgical interventions in real time and provide support to decision-making physician during diagnostic and opera active interventions. The analysis of methods and models of recognition, video segmentation, forecasting the development of situations. A systematic set of stages for predicting the actions of objects in a video stream is determined. The use of deep learning, neural networks for video recognition technology has been investigated. An endoscopic video recognition model was developed using the bounding box method. An endoscopic video recognition model using the segmentation method has been developed. Testing and determining the accuracy of the developed models.

**Keywords:** video recognition, endoscopic video, object action prediction, mining, semantic segmentation, neural network

# ЗМІСТ

# ПЕРЕЛІК УМОВНИХ ПОЗНАЧОК І СКОРОЧЕНЬ

| | |
|---|---|
| CNN | Convoluted neural network |
| DCNN | Deep convoluted neural networks |
| DoG | Difference of Gaussians |
| RCNN | Region-based convoluted neural networks |
| RNN | Recurent Neural Network |
| RPN | Region Proposal Network |
| SHG | Stochastic gradient descent |
| SIFT | The Scale-Invariant Feature Transform |
| SPP | Spatial Pyramid Pooling |
| SVM | Support-vector machine |
| YOLO | You Only Look Once |

# ВСТУП

З розвитком IoT технологій все більш актуальною є тема аналізу медіа даних. Раніше для того, щоб провести дослідження пов’язані з графікою, або відео одним з найбільш трудоємких етапів був етап збору даних, але зараз це не є проблемою, навіть навпаки трапляються так, що не вистачає ресурсів щоб обробити усі накопиченні дані. Саме ці факти й обумовлюють великий інтерес до технологій пов’язаних з семантичним аналізом відеопотоку. Така зацікавленість обґрунтовується тим, що технології які вирішують задачу розпізнавання об'єктів у відеопотоці мають велику цінність, тому як можуть вирішити велику кількість прикладних задач. Методи розпізнавання та прогнозування дій у відеопотоці використовуються у багатьох галузях застосування: автомобілі з автопілотом, робототехніка, системи відеоспостереження тощо.

При використанні систем відеоспостереження при проведенні медичних досліджень, поцедур або оперативних втручань у закладах медичної допомоги частота виявлення патологічних діагностичних елементів або підтримки прийняття рішення під час поведення хірургічної поцедури повинна бути вище, ніж зазвичай, щоб забезпечити своєчасне реагування на події під час операції (виявлення аномалії, патології, рух медичних інструментів тощо). Разом з тим, більшість медичних систем відеозйомки медичних процесів, такі як ендоскоп, лапароскоп, просто передають дані лікарю, який оцінює стан і відповідним чином на нього реагує. Моніторинг та інтерпретація великої кількості даних можуть накласти на лікаря під час медичної процедуре небажане когнітивне навантаження, що може затримати його реакцію на розвиток подій під час проведення медичної процедури.

Розпізнавання взаємодії хірургічних інструментів з тканинами внутрішніх органів є важливим етапом аналізу лапороскопічних операцій. Такі операції здійснюються через природні отвори тіла або невеликі штучні розрізи, за рахунок чого зменшуються травми пацієнтів та скорочується час їх госпіталізації. Разом з тим, під час лапароскопічних операцій можливі обмеження бачення та рухливості, ускладнена координація рук і очей хірурга, що обумовлює створення нових засобів візуалізації, контролю та управління. З цього приводу, спільнота медичних дослідників прагне розширити можливості хірурга за допомогою контекстно-ведених комп'ютерних хірургічних систем. Метою таких систем є забезпечення хірурга актуальною інформацією, аналіз дій під час операції, прогнозування можливих непередбачуваних ситуацій з метою їх уникнення, підтримка прийняття рішення при діагностуванні патологічних аномалій внутрішніх органів, тощо. Одним із завдань, в контексті задачі аналізу дій хірурга під час операції, є аналіз відеозображень та розпізнавання взаємодії

інструментів з тканинами. Якість диференціації об'єктів має вирішальне значення для оцінки продуктивності, оскільки існує багато факторів, що ускладнюють розпізнавання, серед яких роздільна здатність відеокамери, наявність у порожнині газів, диму, запотівання лінзи камери, індивідуальні особливості анатомії та інші. Також, обмежені сегменти реальних операцій можуть не дати всебічну оцінку хірургічного процесу.

Однією з передумов кількісної оцінки взаємодії тканина-інструмент, є сегментація хірургічних епізодів та виявлення деформації тканини у відповідь на рухи інструменту. Для отримання якісної, ефективної моделі необхідна попередня обробка відеозображень для отримання даних об'єктів. Такі дані можуть включати анотації присутності хірургічного інструменту або фази операції, які є мітками для більш детальних завдань, таких як сегментація.

Проблема розпізнавання дій об'єктів (таких як рух інструментів, зміни хірургічного поля, стан внутрішніх органів тощо) по відео (наприклад, що надходять з ендоскопу або лапароскопу) є дуже складною, оскільки одночасно можуть виникати різні ситуації, і оклюзії ще більше ускладнюють розпізнавання. Крім того, відео традиційно обробляється в цілому, тому рішення про те, який клас дії спостерігається, не може бути прийнято в режимі реального часу (миттєво). Це значно знижує потенціал медичної системи відеозйомки; більшість з них використовуються як інструмент аналітики історичних даних замість проактивних і профілактичних інструментів. Більш того, незважаючи на значні зусилля, сучасні системи призначені тільки для розпізнавання одного типу дії на відеокадрі або можуть працювати з покроковим розпізнаванням, але не можуть одночасно знайти необхідну дію в кожному відеокадрі. Найбільш ефективні методи виявлення дій об'єктів у відео на сьогоднішній день по своїй суті є автономними, оскільки вони засновані на виявленні пропозицій по регіонах кадр за кадром і об'єднанні їх в так звану «послідовність фреймів» лише на етапі подальшої обробки.

Сучасний розвиток систем комп'ютерного зору і машинного навчання робить можливим впровадження в медичні системи відеозйомки підтримку прийняття медичних рішень щодо подальшого розвитку сценарію надання медичної допомоги. Однак, алгоритми машинного навчання, як правило, вимагають значних обчислювальних потужностей використовуваного апаратного забезпечення. Замість того, щоб встановлювати більшу комп'ютерну систему в медичній установі, аналіз відеопотоку може проводитись на стороні хмарних платформ, що значно знижує початкові витрати на обладнання та обслуговування. Однак, в цій ситуації конфіденційні дані виходять за межі прямого контролю медичної установи, що є не зовсім прийнятним з точки зору забезпечення конфіденційності.

Тому обґрунтованою є тема магістерської роботи, у якій вирішується **науково-прикладне завдання** удосконалення методів і моделей прогнозування дій об’єктів у відео.

**Об'єкт дослідження** – процеси перетворення відеопотоку у цифрові дані та їх

використання.

**Предмет дослідження –** методи і моделі прогнозування розвитку дій об'єктів у відео.

**Мета і завдання дослідження**. Метою дослідження є підвищення ефективності інтелектуальних медичних систем з використанням відео зйомки, що використовуються у закладах надання медичної допомоги за рахунок розробки методу прогнозування на підставі даних моделей розпізнавання, розроблених з використанням обмежуючої коробки і сегментації, що дозволить виявляти аномальні дії у відео хірургічних втручань в режимі реального часу та надати підтримку прийняття рішень лікарю під час проведення діагностичних та оперативних втручань.

Для досягнення мети дослідження необхідно вирішити такі **завдання**:

– аналіз методів та моделей розпізнавання, сегментації відео, прогнозування розвитку ситуацій;

– розроблення методу систематизованого прогнозування дій об’єктів у відеопотоці та визначення сукупності етапів прогнозування;

– розроблення моделі розпізнавання з використанням методу обмежуючої коробки;

– розроблення моделі розпізнавання з використанням методу сегментації;

– тестування розроблених моделей;

– оцінка ефективності моделей розпізнавання об’єктів у відео.

**Методи дослідження.** Проведені в роботі дослідження основані на методах моделювання зображень, технології глибокого навчання, нейронних мережах, що використовувались для розпізнавання об’єктів відеопотоку з використанням методів обмежуючої коробки та сегментації та прогнозування його дій. Перевірка результатів дослідження ґрунтувалась на методах експерименту та порівняння, які використовувались при розробленні практичної частини дипломного проекту.

**Особистий внесок здобувача** полягає у розробці методу систематизованого прогнозування дій об’єктів у відеопотоці та визначення сукупності етапів прогнозування та розробленні моделей розпізнавання з використанням методу обмежуючої коробки та методу сегментації, що дозволяє вирішити поставлені задачі. Усі основні результати отримані автором особисто.

**Апробація матеріалів магістерської роботи**. Основні положення, ідеї, та висновки магістерської роботи доповідалися та обговорювалися на V регіональному форумі ІТ-Ідея (м. Сєвєродонецьк, 2019) та на Всеукраїнській науково-практичній конференції «Майбутній науковець – 2019» (м. Сєвєродонецьк, 2019).

**Практичне значення отриманих результатів** полягає в тому, що основні наукові положення реалізовані у виді визначеної систематизованої сукупності етапів та моделей розпізнавання, що утворюють метод прогнозування дій об’єктів у відео.

**Публікації.** За темою магістерської роботи з викладенням її результатів опубліковані одна наукова стаття у науковому фахововому виданні України; дві тези доповідей всеукраїнських конференцій.

**Структура та обсяг магістерської роботи.** Кваліфікаційна магістерська робота складається із вступу, чотирьох розділів, висновків, переліку посилань. Загальний обсяг складає 91 сторінку, з яких основний текст на 79 сторінках, список використаних джерел із 35 найменувань на 2 сторінках та 1 додатку на 8 сторінок. Робота містить 10 таблиць, 28 рисунків.

# РОЗДІЛ 1 ТЕОРЕТИЧНІ АСПЕКТИ РОЗПІЗНАВАННЯ ТА ПРОГНОЗУВАННЯ РОЗВИТКУ СИТУАЦІЙ У ВІДЕОПОТОЦІ

## 1.1 Огляд галузі застосування та проблеми розпізнавання, прогнозування розвитку ситуацій у відеопотоці

У комп'ютерному зорі є дві основні теми, розпізнавання та прогнозування дій на основі того що ми бачимо [1]:

Розпізнавання дії: мета цієї задачі розпізнати закінчену дію об'єкта з відео.

Прогнозування дій: передбачення майбутнього становища об’єкту використовуючи неповні відеодані.

Розпізнавання дій - це фундаментальне завдання спільноти комп'ютерного зору, яке розпізнає дії людини на основі повністю виконаних дій у відео. Іншими словами, розпізнавання дій - це завдання відео аналізу за фактом, яке зосереджено на теперішньому стані. Данна тема досліджується десятиліттями і досі залишається дуже популярною темою завдяки широкому застосуванню в реальному світі, включаючи пошук відео візуальне спостереження, тощо. Дослідники доклали великих зусиль для створення інтелектуальної системи, що імітує здатність людини, яка може розпізнавати складні людські. Ця проблема розглядається як представлення та класифікація дій в контексті розпізнаванні дій. Було запропоновано багато способів вирішити цю проблему. Алгоритми розпізнавання та прогнозування дій використовуються багатьма реальним програмами. Сучасні алгоритми значно зменшують людську працю при аналізі масштабних відеоданих і дають інформацію про поточний та майбутній стан об’єктів на відео.

Розвиток інформаційних технологій обумовив широкий спектр галузей застосування технології розпізнавання та прогнозування різноманітних об’єктів, подій по відео записам. Детальніше галузі застосування розглянуті далі.

*Відеоспостереження.* Питання безпеки стає все більш важливим у нашому повсякденному житті, і це одна з найбільш часто обговорюваних тем в наш час. Місця, що знаходяться під наглядом, зазвичай встановлюють певний регламент дій для людини. При введенні мережі камер може бути створена система візуального спостереження, що працює на розпізнаванні дій та їх прогнозуванні. Такий підхід дозволяє збільшити шанси захоплення злочинця на відео та зменшити ризик скоєння злочину.

*Автономний водійський транспортний засіб.* Алгоритми прогнозування дій можуть бути одним з потенційно найважливішими складовими в автономному транспортному засобі. Алгоритми прогнозування дій можуть передбачити наміри людини за короткий проміжок часу. У надзвичайній ситуації транспортний засіб, оснащений алгоритмом прогнозування дій, може передбачити майбутню дію пішохода або траєкторію руху в найближчі кілька секунд, і це може стати критичним для уникнення зіткнення. Аналізуючи характеристики руху людського тіла на ранній стадії дії, використовуючи так звані опорні точки або згорткову нейронну мережу, алгоритми прогнозування дій можуть зрозуміти можливі дії, проаналізувавши розвиток дій без необхідності. спостерігати за виконанням всієї дії.

*Взаємодія людини з роботом.* Взаємодія людини з роботом широко застосовується в домашніх та промислових умовах. Уявіть, що людина взаємодіє з роботом і просить його виконати певні завдання, такі як "принести чашку води" або "прибрати вдома". Така взаємодія вимагає зв’язку між роботами та людьми, а візуальний зв’язок - один із найефективніших способів.

*Галузь Robot-assisted laparoscopy* [2]. Хірургічний робот - це керуючий комп'ютером пристрій, який може бути запрограмований для облегшення позиціювання та маніпуляції з хірургічно інструментами. Хірургічна робототехніка зазвичай використовується в лапароскопії, а не у відкритих хірургічних операціях. С 1980-х років хірургічні роботи були розроблені для рішення задач пов'язані з обмеження в лапароскопії, включаючи двовимірну візуалізацію, неповну артикуляцію інструментів і ергономічні обмеження. Мета роботизованої лапароскопічної хірургії постає в тому, щоб допомогти хірургам поліпшити обслуговування пацієнтів шляхом перетворення процедур які зазвичай виконуються лапаротомією, в мінімально інвазивні процедури.

Галузь застосування Robot-assisted laparoscopy є актуальним та перспективним напрямком розвитку технологій прогнозування розвитку ситуацій.

Непередбачувані медичні помилки в операційній трапляються досить часто, щоб коштувати десятки тисяч людських життів на рік. Щоб зменшити кількість таких випадків, спільнота медичних технологій прагне розширити можливості хірурга за допомогою контекстно відомих комп'ютерних хірургічних систем. Метою такої системи є аналіз дій хірурга у під час операції з метою спрогнозувати можливі непередбачувані ситуації та уникнути їх. Одним з кращих рішень є розпізнавання хірургічних інструментів та прогнозування їх поведінки (наприклад, траєкторії). Та у контексті людського життя покращення задачі прогнозування розвитку ситуацій під час лапароскопічних операцій за участі Robot-assisted є актуальним завданням.

За цим напрямом виділяють наступні задачі.

*Розпізнавання об’єктів у відеопотоці* [3]. Задача розпізнавання образів є актуальною вже не один рік поспіль. Це підтверджується тим, що системи розпізнавання образів інтегруються у все більшу кількість сучасних галузей. Методи та моделі які для цього використовуються вдосконалюються щодня.

*Прогнозування майбутніх дій об’єкта.* Задача прогнозування є не менш актуальною і для її рішення вже розроблена велика кількість алгоритмів. Проблема прогнозування майбутніх дій поширена у робототехніці і для її рішення використовується велика кількість методів, від експертних систем до нейронних мереж.

Слід зауважити, що для рішення задачі поставленою обраним об’єктом дослідження не достатньо розробити модель яка буде вирішувати проблему розпізнавання об’єктів, та модель яка буде вдало прогнозувати дії. Ще слід інтегрувати їх між собою, метою є створення системи яка буде використовувати розробленні моделі разом для вирішення загально задачі.

Щоб краще зрозуміти яким чином можна вирішити поставлену задачу слід розділити її на підзадачі і розглянути їх окремо.

### 1.2 Аналіз методів розпізнавання об’єктів

Розпізнавання об’єктів - це комп’ютерна технологія, пов’язана з комп'ютерним зором та обробкою зображень, яка стосується виявлення екземплярів смислових об'єктів певного класу (наприклад, людей, будівель чи автомобілів) у цифрових зображеннях та відео.

Основною ідеєю при розпізнаванні об’єктів є те, що кожний об’єкт має особливі ознаки завдяки яким його можна віднести до якогось класу або групи класів, які допомагають класифікувати об’єкт. Наприклад, усі кола круглі. Для виявлення класів об'єктів використовуються ці особливості. Таким чином, шукаючи кола, шукають об’єкти, які знаходяться на певній відстані від точки (тобто центру). Аналогічно при пошуку квадратів потрібні предмети, які перпендикулярні по кутах і мають рівні бічні довжини. Але це не усі ознаки які використовують для класифікації об’єкту, більш складні системи використовують дуже велику кількість ознак, щоб якомога точніше обрати клас для об’єкта. Ознакою може бути що завгодно, головне щоб це допомогло ідентифікувати об’єкт.

Якщо казати про розпізнавання об’єктів на відео, то додається ще одна важлива ознака – рух. Завдяки руху на відео ми можемо розпізнавати не тільки об’єкти, але й події. Рух сам по собі є потужною візуальної підказкою. Суть багатьох дій саме в динаміці, та іноді достатньо відстежити рух окремих точок, щоб розпізнати якусь подію.

Рух можна характеризувати як точки спостерігаємо сцени, які рухаються відносно камери. Але потрібно цей рух якось формалізувати, описувати і вимірювати.

З точки зору структурного аналізу відеопоток може бути представлений як набір сцен, де кожна сцена має набір об’єктів які і слід проаналізувати та класифікувати [3].

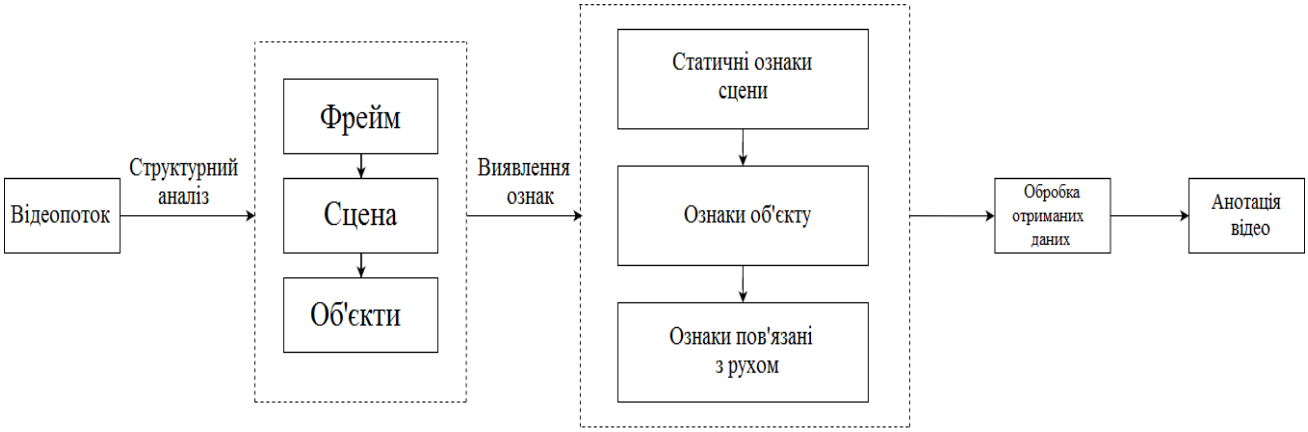


Рисунок 1.1 – Узагальнена схема аналізу відеопотока

Як можна побачити на рисунку 1.1 аналіз відеопотоку має декілька етапів:

*Етап 1.* Структурний аналіз.

На цьому етапі необхідно розділити потік на фрейми, фрейми на сцени, а у сцен виділити об’єкти.

*Етап 2.* Виявлення ознак.

На етапі виявлення ознак необхідно виділити інформацію з структурних одиниць відеопотоку: статичні ознаки сцени, ознаки які характеризують об’єкт, та ознаки пов’язані з рухом.

*Етап 3.* Обробка даних.

Саме на етапі обробки даних і використовується модель яка проаналізує всі ознаки отриманні на попередніх етапах, щоб класифікувати об’єкт.

*Етап 4.* Анотація відео.

На цьому етапі необхідно пов'язати данні отримані з моделі, з підготовленими метаданими які дозволять семантично анотувати об'єкти на відео.

### 1.3 Аналіз методів розпізнавання дій об’єктів

Прогнозування дій викликає все більший інтерес в останні роки завдяки широкому та важливому застосуванню в реальних сценаріях, таких як візуальне спостереження та уникнення

аварій на дорогах. На відміну від розпізнавання дій, в прогнозуванні дії, слід вказати мітку до того, як буде здійснено виконання дії. Що важливіше, потрібно, щоб алгоритм прогнозування міг робити точні прогнози на самому початковому етапі відео, наприклад, коли спостерігається лише кілька кадрів відео.

Типовий алгоритм розпізнавання дій, як правило, поділяється на два етапи, представлення дій та класифікація дій.

Представлення дій перша і найважливіша проблема розпізнавання дій - це відображення дії у відео. Дії людини, що з'являються у відео, відрізняються за своєю швидкістю руху, кутом огляду камери, зовнішнім виглядом та варіаціями постанови, що робить представлення дій справжнім викликом. Успішні методи представлення дій повинні бути ефективними для обчислення, ефективними для характеристики дій та можуть максимально збільшити розбіжність між діями, щоб зменшити помилку класифікації [4].

Метою представлення дій є перетворення відео дії у векторний вигляд, витяг репрезентативної та дискримінативної інформації про дії та мінімізація варіацій, тим самим покращуючи ефективність розпізнавання.

Після обчислення представлення про дії натренований класифікатор повинен буде обрати необхідну мітку для дії. Класифікатори дій можна розділити на наступні категорії:

1. Пряма класифікація.

У цьому випадку класифікатор підсумовує наданні вектори, а потім розпізнає за допомогою стандартних алгоритмів класифікації. (Метод опорних векторі, мето к-найближчих сусідів та інші), У цих методах динаміка дії характеризується цілісно, використовуючи форму дії або використовуючи так звану «сумку слів», яка кодує розподіл локальних моделей руху за допомогою гістограми.

2. Послідовні підходи.

Цей напрямок роботи передбачає часову еволюцію появи або створення за допомогою послідовних моделей, таких як приховані Марковські моделі (HMM), умовні випадкові поля (CRF) та структурований алгоритм опорних векторів (SSVM). Ці підходи трактують відео як композицію часових фрагментів або кадрів. Модель аналізує траєкторію руху об’єкта для класифікації. Останні роботи показують, що якщо розділити дії на ключові моменти то можна навчитися краще представляти людські дії. Цей метод відкидає ряд часових неінформативних позицій за тимчасовим наслідком і створює більш компактну послідовність пози для класифікації. Тим не менш, ці послідовні підходи переважно використовують цілісний характер.

3. Часткові підходи.

Цей метод більш актуальний для структурованих об'єктів. Наприклад, якщо розглядати людину як структурований об’єкт то можна легко моделювати людські дії, використовуючи інформацію про рух від частин тіла. Частково-базові підходи розглядають інформацію про рух як з усього об'єкта так і з його частин. Перевага цієї лінії підходів полягає в тому, що вона по суті фіксує геометричні відносини між частинами об'єкта, що є важливою підказкою для розрізнення дій.[3]

Нещодавні дослідження показали, що особливості дії можуть бути досліджені за допомогою методів глибокого навчання, таких як згорткові нейронні мережі (CNN) та рекурентні нейронні мережі (RNN). Використовуючи RGB-кадри та оптичні потоки кадрів, нейронні мережі показали гарні результати на різних наборах даних про дії. Однак більшість методів, як і очікується, розпізнають дії лише на передчасно завершених відеозаписах. Їх продуктивність в умовах неповної дії має значно гірші результати.[4]

### 1.4 Аналіз методів прогнозування дій

За останні роки розпізнавання дій після фактів було детально вивчене, і були досягнуті плідні результати. Найсучасніші методи здатні точно дотримувати мітки дій після того як вона сталася. Однак у багатьох реальних сценаріях (наприклад, ДТП, чи хірургічна операція) інтелектуальні системи не мають розкоші очікування всього відео, перш ніж реагувати на зміст, що міститься в ньому. Наприклад, вміти передбачити небезпечну ситуацію, що трапиться до того, як це споюється. Крім того, було б чудово, якби спеціальна система могла спрогнозувати траєкторію руху хірургічного інструменту щоб уникнути непередбачуваних пошкоджень. На жаль, більшість існуючих підходів до розпізнавання дій не підходять для таких завдань класифікації, оскільки вони очікують побачити весь набір динаміки дій з повного відео, а потім приймати рішення.

Завдання прогнозування дій можна приблизно класифікувати на два типи - короткострокове прогнозування та довгострокове прогнозування. Перший, короткотерміновий прогноз, зосереджується на відеозаписах із короткою тривалістю дії, які зазвичай тривають протягом декількох секунд. Метою цього завдання є призначення міток дій на основі тимчасово неповних відеозаписів. Довгострокове передбачення чи передбачення намірів, визначає майбутні дії, засновані на поточних спостережуваних діях об'єкту. Цей тип призначений для моделювання переходу становищ об'єкту і таким чином, фокусується на тривалих відео, які тривають протягом декількох хвилин.

Класифікація короткострокової дії це завдання спрямоване на розпізнавання дії об’єкту на ранній стадії, тобто на основі тимчасово неповного відео. Мета полягає в тому, щоб досягти

високої точності розпізнавання, якщо спостерігається лише початкова частина відео. Спостережуване відео містить незакінчену дію, що робить завдання передбачення складним. [3]

Прогнозування наміру - можна представити як класифікацію послідовності дій з метою спрогнозувати кінцевий результат. Мета такої задачі не спрогнозувати чим закінчиться поточна дія, а з'ясувати до чого може привести комплекс дій виконаних об'єктом. В практиці існують певні типи дій, які містять декілька первісних моделей дій і демонструють складні часові домовленості, такі як "приготувати страву". Зазвичай тривалість цих комплексних дій довша, ніж короткочасних дій. Прогнозування цих довгострокових дій викликає приплив інтересу, оскільки це дозволяє нам зрозуміти, «що буде», включаючи кінцеву мету складної людської дії та правдоподібну передбачувану дію людини в найближчому майбутньому.

Одним із способів прогнозування дій є прогнозування траєкторії. Одним із ключових аспектів при прогнозуванні дій об'єкту є прогнозування траєкторії його руху. Прогнозування траєкторії руху, властива для людей здатність, обумовлює можливе призначення та траєкторію руху цільового об'єкту. Ми можемо передбачити з високою впевненістю, що людина рухається по тротуару і не має наміру змінювати свою траєкторію. Оперуючи цими даними ми також можемо допустити, що ця людина має великі шанси уникнути потрапляння у ДТП. Тому цікаво вивчити, як змусити машини робити ту саму роботу.

Однак прогнозування майбутньої траєкторії руху об'єкту справді важке, оскільки прогнозування неможливо передбачити ізольовано. Якщо повернутися до прикладу з людиною на тротуарі то можна побачити, що не всі фактори враховані. Минулі припущення що до руху людини не враховують того, що трапиться в випадку якщо по тротуару будуть іти інші люди. Тепер нам ще необхідно спрогнозувати як людина за якою ми спостерігаємо пристосує свій рух до поведінки інших. Відповісти на всі ці питання стає дуже важкою задачею і щоб її вирішити нам вже необхідно спрогнозувати рух інших пішоходів, винести якість припущення яку з стратегій руху обере кожен з них і що станеться якщо хтось з пішоходів різко захоче змінити свій напрямок.

### 1.5 Аналіз методів розпізнавання об’єктів

В даний час існує безліч завдань, в яких потрібно прийняти деяке рішення в залежності від присутності на зображенні об'єкта або класифікувати його. Здатність «розпізнавати» вважається основною властивістю біологічних істот, в той час як комп'ютерні системи цією властивістю в повній мірі не володіють.

Розглянемо загальні елементи моделі класифікації.

*Клас* - безліч об'єктом мають загальні властивості. Для об'єктів одного класу передбачається наявність «схожості». Для завдання розпізнавання може бути визначено довільну кількість класів, більше 1. Кількість класів позначається числом S. Кожен клас має свою ідентифікує мітку класу.

*Класифікація* - процес призначення міток класу об'єктів, відповідно до деякого опису властивостей цих об'єктів. Класифікатор - пристрій, який в якості вхідних даних отримує набір ознак об'єкта, а в якості результату видає мітку класу.

*Верифікація* - процес зіставлення примірника об'єкта з однією моделлю об'єкта або описом класу.

Під *описом* будемо розуміти найменування області в просторі ознак, в якій відображається безліч об'єктів або явищ матеріального світу. Ознака - кількісний опис тієї чи іншої властивості досліджуваного предмета або явища.

*Простір ознак* це N-мірний простір, певне для даної задачі розпізнавання, де N - фіксоване число вимірюваних ознак для будь-яких об'єктів. Вектор з простору ознак x, відповідний об'єкту завдання розпізнавання це N-мірний вектор з компонентами (x_1, x_2, ..., x_N), які є значеннями ознак для даного об'єкта.

Іншими словами, розпізнавання образів можна визначити, як віднесення вихідних даних до певного класу за допомогою виділення істотних ознак або властивостей, які характеризують ці дані, із загальної маси несуттєвих деталей.

Найчастіше вихідним матеріалом служить отримане з камери зображення. Завдання можна сформулювати як отримання векторів ознак для кожного класу на даному зображенні. Процес можна розглядати як процес кодування, що полягає в присвоєнні значення кожною ознакою з простору ознак для кожного класу.

Другим завданням розпізнавання є виділення характерних ознак або властивостей з вихідних зображень. Це завдання можна віднести до попередньої обробки. Ознака повинна представляти із себе характерну властивість конкретного класу, при цьому загальні для цього класу. Ознаки, що характеризують відмінності між - міжкласовими ознаками. Ознаки загальні для всіх класів не несуть корисної інформації і не розглядаються як ознаки в задачі розпізнавання. Вибір ознак є однією з важливих задач, пов'язаних з побудовою системи розпізнавання.

Розпізнавання образів, як правило, класифікується відповідно до типу навчальної процедури, що використовується для отримання вихідного значення. Контрольоване навчання передбачає, що набір навчальних даних (навчальний набір) був наданий, що складається з набору примірників, які були належним чином позначені вручну з правильним результатом.

Процедура навчання потім створює модель, яка намагається досягти двох іноді суперечливих цілей: виконувати якомога краще на даних про навчання та узагальнювати якнайкраще нові дані. З іншого боку, непідконтрольне навчання передбачає дані про навчання, які не були марковані вручну, і намагається знайти властиві шаблони в даних, які потім можуть бути використані для визначення правильного вихідного значення для нових екземплярів даних. Нещодавно вивчене поєднання двох - це напівконтрольне навчання, яке використовує комбінацію мічених та немечених даних (як правило, невеликий набір мічених даних у поєднанні з великою кількістю незазначених даних). Зауважте, що у випадках непідвладного навчання може взагалі не бути даних про навчання; Іншими словами, і дані, що підлягають маркуванню, - це дані про навчання.

### 1.5.1 Методи машинного навчання

#### 1.5.1.1 Інваріантне масштабне перетворення функції

The scale-invariant feature transform (SIFT) - алгоритм виявлення функцій в комп'ютерному зорі для виявлення та опису локальних особливостей у зображеннях. Цей метод здатен на розпізнавання об'єктів, роботизоване картографування та навігацію, зшивання зображень, 3D-моделювання, розпізнавання жестів, відстеження відео, індивідуальну ідентифікацію дикої природи.

У цьому алгоритмі зображення поєднується з фільтрами Гаусса в різних масштабах, а потім приймають різницю послідовних розмитих гауссових зображень. Після цього ключові точки приймаються як максимуми / мінімуми різниці гауссів (DoG - Difference of Gaussians) , які виникають у декількох масштабах. DoG зображення $D(x, y, \sigma)$ задається наступним чином (1.1).

$$D(x, y, \sigma) = L(x, y, k_i, \sigma) - L\left(x, y, k_j, \sigma\right), \tag{1.1}$$

де $L(x, y, k_i, \sigma)$ - це згортання вихідного образу $I(x, y)$ з розмиттям Гаусса $G(x, y, k\sigma)$ у масштабі $k\sigma$. Після отримання зображень DoG ключові точки ідентифікуються як локальні мінімуми / максимуми зображень DoG в масштабах. Це робиться шляхом порівняння кожного пікселя у зображеннях DoG з його вісьмома сусідами в одному масштабі та дев'ятьма відповідними сусідніми пікселями у кожному із сусідніх масштабів. Якщо значення пікселя є максимальним або мінімальним серед усіх порівняних пікселів, воно вибирається як ключова точка-кандидат.

Виявлення екстремуму в масштабі простору створює занадто багато кандидатів у ключові точки, деякі з яких нестабільні. Наступним кроком алгоритму є детальне пристосування до розташованих поблизу даних для точного розташування, масштабу та співвідношення основних кривих. Ця інформація дозволяє відхиляти точки, які мають низький контраст (і тому чутливі до шуму) або погано локалізовані уздовж краю [7].

Для виявлення точної позиції кандидата використовується інтерполяція довколишніх даних. До цього підхід полягав у тому, щоб просто знайти кожну ключову точку в місці розташування та масштабі ключового пункту-кандидата. Тепер завдяки використанню розрахунку інтерпольованого розташування екстремуму можна значно покращити відповідність та стабільність [8].

Інтерполяція проводиться за допомогою квадратичного розширення Тейлора для DoG функції шкали-простору використовуючи ключову точку кандидата як джерело. Розширення Тейлора задається наступним чином (1.2).

$$D(x) = D + \frac{\delta D^T}{\delta x}\, x + \frac{1}{2}\, x^T\, \frac{\delta^2 D}{\delta x^2}\, x. \qquad (1.2)$$

$D$ та його похідні оцінюються за ключовою точкою-кандидатом, а $x = (x, y, \sigma)^T$ - це зміщення від цієї точки. Розташування екстремуму визначається приймаючи похідну від цієї функції відносно X і встановивши його в нуль. Якщо зміщення екстремуму більше ніж 0.5 в будь якому вимірі, це вказує на те що екстремум лежить ближче до іншого ключового пункту кандидата. У цьому випадку ключову точку кандидата змінюють та замість цього проводять інтерполяцію. В іншому випадку зміщення додається до його ключового пункту-кандидата, щоб отримати інтерпольовану оцінку місця розташування екстремуму [9].

Попередні кроки знаходили місця розташування ключових точок у конкретних масштабах та присвоювали їм орієнтації. Це забезпечило інваріантність розташування зображення, масштабу та обертання зображення. Тепер ми хочемо обчислити вектор дескриптора для кожної ключової точки таким чином, що дескриптор відрізняється високою відмінністю та частково інваріантним для решти варіацій, таких як освітленість, 3D-точка зору тощо. Цей крок виконується на зображенні, найближчому за шкалою до шкали ключової точки. Спочатку створюється набір гістограм орієнтації на 4 × 4 піксельні мікрорайони кожна з яких має 8 секторів. Ці гістограми обчислюються за значеннями величини та орієнтації зразків в області 16 × 16 навколо ключової точки, так що кожна гістограма містить зразки 4 × 4 підрегіону сусідньої області. Величини та орієнтації градієнта зображення відбирають вибірку навколо місця розташування ключової точки, використовуючи шкалу ключової точки, щоб

вибрати рівень розмитості Гаусса для зображення. Для досягнення інваріантності орієнтації координати дескриптора та орієнтації градієнта обертаються відносно орієнтації ключових точок. Величини додатково зважуються функцією, що дорівнює половині ширини вікна дескриптора. Потім дескриптор стає вектором усіх значень цих гістограм. Оскільки існує 4 × 4 = 16 гістограм, кожна з яких має 8 секторів, вектор має 128 елементів. Потім цей вектор нормалізується до одиничної довжини, щоб посилити інваріантність для афінних змін освітленості. Для зменшення ефектів нелінійного освітлення застосовується поріг 0,2 і вектор знову нормалізується. Процес порогування, який також називають затисканням, може покращити відповідність результатів навіть тоді, коли ефектів нелінійного освітлення немає. Поріг 0,2 був вибраний емпіричним шляхом, і замінивши фіксований поріг одним систематично обчисленим, відповідні результати можуть бути покращені [10].

#### 1.5.1.2 Гістограма орієнтованих градієнтів

Гістограма орієнтованих градієнтів (HOG) - це дескриптор ознак, який використовується в комп'ютерному зорі та обробці зображень з метою виявлення об'єктів. Метод підраховує виникнення градієнтної орієнтації в локалізованих ділянках зображення. Цей метод схожий з гістограмами крайової орієнтації, дескрипторами перетворення ознак, інваріантними масштабами, та контекстами форми, але відрізняється тим, що він обчислюється на щільній сітці рівномірно розташованих комірок і використовує перекриття локалізації нормалізації контрасту для підвищення точності.

Основна думка, що стоїть за гістограмою дескриптора орієнтованих градієнтів, полягає в тому, що зовнішній вигляд і форма локального об'єкта в межах зображення можуть бути описані розподілом градієнтів інтенсивності або напрямками краю. Зображення поділяється на невеликі сполучені області, які називаються клітинками, а для пікселів всередині кожної комірки складається гістограма напрямків градієнта. Дескриптор - це конкатенація цих гістограм. Для поліпшення точності локальні гістограми можуть бути нормалізовані на контрасті, обчисливши міру інтенсивності в більшій області зображення, що називається блоком, а потім використовувати це значення для нормалізації всіх осередків блоку. Ця нормалізація призводить до кращої інваріації, до змін освітленості та затінення. Дескриптор HOG має ряд ключових переваг перед іншими дескрипторами. Оскільки він діє на локальних осередках, він інваріантний геометричним та фотометричним перетворенням, за винятком орієнтації на об'єкт. Такі зміни з’являться лише у великих просторових регіонах [11].

Перший крок підрахунку в багатьох детекторах функцій при попередній обробці зображення - це забезпечення нормалізованих значень кольору та гамми. При обчисленні дескриптора HOG цей крок може бути опущений, оскільки наступна нормалізація дескриптора, по суті, досягає того ж результату. Попередня обробка зображення надає незначний вплив на продуктивність. Натомість перший крок - це обчислення значень градієнта. Найпоширеніший метод - нанесення 1-D орієнтованої в точці дискретної похідної маски в одному або обох горизонтальному та вертикальному напрямках. Зокрема, цей метод вимагає фільтрації даних про колір або інтенсивність зображення за допомогою наступних ядер фільтра

$$[-1, 0, 1] \text{ та } [-1, 0, 1]^T$$

Другий крок обчислення - створення гістограм клітинок. Кожен піксель у комірці подає зважений голос для каналу гістограми на основі орієнтації та значень, знайдених у розрахунку градієнта. Самі осередки можуть мати прямокутну або радіальну форму, а канали гістограми рівномірно розподіляються на 0-180 градусів або 0- 360 градусів, залежно від того, градієнт є "непідписаним" або "підписаним". Що стосується ваги голосу, то вклад пікселя може бути або самою величиною градієнта, або деякою функцією величини. Під час тестів величина градієнта сама по собі дає найкращі результати. Інші варіанти ваги голосу можуть включати квадратний корінь або квадрат градієнтної величини або якусь відрізану версію величини. [11]

Щоб врахувати зміни освітленості та контрасту, градієнтні сили повинні бути локально нормалізовані, що вимагає групування комірок у більші, просторово пов'язані блоки. Дескриптор HOG - це зв'язаний вектор компонентів нормалізованих гістограм клітин з усіх блокових областей. Ці блоки, як правило, перекриваються, тобто кожна комірка вносить внесок у кінцевий дескриптор не один раз. Існують дві основні геометрії блоків: прямокутні блоки R-HOG та кругові блоки C-HOG. Блоки R-HOG - це, як правило, квадратні сітки, представлені трьома параметрами: кількість комірок на блок, кількість пікселів на клітинку та кількість каналів на клітинку гістограми. Також поліпшення продуктивності можна досягти, застосувавши Гауссова просторове вікно у кожному блоці перед таблицею голосів гістограми, щоб зменшити вагу пікселів по краю блоків. Блоки R-HOG виглядають досить схожими на масштабно-інваріантні дескриптори перетворення функції (SIFT), однак, незважаючи на подібне формування, блоки R-HOG обчислюються в щільних сітках за деякою шкалою без вирівнювання орієнтації, тоді як дескриптори SIFT зазвичай обчислюються в розріджених, інваріантних масштабах ключових точках зображення і повертаються для вирівнювання

орієнтації. Крім того, блоки R-HOG використовуються спільно для кодування інформації про просторові форми, тоді як дескриптори SIFT використовуються поодиноко.

Кругові блоки HOG (C-HOG) можна знайти у двох варіантах: блоці з одинарною центральною коміркою та центральній комірці з кутовим поділом. Крім того, ці блоки C-HOG можна описати чотирма параметрами: кількістю кутових та радіальних секторів, радіусом центрального сектору та коефіцієнтом розширення для радіусу додаткових радіальних секторів. Блоки C-HOG схожі на дескриптори контексту форми, але сильно відрізняються тим, що блоки C-HOG містять комірки з кількома орієнтаційними каналами, тоді як контексти форм використовують лише один підрахунок присутності краю в їх формулюванні [11].

Дескриптори HOG можуть використовуватися для розпізнавання об'єктів, надаючи їх як функції алгоритму машинного навчання. У деяких експериментах дескриптори HOG використовували як функції в для методу опорних векторів (SVM), однак дескриптори HOG не прив'язані до конкретного алгоритму машинного навчання.

### 1.5.2 Методи глибокого навчання

Розпізнавання об'єктів за допомогою нейронних мереж та глибокого навчання поділяється на два основні підходи: двоступеневий (two-stage) та одноступеневий (one-stage).

Двоступеневий підхід складається з двох частин, де перша генерує розріджений набір кандидатів об'єктних пропозицій, а друга визначає точні області об'єктів та відповідні мітки класу за допомогою згорткових мереж. Зокрема, двоступеневий підхід досягає кращої продуктивності на кількох складних наборах даних. Після цього запропоновано численні ефективні методи для підвищення продуктивності, такі як діаграма архітектури, стратегія навчання, контекстуальне обґрунтування.

Одноступеневий підхід використовує єдину згорнуту мережу для прямого прогнозування об'єктних класів та локацій. Для підвищення точності деякі одноступеневі методи  мають на меті вирішити проблему незбалансованості екстремального класу шляхом повторного проектування функції втрат або стратегій класифікації. Хоча одноступеневі детектори досягли хорошого прогресу, їх точність все-таки не відповідає рівню двоступеневих методів.

#### 1.5.2.1 Мережа пропозицій регіону (RPN)

Останні досягнення в області виявлення об'єктів зумовлені успіхом методів пропозицій регіону  та конволюційних нейронних мереж на основі регіону (R-CNN). Незважаючи на те, що

CNN, що базуються на регіонах, були дорогими, з точки зору швидкості виконання, останнє втілення, Fast R-CNN, досягає майже реального часу, використовуючи дуже глибокі мережі , ігноруючи час, витрачений на пропозиції регіонів. Зараз пропозиції є найвищим місцем для обчислень у найсучасніших системах виявлення. Методи пропозицій регіонів зазвичай покладаються на невмілі функції та економічні схеми вибору. Вибірковий пошук, один із найпопулярніших методів, жадібно об’єднує суперпікселі на основі функцій інженерно низького рівня. Але порівняно з ефективними мережами виявлення, селективний пошук - це на порядок повільніше, на 2 секунди на зображення в процесі здійснення процесора. EdgeBoxes в даний час забезпечує найкращий компроміс між якістю пропозиції та швидкістю - 0,2 секунди на зображення.

Faster R-CNN, складається з двох модулів. Перший модуль являє собою глибоко згорнуту мережу, яка пропонує регіони (Region Proposal Networks – RPN) , а другий модуль - швидкий детектор R-CNN, який використовує запропоновані області.

Мережа пропозицій регіону (RPN) сприймає зображення (будь-якого розміру) як вхідні дані та виводить набір пропозицій прямокутних об'єктів, кожна з оцінкою об'єктивності. Для створення області пропозиції використовується невелика мережа поверх функції виводу за допомогою останнього спільного згорткового шару. Ця невелика мережа приймає на вхід n × n просторове вікно вхідної карти зведених функцій. Кожне розсувне вікно відображається на функцію нижнього розміру, ця функція подається у два повністю пов'язані між собою шари - шар регресії (reg) та шар класифікації (cls). Ця міні-мережа працює в розсувному вікні, її повністю пов'язані шари поділяються у всіх просторових місцях. Ця архітектура, природно, доповнена рівномірним конволюційним шаром з подальшими шарами згортки 1×1.

У кожному місці ковзкого вікна одночасно прогнозується кілька пропозицій регіону, де число максимально можливих пропозицій для кожного місця позначається запитом. Таким чином, у шарі регресії є 4k виходи, що кодують координати боксів k, а шар класифікації видає 2k балів, які оцінюють ймовірність віднесення об'єкта для кожної пропозиції. Ці пропозиції має параметричні відносні бокси для вибору, які називаються якорями. Якір зосереджений у ковзкому вікні залежить від масштабу та аспектним співвідношенням. За замовчуванням ми використовуємо 3 шкали та 3 співвідношення сторін, отримуючи k = 9 якорів у кожному положенні ковзання.

Тренуючи RPN, ми присвоюємо бінарну мітку класу (бути об’єктом чи ні) для кожного якоря. Позитивну мітку ми підписуємо двома видами якорів: перший коли об’єкт можна точно спів поставити з одним із класів. Другий коли має вірогідність спів падання більше, ніж 70%.

Негативну мітку якоря присвоюють у випадку коли вірогідність менша ніж 30%. Функція втрати виглядає наступним чином (1.3).

$$L(\{P_i\},\{t_i\}) = \frac{1}{N_{cls}} \sum_i L_{cls}(p_i, p_i^*) + \lambda \frac{1}{N_{reg}} \sum_i p_i^*(t_i,\ t_i^*) \quad (1.3)$$

Тут $i$ є індексом якоря в міні-батчі, а $p_i$ - це вірогідність прив'язки якоря до об'єкта. $p_i^*$ буде рівнятися 1 якщо якорь позитивний і 0 якщо негативний. $t_i$ - вектор, що представляє 4 параметризовані координати передбачуваного обмежувального поля. Для визначення втрати регресії використовуємо $L_{reg}(t_i, t_i^*) = R(t_i - t_i^*)$. $p_i^* L_{reg}$ означає що функція втрат активується лише для позитивних якорів ($p_i^*=1$) і відключається в іншому випадку ($p_i^*=0$).

RPN можна тренувати за допомогою функції зворотнього розповсюдження помилки та стохастичного градієнтного спуску (SGD). Кожен міні-батч виникає із єдиного зображення, що містить безліч як позитивних так і негативних прикладів якорів. Можна оптимізувати функції втрат усіх якорів, але це буде спрямовано на негативні вибірки, оскільки вони домінують. Замість цього можна випадковим чином відбираємо 256 якорів у зображенні, щоб обчислити функцію втрат міні-батчу, де відбирали позитивні та негативні якорі у співвідношенні один до одного. Якщо на зображенні менше 128 позитивних прикладів, ми прокладаємо міні-партію негативними. Далі випадково ініціалізуємо всі нові шари шляхом нанесення ваг з нульового середнього гауссового розподілу з витриманим відхиленням 0,01. Усі інші шари ініціалізуються за допомогою попередньої підготовки моделі класифікації ImageNet.

Після реалізації RPN необхідно поєднати її з рекурентною нейронною мережею яка буде використовувати створені пропозиції для розпізнавання об'єктів. Для цього існує декілька методів. Перший варіант – це чергове навчання, цей підхід базується на тому, що спочатку тренують RPN і використовують отримані пропозиції для тренування R-CNN. Після цього натренована R-CNN використовується для ініціалізації RPN і цей процес циклічно повторюється. Другий варіант – це спільний тренінг. У цьому рішенні мережі RRN та Fast R-CNN об'єднуються в єдину мережу під час тренінгу. На кожній ітерації стохастичного градієнтного спуску прямий прохід генерує регіональні пропозиції, які трактуються так само, як фіксовані, попередньо обчислені пропозиції під час тренування R-CNN. Поширення зворотного руху відбувається як зазвичай, де для спільних шарів сигнали, що розповсюджуються назад, втрати RPN, і втрати R-CNN поєднуються разом [12].

#### 1.5.2.2 Single Shot Detector

Головою перевагою використання Single Shot Detector (SSD) є  те що потрібно зробити лише один знімок для виявлення декількох об'єктів у зображенні, тоді як регіональна мережа пропозицій (RPN), заснована на підходах, таких як серія R-CNN, потребує двох знімків, одному для створення пропозицій регіону, одному для виявлення Об'єкт кожної пропозиції. Таким чином, SSD набагато швидше порівняно з двосхилим RPN-підходами.

Основна ідея, що лежить в основі SSD, - це концепція вибору вікон за змовчуванням(default boxes). Вікна за змовчуванням представляють з себе ретельно вибрані обмежувальні поля на основі їх розмірів, співвідношення сторін та позицій по всьому зображенню. SSD містить 8732 таких вікон. Мета моделі - визначити, яке вікно використовувати для заданого зображення, а потім передбачити його зміщення для отримання остаточного результату.

Архітектура SSD складається з трьох головних компонентів:

1) Бзова мережа.

Базова мережа по суті є початковими шарами будь-якої стандартної мережі класифікації зображень, попередньо підготовленої на наборі даних ImageNet. Повністю з'єднані шари в кінці реалізуються у вигляді згорткових шарів. Кінцевим висновком базової мережі є карта об'єктів розміром 19 x 19 x 1024.

2) Додаткові шари

Поверх базової мережі додаються 4 додаткові згорткові шари, що дозволяють зменшити розмір карт функцій до тих пір, поки не буде отримана остаточна карта об'єктів розміром 1 x 1 x 256.

3) Шари прогнозування

Шари прогнозування є найважливішим компонентом SSD. Замість того, щоб просто використовувати одну карту функцій для прогнозування класифікаційних балів та координатних обмежувальних вікон, використовуються декілька функціональних карт, що представляють кілька масштабів. Тут використовується ідея про вікна за замовчуванням і роздільну здатність карти, про яку йшлося вище.

Тепер враховуючи кількість класів (включаючи фоновий клас як клас 0), який повинен бути , кожне прогнозування представлено (C + 4) числами: оцінками класифікації C та 4 зсувами: Δcx, Δcy, w і h, що представляють собою компенсації від центру вікна за замовчуванням та його розмірів. Таким чином, для карт функції з k полями за замовчуванням на клітинку шаром прогнозування є 3-х згортковий шар 3 x 3 з k * (C + 4) каналами.

Оскільки SSD сильно покладається на поля за замовчуванням, він є дуже чутливим до вибору вікон за замовчуванням. Межі поля за замовчуванням вибираються вручну. SSD визначає значення масштабу для кожного шару карти функції. Починаючи зліва виявляє об’єкти в найменшій шкалі 0,2 (або іноді 0,1), а потім лінійно збільшується до самого правого шару в масштабі 0,9. Комбінуючи значення масштабу з цільовими співвідношеннями сторін, ми обчислюємо ширину та висоту полів за замовчуванням. Для шарів, що роблять 6 прогнозів, SSD починається з 5 цільових співвідношень сторін: 1, 2, 3, 1/2 та 1/3. Тоді ширина та висота полів за замовчуванням обчислюються наступним чином (1.4, 1.5).

$$w = scale * \sqrt{aspect\ ratio} \tag{1.4}$$

$$h = \frac{scale}{\sqrt{aspect\ ratio}}, \tag{1.5}$$

де w – довжина вікна, а h – висота вікна. Scale – масштаб, aspect ratio - співвідношення сторін.

У SSD використовується функція витрат Multibox loss, яка складається з двох термінів: втрата довіри та втрата локалізації. На вихід мережа подає (C+4) прогнозів для кожного з 8732 вікон: вірогідність для відношення об'єктів до одного з C класів та 4 характеристики локалізації об’єкту *(dx, dy, h, w)*.

Втрата локалізації - це невідповідність між основним полем істини та передбачуваним граничним полем.

Втрата довіри - це втрата при прогнозуванні класу. За кожне правильне прогнозування враховуються втрати відповідно до рівня довіри відповідного класу. За негативні прогнози відповідності штрафуються збитки відповідно до показника довіри класу "0": клас "0" класифікує, що жоден об'єкт не виявлений.

Функція втрат розраховується за формулою (1.6).

$$L(x, c, l, g) = \frac{1}{N}(L_{conf}(x, c) + \alpha L_{loc}(x, l, g)), \tag{1.6}$$

де N - кількість позитивних збігів, а α - вага втрати за локалізацію.

Недоліком SSD є те що мережа робить більше прогнозів, ніж кількість присутніх об’єктів. Тож набагато більше негативних результатів, ніж позитивних. Це створює класовий дисбаланс, який шкодить навчанню. Через це може статися так, що модель навчиться розпізнавати фоновий простір, а не виявляти об'єкти. Однак цю проблему можна вирішити для

деяких випадків якщо замість того, щоб використовувати всі негативні результати разом, відсортувати їх за допомогою функції втрат довіри. SSD вибере негативні результати з найбільшою втратою і гарантує, що співвідношення між вибраними негативами та позитивами не більше 3:1. Це призводить до більш швидкого і стабільного тренування [12].

#### 1.5.2.3 Single-Shot Refinement

Single-Shot Refinement мережа є удосконаленою версією SSD і подібно до неї RefineDet заснований на згорнутої згорткової мережі, яка виробляє фіксовану кількість обмежувальних коробок і балів, що вказують на наявність різних класів об'єктів у цих полях. Мережа формується двома взаємопов'язаними модулями, ARM та ODM. ARM - спрямований на видалення негативних якорів, щоб зменшити простір пошуку для класифікатора, а також грубо відрегулювати місця та розміри якоря, щоб забезпечити кращу ініціалізацію для наступного регресора. ODM спрямований на регресування точних розташувань об'єктів та прогнозування міток для багатьох класів на основі вдосконаленого за допомогою ARM якоря.

ARM побудований шляхом видалення шарів класифікації та додавання деяких допоміжних структур. ODM складається з виходів TCB (блок з'єднання) з подальшими шарами прогнозування (тобто шарами згортання розміром 3 × 3 ядра), який генерує бали для класів об'єктів і зміщення форми відносно уточнених координат коду якоря. SSRD складається з трьох основних компонентів:

1) TCB (Transfer Connection Block) – блок з'єднання, пов'язує між собою блоки ARM та ODM. Щоб зв’язати ARM та ODM, був введен TCB для перетворення функцій різних шарів з ARM у форму, необхідну ODM, щоб ODM могла ділитися функціями з ARM. Також TCB інтегрує масштабний контекст шляхом додавання функцій високого рівня до переданих функцій для підвищення точності виявлення.

2) Двоступінчаста каскадна регресія служить для виявлення точного місця та розміру об'єктів. Цей компонент використовує регресію, у шарах з різними масштабами, щоб передбачити розташування та розміри об'єктів. За допомогою двоступінчастої каскадної регресії можна змінювати місця та розміри об'єктів.

3) Фільтрування негативного якоря - достроково відхилює добре класифіковані негативні якорі для пом’якшення проблеми дисбалансу.

Для поліпшення точності та стабільності у мережі використовується метод збільшення даних. Цей метод заключається в тому, що випадковим чином розширюються та обрізаються оригінальні навчальні зображення з додатковим випадковим фотометричним викривленням для отримання додаткових навчальних зразків.

Для обробки різних масштабних об'єктів використовується чотири функціональних шари із загальними строковими розмірами 8,16,32 та 64 пікселів пов'язаних з декількома різними масштабами прогнозування якорів. Кожен особливий шар асоціюється з однією специфічною шкалою якорів (тобто масштаб становить тричі від величини розміру відповідного шару) та трьома аспектними співвідношеннями (тобто 0,5,1,0 та 2,0). На етапі тренувань, визначається відповідність між якорями.

Функція втрати для RefineDet складається з двох частин, тобто втрати в ARM та втрати в ODM. Для ARM призначається бінарний маркер класу (це передбачуваний об’єкт, чи ні) кожному якіру і регресуєть ся його розташування та розмір одночасно, щоб отримати вдосконалений якір. Після цього очищені якіри пропускаються з негативною впевненістю, що знижують поріг до ODM для подальшого прогнозування категорій об'єктів та точних розташувань і розмірів об'єктів. За допомогою цих визначень функцію втрати визначається наступним чином (1.7).

$$L(\{p_i\},\{x_i\},\{c_i\},\{t_i\}) = \frac{1}{N_{arm}} \left(\sum_i L_b(p_i,[l_i^* \geq 1]) + \sum_i [l_i^* \geq 1]\, L_r(x_i, g_i^*)\right), \qquad (1.7)$$

де $i$ – це індекс якоря в батчі, $l_i^*$ - мітка класу для якоря $i$, $g_i^*$ - місце розташування і розмір якоря $i$, $p_i$ та $x_i$ - передбачувана впевненість у прив'язці якоря до об'єкта та уточнені координати якоря в ARM. $c_i$ та $t_i$ - це передбачуваний клас об'єктів і координати обмежувального поля в ODM. $N_{ARM}$ та $N_{ODM}$ - кількість позитивних якорів в ARM і ODM відповідно [13].

## 1.6 Аналіз методів сегментації зображень

Мета семантичної сегментації зображення - позначити кожен піксель зображення відповідним класом того, що представлено. Оскільки ми прогнозуємо кожен піксель на зображенні, це завдання зазвичай називають щільним прогнозуванням.

Очікуваний вихід у семантичній сегментації - це не лише мітки та параметри обмежувальної рамки. Вихід сам по собі - зображення високої роздільної здатності (як правило, такого ж розміру, як і вхідне зображення), в якому кожен піксель віднесений до певного класу. Таким чином, це класифікація зображень на рівні пікселів.

Ми можемо розділити сегментацію зображення на різні методи. Наприклад, методи, засновані на техніках стиснення, передбачають, що найкращим методом сегментації є той, який мінімізує довжину кодування даних та загальну ймовірну сегментацію. Відомо, що методи, засновані на гістограмах, надзвичайно добре організовані для оцінки додаткових схем

сегментації, оскільки їм потрібно лише однократне перевищення при прогресуванні пікселів [14]. У цій схемі всі пікселі зображення враховуються для відображення гістограми, а долини та піки гістограми використовуються для встановлення кластерів у зображенні. Внутрішня обробка зображення, виявлення ребер - це надійне поле самостійно. Краї області та межі пов'язані безпосередньо, оскільки часто відбувається швидка модифікація сили в області меж [15].

### 1.6.1 Метод порогового значення

Найпростіший метод сегментації зображення називається методом порогового значення. Цей метод заснований на рівні порогового значення для перетворення сірого зображення у бінарне. Ключ цього методу полягає у виборі порогового значення (або значень). Нові методи запропонували використовувати багатовимірні нечіткі правила які створюють нелінійні пороги. У цих роботах рішення щодо належності кожного пікселя до сегменту ґрунтується на багатовимірних правилах, що випливають із нечіткої логіки та еволюційних алгоритмів, заснованих на середовищі та освітлення зображення [16].

### 1.6.2 Кластерний метод

Зазвичай цей метод реалізуються за рахунок алгоритму K-means (к – середнє) - це ітеративна техніка, яка використовується для розділення зображення на К кластерів. Алгоритм можна описати декількома шагами:

1) Спочатку вибирається К кластерних центрів. Кластерні центри вибираються випадковим чином або на основі евристичного методу.

2) Після цього кожен піксель на зображенні присвоюється до одного з кластерів. Кластер до якого буде відноситися піксель обирається за рахунок мінімізації відстані між пікселем та центром кластера.

3) Шляхом усереднення усіх пікселів кластера обчислюються нові більш точніші центри кластерів.

4) Кроки номер 2 і 3 необхідно повторювати доки пікселі не перестануть змінювати кластери.

Відстань між пікселем та центром кластера представлена як квадрат або абсолютна різниця між пікселем та центром кластера. Різниця, як правило, ґрунтується на кольорі пікселів, інтенсивності, текстурі та розташуванні або зваженій комбінації цих факторів. К може бути обраний вручну, випадковим чином або евристикою. Цей алгоритм гарантовано збігається, але

може не повернути оптимальне рішення. Якість рішення залежить від початкового набору кластерів і значення K [16].

### 1.6.3 Сегментація заснована на компресії

Ідея цього методу полягає втому, що оптимальною сегментацією вважається та, яка мінімізує за всіма можливими сегментаціями довжину кодування даних. Цей метод під час роботи за допомогою сегментації намагається знайти шаблони в зображенні, і будь-яка закономірність зображення може бути використана для його стиснення. Метод описує кожен сегмент за його фактурою та граничною формою. Кожен з цих компонентів моделюється функцією розподілу ймовірностей, і його довжина кодування обчислюється.

Для будь-якої заданої сегментації зображення алгоритм алгоритм розраховує деяку кількість бітів, необхідних для кодування цього зображення на основі заданої сегментації. Таким чином, серед усіх можливих сегментацій зображення, мета - знайти сегментацію, яка створює найкоротшу довжину кодування. Цього можна досягти простим агломераційним методом кластеризації [17].

### 1.6.4 Методи засновані на гістограмах

Методи на основі гістограм є дуже ефективними порівняно з іншими методами сегментації зображень, оскільки вони, як правило, потребують лише одного проходу через пікселі. У цій техніці гістограма обчислюється з усіх пікселів зображення, а піки та долини в гістограмі використовуються для визначення кластерів на зображенні. Колір або інтенсивність можна використовувати як міру.

Удосконалення цієї методики полягає в рекурсивному застосуванні методу пошуку гістограми до кластерів зображення, щоб розділити їх на менші кластери. Ця операція повторюється з меншими та меншими кластерами до тих пір, поки не сформується більше кластерів.

Одним з недоліків методу пошуку гістограм є те, що може бути важко визначити значні вершини та долини на зображенні.

Підходи, засновані на гістограмах, також можуть бути швидко адаптовані до застосування до декількох кадрів, зберігаючи ефективність їх одноразового проходження. Гістограма може бути виконана декількома способами, якщо розглянуто кілька кадрів. Той самий підхід, який застосовується з одним кадром, може бути застосований до декількох, і після об'єднання результатів піки та долини, які раніше було важко визначити, є більш імовірними. Гістограма також може бути застосована на основі пікселя, де отримана інформація

використовується для визначення найбільш частого кольору для місця пікселя. Цей сегмент підходу базується на активних об'єктах та статичному середовищі, в результаті чого сегментація різного типу корисна для відстеження відео [18].

### 1.6.5 Виявлення країв

Виявлення країв - це добре розроблене поле в межах обробки зображень. Межі регіонів та краї тісно пов'язані, оскільки на кордонах області часто відбувається різке регулювання інтенсивності. Тому методи виявлення країв використовувались як основа іншої методики сегментації. Краї, визначені методом виявлення ребер, часто від’єднуються. Щоб сегментувати об'єкт із зображення, потрібні обмежені межі області. Бажані ребра - це межі між такими об'єктами або просторовими таксонами [19]. Просторові таксони - це інформаційні гранули, що складаються з чіткої піксельної області, розміщеної на рівнях абстракції в межах ієрархічної вкладеної архітектури сцен. Вони схожі на гештальт-психологічне позначення фігури-землі, але поширюються на передній план, групи об'єктів, предмети та важливі частини об'єкта. Методи виявлення країв можуть бути застосовані до просторово-таксонової області таким же чином, як і до силуету. Цей метод особливо корисний, коли від'єднаний край є частиною ілюзорного контуру. Методи сегментації також можуть застосовуватися до країв, отриманих від крайових детекторів [20].

### 1.6.6 Методи вирощуваних регіонів

Методи вирощування регіонів в основному покладаються на припущення, що сусідні пікселі в одному регіоні мають аналогічні значення. Загальна процедура полягає в порівнянні одного пікселя з сусідами. Якщо критерій подібності задовольняється, піксель може бути встановлений так, щоб він належав тому ж кластеру, що і один або кілька його сусідів. Вибір критерію подібності є вагомим, і на результати впливає шум у всіх випадках.

Метод статистичного об'єднання регіонів (SRM) починається з побудови графіка пікселів, використовуючи 4-зв'язаність з ребрами, зваженими на абсолютне значення різниці інтенсивності. Спочатку кожен піксель утворює одну піксельну область. Потім SRM сортує ці краї в черзі пріоритету і вирішує, чи слід об’єднувати поточні регіони, що належать крайовим пікселям, використовуючи статистичний предикат[21].

## 1.7 Постановка наукової задачі та обґрунтування методики досліджень

Проблема розпізнавання дій об'єктів (таких як рух інструментів, зміни хірургічного поля, стан внутрішніх органів тощо) по відео (наприклад, що надходять з ендоскопу або лапароскопу) є дуже складною, оскільки одночасно можуть виникати різні ситуації, і оклюзії ще більше ускладнюють розпізнавання. Крім того, відео традиційно обробляється в цілому, тому рішення про те, який клас дії спостерігається, не може бути прийнято в режимі реального часу (миттєво). Це значно знижує потенціал медичної системи відеозйомки; більшість з них використовуються як інструмент аналітики історичних даних замість проактивних і профілактичних інструментів. Більш того, незважаючи на значні зусилля, сучасні системи призначені тільки для розпізнавання одного типу дії на відеокадрі або можуть працювати з покроковим розпізнаванням, але не можуть одночасно знайти необхідну дію в кожному відеокадрі. Найбільш ефективні методи виявлення дій об'єктів у відео на сьогоднішній день по своїй суті є автономними, оскільки вони засновані на виявленні пропозицій по регіонах кадр за кадром і об'єднанні їх в так звану «послідовність фреймів» лише на етапі подальшої обробки.

Сучасний розвиток систем комп'ютерного зору і машинного навчання робить можливим впровадження в медичні системи відеозйомки підтримку прийняття медичних рішень щодо подальшого розвитку сценарію надання медичної допомоги.

За результатами проведеного аналізу моделей, методів й інструментальних засобів, що використовуються для розпізнавання об'єктів та моделювання можливого розвитку ситуацій слід визначити наступні задачі магістерської роботи:

1) аналіз методів та моделей розпізнавання, сегментації відео, прогнозування розвитку ситуацій;
2) розроблення методу систематизованого прогнозування дій об’єктів у відеопотоці та визначення сукупності етапів прогнозування;
3) розроблення моделі розпізнавання з використанням методу обмежуючої коробки;
4) розроблення моделі розпізнавання з використанням методу сегментації;
5) тестування розроблених моделей;
6) оцінка ефективності моделей розпізнавання об’єктів у відео.

Для проведення досліджень доцільно використовувати методи моделювання відеопотоку, технології глибокого навчання, нейронні мережі.

## 1.8 Висновки до першого розділу

Аналіз досліджень методів та моделей розпізнавання, сегментації відео, прогнозування розвитку ситуацій дозволив сформулювати наступні висновки.

1. Актуальною галуззю застосування технології розпізнавання об'єктів у відео та прогнозування розвитку ситуацій є медичні системи відеозйомки, що використовуються під час проведення діагностичного або оперативного втручання у закладах надання медичної допомоги з використанням спеціального медичного обладнання.

2. Для розпізнавання об'єктів у відео та подальшого прогнозування дій об'єктів доцільно використовувати технологію глибокого навчання та нейронні мережі.

3. Для отримання комплексної оцінки розвитку дій під час проведення діагностичного або оперативного втручання з використанням відеозйомки доцільно застосовувати комплексну оцінку на підставі моделей, розроблених з використанням методів обмежувальної коробки та сегментації.

4. Сформульована загальна науково-технічна задача дослідження, як задача удосконалення методів і моделей прогнозування дій об’єктів у відео за рахунок підвищення їх точності методів і моделей.

# РОЗДІЛ 2 ДОСЛІДЖЕННЯ МЕТОДІВ, МОДЕЛЕЙ РОЗПІЗНАВАННЯ ТА СЕГМЕНТАЦІЇ ОБ'ЄКТІВ У ВІДЕО

## 2.1 Загальна структура методу прогнозування дій об'єктів у медичних відео зображень

Структура методу прогнозування дій об'єктів у медичних відео зображень

Підготовка і розпізнавання відеозображень містить наступні етапи, як це представлено на рисунок 2.1.

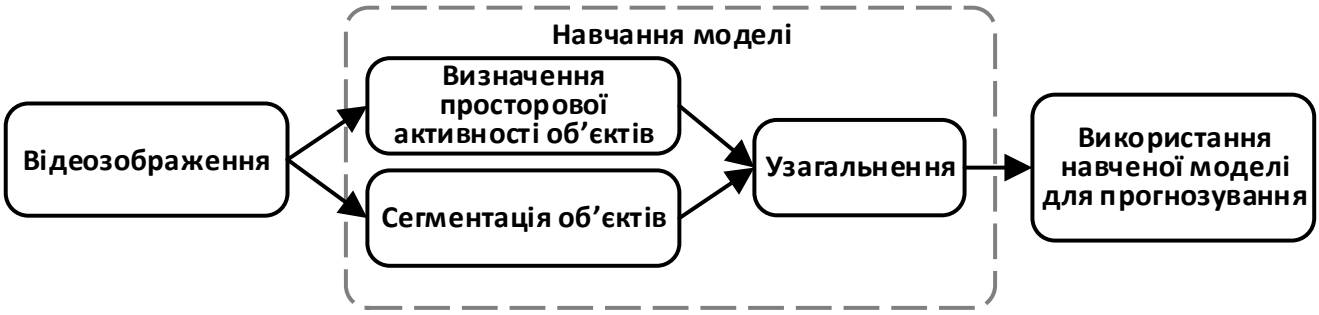


Рисунок 2.1 - Етапи розпізнавання об'єктів та прогнозування

1. Навчання моделі для розпізнавання об'єктів, що передбачає визначення просторової активності об'єктів з використанням анотованих координат обмежуючого прямокутника, відповідних кожному класу, сегментацію з використанням даних масок анотованих об'єктів, цілочисельне значення для кожного зображення та узагальнення – об'єднання узагальненого уявлення об'єктів.

2. Використання навченої моделі для прогнозування дій об'єктів нових відеозображень в режимі реального часу.

## 2.2 Розпізнавання об'єктів

Розпізнавання об'єктів - це комп'ютерна технологія, пов'язана з комп'ютерним зором та обробкою зображень, яка стосується виявлення екземплярів смислових об'єктів певного класу (наприклад, людей, будівель чи автомобілів) у цифрових зображеннях та відео. Добре досліджені області виявлення об'єктів включають в себе виявлення обличчя та виявлення

пішоходів. Для виявлення об'єктів є програми в багатьох областях комп'ютерного зору, включаючи пошук зображень та відеоспостереження.

Методи виявлення об'єктів зазвичай підпадають або до підходів, заснованих на машинному навчанні, або до методів глибокого навчання. Для підходів до машинного навчання необхідно спочатку визначити особливості за допомогою одного із наведених нижче методів, а потім використати таку техніку, як машина підтримки векторів (SVM) для класифікації. З іншого боку, методи глибокого навчання здатні робити виявлення об'єктів в кінці без конкретного визначення особливостей і, як правило, засновані на згорткових нейронних мережах (CNN).

Розпізнавання об’єктів проводилось з використанням You Only Look Once (YOLO). Мережа YOLO постійно розвивається та має вже декілька версій найостанніша з яких YOLO3.

Під час роботи YOLO ділить вхідне зображення на сітку S × S. Кожна комірка сітки передбачає лише один об’єкт. Коли мережа змогла розпізнати об'єкт вона поміщає його у граничне поле (bounding box). Граничне поле використовується, щоб обмежити об'єкт. Кожне граничне поле містить 5 елементів: (x, y, w, h) де $x$ та $y$ – координати боксу, а $w$ та $h$ його висота та ширина. П'ятий елемент  це показник достовірності поля який відображає достовірність, вірогідна того, що у граничному полі міститься очікуваний об'єкт та наскільки точні його межі.

Основна концепція YOLO - побудувати мережу CNN для прогнозування (7, 7, 30) тензора. YOLO використовує мережу CNN для зменшення просторових розмірів до 7 × 7 з 1024 вихідними каналами в кожному місці. YOLO виконує лінійну регресію за допомогою двох повністю з'єднаних шарів, щоб зробити передбачення граничного поля 7 × 7 × 2. Для остаточного прогнозування зберігаються ті результати, які мають високі показники достовірності поля (більше 0,25) як остаточні прогнози.

YOLO має 24 згорткових шара, а також ще 2 повністю з'єднаних шари. Деякі з шарів згортки альтернативно використовують шари відновлення 1 × 1 для зменшення глибини карт ознак. Останній шар згортки подає на вихід тензор з формою (7, 7, 1024). Потім тензор сплющується. Використовуючи 2 повністю з’єднаних шари як форму лінійної регресії, і на виході отримується тензор (7 × 7 × 30). Також існує більш швидка але менш точна версія мережі яка має назву FastYOLO. Вона використовує лише дев'ять згорткових шарів.

YOLO прогнозує декілька граничних полів на одну комірку сітки. Необхідно щоб лише одна з них відповідала за об'єкт. Для цього ми вибираємо поле, що має найвищий коефіцієнт IoU( intersection over union - перетин над з'єднанням). Кожен прогноз стає кращим при прогнозуванні певних розмірів і співвідношення сторін. YOLO використовує іфункцію

залишкової суми квадратів між прогнозами істинними даними для обчислення втрат. Функція втрат складається з трьох частин:

1) Функція втрати класифікації – розраховується якщо об'єкт виявлений. У цьому випадку втрата класифікації в кожній комірці - це помилка квадрата умовної ймовірності класу відносно до кожного класу.

$$\sum_{i=0}^{S^2} \mathbf{1}_i^{obj} \sum_{c \in classes} (p_i(c) - \hat{p}_i(c))^2 \quad (2.1)$$

У цій формулі $1_i^{obj}$ дорівнює 1 якщо об'єкт є у *i* комірці, інакше 0. $p_i\hat{}(c)$ – умовне позначення вірогідності класу *c* у *i* комірці.

2) Локалізаційні втрати - вимірює помилки у передбачуваних місцях та розмірах вікон.

$$\lambda_{coord} \sum_{i=0}^{S^2} \sum_{j=0}^{B} \mathbf{1}_i^{obj} [(x_i - \hat{x}_i)^2 + (y_i - \hat{y}_i)^2] + \lambda_{coord} \sum_{i=0}^{S^2} \sum_{j=0}^{B} \mathbf{1}_i^{obj} [(\sqrt{w_i} - \sqrt{\hat{w}_i})^2 + \left(\sqrt{h_i} - \sqrt{\hat{h}_i}\right)^2] \quad (2.2)$$

$1_{ij}^{obj}$ дорівнює 1 якщо *j* граничне поле відповідальне за виявлення *i* об'єкту, інакше 0. $\lambda_{coord}$ – збільшує вагу втрати для координат граничного поля.

Помилка у 2 пікселі може мати різну вагу залежно від розміру поля. Щоб частково вирішити це, YOLO використовує квадратний корінь ширини та висоти граничного поля замість ширини та висоти. Крім того, щоб зробити більше уваги на точності граничного поля, втрата помножується на додатковий коефіцієнт $\lambda_{coord}$ (за замовчуванням: 5).

3) Втрата впевненості – розраховується для виявленого об'єкта до якого втрачена довіра, іншими словами вимірює об'єктивність обраного для поля класу.

$$\sum_{i=0}^{S^2} \sum_{j=0}^{B} \mathbf{1}_i^{obj} (C_i - \hat{C}_i)^2 \quad (2.3)$$

$C_i$ – оцінка достовірності граничного поля. $1_{ij}^{obj}$ як і у випадку з втратою локалізації дорівнює 1 якщо *j* граничне поле відповідальне за виявлення *i* об'єкту, інакше 0. Якщо довіра до об'єкта не була втрачена то втрата розраховується за іншою формулою.

$$\lambda_{noobj} \sum_{i=0}^{S^2} \sum_{j=0}^{B} \mathbf{1}_i^{obj} (C_i - \widehat{C_\iota})^2 \quad (2.4)$$

$1_{ij}^{noobj}$ – доповнення $1_{ij}^{obj}$. $C_i$ – оцінка достовірності граничного поля. $\lambda_{noobj}$ – коефіцієнт зниження втрати при розпізнаванні фону.

Більшість полів не містить жодних об'єктів. Це викликає проблему дисбалансу класів, тобто модель навчається виявляти фон частіше, ніж виявляти об'єкти. Щоб виправити це, втрата зменшується на коефіцієнт $\lambda_{noobj}$ (за замовчуванням: 0,5).

В результаті об'єднання всіх трьох частин отримаємо повну функцію втрат яка використовується у YOLO [14].

$$\lambda_{coord} \sum_{i=0}^{S^2} \sum_{j=0}^{B} \mathbf{1}_i^{obj} [(x_i - \widehat{x_\iota})^2 + (y_i - \widehat{y_\iota})^2] + \lambda_{coord} \sum_{i=0}^{S^2} \sum_{j=0}^{B} \mathbf{1}_i^{obj} \left[ \left(\sqrt{w_i} - \sqrt{\widehat{w_\iota}}\right)^2 + \left(\sqrt{h_i} - \sqrt{\widehat{h_\iota}}\right)^2 \right] + \sum_{i=0}^{S^2} \sum_{j=0}^{B} \mathbf{1}_i^{obj} \left(C_i - \widehat{C_\iota}\right)^2 + \lambda_{noobj} \sum_{i=0}^{S^2} \sum_{j=0}^{B} \mathbf{1}_i^{obj} (C_i - \widehat{C_\iota})^2 + \sum_{i=0}^{S^2} \mathbf{1}_i^{obj} \sum_{c \in classes} (p_i(c) - \widehat{p_\iota}(c))^2 \quad (2.5)$$

Перевагою використання YOLO є велика швидкість в порівнянні з іншими методами, ця мережа добре підходить для використання в реальному часі.

### 2.3 Нейронні мережі для сегментації зображень

У комп'ютерному зорі сегментація зображення - це процес розподілу цифрового зображення на кілька сегментів (наборів пікселів, також відомих як об’єкти зображення). Метою сегментації є спрощення та / або зміна подання зображення на щось більш змістовне та просте для аналізу. Сегментація зображення зазвичай використовується для пошуку об’єктів та меж (ліній, кривих тощо) у зображеннях. Точніше, сегментація зображення - це процес

присвоєння мітки кожному пікселю зображення таким чином, що пікселі з однаковою міткою мають певні характеристики.

Вже було розроблено велика кількість методів які вирішують проблему сегментації зображення, деякі з цих методів досягли великих результатів, інші більш підходять для специфічних задач та більш якісно вирішують проблему у конкретних предметних областях.

### 2.3.1 Повністю згорткова нейронна мережа для семантичної сегментації

Повністю згорткові мережі на вхід приймають зображення будь-якого розміру та генерують на вихід відповідні просторові розміри. У цій моделі класифікатори ILSVRC вводяться у повністю пов'язаній мережі та доповнюються для використанням піксельної втрати для щільного прогнозування. Навчання сегментації проводиться за допомогою тонкої настройки. Тонка настройка виконується шляхом зворотного розповсюдження по всій мережі.

В цій мережі використовується поєднання ієрархії функції, це означає що функції поєднуються поперек шарів щоб визначити нелінійне представлення у локальному та глобальному масштабах.

Кожен вихідний шар у згортці представляє собою тривимірний масив розміром $h \times w \times d$, де застосовуються просторові розміри, і відображається розмір функції або каналу. Перший шар - зображення, розміром пікселя $h \times w$, і $d$ каналами. Місцеположення в більш високих шарах відповідають місцям зображення, до якого вони пов'язані шляхом, які називаються рецептивними полями.

Функція втрат, визначається завданням. Якщо функція втрати - це сума над просторовими розмірами кінцевого шару його параметр градієнта буде сумою над градієнтом параметра для кожної його просторової складової (2.6).

$$l\,(x;\theta) = \sum_{ij} l`(x_{ij};\theta) \tag{2.6}$$

Таким чином, стохастичний градієнт, що спускається на $l$ обчислений на рівному зображенні, буде таким самим, як і стохастичний градієнт спуску на $l`$, приймаючи всі кінцеві поля прийому в якості міні-батчу.

Типові мережі розпізнавання приймають на вхід данні фіксованого розміру та повертають на вихід непросторові данні. Повністю з'єднані шари цих мереж мають фіксовані розміри і відкидають просторові координати. Однак повністю пов'язані шари також можна розглядати як згортки з ядрами, які охоплюють всі їхні області введення. Це відносить ці

мережі повністю згорткових мереж, які беруть на увазі аналіз і складають просторові вихідні карти.

Данна мережа використовує класифікатори ILSVRC (ImageNet Large Scale Visual Recognition Challenge) у FCN та доповнюємо їх для щільного прогнозування з підсиленням у мережі. Також у мережі додані пропуски між шарами, щоб передавати грубу, семантичну та локальну інформацію про зовнішній вигляд об'єктів. Ця архітектура пропуску навчилася в кінці кінців удосконалювати семантику та просторову точність результату. У FCN не нормалізуються втрати, так щоб кожен піксель мав однакову вагу, незалежно від розмірів зображення. Таким чином, під час роботи використовуються невеликі показники навчання, оскільки просторові втрати підсумовуються на всіх пікселях.

Ми розглядаємо два режими розміру партії. По-перше, градієнти накопичуються понад 20 зображень. Накопичення зменшує необхідну пам'ять і враховує різні розміри кожного входу шляхом перестановки мережі. Ми вибрали цей розмір партії емпіричним шляхом, щоб привести до розумного зближення. Навчання таким чином схоже на стандартне класифікаційне навчання: кожна міні-партія містить кілька зображень і має різноманітний розподіл міток класу. Мережі, порівняні в Таблиці 1, оптимізовані таким чином [22].

### 2.3.2 Fully Convolutional DenseNets

Головна ідея Fully Convolutional DenseNets (FCDN) ґрунтується на спостереженні, що якщо кожен шар буде прямо пов'язаний з будь-яким іншим шаром в режимі передачі даних між шарами, мережа буде більш точною і простішою в навчанні.

FCDN також використовує архітектуру FCN (Повністю згорткова мережа). Зазвичай такі мережі використовують швидкі з'єднання які допомагають прискорити розбір даних та відновити детально детальну інформацію шляхом повторного використання карт функцій. Мета FCDN полягає у подальшому використанні повторних функцій шляхом розширення більш досконалої архітектури, уникаючи при цьому експлуатування функцій на шляху прокладки мережі.

Так як дане рішення побудовано на основі DensNet слід розглянути його біль детально. Нехай $x_l$ – це результат який ми отримуємо на виході шару $l^{th}$. У стандартному CNN $x_l$ обчислюється шляхом застосування нелінійного перетворення $H_l$ на вихід попереднього шару $x_l$ *-1*.

$$x_l = H_l(x_{l-1}) \tag{2.7}$$

У формулі (1.9) *H* визначають як згортку з наступною нелінійністю випрямляча (ReLU).

Щоб полегшити навчання дуже глибоких мереж, ResNets вводить залишковий блок, який підсумовує ідентифікаційне відображення вхідних даних до виходу шару. Повторне отримання результату стає

$$x_l = H_l(x_{l-1}) + x_{l-1} \tag{2.8}$$

Це дозволяє повторно використовувати функції, а також дозволяє градієнту надходити безпосередньо до попередніх шарів. Використавши цю ідею у DenseNets розробили більш вдосконалений зразок підключення, який ітеративно поєднує в собі всі виходи функцій зворотнім способом. Таким чином, вихідний рівень шару визначається наступним чином (2.9).

$$x_l = H_l([x_{l-1}, x_{l-2}, \dots, x_0) \tag{2.9}$$

Позначка «...» являє собою операцію конкатенації. Така схема підключення сильно заохочує повторне використання функцій і змушує всі шари в архітектурі отримувати прямий сигнал. Можна помітити, що ця схема передбачує лінійний приріст кількості ознак. Цей приріст компенсується зменшенням просторової роздільної здатності кожної особливості карти після операції об'єднання. Останній шар шляху зменшення тиску називається вузьким місцем.

Для відновлення вхідної просторової роздільної здатності у звичайній FCN вводять шлях перебігу збору, що складається з згортки, операцій з відбору (транспонованих згортків або відключення операцій) та пропускання з'єднань. У FC-DenseNets заміняють операцію згортання щільним блоком, а операцію по підвищенню потужності називають відстороненням. Модулі трансляції складаються з транспонованої згортки. Після цього переглянуті карти властивостей приєднуються до тих, що надходять із пропускного з'єднання для формування входу нового щільного блоку. Оскільки шлях до розгортання збільшує функцію карт просторової роздільної здатності, лінійне зростання кількості функцій буде занадто пам'ятним вимогливі, особливо до функцій повної роздільної здатності в попередньому шарі [23].

### 2.3.3 Gated-SCNN: Gated Shape CNNs for Semantic Segmentation

Gated Shape CNNs for Semantic Segmentation **(**Gated-SCNN) пропонує нову двопотокову архітектуру CNN для семантичної сегментації, яка явно проводить форму інформації як окрему гілку обробки, яка обробляє інформацію паралельно класичному потоку. Ключ до цієї архітектури - це новий тип воріт, що з'єднують проміжні шари двох потоків. В класичному

потоці використовуються активації вищого рівня, що тонують активації нижнього рівня у потоці форми, ефективно видаляючи шум і допомагаючи потоку форми лише зосереджуватися на обробці відповідної інформації, що стосується меж. Це дозволяє використовувати неглибоку архітектуру для потоку фігур, який працює на рівні зображення. Експерименти показують, що це призводить до високоефективної архітектури, яка дає чіткіші прогнози навколо меж об'єкта і значно підвищує продуктивність на менших та менших об'єктах.

Як вже казалося вище дане рішення включає в себе два потоки обробки даних. Перший потік можна умовно назвати класичним потоком, в ньому реалізована стандартна сегментація за допомогою CNN. Для цього потоку використовується ResNet подібна архітектура мережі. Другий потік називається «потік форм», він обробляє інформацію яка включає в себе опис форми у вигляді семантичних меж. Потік форми обробляє лише інформацію, пов'язану з межами, завдяки спеціально розробленому шару згортання (GCL – Gated Convolutional Layer) та локальному нагляду. Цей потік, приймає градієнти зображення, а також вихід першого шару звичайного потоку як вхідні данні і видає смислові межі на виході. Мережева архітектура складається з декількох залишкових блоків, переплетених із закритими шарами згортки (GCL). GCL обробляє лише інформацію, що стосується кордонів. Для контролю за потоком форми використовується контрольовані межі вихідних поперечних ентропійних втрат. Потім риси семантичного регіону з’єднуються, у модулі злиття, з класичним потоком та межовими особливостями з потоку форм, щоб отримати вишуканий результат сегментації, особливо навколо меж. Модуль злиття приймає як вхід щільне представлення функції, що надходить від звичайної гілки, і зливає його з виведенням границь, виводиться за допомогою гілки форми таким чином, щоб зберігалася багато масштабна контекстна інформація. Він поєднує особливості регіону з прикордонними особливостями та дає на вихід результати сегментації. Більш формально для сегментаційного прогнозування K семантичних класів він виводить категоричний розподіл, що представляє ймовірність того, що пікселі належать кожному класу K. Це дозволяє зберегти багато масштабну контекстуальну інформацію та виявляється важливою складовою у сучасних семантичних мережах сегментації.

Оскільки завдання оцінки семантичної сегментації та семантичних меж тісно пов'язані між собою, був розроблений новий шар GCL, який полегшує потік інформації з регіонального потоку в потік форми. GCL є основним компонентом представленої архітектури і допомагає потоку фігур обробляти лише відповідну інформацію, фільтруючи решту. Зауважте, що потік фігури не містить функцій регіонального потоку. Швидше, він використовує GCL для деактивації власних заходів, які не вважаються релевантними інформацією вищого рівня, що міститься у регулярному потоці. Можна розрізнити це як співпрацю між двома потоками, де

більш потужний, що сформував семантичне розуміння сцени вищого рівня, допомагає іншому потоку зосередитися лише на відповідних частинах з самого початку. Це дає можливість потоку фігур прийняти ефективну дрібну архітектуру, яка обробляє зображення з дуже високою роздільною здатністю [24].

### 2.3.4 U-Net

Ця модель була розроблена з урахуванням того, що даних для тренувань може бути недостатньо. Тому U-Net використовує розширення даних, застосовуючи еластичні деформації для наявних даних. Архітектура мережі складається з узгодженого шляху зліва та розширюваного шляху праворуч. У цій моделі навчання проводиться за допомогою вхідних зображень, карти їх сегментації та реалізації стохастичного градієнтного спуску Caffe. Розширення даних використовується для навчання мережі необхідної стійкості та інваріантності, коли використовується дуже мало навчальних даних.

Мережа U-Net складається з контрактної доріжки та експансивного шляху. Шлях контрактування проходить за типовою архітектурою згорткової мережі. Він складається з повторного застосування двох згортків 3x3 (нерозкладені згортки), за якими слідує випрямлена лінійна одиниця (ReLU), і операції об'єднання 2x2 максимуму з кроком 2 для зменшення кемпінгу. На кожному кроці зниження потужності подвоюється кількість каналів функцій. Кожен крок на експансивному шляху складається з розгортання карти з подальшим згортанням 2x2 (“згортання вгору”), що вдвічі зменшує кількість функціональних каналів, конкатенацію з відповідною обрізаною картою особливості з контрактного шляху та два 3x3 згортки, кожен дотримуючись ReLU. Обрізка необхідна через втрату нескінченної згортки прикордонних пікселів. На завершальному шарі використовується згортка 1x1 для зіставлення кожної 64-компонентної векторної ознаки на потрібну кількість класів. Всього в мережі є 23 згорткових шару.

Вхідні зображення та їх відповідні маски сегментації використовуються для відстеження мережею із застосуванням стохастичного градієнтного спуску Caffe. Зважаючи на нерозкладені згортки, вихідне зображення менше, ніж на вході. Щоб мінімізувати накладні витрати та максимально використовувати пам'ять графічного процесора, перевага надається великим розмірам партії і, таким чином, зменшує пакет до одного зображення. Відповідно до цього використовується високий коефіцієнт імпульсу (0,99), такий, що велика кількість попередньо бачених навчальних зразків визначає актуалізацію на поточному етапі оптимізації.

Енергетична функція обчислюється за допомогою алгоритму soft-max по мапі кінцевої характеристики в поєднанні з функцією втрати поперечної ентропії.

$$p_k(x) = \exp\big(a_k(x)\big) / (\sum_{k`=1}^{K} \exp(a_{k`}(x))), \quad (2.10)$$

де $a_к(x)$ позначає активацію в функції каналу положення пікселя $x$. $K$ - кількість класів, а $p_k(x)$ - приблизна максимальна функція. Потім поперечна ентропія штрафує у кожному положенні відхилення використовуючи (2.11).

$$E = \sum_{x \in \Omega} w(x)\log(p_{l(x)}(x)), \quad (2.11)$$

де $l : \Omega \rightarrow \{1, \ldots, K\}$ істина мітка для кожного пікселя та w карта ваги, яку представили, щоб надати деяким пікселям більше значення у навчанні.

Карта ваги обчислюється за формулою (2.12).

$$w(x) = w_c(x) + w_0 * \exp(-\frac{(d_1(x)+ d_2(x))^2}{2\sigma^2}), \quad (2.12)$$

де $w_c$ - карта ваги для балансування частот класу. $d_1$ - позначає відстань до межі найближчої комірки. $d_2$ – відстань до другої найближчої комірки [25].

### 2.4 Висновок до розділу 2

В розділі представлена загальна структура запропонованого методу прогнозування дій об'єктів у медичних відео зображень на підставі моделей розпізнавання. Підготовка і розпізнавання відеозображень містить наступні етапи:

1. Навчання моделі для розпізнавання об'єктів, що передбачає визначення просторової активності об’єктів з використанням анотованих координат обмежуючого прямокутника, відповідних кожному класу, сегментацію з використанням даних масок анотованих об'єктів, цілочисельне значення для кожного зображення та узагальнення – об’єднання узагальненого уявлення об'єктів.

2. Використання навченої моделі для прогнозування дій об'єктів нових відеозображень в режимі реального часу.

Розглянуті та виведені основні математичні моделі нейронних мереж, що дозволяють отримувати найбільш якісний результат розпізнавання та прогнозування під час обробки великої кількості даних відеозображень.

# РОЗДІЛ 3 ПРАКТИЧНА РЕАЛІЗАЦІЯ МЕТОДІВ, МОДЕЛЕЙ РОЗПІЗНАВАННЯ ТА ПРОГНОЗУВАННЯ ДІЙ ОБ'ЄКТА У ВІДЕОПОТОЦІ

Практична реалізація запропонованих методів, моделей представлена шляхом проведення експериментів з розпізнавання та прогнозування дій об'єкта у відеопотоці з відео данних, отриманих в процесі поведення ендоскопічних операцій та дослідження. Ендоскопія - широко застосовувана клінічна процедура для раннього виявлення численних онкологічних захворювань (наприклад, новоутворень носоглотки, аденокарциноми стравоходу, раку шлунка, колоректального каналу, раку сечового міхура тощо), терапевтичних процедур та малоінвазивної хірургії (наприклад, лапароскопії). Під час цієї процедури застосовується ендоскоп; довга тонка, жорстка або гнучка трубка, що має джерело світла і камеру на кінці, що дозволяє візуалізувати всередині уражених органів на екрані, що представлений на рисунку 3.1.

Рисунок 3.1 – Ендоскоп для проведення ендоскопічних досліджень та операцій

Основним недоліком цих відеокадрів є те, що вони сильно зіпсовані безліччю артефактів (наприклад, насиченість пікселів, розмиття руху, розфокусування, дзеркальні відбиття, бульбашки, рідина, сміття тощо). Ці артефакти не лише становлять труднощі у візуалізації основної тканини під час діагностики, але й впливають на будь-які методи постаналізу, необхідні для подальшого спостереження (наприклад, відеомозаїка, зроблена для спостережень та архівних цілей, та пошук відеокадрів, необхідний для звітування).

## 3.1 Апаратне та програмне забезпечення

Апаратне та програмне забезпечення, що використовувалось в процесі практичної реалізації методів та моделей прогнозування дій об’єктів у відео, та його параметри представлені в таблиці 3.1.

Таблиця 3.1 – Апаратне та програмне забезпечення

| Найменування | Параметри |
|---|---|
| CPU | Intel Core i7 9700K |
| RAM | 32 GB 3000 MHz DDR4 |
| GPU | GeForce RTX 2080 Ti \| NVIDIA |
| Накопичувач | Kingston SSDNow A400 240GB |
| Операційна система | Microsoft Windows 10 |
| Python | 3.7.2 |

## 3.2 Розпізнавання медичних зображень

### 3.2.1 Опис досліджуваних даних

Для проведення дослідження використано відео дані, зібрані під час ендоскопії. На зображеннях присутні внутрішні органи людини, хірургічні інструменти, кров та різні артефакти які слід класифікувати.

Точне виявлення артефактів є основним завданням у широкому діапазоні ендоскопічних застосувань, що стосуються безлічі різних областей хвороби. Важливість точного виявлення цих артефактів має важливе значення для якісної реставрації ендоскопічного каркасу і має вирішальне значення для реалізації надійних засобів комп'ютерної ендоскопії для поліпшення діагностики та догляду за пацієнтами. Існуючі ендоскопічні робочі процеси виявляють лише один клас артефактів, який недостатній для отримання якісної реставрації кадру. Загалом, один і той же кадр відео може бути зіпсований множиною артефактів, наприклад розмиття руху, дзеркальні відображення та низький контраст можуть бути присутніми в одному кадрі. Крім того, не всі типи артефактів забруднюють раму однаково. Отже, якщо безліч артефактів, наявних у кадрі, не відомі з їх точним просторовим розташуванням, якість відновлення кадру не може бути гарантована. Ще однією перевагою такого виявлення є те, що в оцінці якості кадру

можна керуватися, щоб мінімізувати кількість кадрів, які відкидаються під час автоматичного відео аналізу.

Метою розпізнавання відео об'єктів є завдання визначення локалізації обмежувальних коробок, прогнозування міток класів та піксельна сегментація 8 різних артефактів для заданих кадрів та відеокліпів клінічної ендоскопії.

Класи артефактів мають наступні мітки: дзеркальне відображення, бульбашки, насиченість, контрастність, кров, інструмент, розмиття та зображення артефактів.

Для навчання мережі було використано 2200 зображень. Нижче приведені приклади деяких з них.

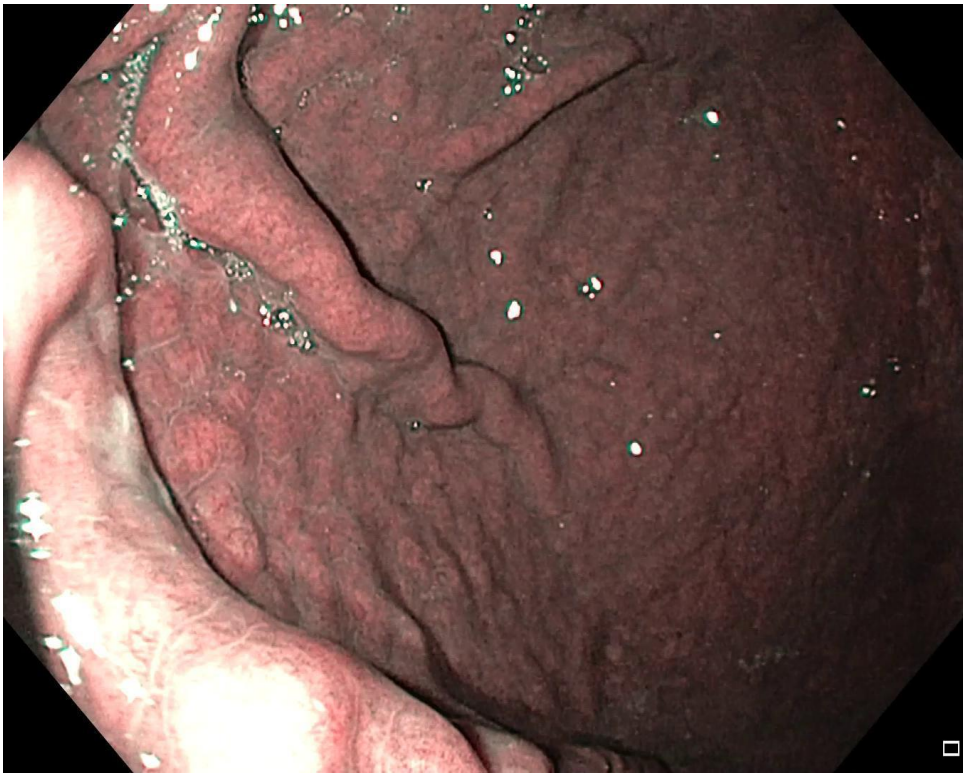

Рисунок 3.2 - Приклад відео кадру зображення з наявністю віддзеркалювань

На рисунку 3.2 добре видно велику кількість маленьких віддзеркалювань, а на рисунку 3.3 окрім декількох віддзеркалювань, ще є кров, яка ускладнює поцес діагностики, скриваючі за собою можливі діагностичні елементи.

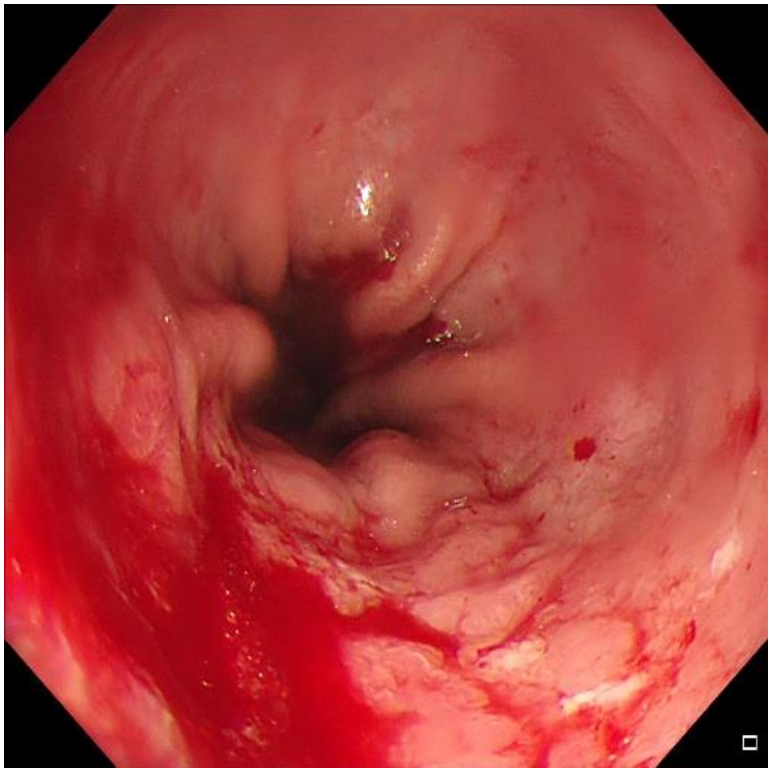

Рисунок 3.3 - Приклад відео кадру зображення з кров'ю та віддзеркалюванням

### 3.1.2 Параметри конфігурації мережі

Для вирішення завдання розпізнавання зображень використано нейронну мережу YOLOv3. Головною перевагою такого вибору є швидкість роботи яку дає YOLO. Найпростіший спосіб її використання це завантаження додатку з офіційного репозиторію на GitHub [https://github.com/AlexeyAB/darknet].

Після того як усі необхідні файли були завантажені та установлені можна перейти до наступного кроку. Перш за все необхідно створити файл yolo.cfg, у цьому файлі зберігається інформація стосовно конфігурації нейронної мережі. Стандартні конфігурації можна також завантажити з репозиторію на GitHub. Слід визначитися з правильними конфігураціями, щоб отримати найбільш точні результати. Файл конфігурації передбачає налаштування наступних параметрів.

Параметри мережі:

- *Width, height*– довжина та висота тензорів зображень. Усі зображення будуть автоматично приведені до вказаних розмірів.
- *Classes* – кількість класів для прогнозування.
- *Batch* – відповідає кількість зображень та міток які використовуються в прямому проході для обчислення градієнта та оновлення ваг за допомогою зворотного розповсюдження.

- *Max_batch* – встановлює максимальний розмір партії. Цей параметр розраховується за формулою

*MAX_BATCH = КІЛЬКІСТЬ КЛАССІВ * 2000*

- *Subdivisions* – кількість підрозділів на які розділяється *batch.*
- *Decay* – параметр для зменшення ваг, щоб уникнути великих значень.
- *Channels* - позначає розмір каналу вхідного зображення.
- *Momentum* - є навчальним параметром який визначає стійкість градієнту.
- *Adadm* – визначає чи слід використовувати оптимізатор Adam.
- *burn_in* - для перших *x* партій повільно збільшуйте ступінь навчання до остаточного значення. Використовується щоб визначити рівень навчання, відстежуючи, до якого значення зменшуються збитки (до того, як вони почнуть розходитися).
- *Policy* – використовується, щоб регулювати рівень навчання (*Learning rate)*
- *Steps* – регулює рівень навчання після *x* батчу.
- *Scales* – перечислюється коефіцієнт на який слід помножити рівень навчання після кожних 500 епох.
- *Angle* – збільшує зображення поворотом до вказаного кута (у градусах).

Параметри окремих шарів:

- *Filters* – кількість згорткових ядер у шарі. Підраховується за формулою

*Filters = (Кількість класів +5)*3*

- *Activation* – функція активації яка використовється у данному шарі
- *Stopbackword* – бульовий параметр, який використовується, щоб тренувати лише шари позаду, наприклад, при використанні перевірених ваг.
- *Random* – використовується для збільшення даних, змінює розміри зображень на різні розміри кожні кілька партій.

Окрім наведених вище параметрів існують і інші, але вони більш ситуативні, а не обов’язкові. Нижче представленні конфігурації мережі які використовувалися для проведення експерименту.

*batch=64*

*subdivisions=64*

*width=512*

*height=512*

*channels=3*

*momentum=0.9*

*decay=0.0005*

*angle=0*

*saturation = 1.5*

*exposure = 1.5*

*hue=.1*

*learning_rate=0.0005*

*burn_in=2000*

*max_batches = 160200*

*policy=steps*

*steps=40000,45000*

*scales=.1,.1*

Після визначення параметрів конфігурації мережі слід зробити окремий текстовий файл в якому будуть перераховані усі класи для класифікації. Класи які використовувалися у експерименті представленні у таблиці 3.2.

Таблиця 3.2 – Класи, використані для проведення екперименту

| Назва класу | Переклад |
|---|---|
| Saturation | Насичення |
| Artifact | Артефакт |
| Blur | Розмиття |
| Bubbles | Бульбашки |
| Instrument | Інструмент |
| Blood | Кров |
| Contrast | Контрастність кольорів |
| Specularity | Віддзеркалення |

Наступним кроком необхідно додати усі зображення до директорії проекту у папку *obj*. Також до цієї папки слід додати текстові файли з координатами об’єктів класів для кожного зображення.

### 3.1.3 Оцінка отриманих результатів розпізнавання

Для оцінки якості моделі використовується оцінка mAP. У статистиці Байєса максимальна оцінка задньої ймовірності (mAP) - це оцінка невідомої величини, яка дорівнює режиму заданого розподілу. mAP може бути використаний для отримання точкової оцінки непоміченої кількості на основі емпіричних даних. Він тісно пов'язаний з методом оцінки максимальної вірогідності (ML), але використовує розширену оптимізаційну мету, яка включає попередній розподіл (який кількісно визначає додаткову інформацію, наявну за попередніми знаннями про пов’язану подію) над кількістю, яку хочеться оцінити. Оцінка mAP може, таким чином, розглядатися як регуляризація оцінки ML.

Середня середня точність розраховується наступним чином.

$$MAP = \frac{\sum_{q=1}^{Q} AveP(q)}{Q},$$

де $Q$ - кількість запитів в наборі, а $AveP\ (q)$ - середня точність (Average Precision) для даного запиту $q$.

Середня оцінка mAP для мережі на п’ятнадцяти тисячах ітерацій дорівнювала 0.298547 або 29,8%. Однак, якщо поглянути на середні оцінки для кожного класу окремо (таблиця 3.3) то можна помітити, що деякі з класів модель навчилася визначати значно точніше ніж інші.

Таблиця 3.3 - Результати оцінки mAP для кожного класу окремо

| Назва класу | mAP |
|---|---|
| Saturation | 0,054264 |
| Artifact | 0,314134 |
| Blur | 0,078622 |
| Bubbles | 0,194705 |
| Instrument | 0,210045 |
| Blood | 0,185895 |
| Contrast | 0,088984 |
| Specularity | 0,380056 |

На рисунках 3.4 та 3.5 представлені результати роботи мережі на тестових зображеннях.

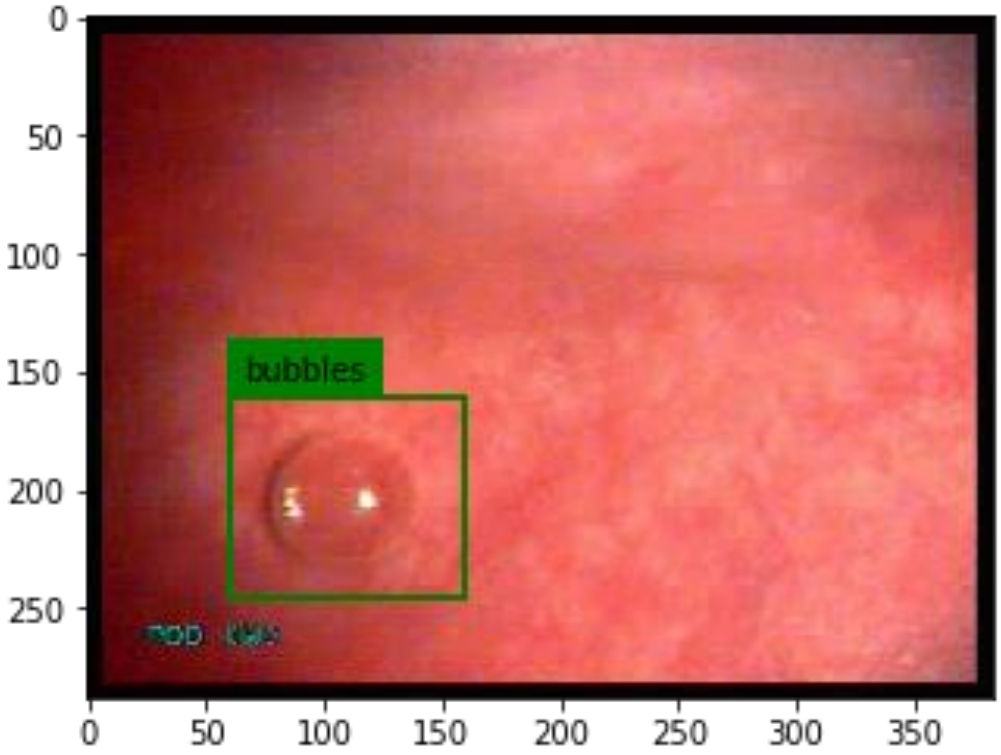


Рисунок 3.4 – Розпізнавання класу bubbles

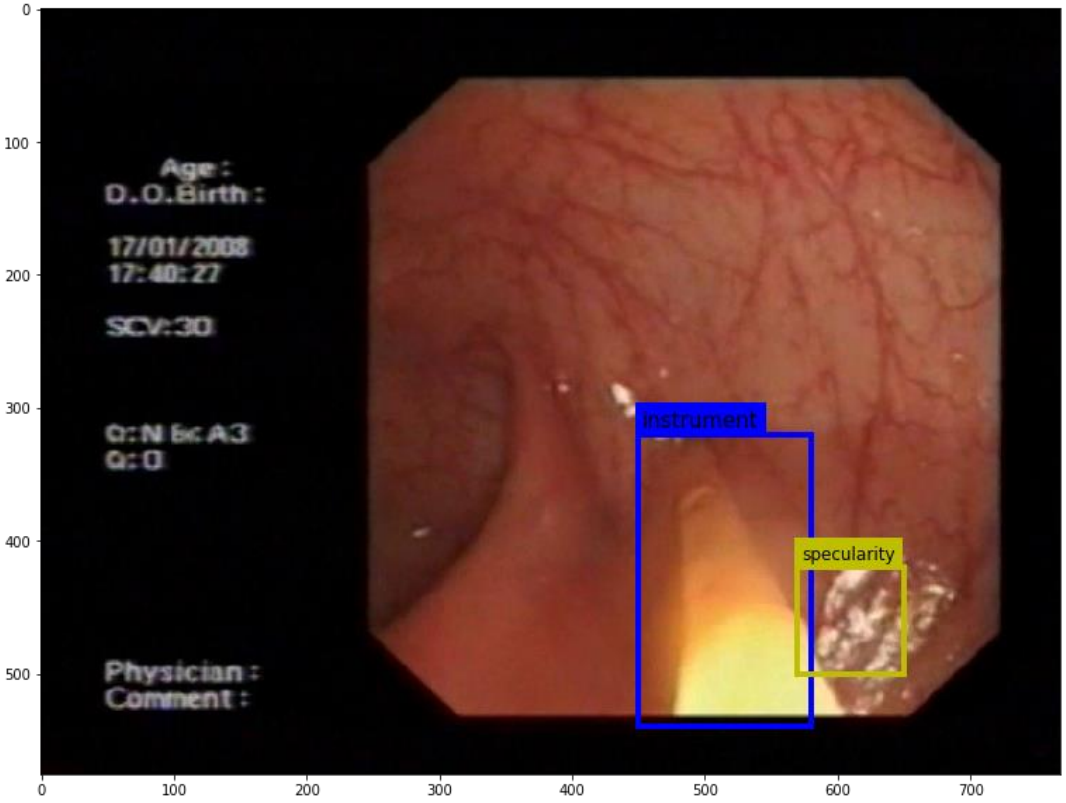


Рисунок 3.5 – Розпізнавання класів instrument та specularity

На рисунку 3.4 модель явно змогла розпізнати клас bubbles, але якщо придивитися то зверху справа на зображенні є невелике розмиття, яке мережа не змогла розпізнати. На рисунку 3.5 є одразу два об’єкти які побачила мережа – це інструмент та віддзеркалювання. Однак модель не змогла розпізнати менші віддзеркалювання, що теж присутні на зображенні.

## 3.2 Сегментація медичних зображень

### 3.2.1. Опис досліджувани даних

Для проведення експерименту пов'язаного з сегментацією були використанні зображення  добуті під час ендоскопії. Ці зображення мають ряд артефактів які створюють труднощі під час візуалізації. Метою експерименту є виявлення таких артефактів для подальшої обробки.

Навчальний набір даних включав в себе 948 зображень відеокадрів. Половина з цих зображень були кадрами отриманими під час ендоскопії, а друга половина це маски. Ці маски використовуються для виділення певного класу на зображенні. Нижче на рисунках 3.6 та 3.7 наведені приклади зображення відеокадру та маски, яка для нього використовується.

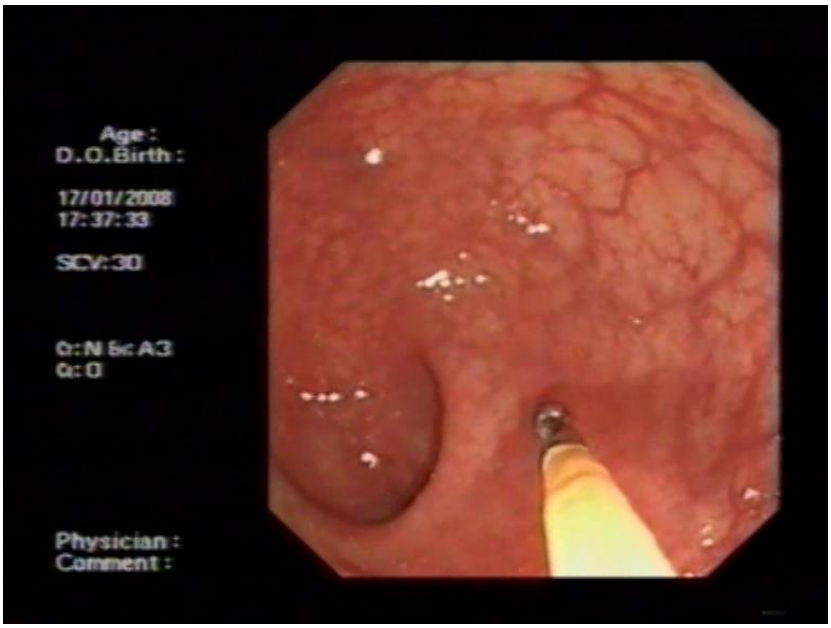


Рисунок 3.6 - Приклад відеокадру відео ендоскопічного дослідження

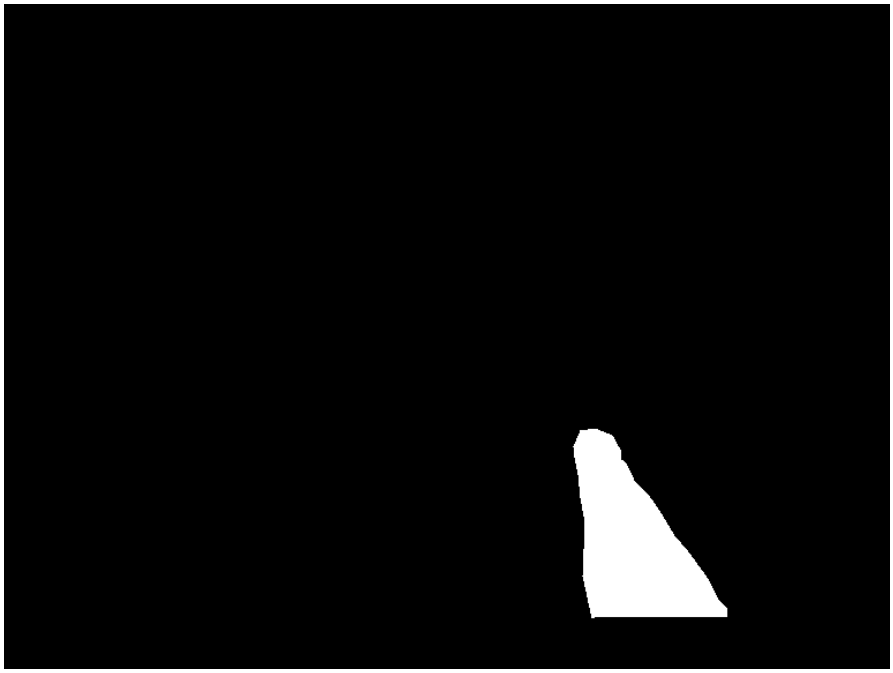

Рисунок 3.7 - Маска об’єкту, представленого на рисунку 3.6

Рисунок. 3.5 є відео кадром операції, на якому присутній хірургічний інструмент. Можна побачити, що на рисунку 3.7 зображені контури інструмента з рисунка 3.6. Для кожного кадру є своя маска.

На кожному зображенні є об'єкт який слід визначити та виділити. Всього на зображеннях є п'ять типів об'єктів (таблиця 3.4).

Таблиця 3.4 - Опис класів об’єктів

| Назва об'єкту | Пояснювання |
|---|---|
| Instrument | Хіургічний інструмент |
| Artefact | Зіпсований регіон у зображенні |
| Saturation | Насичені пікселі |
| Bubbles | Бульбашки |
| Blood | Кров |

### 3.2.2 Параметри конфігурації мережі

Для вирішення поставленої задачі було використано нейронну мережу U-Net. Її перевага в тому, що вона швидко навчається та може навчатися на меншій кількості прикладів. Реалізація була виконана на мові програмування Python, з використанням спеціалізованих бібліотек для обробки зображень та глибокого навчання глибокого навчання.

Для реалізації U-Net була використана бібліотека Tensorflow. TensorFlow — відкрита програмна бібліотека для машинного навчання розроблена компанією Google для задоволення її потреб у системах, здатних будувати та тренувати нейронні мережі для виявлення та розшифровування образів та кореляцій. Нижче представлений програмний код реалізації нейронної мережі.

```
def get_unet(input_img, n_filters=16, dropout=0.5, batchnorm=True):
    # шлях скорочення
    c1 = conv2d_block(input_img, n_filters=n_filters*1, kernel_size=3, batchno
rm=batchnorm)
    p1 = MaxPooling2D((2, 2)) (c1)
    p1 = Dropout(dropout*0.5)(p1)

    c2 = conv2d_block(p1, n_filters=n_filters*2, kernel_size=3, batchnorm=batc
hnorm)
    p2 = MaxPooling2D((2, 2)) (c2)
    p2 = Dropout(dropout)(p2)

    c3 = conv2d_block(p2, n_filters=n_filters*4, kernel_size=3, batchnorm=batc
hnorm)
    p3 = MaxPooling2D((2, 2)) (c3)
    p3 = Dropout(dropout)(p3)

    c4 = conv2d_block(p3, n_filters=n_filters*8, kernel_size=3, batchnorm=batc
hnorm)
    p4 = MaxPooling2D(pool_size=(2, 2)) (c4)
    p4 = Dropout(dropout)(p4)

    c5 = conv2d_block(p4, n_filters=n_filters*16, kernel_size=3, batchnorm=bat
chnorm)

    # шлях розширення
```

```
    u6 = Conv2DTranspose(n_filters*8, (3, 3), strides=(2, 2), padding='same')
(c5)
    u6 = concatenate([u6, c4])
    u6 = Dropout(dropout)(u6)
    c6 = conv2d_block(u6, n_filters=n_filters*8, kernel_size=3, batchnorm=batc
hnorm)

    u7 = Conv2DTranspose(n_filters*4, (3, 3), strides=(2, 2), padding='same')
(c6)
    u7 = concatenate([u7, c3])
    u7 = Dropout(dropout)(u7)
    c7 = conv2d_block(u7, n_filters=n_filters*4, kernel_size=3, batchnorm=batc
hnorm)

    u8 = Conv2DTranspose(n_filters*2, (3, 3), strides=(2, 2), padding='same')
(c7)
    u8 = concatenate([u8, c2])
    u8 = Dropout(dropout)(u8)
    c8 = conv2d_block(u8, n_filters=n_filters*2, kernel_size=3, batchnorm=batc
hnorm)

    u9 = Conv2DTranspose(n_filters*1, (3, 3), strides=(2, 2), padding='same')
(c8)
    u9 = concatenate([u9, c1], axis=3)
    u9 = Dropout(dropout)(u9)
    c9 = conv2d_block(u9, n_filters=n_filters*1, kernel_size=3, batchnorm=batc
hnorm)

    outputs = Conv2D(1, (1, 1), activation='sigmoid') (c9)
    model = Model(inputs=[input_img], outputs=[outputs])
    return model
```

- Шари *conv2d_block* означають, що застосовуються два послідовних шару згортки.
- c1, c2,…. c9 - вихідні тензори згорткових шарів
- p1, p2, p3 і p4 - вихідні тензори шарів Max Pooling
- u6, u7, u8 і u9 - вихідні тензори верхніх вибіркових (транспонованих згорткових) шарів.

Архітектуру нейронної мережі U-net можна розділити на дві частини.

Перша частина мережі (#шлях скорочення) - це шлях скорочення який називається Енкодер, де ми застосовуємо регулярні згортки та шари Max Pooling. Max Pooling - вбирає максимум корисної інформації з невеликих блоків попереднього шару. На вихід ці шари повертають інформацію чи був присутній корисний сигнал функції в попередньому шарі, але не уточнює де саме. У Енкодері розмір зображення поступово зменшується, в той час як глибина поступово збільшується. На цьому етапі мережа навчається розпізнавати об’єкти, однак вона не враховує їх місце знаходження (координати).

Друга частина це шлях розширення - декодер, де застосовуються транспоновані згортки разом з регулярними згортками. У декодері навпаки розмір зображення поступово збільшується, а глибина поступово зменшується. Також декодер в свою чергу відповідає за вилучення інформації про місце знаходження об’єкту на зображенні. Для отримання більш точної інформації про розташування об’єкту, на кожному кроці декодера використовуються пропускні з'єднання. Вони реалізуються шляхом об'єднання виводу транспонованих шарів згортки з картами функцій отриманих з енкодеру:

$u6 = u6 + c4$

$u7 = u7 + c3$

$u8 = u8 + c2$

$u9 = u9 + c1$

Після кожної конкатенації знову застосовується дві послідовних регулярних згортки, щоб модель могла навчитися повертати більш точні данні на вихід.

При ініціалізації моделі їй слід передати чотири параметри:

1) input_img – це кортеж який містить три значення в яких надається інформація про розміри тензору до якого були приведені зображення (Висота зображення, Ширина зображення, Кількість каналів кольору).

2) n_filters – кількість використовуваних фільтрів.

3) dropout – коефіцієнт який буде враховуватися при підрахунку функції зворотнього розповсюдження помилки.

4) barchnorm – булевий параметр який визначає чи слід використовувати нормалізацію партії, чи ні.

### 3.2.3 Попередня обробка даних

Перед навчанням мережі проведена попередня обробка зображень, а саме: структурувати розділити зображення та їх маски, поділити вибірку на навчальну та тестову, привести усі зображення до єдиної форми тензора та інше.

Перш за все необхідно для того, щоб програмно автоматизувати обробку зображень їх слід структурно розділити по каталогам, щоб було легше з ними працювати. Нижче представлена структура зберігання даних (діаграма 3.1).

```
      data
|       |       |------train
|       |       |       |-----|Folder_1
|       |       |       |     |------|images
|       |       |       |     |------|------|Folder_1.jpg
|       |       |       |     |------|masks
|       |       |       |     |------|------|Folder_1_mask_1.tif
|       |       |       |-----|Folder_n
|       |       |       |     |------|images
|       |       |       |     |------|------|Folder_n.jpg
|       |       |       |     |------|masks
|       |       |       |     |------|------|Folder_n_mask_1.tif
|       |       |------test
|       |       |       |-----|Folder_1
|       |       |       |     |------|images
|       |       |       |     |------|------|Folder_1.jpg
|       |       |       |     |------|masks
|       |       |       |     |------|------|Folder_1_mask_1.tif
|       |       |       |-----|Folder_n
|       |       |       |     |------|images
|       |       |       |     |------|------|Folder_n.jpg
|       |       |       |     |------|masks
|       |       |       |     |------|------|Folder_n_mask_1.tif
```

Діаграма 3.1 - Структура зберігання даних

Як видно з діаграми усі зображення зберігаються в ієрархічній структурі. Окремо зберігається тренувальний та тестовий набори даних. Усі зображення зберігаються в окремих пронумерованих директоріях, окремо від їх масок. Така структура дозволяє легко обходити усі зображення та маски.

Після цього необхідно усі зображення та привести їх до єдиної тензорної форми. Для обробки зображень використовувалась бібліотека Skimage для мови програмування Python, ця бібліотека є продуктом з відкритим програмним кодом та розробляється спільнотою волонтерів. Skimage - це сукупність алгоритмів для обробки зображень. Нижче переведена реалізація скрипта для обробки зображень.

```
IMG_WIDTH = 512
IMG_HEIGHT = 512
IMG_CHANNELS = 3
```

Параметри `IMG_WIDTH,IMG_HEIGHT,IMG_CHANNELS` є константами які задають висоту, ширину та кількість каналів кольору для зображення. По суті це є розмірність тензору до якого будуть приведені усі зображення. У приведеному прикладі ми отримаємо тензор розміром 512 x 512 x 3 для кожного зображення.

```
TRAIN_PATH = '/content/drive/My Drive/data/train'
TEST_PATH = '/content/drive/My Drive/data/test'
```

`TRAIN_PATH` та `TEST_PATH` також є константами які зберігають інформацію пор місцезнаходження зображень на жорсткому накопичувачі.

```
warnings.filterwarnings('ignore', category=UserWarning, module='skimage')
seed = 42
random.seed = seed
np.random.seed = seed

print("Imported all the dependencies")


train_ids = next(os.walk(TRAIN_PATH))[1]
test_ids = next(os.walk(TEST_PATH))[1]
```

У змінні `train_ids, test_ids` записують індекси усіх директорій для основної та тестової вибірок.

```
X_train = np.zeros((len(train_ids), IMG_HEIGHT, IMG_WIDTH, IMG_CHANNELS),
dtype=np.uint8)
```

```
Y_train = np.zeros((len(train_ids), IMG_HEIGHT, IMG_WIDTH, 1), dtype=np.bool)
```

Після цього створюються пусті масиви тензорів в які будуть зберігатися оброблені зображення. Насправді тензори не є пустими, поки що вони будуть заповнені нулями.

`X_train` - ініціалізується довжиною яка відповідає кількості зображень які будуть використовуватися для навчання, в ньому будуть зберігатися зображення. Розміри тензору відповідають константним розмірам зображення які були визначені спочатку. Тип даних int8.

`Y_train` – також ініціалізується довжиною яка відповідає кількості зображень які будуть використовуватися для навчання, в цьому тензорі будуть зберігатися маски. Висота і ширина цього тензору така сама як і в попередньому випадку, але він має лише один канал для відображення кольору так як маски мають усього два кольори (чорний – для зображення фону, білий – для самої маски). Враховуючи це для зберігання інформації про маски використовується булевий тип даних.

Після цього цикл обходить усі директорії, зчитує зображення, змінює його розмір та додає до масиву тензорів.

```
for n, id_ in tqdm(enumerate(train_ids), total=len(train_ids)):
  path = TRAIN_PATH + id_
  img = imread(path + '/images/' + id_ + '.jpg')[:,:,:IMG_CHANNELS]
```

Функція `imread` зберігає зображення до тимчасової змінної `img`.

```
  img = resize(img, ( IMG_HEIGHT, IMG_WIDTH), mode='constant', preserve_range=True)
```

Після `resize` цього функція змінює розміри зображення до бажаних.

```
  X_train[n] = img
```

Останнім кроком буде додавання обробленого зображення до масиву.

```
  mask = np.zeros((IMG_HEIGHT, IMG_WIDTH, 1), dtype=np.bool)
  for mask_file in next(os.walk(path + '/masks/'))[2]:
```

Для перетворення маски до тензорного виду використовується більш складний алгоритм. Це обумовлено тим, що зображення з масками зберігаються у форматі *tif*, на відміну від

звичайних зображень які мають формат *jpg*. Для їх зчитування використовується інша функція, а саме `tiffread`.

```
mask_ = tiffread(path + '/masks/' + mask_file)[:1,:,:]
```

Окрім цього *tif* – зображення, мають інший формат тензору при зчитування. Якщо *jpg* має стандартну форму тензора (ШИРИНА x ВИСОТА x КІЛЬКІСТЬ КАНАЛІВ КОЛЬОРУ), то *tif* навпаки зберігається як (КІЛЬКІСТЬ КАНАЛІВ КОЛЬОРУ x ВИСОТА x ШИРИНА). Тобто у віддзеркаленому вигляді.

Щоб привести таке зображення до тіє їж форми яку мають *jpg* – зображення слід по-перше транспонувати отриманий тензор за допомогою функції з пакету NumPy transpose.

```
mask_transpose = np.transpose(mask_)
```

Після цього повернути тензор на дев'яносто градусів функцією rot90.

```
mask_rotate = np.rot90(mask_transpose, 3)
```

Останнім кроком буде відзеркалювання тензору по діагональній осі, функція fliplr

```
mask_flip = np.fliplr(mask_rotate)
```

Тепер отриманий тензор з *tif* – зображенням має таку ж форму як і *jpg*, і до нього можна примінити функцію resize, щоб привести до необхідних розмірів.

```
mask_ = resize(mask_flip, (IMG_HEIGHT, IMG_WIDTH), mode='constant',pr
eserve_range=True)
```

### 3.2.4 Навчання мережі

Для навчання мережі проводилося на платформі Google Colab. Goolge Colab – це сервіс від компанії Google який дає доступ до використання потужних графічних та тензорних процесорів які можна використовувати у дослідницьких цілях.

Для того, щоб розпочати навчання мережі необхідно скомпілювати мережу та настроїти її гіпер-параметри. Цей процес можна представити у вигляді наступних етапів.

*Eman 1.* Створення об'єкт класу Input в який передається інформація про зображення: висота, ширина, кількість графічних каналів.

```
input_img = Input((IMG_HEIGHT, IMG_WIDTH, IMG_CHANNELS), name='img')
```

*Eman 2.* Ініціалізація самої моделі, та передача параметрів.

```
model = get_unet(input_img, n_filters=16, dropout=0.2, batchnorm=True)
```

*Eman 3.* Компілювання моделі.

```
model.compile(optimizer=Adam(), loss="binary_crossentropy", metrics=["acc
uracy"])
```

*Eman 4.* Передача параметрів оптимізатора, функції втрат, та метрики для оцінки якості моделі.

Був обран оптимізатор *Aadam* - це адаптивний алгоритм оптимізації швидкості навчання, який був розроблений спеціально для навчання глибоких нейронних мереж. Функцією втрат була визначена бінарна кросентропія, це функція втрат, яка використовується при проблемах, пов'язаних з рішеннями так або ні (бінарні). Метрика для оцінки точності -*Accuracy* - показник для оцінки класифікаційних моделей. *Accuracy* - це частка прогнозів, яку наша модель отримала правильно.

*Eman 5.* Визначення функції зворотнього виклику які будуть використовуватися при навчанні.

```
callbacks = [
    EarlyStopping(patience=10, verbose=1),
    ReduceLROnPlateau(factor=0.1, patience=5, min_lr=0.00001, verbose=1),
ModelCheckpoint('unetmodel.h5', verbose=1, save_best_only=True, save_weig
hts_only=True)
]
```

Функція *EarlyStopping* використовується для того, щоб слідкувати за навчанням моделі та зупинити якщо результати не будуть поліпшені. Ця функція приймає параметр *patient* в якому вказується після якої кількості епох слід завершити навчання, якщо результат не буде змінюватися.

Функція *ReduceLROnPlateau* знизити швидкість навчання, коли показник перестав покращуватись. Моделі часто отримують вигоду від зниження швидкості навчання в 2–10 разів,

коли навчання застоюється. Цей зворотний виклик відстежує кількість, і якщо не спостерігається поліпшення швидкість навчання знижується.

*ModelCheckpoint* відповідає за зберігання отриманих ваг в окремий файл.

*Eman 6.* Навчання моделі.

Для того, щоб почати навчання моделі слід викликати метод fit.

```
results = model.fit(X_train, Y_train, validation_split=0.1, batch_size=16
, epochs=500, callbacks=callbacks)
```

Цей метод приймає зображення (X_train), маски (Y_train), коефіцієнт розділення датасету на тренувальний та перевіряючий (validation_split), розмір партії (кількість зображень яка подається до моделі за оди раз, batch_size), кількість епох навчання (epohs), та функції зворотнього виклику (callbacks).

Для поліпшення якості моделей були проведенні експерименти з різними параметрами моделі та визначенням різних гіпер-параметрів навчання. Результати приведені у таблиці 3.4.

Таблиця 3.5 - Результати мережі при різних конфігураціях

| n_filters | dropout | validation_split | batch_size | loss | accuracy | validation-loss | validation-accuracy |
|---|---|---|---|---|---|---|---|
| 16 | 0.2 | 0.1 | 16 | 0.0611 | 0.9410 | 0.1091 | 0.9278 |
| 16 | 0.2 | 0.1 | 8 | 0.1233 | 0.8211 | 0.1865 | 0.7933 |
| 16 | 0.2 | 0.1 | 32 | 0.1022 | 0.8425 | 0.1631 | 0.8024 |
| 16 | 0.1 | 0.2 | 16 | 0.1611 | 0.7514 | 0.2147 | 0.7013 |
| 16 | 0.1 | 0.2 | 8 | 0.1770 | 0.7452 | 0.2267 | 0.6923 |
| 16 | 0.1 | 0.2 | 32 | 0.1819 | 0.7011 | 0.2431 | 0.6433 |
| 8 | 0.2 | 0.1 | 16 | 0.1211 | 0.8910 | 0.1458 | 0.8613 |
| 8 | 0.2 | 0.1 | 8 | 0.1323 | 0.8412 | 0.1765 | 0.8232 |
| 8 | 0.2 | 0.1 | 32 | 0.1149 | 0.8848 | 0.1531 | 0.8595 |

Дані таблиці 3.5 пояснюються наступним чином.

Параметри можна пояснити наступним чином:

- n_filters - кількість фільтрів
- dropout - коефіцієнт зворотнього розповсюдження помилки
- validation_split - процент розділення вибірки на навчальну та перевірочну
- batch_size - кількість зображень яка подається до моделі

Результати:

- loss - коефіцієнт помилки на навчальній виборці
- accuracy – точність на навчальній виборці
- validation-loss - коефіцієнт помилки на перевірочній виборці
- validation-accuracy - точність на перевірочній виборці

Найточніші результати були досягнені використовуючи наступні конфігурації, представлені у таблиці 3.6.

Таблиця 3.5 – Рконфігурація мережі з найточнішими результатими

| n_filters | dropout | validation_split | batch_size | loss | accuracy | validation-loss | validation-accuracy |
|---|---|---|---|---|---|---|---|
| 16 | 0.2 | 0.1 | 16 | 0.0611 | 0.9410 | 0.1091 | 0.9278 |

На рисунку 3.8 наведений приклад роботи мережі.

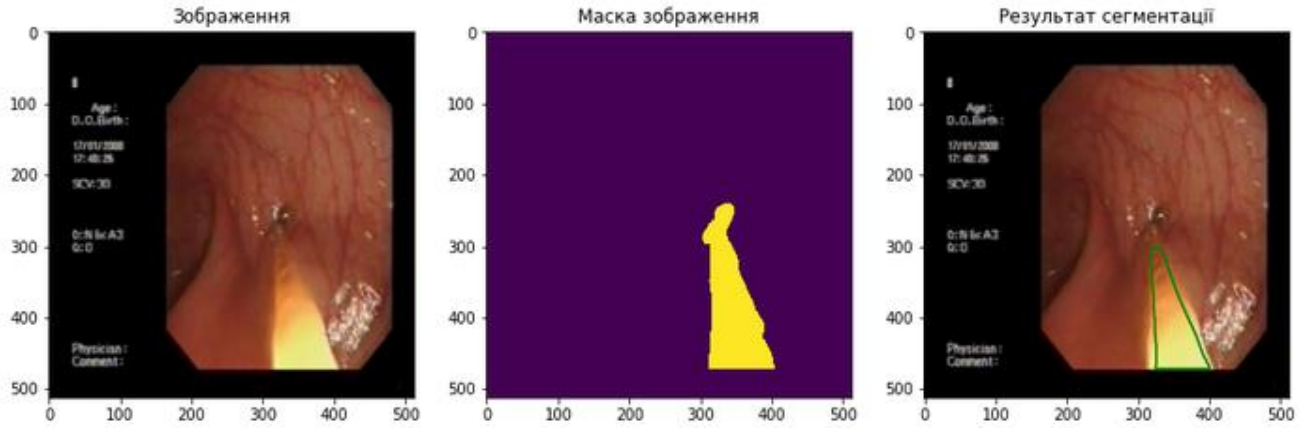


Рисунок 3.8 - Сегментація контуру хірургічного інструменту

### 3.2.5 Оцінка отриманих результатів розпізнавання

Для оцінки якості отриманих результатів сегментації було використано дві метрики.

1) Accuracy – це одна з найросповсюдженіших метрик для перевірки якості моделей машинного та глибокого навчання. Accuracy представляє собою відношення кількості правильних прогнозів до загальної кількості прогнозів і рораховується за формулою (3.1).

$$Accuracy = \frac{\text{Кількість правильних прогнозів}}{\text{Загальна кількість прогнозів}} \quad (3.1)$$

Для отримання оцінки Accuracy необхідно викликати метод *evaluate* у моделі і в якості параметрів передати відложену вибірку та правельні відповіді для цієї вибірки.

```
model.evaluate(X_test, Y_test)
```

після чого модель зробить прогнози для усіх зразків з тестової вибірки та видасть свій верикт. Розроблена модель отримала результат оцінки Accuracy - 0.6222770105126083 або 62,2%.

2) Друга метрика яка була використана для оцінки моделі – MeanIoU.

MeanIoU (Mean Intersection over Union, з англ. Середнє значення від перехрестя над об'єднанням) - це загальна метрика оцінювання для сегментації зображення, яка спочатку обчислює IOU для кожного зразку у вибірці, а потім обчислює середнє значення для всіх зразків. IOU в свою чергу є методом кількісного визначення відсоткового перекриття між цільовою маскою та отриманим результатом прогнозування. Розраховується наступним чином (3.2).

$$IOU = \frac{\text{ВПП}}{\text{ВПП+НПП+ННП}}, \tag{3.2}$$

де ВПП – вірні правильні прогнози, НПП – невірні правильні прогнози, ННП – невірні неправильні прогнози. Під ВПП та НПП розуміються прогнози для тих пікселів, які мережа віднесла до об'єкту сегментаціїї, тобто порахувала їх як істинні. ННП – це ті пікселі, які мережа не зарахувала які істинні хоча вони такими являються.

Нижче преставлена реалізація функції підрахунку MeanIoU на мові Python

```
from sklearn.metrics import confusion_matrix
def compute_iou(y_pred, y_true):
    # ytrue, ypred is a flatten vector
    y_pred = y_pred.flatten()
    y_true = y_true.flatten()
    current = confusion_matrix(y_true, y_pred, labels=[0, 1])
    # compute mean iou
    intersection = np.diag(current)
    ground_truth_set = current.sum(axis=1)
    predicted_set = current.sum(axis=0)
```

```
union = ground_truth_set + predicted_set - intersection
IoU = intersection / union.astype(np.float32)
return np.mean(IoU)
```

Розроблена мережа отримала оцінку 0.4453033815580889 або 44,5 % для метрики MeanIoU.

### 3.3 Висновки до розділу 3

В розділі представлена практична реалізаціїю методів прогнозування дій об’єктів у відео на підставі даних моделей розпізнавання з використанням методів обмежувальної коробки та сегментації відеозображень. Практична реалізація виконана з використанням технології глибокого навчання, нейронних мереж, мови програмування Python, бібліотеки TensorFlow.

Конфігурації мережі при розробленні моделі розпізнавання відеозображень з використанням методу обмежувальної коробки визначена наступними параметрами мережі: width, height, classes, batch, max_batch, subdivisions, decay, channels, momentum, adadm, burn_in, police, steps, scales, angle.

Розроблення моделі розпізнавання відеозображень з використанням методу сегментації складається з наступних етапів: створення об’єкт класу Input в який передається інформація про зображення: висота, ширина, кількість графічних каналів; ініціалізація самої моделі, та передача параметрів; компілювання моделі; передача параметрів оптимізатора, функції втрат, та метрики для оцінки якості моделі; визначення функції зворотнього виклику які будуть використовуватися при навчанні; навчання моделі.

Представлено проведення експерименту з розпізнавання відеозображень ендоскопічного дослідження з використанням методів обмежувальної коробки і сегментації. Якість розроблених моделей оцінено з використанням параметрів mean average precision (mAP), Accuracy та MeanIoU.

.

# РОЗДІЛ 4 ОХОРОНА ПРАЦІ ТА БЕЗПЕКА В НАДЗВИЧАЙНИХ СИТУАЦІЯХ. ЕКОЛОГІЯ

В даному розділі проведено аналіз потенційних небезпечних та шкідливих виробничих факторів, причин пожеж. Розглянуті заходи, які дозволяють забезпечити гігієну праці і виробничу санітарію. На підставі аналізу розроблені заходи з техніки безпеки та рекомендації з пожежної профілактики.

## 4.1. Загальні питання з охорони праці

Умови праці на робочому місці, безпека технологічних процесів, машин, механізмів, устаткування та інших засобів виробництва, стан засобів колективного та індивідуального захисту, що використовуються працівником, а також санітарно-побутові умови повинні відповідати вимогам нормативних актів про охорону праці. В законі України «Про охорону праці» [26] визначається, що охорона праці - це система правових, соціально-економічних, організаційно-технічних, санітарно-гігієнічних і лікувально-профілактичних заходів та засобів, спрямованих на збереження життя, здоров'я і працездатності людини у процесі трудової діяльності.

При роботі з обчислювальною технікою змінюються фізичні і хімічні фактори навколишнього середовища: виникає статична електрика, електромагнітне випромінювання, змінюється температура і вологість, рівень вміст кисню і озону в повітрі.

## 4.2. Аналіз стану умов праці

Робота над розробкою мктодів для прогнозування дій на відео в робочому приміщенні. Для даної роботи достатньо однієї людини, для якої надано робоче місце зі стаціонарним комп’ютером.

Геометричні розміри приміщення зазначені в таблиці 4.1. ті відповідають нормам згідно з ДСН 3.3.6.042-99 «Санітарні норми мікроклімату виробничих приміщень» [27].

Таблиця 4.1 – Розміри приміщення

| Найменування | Значення |
|---|---|
| Довжина, м | 6 |
| Ширина, м | 4 |
| Висота, м | 3 |
| Площа, $м^2$ | 24 |
| Об’єм, $м^3$ | 72 |

При порівнянні відповідності характеристик робочого місця нормативним основні вимоги до організації робочого місця за ДСанПіН 3.3.2.007-98 «Правила і норми роботи з візуальними дисплейними терміналами електронно-обчислювальних машин» [28] (табл. 4.2) і відповідними фактичними значеннями для робочого місця, констатуємо повну відповідність.

Таблиця 4.2 - Характеристики робочого місця

| Найменування параметра | Фактичне значення | Нормативне значення |
|---|---|---|
| Висота робочої поверхні, мм | 750 | 680 ÷ 800 |
| Висота простору для ніг, мм | 730 | не менше 600 |
| Ширина простору для ніг, мм | 660 | не менше 500 |
| Глибина простору для ніг, мм | 700 | не менше 650 |
| Висота поверхні сидіння, мм | 470 | 400 ÷ 500 |
| Ширина сидіння, мм | 400 | не менше 400 |
| Глибина сидіння, мм | 400 | не менше 400 |
| Висота поверхні спинки, мм | 600 | не менше 300 |
| Ширина опорної поверхні спинки, мм | 500 | не менше 380 |
| Радіус кривини спинки в горизонтальній площині, мм | 400 | 400 |
| Відстань від очей до екрану дисплея, мм | 800 | 700 ÷ 800 |

### 4.2.1. Навантаження та напруженість процесу праці

Під час виконання робіт використовують ПК та периферійні пристрої (лазерні та струменеві), що призводить до навантаження на окремі системи організму. Такі перекоси у напруженні різних систем організму, що трапляються під час роботи з ПК, зокрема, значна напруженість зорового аналізатора і довготривале малорухоме положення перед екраном, не тільки не зменшують загального напруження, а навпаки, призводять до його посилення і появи стресових реакцій.

Роботу за дипломним проектом визнано, таку, що займає 50% часу робочого дня та за восьмигодинної робочої зміни рекомендовано встановити додаткові регламентовані перерви: (потрібне вибрати):

- для розробників програм тривалістю 15 хв через кожну годину роботи;
- для операторів персональних комп'ютерів тривалістю 15 хв через дві години роботи;
- для операторів комп'ютерного набору тривалістю 10 хв через кожну годину роботи.

### 4.3. Виробнича санітарія

Аналіз небезпечних та шкідливих виробничих факторів виконується у табличній формі (табл. 4.3). Роботу, пов'язану з ЕОП з ВДТ, у тому числі на тих, які мають робочі місця, обладнані ЕОМ з ВДТ і ПП, виконують із забезпеченням виконання НПАОП 0.00.-7.15-18 «Вимоги щодо безпеки та захисту здоров`я працівників під час роботи з екранними пристроями» [29].

Таблиця 4.3 – Аналіз небезпечних і шкідливих виробничих факторів

| Небезпечні і шкідливі виробничі фактори | Джерела факторів (види робіт) | Кількісна оцінка | Нормативні документи |
|---|---|---|---|
| 1 | 2 | 3 | 4 |
| ***фізичні*** | | | |
| - підвищена температура поверхонь обладнання | експлуатація ЕОМ | 2 | [27] |
| - підвищена або знижена вологість повітря | -//- | 2 | [27] |
| - підвищений рівень електромагнітного випромінення | -//- | 2 | [32] |
| - підвищений рівень напруги електричної мережі, замикання якої може відбутися через тіло людини | -//- | 4 | [33] [34] |
| - підвищений рівень статичної електрики | -//- | 2 | [33] |
| - недостатність природного світла | порушення умов праці (вимог до приміщень) | 2 | [30] |

Продовження таблиці 4.3

| | | | |
|---|---|---|---|
| ***психофізіологічні:*** | | | |
| - нервово-психічна перевантаження (розумове, перенапруження аналізаторів-зорових) | - пошук інформації для постановки теми;<br>- пошук та аналіз аналогів і літератури;<br>- пошук наявних технологій, моделювання та аналіз алгоритмів;<br>- виконання роботи за темою диплома, тестування;<br>- оформлення роботи | 4 | [34] [28] |

### 4.3.1. Пожежна безпека

Для гасіння пожеж в робочому присміщені приміщенні пропонується використовувати порошкові або вуглекислотні вогнегасники, так як вони є універсальними.

Згідно НАПБ А. 01.001-2014 «Правила пожежної безпеки в Україні» [29] таке приміщення, площею 24 м$^2$, відноситься до категорії "В" (пожежонебезпечної). Відповідно до норм первинних засобів пожежогасінні пропонується використовувати:

- повсть 1 1 м$^2$, кошму 2×1,5 м2 в кількості 1 шт.

### 4.3.2. Електробезпека

На робочому місці виконуються наступні вимоги електробезпеки: ПК, периферійні пристрої та устаткування для обслуговування, електропроводи і кабелі за виконанням та ступенем захисту відповідають класу зони за ПУЕ (правила улаштування електроустановок), мають апаратуру захисту від струму короткого замикання та інших аварійних режимів. Лінія електромережі для живлення ПК, периферійних пристроїв і устаткування для обслуговування, виконана як окрема групова три-провідна мережа, шляхом прокладання фазового, нульового робочого та нульового захисного провідників. Нульовий захисний провідник використовується для заземлення (занулення) електроприймачів. Штепсельні з'єднання та електророзетки крім контактів фазового та нульового робочого провідників мають спеціальні контакти для підключення нульового захисного провідника. Електромережа штепсельних розеток для живлення персональних ПК, укладено по підлозі поруч зі стінами відповідно до затвердженого плану розміщення обладнання та технічних характеристик обладнання. Металеві труби та гнучкі металеві рукави заземлені. Захисне заземлення включає в себе заземлюючих пристроїв і

провідник, який з'єднує заземлюючий пристрій з обладнанням, яке заземлюється - заземлюючий провідник.

### 4.4. Гігієнічні вимоги до параметрів виробничого середовища

#### 4.4.1. Параметри мікроклімату

Мікроклімат робочих приміщень – це клімат внутрішнього середовища цих приміщень, що визначається діючої на організм людини з'єднанням температури, вологості, швидкості переміщення повітря. Отже оптимальні значення для температури, відносної вологості й рухливості повітря для зазначеного робочого місця відповідають ДСН 3.3.6.042-99 «Санітарні норми мікроклімату виробничих приміщень» [27] і наведені в таблиці 4.4:

Таблиця 4.4 – Норми мікроклімату робочої зони об’єкту

| **Період року** | **Категорія робіт** | **Температура $C^0$** | **Відносна вологість %** | **Швидкість руху повітря, м/с** |
|---|---|---|---|---|
| Холодна | легка-1 а | 22 - 24 | 40 – 60 | 0,1 |
| Тепла | легка-1 а | 23 - 25 | 40 – 60 | 0,1 |

#### 4.4.2. Освітлення

У приміщенні, де розташовані ЕОМ передбачається природне бічне освітлення, рівень якого відповідає ДБН В.2.5-28:2018 «Природне і штучне освітлення» [30]. Джерелом природного освітлення є сонячне світло. Регулярно повинен проводитися контроль освітленості, який підтверджує, що рівень освітленості задовольняє ДБН В.2.5-28:2018 [30] і для даного приміщення в світлий час доби достатньо природного освітлення.

*Розрахунок освітлення.*

Для виробничих та адміністративних приміщень світловий коефіцієнт приймається не менше -1/8, в побутових – 1/10:

$$S_b = \left(\frac{1}{5} \div \frac{1}{10}\right) \cdot S_n, \tag{4.1}$$

де $S_b$ – площа віконних прорізів, $м^2$;

$S_n$ – площа підлоги, м$^2$.

$S_n = a \cdot b = 6 \cdot 4 = 24$ м$^2$,

$S = 1/8 \cdot 24 = 3$ м$^2$.

Приймаємо 2 вікна площею S=2,2 м$^2$ кожне.

Розрахунок штучного освітлення виробляється по коефіцієнтах використання світлового потоку, яким визначається потік, необхідний для створення заданої освітленості при загальному рівномірному освітленні. Розрахунок кількості світильників n виробляється по формулі (4.2):

$$n = \frac{E \cdot S \cdot Z \cdot K}{F \cdot U \cdot M}, \tag{4.2}$$

де E – нормована освітленість робочої поверхні, визначається нормами – 300 лк;

S – освітлювана площа, м$^2$; S = 24 м$^2$;

Z – поправочний коефіцієнт світильника (Z = 1,15 для ламп розжарювання та ДРЛ; Z = 1,1 для люмінесцентних ламп) приймаємо рівним 1,1;

K – коефіцієнт запасу, що враховує зниження освітленості в процесі експлуатації – 1,5;

U – коефіцієнт використання, залежний від типу світильника, показника індексу приміщення і т.п. – 0,575

M – число люмінесцентних ламп в світильнику – 4;

F – світловий потік лампи – 5400лм (для ЛБ-80).

Підставивши числові значення у формулу (4.2), отримуємо:

$$n = \frac{300 \cdot 24 \cdot 1{,}1 \cdot 1{,}5}{5400 \cdot 0{,}575 \cdot 2} \approx 2{,}0$$

Приймаємо освітлювальну установку, яка складається з 2-х світильників, які складаються з двох люмінесцентних ламп загальною потужністю 160 Вт, напругою – 220 В.

### 4.4.3. Вентилювання

У приміщенні, де знаходяться ЕОМ, повітрообмін реалізується за допомогою природної організованої вентиляції (вентиляційні шахти), тобто при

V приміщення > 40 м$^3$ на одного працюючого допускається природна вентиляція. Цей метод забезпечує приток потрібної кількості свіжого повітря.

## 4.5. Заходи з організації виробничого середовища та попередження виникнення надзвичайних ситуацій

Відповідно до санітарно-гігієнічних нормативів та правил експлуатації обладнання наводимо приклади деяких заходів безпеки.

Розрахунок захисного заземлення (забезпечення електробезпеки будівлі).

Згідно з класифікацією приміщень за ступенем небезпеки ураження електричним струмом НПАОП 40.1-1.01-97 «Правила безпечної експлуатації електроустановок» [31], НПАОП 40.1-1.21-98 «Правила безпечної експлуатації електроустановок споживачів» [35] приміщення в якому проводяться всі роботи відноситься до першого класу (без підвищеної небезпеки). Під час роботи використовуються електроустановки з напругою живлення 36 В, 220 В, та 360 В. Опір контура заземлення повинен мати не більше 4 Ом.

Розрахунок проводять за допомогою методу коефіцієнта використання (екранування) електродів. Коефіцієнт використання групового заземлювача η – це відношення діючої провідності цього заземлювача до найбільш можливої його провідності за нескінченно великих відстаней між його електродами. Коефіцієнт використання вертикальних заземлювачів ηв в залежності від розміщення заземлювачів та їх кількості знаходиться в межах 0,4…0,99. Взаємну екрануючу дію горизонтального заземлювача (з'єднувальної смуги) враховують за допомогою коефіцієнта використання горизонтального заземлювача $\eta_c$.

Послідовність розрахунку.

1) Визначається необхідний опір штучних заземлювачів $R_{шт.з.}$:

$$R_{шт.з.} = \frac{R_д \cdot R_{пр.з.}}{R_{пр.з.} - R_д}, \tag{4.3}$$

де $R_{пр.з.}$ – опір природних заземлювачів; $R_д$ – допустимий опір заземлення. Якщо природні заземлювачі відсутні, то $R_{шт.з.}=R_д$.

Підставивши числові значення у формулу (рис.4.3), отримуємо:

$$R_{шт.з.} = \frac{4 \cdot 40}{40 - 4} \approx 4 \text{ Ом}$$

3) Опір заземлення в значній мірі залежить від питомого опору ґрунту ρ, Ом·м. Приблизне значення питомого опору глини приймаємо ρ=40 Ом•м (табличне значення).

3) Розрахунковий питомий опір ґрунту, $\rho_{розр}$, Ом·м, визначається відповідно для вертикальних заземлювачів $\rho_{розр.в}$, і горизонтальних $\rho_{розр.г}$, Ом·м за формулою:

$$\rho_{розр..} = \psi \cdot \rho, \tag{4.4}$$

де $\psi$ – коефіцієнт сезонності для вертикальних заземлювачів I кліматичної зони з нормальною вологістю землі, приймається для вертикальних заземлювачів $\rho_{розр.в}$=1,7 і горизонтальних $\rho_{розр.г}$=5,5 Ом·м.

$$\rho_{розр.в} = 1,7 \cdot 40 = 68 \text{ Ом•м}$$

$$\rho_{розр.г} = 5,5 \cdot 40 = 220 \text{ Ом•м}$$

4) Розраховується опір розтікання струму вертикального заземлювача $R_в$, Ом, за (4.5).

$$R_в = \frac{\rho_{розр.в}}{2 \cdot \pi \cdot l_в} \cdot \left( \ln \frac{2 \cdot l_в}{d_{ст}} + \frac{1}{2} \cdot \ln \frac{4 \cdot t + l_в}{4 \cdot t - l_в} \right), \tag{4.5}$$

де $l_в$ – довжина вертикального заземлювача (для труб - 2−3 м; $l_в$=3 м);

$d_{ст}$ – діаметр стержня (для труб - 0,03−0,05 м; $d_{ст}$=0,05 м);

t – відстань від поверхні землі до середини заземлювача, яка визначається за ф. (4.6):

$$t = h_в + \frac{l_в}{2}, \tag{4.6}$$

де $h_в$ – глибина закладання вертикальних заземлювачів (0,8 м); тоді $t = 0,8 + \frac{3}{2} = 2,3$ м

$$R_в = \frac{68}{2 \cdot \pi \cdot 3} \cdot \left( \ln \frac{2 \cdot 3}{0,05} + \frac{1}{2} \cdot \ln \frac{4 \cdot 2,3 + 3}{4 \cdot 2,3 - 3} \right) = 18,5 \text{ Ом}$$

5) Визначається теоретична кількість вертикальних заземлювачів n штук, без урахування коефіцієнта використання $\eta_в$:

$$n = \frac{2 \cdot R_в}{R_д} = \frac{2 \cdot 18,5}{4} = 9,25 \tag{4.7}$$

I визначається коефіцієнт використання вертикальних електродів групового заземлювача без врахування впливу з’єднувальної стрічки $\eta_в$ =0,57 (табличне значення).

6) Визначається необхідна кількість вертикальних заземлювачів з урахуванням коефіцієнта використання $n_в$, шт:

$$n_в = \frac{2 \cdot R_в}{R_д \cdot \eta_в} = \frac{2 \cdot 18,5}{4 \cdot 0,57} = 16,2 \approx 16 \tag{4.8}$$

7) Визначається довжина з'єднувальної стрічки горизонтального заземлювача $l_с$, м:

$$l_с = 1,05 \cdot L_в \cdot (n_в - 1), \tag{4.9}$$

де $L_в$ – відстань між вертикальними заземлювачами, (прийняти за $L_в$ = 3м);

$n_в$ – необхідна кількість вертикальних заземлювачів.

$$l_с = 1,05 \cdot 3 \cdot (16 - 1) \approx 48 \text{ м}$$

8) Визначається опір розтіканню струму горизонтального заземлювача (з'єднувальної стрічки) $R_г$, Ом:

$$R_г = \frac{\rho_{розр.г}}{2 \cdot \pi \cdot l_с} \cdot \ln \frac{2 \cdot l_с^{\,2}}{d_{см} \cdot h_г}, \tag{4.10}$$

де $d_{см}$ – еквівалентний діаметр смуги шириною b, $d_{см}$ =0,95b, b = 0,15 м;

$h_г$ – глибина закладання горизонтальних заземлювачів (0,5 м);

$l_с$ - довжина з'єднувальної стрічки горизонтального заземлювача $l_с$, м

$$R_г = \frac{220}{2 \cdot \pi \cdot 48} \cdot \ln \frac{2 \cdot 48^2}{0,95 \cdot 0,15 \cdot 0,5} = 8,1 \text{ Ом}$$

9) Визначається коефіцієнт використання горизонтального заземлювача $\eta_{с.}$ відповідно до необхідної кількості вертикальних заземлювачів $n_в$.

Коефіцієнт використання з'єднувальної смуги $\eta_с$=0,3 (табличне значення).

10) Розраховується результуючий опір заземлювального електроду з урахуванням з'єднувальної смуги:

$$R_{заг} = \frac{R_в \cdot R_г}{R_в \cdot \eta_с + R_г \cdot n_в \cdot \eta_в} \le R_д. \tag{4.11}$$

Висновок: дане захисне заземлення буде забезпечувати електробезпеку будівлі, так як виконується умова: $R_{заг}$ <4 Ом, а саме:

$$R_{заг} = \frac{18,5 \cdot 8,1}{18,5 \cdot 0,3 + 8,1 \cdot 16 \cdot 0,57} = 1,9 \leq R_{д}$$

### 4.6 Охорона навколишнього природного середовища

Діяльність за темою магістерської роботи, а саме: робота за компьютером, в процесі її виконання є фактори що впливать на навколишнє природнє середовище.

Основним екологічним аспектом в процесі діяльності за даними спеціальностями є процеси впливу на атмосферне повітря та процеси поводження з відходами, які утворюються, збираються, розміщуються, передаються на відалення (знешкодження), утилізацію, тощо в ІТ галузі.

Вплив на атмосферне повітря при нормальних умовах праці не оказує, бо не має в приміщенні сканерів, принтерів та інших джерел викиду забруднюючих речовин в повітря робочої зони.

В процесі діяльності компьютера виникають процеси поводження з відходами ІТ галузі. Нижче надано перелік відходів, що утворюються в процесі роботи:

Відпрацьовані люмінесцентні лампи - I клас небезпеки

Батарейки та акумулятори (малі) -III клас небезпеки

Змінні носії інформації - IV клас небезпеки

Відпрацьований ізолюючий матеріал, дроти та кабелі - IV клас небезпеки

Макулатура - IV клас небезпеки

Побутові відходи - IV клас небезпеки

### Висновки до розділу 4

В результаті проведеної роботи було зроблено аналіз умов праці, шкідливих та небезпечних чинників, з якими стикається робітник. Було визначено параметри і певні характеристики приміщення для роботи над розробкою методів прогнозуваннія дій об’єктів на відео описано, які заходи потрібно зробити для того, щоб дане приміщення відповідало необхідним нормам і було комфортним і безпечним для робітника. Приведені рекомендації щодо організації робочого місця, а також важливу інформацію щодо пожежної та електробезпеки. Були наведені розміри приміщення та наведено значення температури,

вологості й рухливості повітря, необхідна кількість і потужність ламп та інші параметри, значення яких впливає на умови праці робітника.

А також визначені основні екологічні аспекти впливу на навколишнє природнє середовище та зазначені заходи щодо поводження з ними.

# ВИСНОВКИ

Метою дипломної роботи визначено підвищення ефективності інтелектуальних медичних систем з використанням відео зйомки, що використовуються у закладах надання медичної допомоги за рахунок розробки методу прогнозування на підставі даних моделей розпізнавання, розроблених з використанням обмежуючої коробки і сегментації, що дозволить виявляти аномальні дії у відео хірургічних втручань в режимі реального часу та надати підтримку прийняття рішень лікарю під час проведення діагностичних та оперативних втручань.

В ході дослідницької частини роботи були отримані наступні результати:

1. Проведено аналіз методів та моделей розпізнавання, сегментації відео, прогнозування розвитку ситуацій.

2. Визначена систематизована сукупність етапів прогнозування дій об’єктів у відеопотоці.

3. Досліджено використання глибокого навчання, нейронних мереж для технології розпізнавання відео.

В ході практичної частини роботи були отримані наступні результати:

1. Розроблена модель розпізнавання відео ендоскопічного дослідження з використанням методу обмежуючої коробки.

2. Розроблена модель розпізнавання відео ендоскопічного дослідження з використанням методу сегментації.

3. Протестовано розроблені моделі.

4. Визначено оцінку точності моделей розпізнавання об’єктів у відео.

Таким чином, вирішено поставлене завдання - удосконалення методів і моделей прогнозування дій об’єктів у відео.

# ПЕРЕЛІК ПОСИЛАНЬ

1. Human Action Recognition and Prediction:A Survey. Yu Kong,Member, IEEE,and Yun Fu,Senior Member, IEEE. [Електронний ресурс] / Режим доступу до ресурсу: https://arxiv.org/pdf/1806.11230.pdf

2. Detecting Surgical Tools by ModellingLocal Appearance and Global Shape, David Bouget*, Rodrigo Benenson, Mohamed Omran, Laurent Riffaud, Bernt Schiele, and Pierre Jannin. [Електронний ресурс] / Режим доступу до ресурсу: https://rodrigob.github.io/documents/2015_tmi_bouget_et_al_detecting_surgical_tools.pdf

3. Robust Real-Time Detection of LaparoscopicInstruments in Robot Surgery Using ConvolutionalNeural Networks with Motion Vector Prediction. Kyungmin Jo, Yuna Choi, Jaesoon Choi and Jong Woo Chung. [Електронний ресурс] / Режим доступу до ресурсу:

https://res.mdpi.com/d_attachment/applsci/applsci-09-02865/article_deploy/applsci-09-02865.pdf

4. Action Prediction from Videos viaMemorizing Hard-to-Predict Samples, Yu Kong, Shangqian Gao, Bin Sun,Yun Fu. [Електронний ресурс] / Режим доступу до ресурсу:

https://pdfs.semanticscholar.org/c773/60f0bcdddd30e3f714b9ee127989f633a773.pdf

5. Tool Detection and Operative Skill Assessment in Surgical Videos UsingRegion-Based Convolutional Neural Networks, Amy Jin, Serena Yeung, Jeffrey Jopling, Jonathan Krause, Dan Azagury,Arnold Milstein, and Li Fei-Fei. [Електронний ресурс] / Режим доступу до ресурсу: https://arxiv.org/pdf/1802.08774.pdf

6. Robot-assisted laparoscopy. Marie Fidela R Paraiso, MD, FACOG, FPMRS Tommaso Falcone, MD, FRCSC, FACOG. [Електронний ресурс] / Режим доступу до ресурсу: https://www.uptodate.com/contents/robot-assisted-laparoscopyLindeberg, Tony (1998). "Feature detection with automatic scale selection". International Journal of Computer Vision. 30 (2): 79–116. doi:10.1023/A:1008045108935.

7. Lowe, David G. (1999). "Object recognition from local scale-invariant features" (PDF). Proceedings of the International Conference on Computer Vision. 2. pp. 1150–1157. doi:10.1109/ICCV.1999.790410.

8. Lindeberg, Tony & Bretzner, Lars (2003). Real-time scale selection in hybrid multi-scale representations. Proc. Scale-Space'03, Springer Lecture Notes in Computer Science. 2695. pp. 148–163. doi:10.1007/3-540-44935-3_11. ISBN 978-3-540-40368-5.

9. Kirchner, Matthew R. "Automatic thresholding of SIFT descriptors." In Image Processing (ICIP), 2016 IEEE International Conference on, pp. 291-295. IEEE, 2016.

10. Histograms of Oriented Gradients for Human Detection, Navneet Dalaland Bill Triggs.

11. Faster R-CNN: Towards Real-Time ObjectDetection with Region Proposal NetworksShaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun

12. SSD: Single Shot MultiBox DetectorWei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy,Scott Reed, Cheng-Yang Fu, Alexander C. Berg

13. Single-Shot Refinement Neural Network for Object DetectionShifeng Zhang, Longyin Wen, Xiao Bian, Zhen Lei, Stan Z. Li

14. You Only Look Once:Unified, Real-Time Object DetectionJoseph Redmon, Santosh Divvala, Ross Girshick, Ali Farhadi

15. Kashanipour, A.; Milani, N; Kashanipour, A.; Eghrary, H. (May 2008). "Robust Color Classification Using Fuzzy Rule-Based Particle Swarm Optimization"

16. Barghout, Lauren; Sheynin, Jacob (2013). "Real-world scene perception and perceptual organization: Lessons from Computer Vision". Journal of Vision.

17. Hossein Mobahi; Shankar Rao; Allen Yang; Shankar Sastry; Yi Ma. (2011). "Segmentation of Natural Images by Texture and Boundary Compression"

18. Ohlander, Ron; Price, Keith; Reddy, D. Raj (1978). "Picture Segmentation Using a Recursive Region Splitting Method"

19. Barghout, Lauren. Visual Taxometric approach Image Segmentation using Fuzzy-Spatial Taxon Cut Yields Contextually Relevant Regions. Communications in Computer and Information Science (CCIS). Springer-Verlag. 2014

20. Barghout, Lauren (2014). Vision. Global Conceptual Context Changes Local Contrast Processing (Ph.D. Dissertation 2003).

21. R. Nock and F. Nielsen, Statistical Region Merging, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol 26, No 11, pp 1452-1458, 2004.

22. Fully Convolutional Networksfor Semantic SegmentationEvan Shelhamer, Jonathan Long, and Trevor Darrell,Member, IEEE

23. The One Hundred Layers Tiramisu:Fully Convolutional DenseNets for Semantic Segmentation, Simon J´egou1Michal Drozdzal, David Vazquez, Adriana Romero, Yoshua Bengio

24. Gated-SCNN: Gated Shape CNNs for Semantic Segmentation Towaki Takikawa, David Acuna, Varun Jampani, Sanja Fidler.

25. U-Net: Convolutional Networks for BiomedicalImage Segmentation

26. Закон України «Про охорону праці». Вводиться в дію Постановою ВР № 2695-XII від 14.10.92, ВВР, 1992, № 49, ст.669. – Режим доступу: www. URL: https://zakon.rada.gov.ua/laws/show/2694-12

27. ДСН 3.3.6.042-99 «Санітарні норми мікроклімату виробничих приміщень». Вводиться в дію Постановою ВР № 42 від 01.12.1999. – Режим доступу: www. URL: https://zakon.rada.gov.ua/rada/show/va042282-99

28. ДСанПіН 3.3.2.007-98 «Державні санітарні правила і норми роботи з візуальними дисплейними терміналами електронно-обчислювальних машин». Вводиться в дію Постановою ВР № 7 від 10.12.1998. – Режим доступу: www. URL: https://zakon.rada.gov.ua/rada/show/v0007282-98

29. НАПБ А. 01.001-2014 «Правила пожежної безпеки в Україні». Затверджено Наказом Міністерства внутрішніх справ України № 1417 від 30.12.2014. – Режим доступу: www. URL: https://zakon4.rada.gov.ua/laws/show/z0252-15

30. ДБН В.2.5-28:2018 «Природне і штучне освітлення». – Режим доступу: www. URL: http://www.minregion.gov.ua/wp-content/uploads/2018/12/V2528-1.pdf

31. НПАОП 40.1-1.01-97 «Правила безпечної експлуатації електроустановок». Затверджено наказом Державного комітету України по нагляду за охороною праці № 257 від 6 жовтня 1997 р. – Режим доступу: www. URL: https://zakon.rada.gov.ua/laws/show/z0011-98

32. ДСТУ 7237:2011 «Система стандартів безпеки праці. Електробезпека. Загальні вимоги та номенклатура видів захисту». Затверджено Держспоживстандартом України № 37 від 02.02.2011. – Режим доступу: www. URL: http://online.budstandart.com/ru/catalog/doc-page.html?id_doc=30045

33. ГОСТ 13109-97 «Електрична енергія. Сумістність технічних засобів електромагнітних». Дата введення 01.01.1999. – Режим доступу: www. URL: https://dnaop.com/html/42313/doc-ГОСТ_13109-97

34. НПАОП 0.00-7.15-18 «Вимоги щодо безпеки та захисту здоров`я працівників під час роботи з екранними пристроями». Зареєстровано в Міністерстві юстиції України 25 квітня 2018 р. за № 508/31960. – Режим доступу: www. URL: https://zakon.rada.gov.ua/laws/show/z0508-18

35. НПАОП 40.1-1.21-98 «Правила безпечної експлуатації електроустановок споживачів». Затверджено Наказом Держнаглядохоронприці України № 4 від 09.01.98 – Режим доступу: www. URL: https://zakon.rada.gov.ua/laws/show/z0093-98

**Додаток А Слайди презентації**

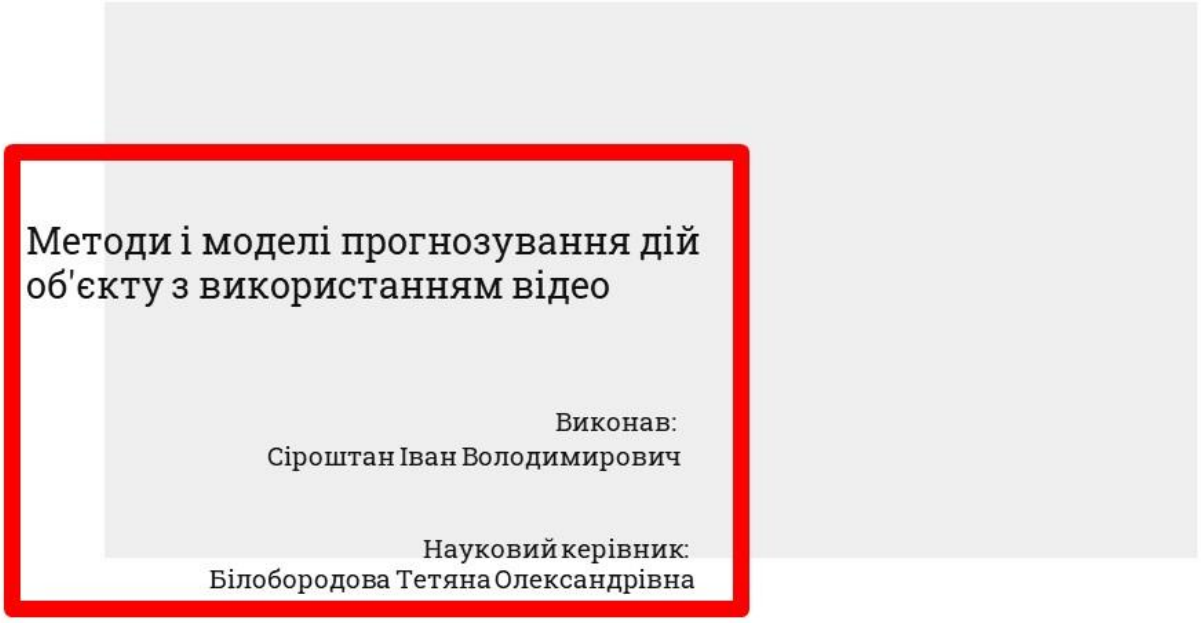


Рисунок А.1 – Слайд №1. Титульна сторінка

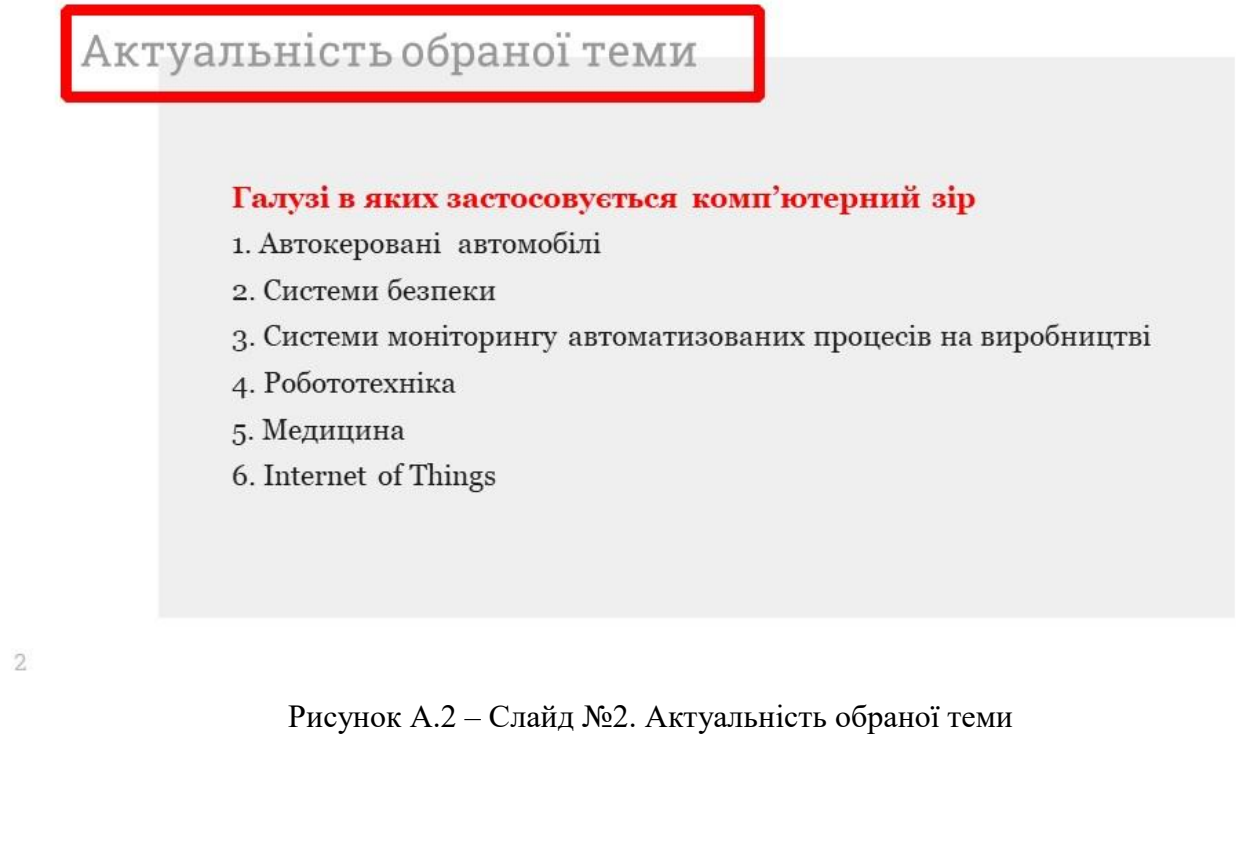


Рисунок А.2 – Слайд №2. Актуальність обраної теми

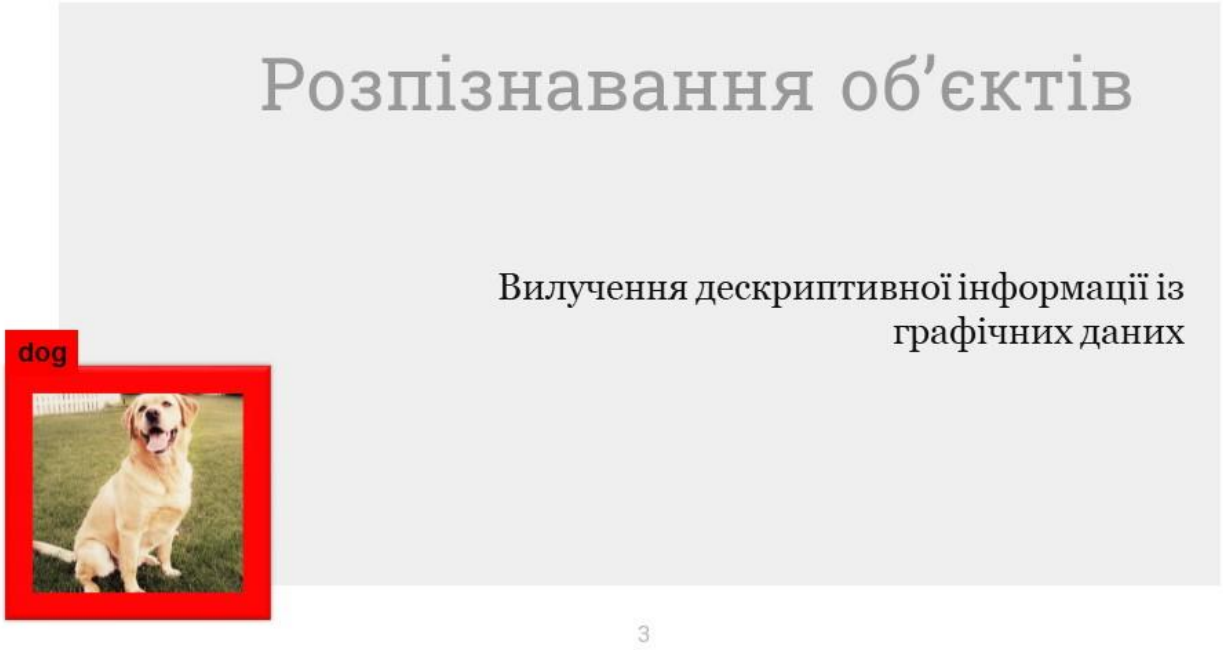


Рисунок А.3 – Слайд №3. Огляд проблем у галузі

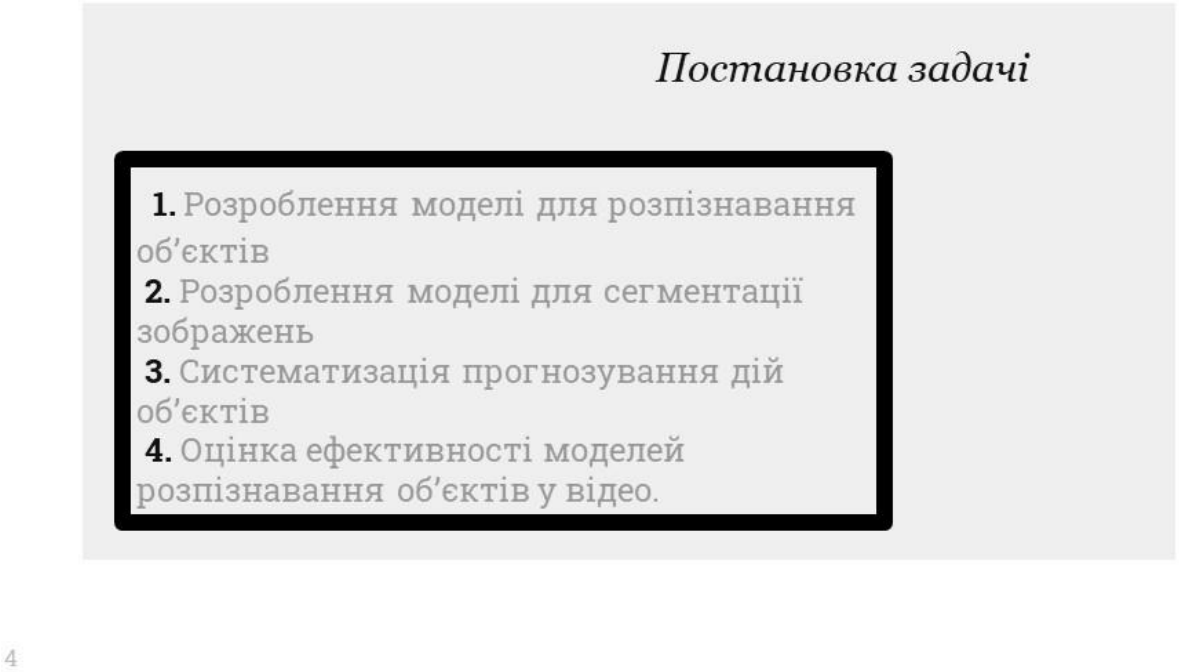


Рисунок А.4 – Слайд №4. Постановка задачі


- ***Об'єкт дослідження*** *– процеси перетворення відеопотоку у цифрові дані та їх використання.*
- ***Предмет дослідження –*** *методи і моделі прогнозування розвитку дій об'єктів у відео.*
- ***Мета*** *-підвищення ефективності інтелектуальних медичних систем з використанням відеозйомки*

5

Рисунок А.5 – Слайд №5. Визначення напрямку дослідження

***Для досягнення мети дослідження необхідно вирішити такі завдання***

- аналіз методів та моделей розпізнавання, сегментації відео, прогнозування розвитку ситуацій
- розроблення методу систематизованого прогнозування дій об'єктів у відеопотоці та визначення сукупності етапів прогнозування
- розроблення моделі розпізнавання з використанням методу обмежуючої коробки;
- розроблення моделі розпізнавання з використанням методу сегментації;
- оцінка ефективності моделей розпізнавання об'єктів у відео.

6

Рисунок А. 6 – Слайд №6. Постановка задач


Рисунок А.7 – Слайд №7. Розпізнавання дій


Рисунок А.8 - Слайд №8. Огля галузей застосування


Рисунок А.9 – Слайд №9. Метод прогнозування дій


Рисунок А.10 – Слайд №10. Методи розпізнавання об’єктів

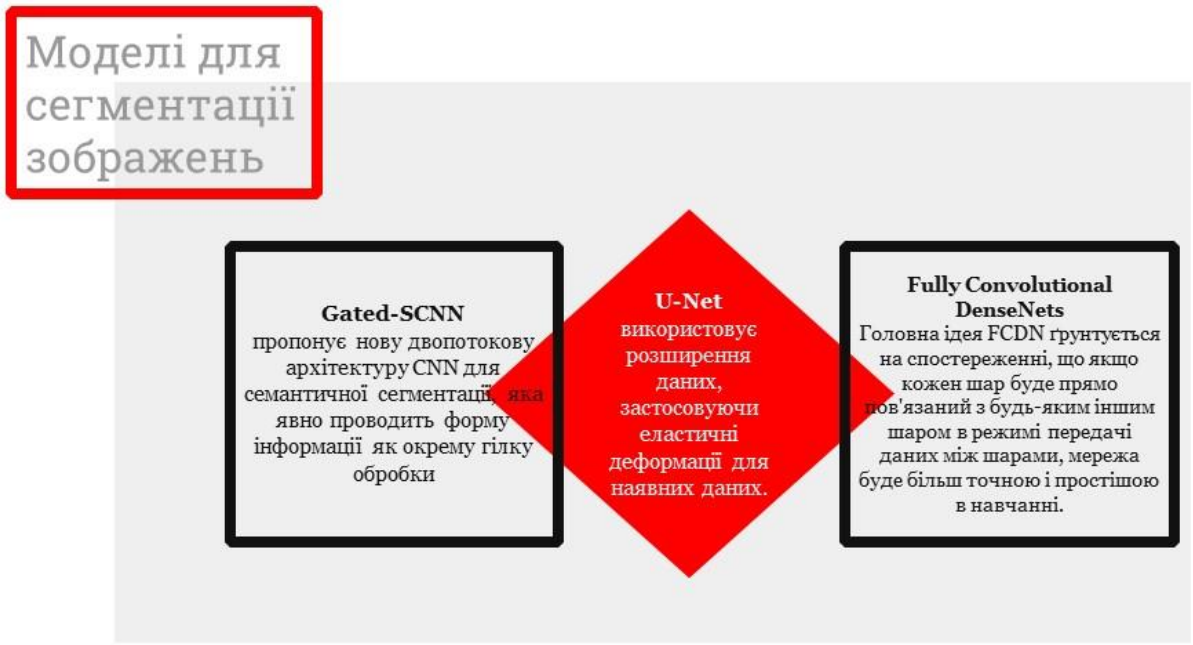


Рисунок А.11 – Слайд №11. Моделі для сегментації зображень


Рисунок А.12 – Слайд №12. Ендоскоп

## Класи які розпізнає модель

| Назва класу | Переклад |
|---|---|
| Saturation | Насичення |
| Artifact | Артефакт |
| Blur | Розмиття |
| Bubbles | Бульбашки |
| Instrument | Інструмент |
| Blood | Кров |
| Contrast | Контрастність кольорів |
| Specularity | Віддзеркалення |

13

Рисунок А.13 – Слайд №13. Класи які використовує модель

## Для тренування U-Net використовуються маски зображень


14

Рисунок А.14 – Слайд №14. Приклад даних для навчання моделі

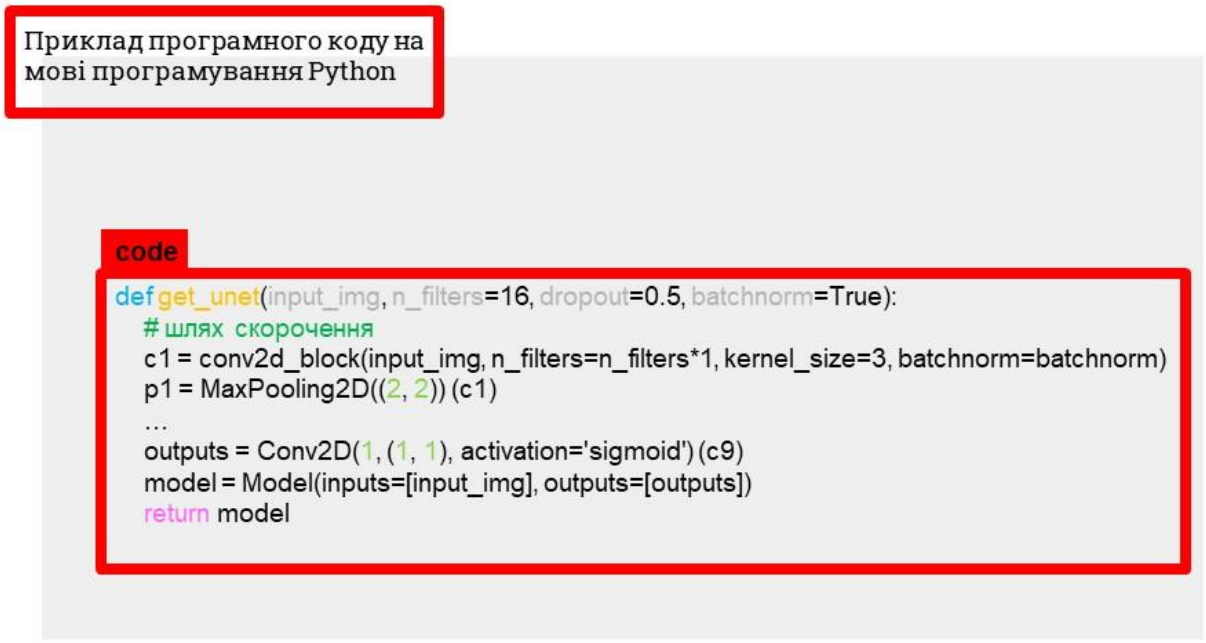


Рисунок А.15 – Слайд №15. Приклад програмного коду


Рисунок А.16 – Слайд №16. Отримані результати

***В ході дослідницької частини роботи були отримані наступні результати:***

1. Проведено аналіз методів та моделей розпізнавання, сегментації відео, прогнозування розвитку ситуацій.
2. Визначена систематизована сукупність етапів прогнозування дій об'єктів у відеопотоці.
3. Досліджено використання глибокого навчання, нейронних мереж для технології розпізнавання відео.

***В ході практичної частини роботи були отримані наступні результати:***

17

1. Розроблена модель розпізнавання відео ендоскопічного дослідження з використанням методу обмежуючої коробки.
2. Розроблена модель розпізнавання відео ендоскопічного дослідження з використанням методу сегментації.
3. Протестовано розроблені моделі.
4. Визначено оцінку точності моделей розпізнавання об'єктів у відео.

Рисунок А.17 – Слайд №17. Висновки

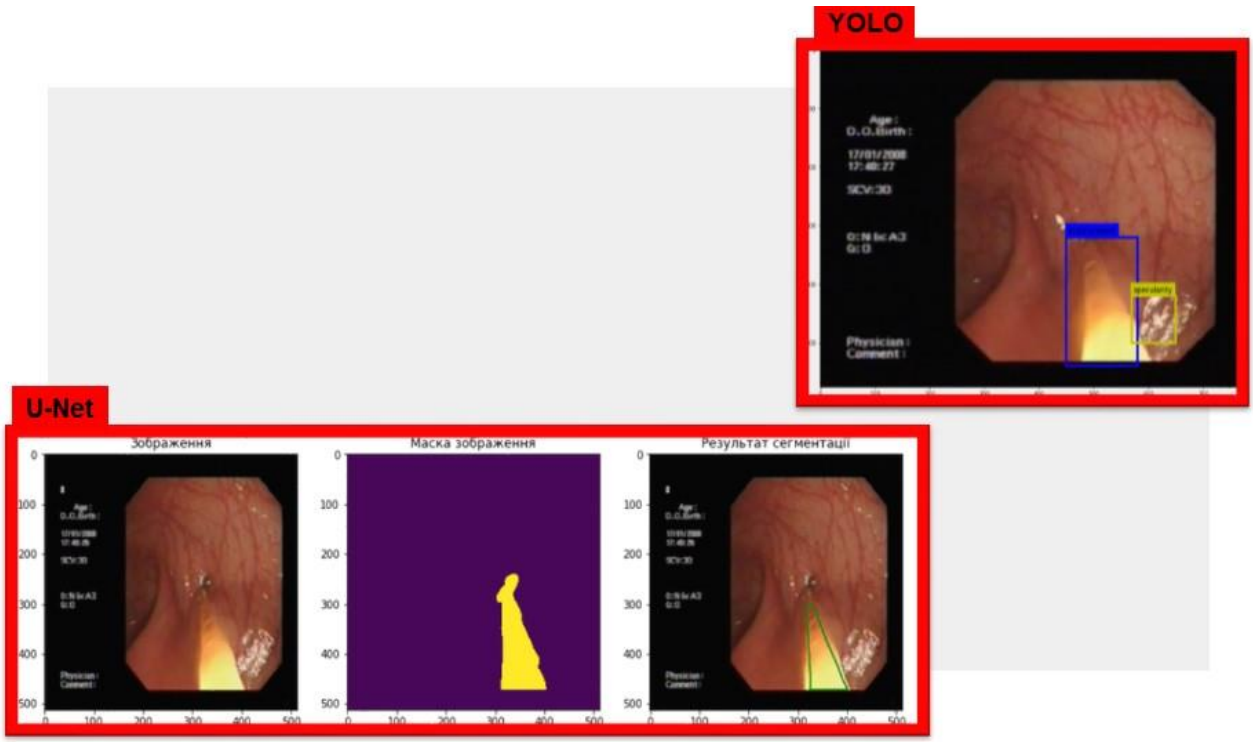


Рисунок А.18 – Слайд №18. Приклад отриманих результатів