Міністерство освіти і науки України
Тернопільський національний технічний університет імені Івана Пулюя

Факультет комп'ютерно-інформаційних систем і програмної інженерії
(повна назва факультету )
Кафедра комп'ютерних систем та мереж
(повна назва кафедри)

# КВАЛІФІКАЦІЙНА РОБОТА

на здобуття освітнього ступеня

**Магістр**
(назва освітнього ступеня)

на тему: Методи та засоби виявлення небезпечних ситуацій для людей комп'ютеризованою системою відеонагляду

Виконав: студент(ка) 6 курсу, групи СІм-61
спеціальності 123 «Комп'ютерна інженерія»
(шифр і назва спеціальності)

(підпис) Величко Д.В. (прізвище та ініціали)

Керівник (підпис) Луцик Н.С. (прізвище та ініціали)

Нормоконтроль (підпис) Тиш Є.В. (прізвище та ініціали)

Завідувач кафедри (підпис) Осухівська Г.М. (прізвище та ініціали)

Рецензент (підпис) Сверстюк А.С. (прізвище та ініціали)

Тернопіль
2024

Міністерство освіти і науки України
Тернопільський національний технічний університет імені Івана Пулюя

Факультет комп'ютерно-інформаційних систем і програмної інженерії
(повна назва факультету)

Кафедра комп'ютерних систем та мереж
(повна назва кафедри)

ЗАТВЕРДЖУЮ
Завідувач кафедри
_____ *Осухівська Г.М.*
(підпис) (прізвище та ініціали)
«11» листопада 2024 р.

# З А В Д А Н Н Я

## НА КВАЛІФІКАЦІЙНУ РОБОТУ

на здобуття освітнього ступеня *магістр*
(назва освітнього ступеня)

за спеціальністю *123 «Комп'ютерна інженерія»*
(шифр і назва спеціальності)

студенту *Величко Діані Вадимівні*
(прізвище, ім'я, по батькові)

1. Тема роботи *Методи та засоби виявлення небезпечних ситуацій для людей комп'ютеризованою системою відеонагляду*

Керівник роботи *Луцик Надія Степанівна, PhD, доцент*
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

Затверджені наказом ректора від « *29* » *листопада 2024* року № 4/7-1144
2. Термін подання студентом завершеної роботи *20.12.2024 р.*
3. Вихідні дані до роботи *наукові літературні джерела*

4. Зміст роботи (перелік питань, які потрібно розробити) *Опис науково-технічної області досліджень. Огляд та аналіз методів і систем для виявлення небезпечних ситуацій. Архітектура комп'ютеризованої системи для виявлення небезпечних ситуацій. Обгрунтування вибору типу моделі для виявлення небезпечних ситуацій. Модель YOLOv8, YOLOv10, YOLOv11. Формування набору даних. Отримання даних. Підключення датасету та YOLO для тренування та використання.Тренування моделей YOLOv8, YOLOv10, YOLOv11. Результати виявлення небезпечних ситуацій. Охорона праці. Безпека в надзвичайних ситуаціях.*
5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень, слайдів)
1. *Актуальність, мета, завдання роботи*
2. *Об'єкт, предмет, наукова новизна*
3. *Архітектура комп'ютеризованої системи відеонагляду для виявлення небезпечних ситуацій*
4. *Моделі YOLO*
5. *Набір даних*
6. *Перебіг тренування моделей*
7. *Результати*
8. *Висновки*

6. Консультанти розділів роботи

| Розділ | Прізвище, ініціали та посада консультанта | Підпис, дата | |
|---|---|---|---|
| | | завдання видав | завдання прийняв |
| *Охорона праці* | *Пилипець М.І., проф.каф. МТ* | *05.12.2024* | *18.12.2024* |
| *Безпека в надзвичайних ситуаціях* | *Стручок В.С., ст.викл.каф.ОХ* | *10.122024* | *18.12.2024* |

7. Дата видачі завдання *11.11.2024*

## КАЛЕНДАРНИЙ ПЛАН

| № з/п | Назва етапів роботи | Термін виконання етапів роботи | Примітка |
|---|---|---|---|
| 1 | *Огляд та аналіз методів і систем для виявлення небезпечних ситуацій.* | | |
| 2 | *Аналіз науково-технічної області досліджень.* | | |
| 3 | *Архітектура комп'ютеризованої системи для виявлення небезпечних ситуацій.* | | |
| 4 | *Обгрунтування вибору типу моделі для виявлення небезпечних ситуацій.* | | |
| 5 | *Модель YOLOv8, YOLOv10, YOLOv11.* | | |
| 6 | *Формування набору даних.* | | |
| 7 | *Отримання даних.* | | |
| 8 | *Підключення датасету та YOLO для тренування та використання.* | | |
| 9 | *Тренування моделей YOLOv8, YOLOv10, YOLOv11.* | | |
| 10 | *Результати виявлення небезпечних ситуацій.* | | |
| 11 | *Охорона праці. Безпека в надзвичайних ситуаціях.* | | |
| 12 | *Оформлення пояснювальної записки та графічного матеріалу.* | | |
| 13 | *Попередній захист* | | |
| 14 | *Захист кваліфікаційної роботи* | | |
| | | | |
| | | | |

Студент _____ *Величко Д.В.*
(підпис) (прізвище та ініціали)

Керівник роботи _____ *Луцик Н.С.*
(підпис) (прізвище та ініціали)

# АНОТАЦІЯ

Величко Д. В. Методи та засоби виявлення небезпечних ситуацій для людей комп’ютеризованою системою відеонагляду: робота на здобуття кваліфікаційного ступеня магістра: спец. 123 — комп’ютерна інженерія / наук.кер. Н. С. Луцик. Тернопіль: Тернопільський національний технічний університет імені Івана Пулюя, 2024. — 84 с.

Ключові слова: комп’ютеризована система відеонагляду, небезпечна ситуація, розпізнавання пози, модель YOLO.

Кваліфікаційна робота присвячена дослідженню методів і засобів виявлення небезпечних ситуацій для людей комп’ютеризованою системою відеонагляду. Робота включає в себе аналіз методів і систем для виявлення небезпечних ситуацій, а також обгрунтування використання YOLO, як ефективного засобу для виявлення небезпечних ситуацій у системах відеонагляду завдяки їх здатності працювати в реальному часі та високій точності виявлення об'єктів.

На основі експериментального дослідження та комплексного аналізу всіх характеристик встановлено, що модель YOLOv11s демонструє оптимальне співвідношення між точністю та швидкодією, маючи найменший час попереднього опрацювання (0.6 мс), високі показники точності (mAP50 = 0.992) та помірні вимоги до пам'яті (19.5 Мб). Тому для практичної реалізації системи виявлення небезпечних ситуацій рекомендовано використовувати модель YOLOv11s, що забезпечить оптимальний баланс між точністю розпізнавання та швидкодією системи в реальних умовах експлуатації.

# ANNOTATION

Velychko D. V. Methods and tools for detecting dangerous situations for people using a computerized video surveillance system. Master's Graduation Thesis: speciality 123 - Computer engineering / supervisor N.S. Lutsyk. Ternopil: Ternopil Ivan Puluj National Technical University, 2024. — 84 p.

Keywords: computerized video surveillance system, dangerous situation, pose recognition, YOLO model.

The Master's graduation thesis is dedicated to the exploration of methods and tools for detecting dangerous situations for people using a computerized video surveillance system. The work includes an analysis of methods and systems for detecting dangerous situations, as well as a justification for the use of YOLO as an effective means for detecting dangerous situations in video surveillance systems due to its ability to operate in real time with high object detection accuracy.

Experimental research and a comprehensive analysis revealed that the YOLOv11s model demonstrates the optimal ratio between accuracy and speed, having the smallest pre-processing time (0.6 ms), high accuracy rates (mAP50 = 0.992) and moderate memory requirements (19.5 MB). Therefore, for the practical implementation of the dangerous situations detection system, it is recommended to use the YOLOv11s model, which will provide an optimal balance between recognition accuracy and system speed in real operating conditions.

ЗМІСТ

# ВСТУП

**Актуальність роботи.** У сучасному світі зростання кількості терактів, нападів, крадіжок та інших надзвичайних ситуацій, які становлять загрозу безпеці людей, стало серйозним викликом для правоохоронних органів, спецслужб, бізнесу та громадян. Існуючі системи не завжди здатні забезпечити своєчасне виявлення та оповіщення про небезпечну ситуацію. Комп'ютеризована система, яка виявляє небезпечні ситуації, може бути ефективним інструментом для запобігання надзвичайних ситуацій. Використання штучного інтелекту дозволить у режимі реального часу виявляти надзвичайні ситуації.

Методам виявлення небезпечних ситуацій присвячено ряд досліджень науковців, зокрема, P.Kuppusamy, V.C.Bharathi, S.Y.Park, A.Prati, Jang Sein, Battulga Lkhagvadorj, Nasridinov Aziz, M.Nakib, R.T.Khan, M.S.Hasan, J.Uddi, A.Yadav, N.Thaker, D.,Makwana та ін. В їх роботах запропоновані різні підходи та методи до виявлення небезпечних (аномальних) ситуацій. В цьому дослідженні ідентифікація надзвичайної ситуації здійснюється за позою людини «руки вгору» з використанням штучного інтелекту. Важливим при цьому є точність та швидкість розпізнавання комп’ютеризованою системою такої ситуації. Тому дослідження методів та засобів виявлення небезпесних ситуацій комп’ютеризованою системою є актуальною задачею.

**Метою кваліфікаційної роботи** є підвищення ефективності виявлення небезпечних ситуацій для людей комп’ютеризованими системати відеонагляду.

**Завдання кваліфікаційної роботи:**

- Проаналізувати сучасні методи та підходи до виявлення небезпечних ситуацій для людей системами віденагляду.

- На основі аналізу обрати методи та моделі, придатні для виявлення небезпечних ситуацій комп’ютеризованими системами.

- Сформувати репрезентативний набір даних для навчання моделі штучного інтелекту з метою виявлення небезпечних ситуацій за позою людини «руки вгору».

- Провести навчання моделей на сформованому наборі даних.

- Провести експериментальні дослідження обгрунтованих методів та оцінити їх ефективність.

- Розробити рекомендації щодо впровадження отриманих результатів.

*Об’єктом дослідження є:* процес виявлення небезпечних ситуацій для людей комп’ютеризованою системою відеонагляду.

*Предметом дослідження є:* методи та засоби виявлення небезпечних ситуацій, зокрема, пози "руки вгору", як індикатора екстреної ситуації, на основі алгоритмів машинного навчання.

**Методи дослідження:** для аналізу наукової літератури та технічної документації, для вивчення існуючих підходів до виявлення небезпечних ситуацій використано теоретичні методи, методи системного аналізу - для визначення основних компонентів та взаємозв'язків у системі відеонагляду, методи порівняння - для аналізу різних версій архітектури YOLO та вибору оптимального рішення, емпіричні методи - для експериментального дослідження характеристик різних моделей YOLO, методи машинного навчання, зокрема глибокого навчання з використанням згорткових нейронних мереж архітектури YOLO.

**Наукова новизна дослідження**

- Вперше обґрунтовано та запропоновано використання моделі YOLOv11 для виявлення небезпечних ситуацій комп’ютеризованими системати відеонагляду шляхом розпізнавання пози “руки вгору” як єдиного об’єкта.

- Удосконалено архітектуру комп'ютеризованої системи відеонагляду для виявлення небезпечних ситуацій шляхом використання сучасних архітектур нейронних мереж YOLO.

**Практичне значення результатів кваліфікаційної роботи:** використання сучасних архітектур нейронних мереж YOLO дозволяє підвищити ефективність виявлення небезпечних ситуацій для людини комп’ютеризованими системати відеонагляду.

**Публікації**. За тематикою досліджень опубліковано статтю у міжнародному журналі «Advances in Science and Technology Research Journal» [1] та проведено апробацію на VI Міжнародній студентській науково-технічній конференції

«Природничі та гуманітарні науки. Актуальні питання» і XI науково-технічній конференції «Інформаційні моделі, системи та технології» [2, 3].

**Структура роботи.** Робота складається з пояснювальної записки та графічної частини. Пояснювальна записка складається із вступу, 4 розділів, висновків, списку використаних джерел та додатків. Обсяг роботи: пояснювальна записка – 84 арк. формату А4, графічна частина – 8 аркушів формату А1.

# РОЗДІЛ 1
# АНАЛІТИЧНА ЧАСТИНА

## 1.1. Опис науково-технічної області досліджень

Спостереження за допомогою відеокамер широко використовується в теперішній час. Комп'ютеризовані системи відеонагляду (КСВ) впроваджуються для забезпечення безпеки шляхом реагування та прогнозування різноманітних можливих небезпечних (непередбачуваних) ситуацій у публічних та приватних місцях. Серед основних завдань КСВ є нагляд за об'єктом спостереження за допомогою камер, виявлення і класифікація подій та інформування про них. Зазвичай такі системи передбачають наявність IoT приладу (відеокамери чи комбінацію камер відеоспостереження) та аналітичної складової, що проводить попередній аналіз [4].

Науково-технічна область досліджень пов'язана із вивченням методів та засобів виявлення небезпечних обставин для людей комп'ютеризованою системою відеонагляду. На сьогоднішній день з цією метою використовують сучасні технології, а саме, штучний інтелект (ШІ). ШІ виявився одним із найперспективніших інструментів для автоматизації процесів виявлення і оповіщення про непередбачувані (небезпечні) ситуації, які можуть виникнути з різних причин, та сприяти швидкому реагуванню на них. Використання технологій машинного навчання дозволяє створювати системи, які здатні розпізнавати загрози в реальному часі з високою точністю. Великий інтерес викликають моделі, такі як Convolutional Neural Networks(CNN), які використовуються для опрацювання зображень, і як результат, можуть бути використані для аналізу відеопотоків, завдяки можливості виділяти важливі особливості, що вказують на наявність потенційної загрози. Важливу роль в опрацюванні відеоданих відіграють підходи глибокого навчання CNN. Розпізнавання людини здійснюється за допомогою різних CNN: 2D CNN і 3D CNN, зокрема, таких як моделі VGG-16, AlexNet, VGG-19, Yolo. Зображення містять людей та різні

небажані об’єкти з елементами фону. Вони ідентифікуються та “витягуються” з послідовності кадрів за допомогою CNN з високою точністю.

Комп’ютеризовані системи відеонагляду, що використовують технології ШІ, надають можливість автоматизованого моніторингу та аналізу відеопотоків з камер відеонагляду. Такі системи здатні виявляти небезпечні ситуації, такі як напад на людину, агресивна поведінка та інші події. Вони дозволяють значно зменшити навантаження на операторів відеонагляду, автоматично повідомляючи про виявлення потенційних загроз.

Існують різноманітні підходи до виявлення небезпек для людей із зображень за допомогою різних методів ШІ. Відповідно до них виокремлюють два основні: класифікація зображення та виявлення об’єктів. Підходи до класифікації зображень зазвичай аналізують зображення за допомогою різних методів і визначають, чи є небезпечна ситуація. В цьому дослідженні (кваліфікаційній роботі) небезпечною ситуацією запропоновано вважати таку, коли у людини підняті вгору обидві руки. Тобто завданням КСВ є розпізнати позу людини “руки вгору”. Така “поза жертви” для виявлення небезпеки вибрана з причини, що якщо є реальна загроза для людини, наприклад, напад зловмисників, пограбування, залякування тощо, то підняття рук догори є найчастішою реакцією на такі дії, а також, як сигнал системі, - не викликатиме підозри. КСВ, яка виявлятиме небезпечну ситуацію за “сигнальною позою” - “руки вгору”, може бути складовою систем “розумного будинку”, “розумного міста”, тощо.

Оцінка пози є складним завданням, яке має високу складність та вимагає значних обчислювальних потужностей. Використання ШІ для розпізнавання людини та її пози є актуальним напрямком досліджень у галузі комп'ютерного зору. Незважаючи на значні досягнення у сфері ШІ та комп'ютерного зору, все ще існують певні виклики. До них належать проблеми з опрацюванням даних у реальному часі, забезпечення конфіденційності та безпеки даних, а також адаптація систем до роботи у різних умовах. Проте розвиток технологій та збільшення обчислювальних потужностей відкривають нові можливості для покращення існуючих рішень. Відомі різні підходи, методи та засоби, які використовуються з метою розпізнавання

різноманітних об’єктів у системах відеонагляду. Деякі з популярних методів включають використання глибоких нейронних мереж і методів, які дозволяють виявляти та класифікувати ключові точки на тілі людини, які потім використовуються для визначення положення (пози).

Таким чином, науково-технічна область пов'язана з дослідженнями методів та засобів виявлення небезпечних ситуацій комп'ютеризованими системами відеонагляду є актуальною та продовжує розвиватися, пропонуючи нові ефективні рішення, які можуть значно підвищити рівень безпеки у різних сферах життя

## 1.2. Огляд та аналіз методів і систем для виявлення небезпечних ситуацій

Дослідженням методів та систем для виявлення небезпечних ситуацій присвячено роботи цілого ряду науковців, зокрема, у роботах [1,5-13] описані системи, які дозволяють виявляти аномальну поведінку та різного роду небезпечні ситуації.

В [5] досліджується різноманітна аномальна поведінка людини та розпізнавання цієї поведінки за допомогою моделей на основі CNN з різними загальнодоступними та приватними наборами даних, розміром вхідного зображення, показниками продуктивності та обмеженнями.

Авторами в [5] аномальна поведінка людини прокласифікована відповідно до різних подій та описано методи розпізнавання таких її видів як: бездіяльність, падіння, зчинення насильства, паніка, пограбування, поведінка злочинця. Зокрема, представлено методи вилучення просторових і часових характеристик, різноманітні набори даних, наведено точність і обмеження методів із різною роздільною здатністю зображення. Також зауважено, що через коливання освітлення та накладання об'єктів здійснюється розпізнавання аномальної поведінки на віддалених зображеннях з низькою точністю. Загальна архітектура моделі розпізнавання поведінки людини з різними технічними компонентами показана на рис. 1.1 [5].

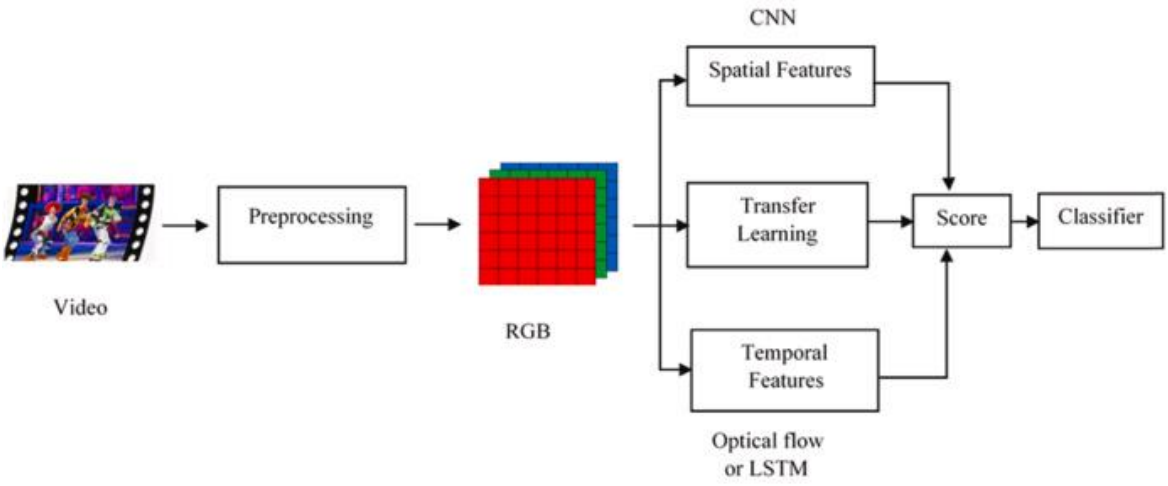


Рис. 1.1 Загальна структурна схема моделі розпізнавання поведінки людини [5]

У роботі [6] авторами запропонована модель, яка використовує ШІ для виявлення на зображеннях з камер відеонагляду об'єктів і дій людини, які могли б спровокувати небезпечну ситуацію, а також взаємозв'язок між ними. Зокрема, описано мережу згорткової довготривалої короткочасної пам’яті (Conv-LSTM) для виявлення аномалій, яка передбачає майбутні кадри шляхом реконструкції зображення. Цей метод демонструє конкурентоспроможність порівняно з іншими найсучаснішими методами виявлення небезпек. У [6] пропонується модель глибокого навчання для виявлення небезпеки, яка одночасно виконує класифікацію зображень і виявлення об’єктів, може підвищити точність виявлення небезпечної ситуації. Це пояснюється тим, що навіть якщо ті самі об’єкти виявляються на різних зображеннях камер відеонагляду, то ситуація відрізняється залежно від розташування об’єктів. Наприклад, якщо на зображеннях камер відеоспостереження виявлено “ніж”, ситуація, в якій людина тримає ніж, і ситуація, в якій ніж лежить на підлозі, відрізняються. Тому доцільно враховувати зв’язок (тобто відстань і розташування) для кожного виявленого небезпечного об’єкта.

На рис. 1.2. показано чотири класи даних, які використано для виявлення небезпечних ситуацій: (а) безпечно; (б) небезпечний інструмент; (в) потенційний злочин; (г) небезпека [6].

Рис. 1.2. Чотири класи даних виявлення небезпечних ситуацій: (а) безпечно; (б) небезпечний інструмент; (в) потенційний злочин; (г) небезпека [6]

На рис. 1.3 показано модель, яка використовує ШІ для виявлення об'єктів і дій людини [6].

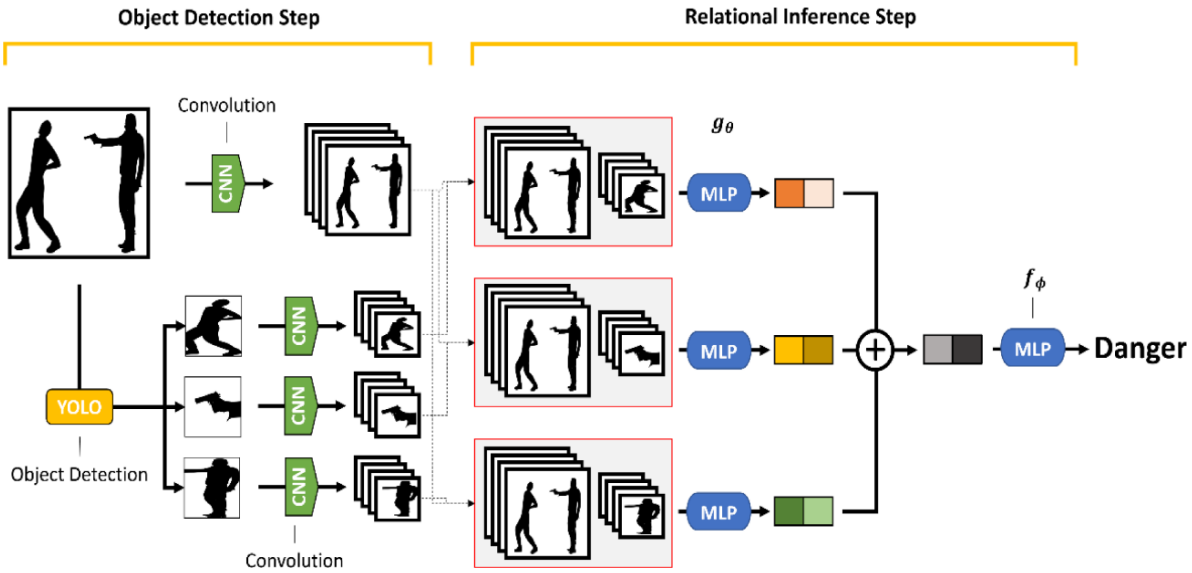


Рис. 1.3. Модель, яка використовує ШІ для виявлення об'єктів і дій людини [6]

У [7] запропоновано систему виявлення аномальної поведінки в реальному світі, яка використовує глибоке навчання. Цей метод моделює нормальні та аномальні події як пакети “екземплярів”, навчає ці “екземпляри” за допомогою моделі глибокого ранжирування аномалій і передбачає високі оцінки аномалій для аномальних сегментів відео. В даному випадку аномальна поведінка виявляється шляхом виділення ознак, пов’язаних із кількістю змін дій у всьому кадрі. Авторами у [8] запропоновано виявляти аномалії руху у відеоспостереженнях за допомогою графіка асоціації регіонів (RAG). Автори отримали статистичні ознаки з даної сцени та класифікували два типи аномалій за допомогою алгоритму SVM. Результати експерименту з використанням двох контрольних наборів даних показали ефективність запропонованого методу.

При підході розпізнавання об’єктів для виявлення небезпеки використовують методи виявлення небезпечних об’єктів і дій із зображень із камер відеонагляду. В [9] запропоновано простий метод, який виявляє “ножі”, “кров” і “зброю”, як предмети,

що тісно пов'язані зі злочинністю. Для цієї мети автори реалізовують CNN, який містить нелінійність Rectified Linear Unit (ReLU), шар згортки, повністю зв’язаний шар і випадання. Їх модель виявлення об'єктів демонструє точність тестування 90,2%. Існує також кілька проблем, пов'язаних з виявленням об'єктів на місці злочину. Наприклад, є можливість оклюзії при відстеженні об'єктів, що може значно знизити розкриття злочинів.

У [10] запропоновано використання дронів для спостереження за небезпечними ситуаціями в громадських місцях. Виявлення таких ситуацій здійснюється шляхом визначення п’яти насильницьких дій, тобто; ударів руками, ножем, пострілів, ударів ногами та удушення. Для цієї мети використано мережу піраміди функцій для виявлення людини та гібридну мережу глибокого навчання ScatterNet для оцінки пози кожної виявленої людини [10].

Однією з проблем, яка виникає при виявленні небезпечних ситуацій є те, що підходи до класифікації зображень і виявлення об’єктів сильно залежать від продуктивності кожного методу. Наприклад, у підходах до класифікації зображень, якщо роздільна здатність камер відеонагляду є низькою, дані, отримані через зображення, можуть бути неякісні, і, отже, точність візуального спостереження стає низькою. У підходах до виявлення об’єктів, якщо системі не вдасться виявити об’єкт, який визначено як небезпечний, то система не може сформувати попередження про небезпеку, тому що не розпізнає наявність небезпечного об’єкта на зображенні. Тому моделі виявлення об'єктів повинні мати високу точність. Модель виявлення об’єктів може забезпечити швидкість у реальному часі з низькою точністю порівняно з іншими моделями. Крім того, продуктивність виявлення об’єктів зменшується з роздільною здатністю зображення [11]. Однак модель повинна одночасно виробляти високу швидкість і точність для автоматичного виявлення небезпеки.

У роботі [12] була запропонована система, що дозволяє виявляти небезпечні ситуації, такі як падіння людини, на основі визначення пози. Загальна структура системи складається з кількох модулів. Зокрема, з трьох внутрішніх модулів опрацювання: детектор людей, який створює для кожного кадру список людей, присутніх на сцені; засіб оцінки пози, що використовується для обчислення поточної

пози для кожної людини, виявленої на попередньому кроці; детектор небезпечних ситуацій, здатний ідентифікувати аномальні ситуації за допомогою часового аналізу пози людей і генерувати відповідну тривогу. Модуль оцінки постави забезпечує класифікацію пози кадр за кадром. У [12] жодна попередня інформація не використовується для визначення поточної пози, і розглянуто лише чотири основні пози: стоячи, сидячи, присівши і лежачи. Історія розташування використовується в модулі детектора небезпечних ситуацій. Передбачається, що спочатку людину виявляють у положенні стоячи (у приміщенні люди часто заходять у кімнати стоячи). Система [12] була розроблена з урахуванням обмежень у реальному часі і можливостей опрацювати орієнтовно 10 кадрів за секунду .

На рис.1.4. показано результати, отримані в різних контекстах для розпізнавання падіння [12].

| Luca1 | Luca2 | Roberto1 | Roberto2 |
|---|---|---|---|
|  |  |  |  |
| 1209 frames | 735 frames | 294 frames | 416 frames |
| 320 x 240 | 320 x 240 | 360 x 288 | 384 x 288 |

Рис. 1.4. Результати, отримані в різних контекстах для розпізнавання падіння [12]

У статті [13] представлено опис різних систем та технологій автоматизації безпеки. Зокрема авторами проаналізовано різні системи, такі як інтелектуальні системи акустичного та вібророзпізнавання/оповіщення для виявлення порушень безпеки, ідентифікації небезпеки поблизу та захисту периметра, система домашньої безпеки з використанням Arduino UNO, детектор вторгнень та тампера на основі акселерометра малої потужності, система виявлення зловмисників на основі IoT за допомогою GSM, ідентифікація шкідливих дій для охорони периметра,

інтелектуальна система безпеки IoT для “розумного дому” з використанням детекції руху та обличчя, система розпізнавання, Ambient Intelligence та IoT Decision Support System для виявлення зловмисників, система виявлення крадіжок з використанням PIR-датчика, Ambient Intelligence та IoT Decision Support System для виявлення зловмисників, система виявлення крадіжки з використанням PIR-датчика та інші.

Отже, проведений аналіз підтверджує актуальність досліджень у напрямку вдосконалення методів та засобів виявлення небезпечних ситуацій комп'ютеризованими системами відеонагляду з використанням штучного інтелекту. Незважаючи на напрацювання цілого ряду науковців щодо проблеми виявлення небезпечних ситуацій для людей, використовуючи різні методи та засоби, все ще залишається важливою проблема точності та швидкості розпізнавання комп’ютеризованою системою такої ситуації. Для її вирішення в цьому дослідженні використано ідею, що ідентифікація небезпечної ситуації здійснюється за позою людини «руки вгору».

## 1.3. Архітектура комп'ютеризованої системи для виявлення небезпечних ситуацій

Узагальнена архітектура комп'ютеризованої системи виявлення небезпечних ситуацій, в якій ідентифікація небезпеки здійснюється за допомогою розпізнавання пози людини "руки вгору", представлена на рис. 1.5. КСВ, зокрема, її узагальнена архітектура, обгрунтування методу визначення небезпечної ситуації (повідомлення про напад зловмисників чи ін.), а також отримані результати прогнозу описані в статті [1].

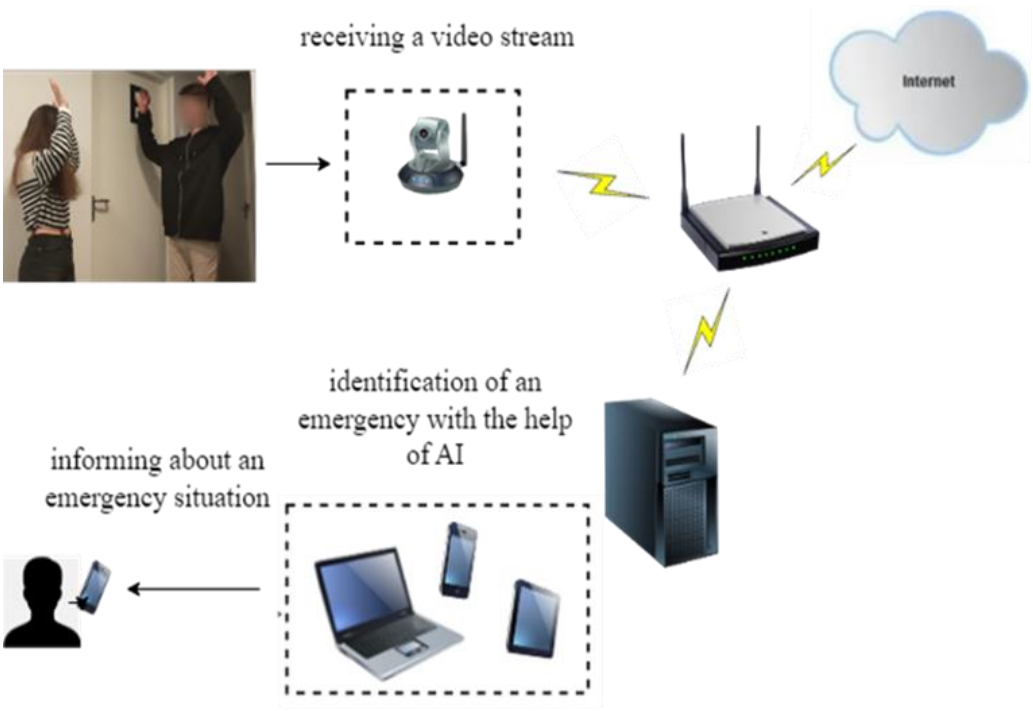


Рис. 1.5. Узагальнена архітектура КСВ для виявлення небезпечних ситуацій [1]

В цій системі для виявлення нападу запропоновано здійснити розпізнавання пози “руки вгору”, використовуючи 6-ть ключових точок в моделі PoseNet [1]. Ці точки показано на рис. 1.6.

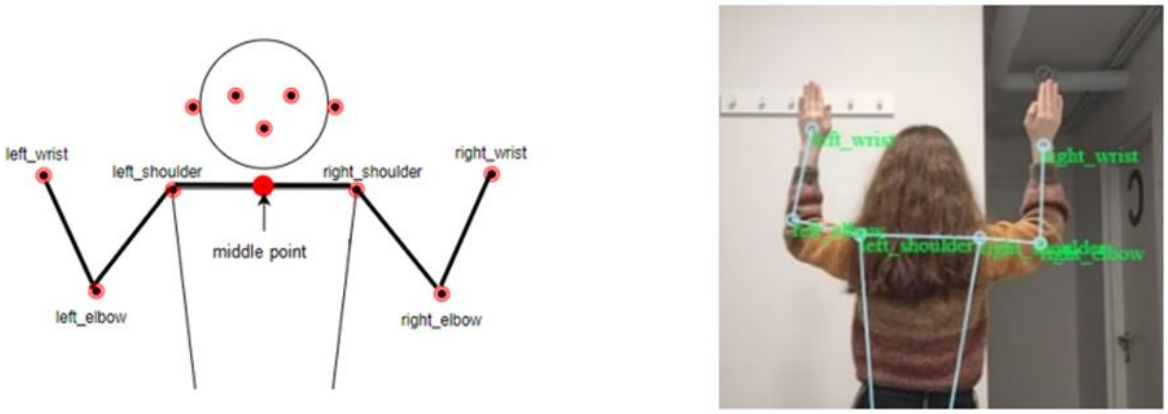


Рис. 1.6. Візуалізація 6-ти ключових точок, отриманих за допомогою моделі PoseNet для виявлення небезпечної ситуації

Для дослідження [1] було створено із 1510 зображень набір даних для навчання моделі ШІ PoseNet. Отримані результати розпізнавання - наведені в табл. 1.1, 1.2 [1].

*Таблиця 1.1*

**Результати розпізнавання пози [1]**

| Назва методу класифікації | Поза «руки вгору» | | Інші пози | |
|---|---|---|---|---|
| | всього: 930 | | всього: 580 | |
| | правильно класифіковано | неправильно класифіковано | правильно класифіковано | неправильно класифіковано |
| SVM з використанням ядра Гауса | 930 | 0 | 580 | 0 |
| Логістична регресія | 856 | 74 | 525 | 55 |
| Наївний баєсів класифікатор | 862 | 68 | 504 | 76 |
| Дискримінантний аналіз | 851 | 79 | 526 | 54 |
| Метод k-найближчих сусідів | 862 | 68 | 504 | 76 |

*Таблиця 1.2*

**Результати прогнозування розпізнавання пози [1]**

| Назва класифікатора | RECALL | PRECISION |
|---|---|---|
| SVM з використанням ядра Гауса | 100 % | 100 % |
| Логістична регресія | 94.0 % | 92.0 % |
| Наївний баєсів класифікатор | 91.9 % | 92.7 % |
| Дискримінантний аналіз | 94.0 % | 91.5 % |
| Метод k-найближчих сусідів | 91.9 % | 92.7 % |

Цей підхід дозволяє зменшити обсяг обчислювальних ресурсів, необхідних для опрацювання відеопотоку, і забезпечити високу точність розпізнавання, але недоліком використання моделі PoseNet є точність розпізнавання ключових точок (суглобів), якщо людина не знаходиться безпосередньо перед камерою.

Варто відмітити, що для швидкого реагування та інформування про небезпеку, системі немає необхідності розпізнавати інші пози. Загалом, передбачено, що користувачам запропонованої в [1] системи, тобто потенційній “жертві”, є відомо, що при небезпеці потрібно прийняти позу “руки вгору” і знайти можливість стояти в полі видимості камери відеонагляду, що є певним обмеженням використання системи.

Отже, необхідність покращення існуючих методів виявлення небезпечних ситуацій КСВ зумовлена такими факторами як: швидкість та ефективність роботи в реальному часі, а також підвищення точності розпізнавання у складних умовах.

## 1.4. Висновки до розділу 1

В результаті проведеного аналітичного огляду літератури встановлено, що комп'ютеризовані системи відеонагляду (КСВ) широко використовуються для забезпечення безпеки шляхом виявлення та прогнозування небезпечних ситуацій у публічних та приватних місцях.

Виявлено, що найперспективнішим інструментом для виявлення небезпечних ситуацій є ШІ, зокрема технології машинного навчання та згорткові нейронні мережі (CNN).

На основі аналізу існуючих рішень встановлено, що для розпізнавання людини використовуються різні типи CNN (2D CNN і 3D CNN), які здатні ідентифікувати об'єкти з послідовності кадрів з високою точністю.

Встановлено основні недоліки існуючих систем, а саме: складність опрацювання даних у реальному часі, необхідність адаптації систем до роботи у різних умовах, залежність точності розпізнавання від якості зображення та роздільної здатності камер, проблеми з розпізнаванням при низькій освітленості та накладанні об'єктів.

На основі проведеного аналізу визначено необхідність подальшого вдосконалення існуючих методів виявлення небезпечних ситуацій для забезпечення підвищення швидкості та ефективності роботи в реальному часі, а також точності розпізнавання у складних умовах.

# РОЗДІЛ 2
# ТЕОРЕТИЧНА ЧАСТИНА

## 2.1. Обгрунтування вибору типу моделі для виявлення небезпечних ситуацій

В даний час існують можливості для подальшого підвищення ефективності та адаптації КСВ до різних умов. Однією з перспективних моделей, яка може бути використана для виявлення небезпечних ситуацій, які ідентифікуються піднятими руками, є YOLO [14].

Протягом останніх років для забезпечення виявлення об'єктів у реальному часі дослідники зосереджувалися на розробці детекторів на основі згорткових нейронних мереж (CNN). Серед яких алгоритм You Only LookOnce(YOLO), представлений на CVPR у 2016р. Джозефом Редмоном разом із дослідниками з FAIR(Facebook AI Research) та Інституту ШІ Аллена, здобув широку популярність завдяки своєму збалансованому підходу до продуктивності та ефективності [15]. Тому виявлення небезпеки для людини КСВ за позою “підняті руки” доцільно здійснювати шляхом використання таких моделей, як YOLO.

Уся серія моделей YOLO — це сукупність новаторських концепцій та інженерних інновацій, які сформували сучасні методи виявлення об’єктів. Вона пропонує численні прикладні ідеї машинного навчання, які можна використовувати в конкретних сценаріях. На відміну від звичайних методів, що використовують двоетапні процеси виявлення, YOLO розглядає виявлення об'єктів як проблему регресії [15]. У парадигмі YOLO єдина згорткова нейронна мережа використовується для прогнозування обмежувальних рамок та визначення ймовірностей класів для всього зображення. Цей спрощений підхід відрізняється від традиційних методів більш складними конвеєрами[16].

Моделі YOLO досягають високої точності виявлення з мінімальними обчислювальними вимогами. Деякі моделі YOLO адаптовані до конкретних можливостей опрацювання даних пристроєм. Більшість моделей YOLO розроблено

для різних масштабів, таких як малі, середні та великі, які можна легко серіалізувати в ONNX, TensorRT, OpenVINO тощо [14, 15]. Це дає користувачам можливість підібрати найоптимальнішу для їх застосування.

Архітектура YOLO представила наскрізний, диференційований підхід до виявлення об’єктів використовуючи уніфікацію завдань регресії обмежувальної рамки та класифікації об’єктів в єдину нейронну мережу [15]. По суті, YOLO складається з трьох основних компонентів: хребет, шия, голова. Хребет витягує просторові та семантичні характеристики з вхідного зображення за допомогою згорткових шарів. Шия агрегує багатомасштабні елементи та покращує їхнє представлення. Загальні механізми включають мережі піраміди функцій (FPN), мережі агрегації шляхів (PAN) або рівень об’єднання просторових пірамід (SPP). PAN (мережі агрегації шляхів) використовується в YOLOv10. А голова виводить координати обмежувальної рамки, оцінки достовірності та прогнози класу. Наприклад, YOLOv10 має голови «один-до-одного» та «один-до-багатьох», які служать прогнозом для найкращих і для всіх можливих обмежувальних рамок для того самого об’єкта (служачи альтернативою NMS (немаксимальне придушення), використовуваному в попередніх версіях, щоб забезпечити найкращу відповідність і єдину обмежувальну рамку, яка посилається на об’єкт [17].

Хребет, згорткова нейронна мережа, відповідає за кодування інформації про зображення в карти функцій у різних масштабах. Потім ці карти об’єктів опрацьовуються шиєю, серією шарів, призначених для інтеграції та вдосконалення представлень об’єктів. Нарешті, головний модуль генерує прогнози для обмежувальних рамок об’єктів і міток класів на основі опрацьованих функцій.

За своєю суттю архітектура YOLO складається з трьох основних компонентів. По-перше, хребет служить основним екстрактором ознак, використовуючи згорточні нейронні мережі для перетворення необроблених даних зображення в багатомасштабні карти функцій [14, 15].

По-друге, компонент шиї діє як проміжний етап опрацювання, використовуючи спеціалізовані шари для агрегування та покращення репрезентації ознак у різних масштабах. По-третє, головний компонент функціонує як механізм прогнозування,

генеруючи остаточні результати для локалізації та класифікації об’єктів на основі вдосконалених карт ознак.

Основний процес виявлення є незмінним у всіх моделях: обмежувальні рамки прогнозуються за допомогою опорних рамок, які є попередньо визначеними прямокутними формами, що представляють типові пропорції та розміри об’єктів. Далі процес включає зіставлення прив’язки - кожна клітинка сітки на карті об’єктів пов’язана з декількома прив’язками. Модель призначає базовий об’єкт істинності опорному блоку, який найкраще відповідає його розмірам, використовуючи порогове значення перетину через з’єднання (IoU). А також передбачає зміщення відносно координат рамки прив’язки, для точного налаштування розташування та розмір обмежувальної рамки. Зокрема, він передбачає коригування координат центру, ширини та висоти. під час навчання для оптимізації прогнозованих обмежувальних рамок обчислюються комбінація втрат: локалізації (наприклад, CIoU або GIoU), достовірності та класифікації [14, 16, 18].

Процес виявлення об'єктів YOLO показано на рис. 2.1 [18].

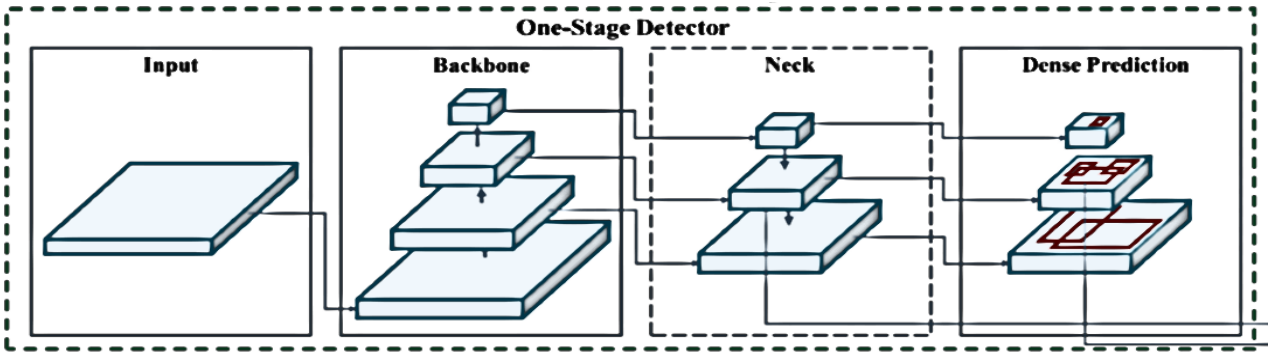


Рис. 2.1. Процес виявлення об'єктів YOLO [18]

На рис. 2.2 показано порівняння параметрів точності та швидкості різних моделей сімейства YOLO TensorRT FP16 на GPU T4 на наборі даних COCO [14].

Еталонні показники надають інформацію про розмір експортованого формату, його показники mAP (середня точність), а також час визначення в мілісекундах на зображення в форматі TensorRT. Ця інформація може допомогти вибрати оптимальний формат експорту для конкретного випадку використання на основі їхніх

вимог до швидкості та точності. Середня точність mAP є однією з найважливіших метрик для оцінки моделей об'єктного детектування і визначається площею під лініями на позначення влучності та запам’ятовування [14]. mAP є узагальненою мірою, яка дозволяє оцінити, наскільки добре модель справляється з усіма класами і, обчислюється як середнє значення AP для всіх класів (див.формули 2.1-2.2.).

$$\mathrm{mAP} = \frac{1}{N} \sum_{i=1}^{N} \mathrm{AP}_i \tag{2.1}$$

де $N$ — загальна кількість класів, AP$i$ — середня точність (Average Precision) для класу $i$.

$$AP = \int_0^1 p(r)\, dr \tag{2.2}$$

де $p(r)$ — залежність точності (precision) від повноти (recall). Precision вказує на відсоток коректно передбачених об'єктів серед усіх передбачених (істинно позитивні/(істинно позитивні + хибно позитивні)), а recall - на відсоток коректно передбачених об'єктів серед усіх реальних об'єктів (істинно позитивні/(істинно позитивні + хибно негативні)).

Формула (2.2) відображає площу під кривою Precision-Recall (PR). Варто зауважити, що модель об'єктного детектування прагне до високих значень як precision, так і recall, а крива PR показує компроміс між точністю (precision) і повнотою (recall). Необхідно відмітити, що чим більша площа, тим краща продуктивність моделі для конкретного класу.

На рис. 2.2. наведено порівняння параметрів точності та швидкості різних моделей сімейства YOLO [14].

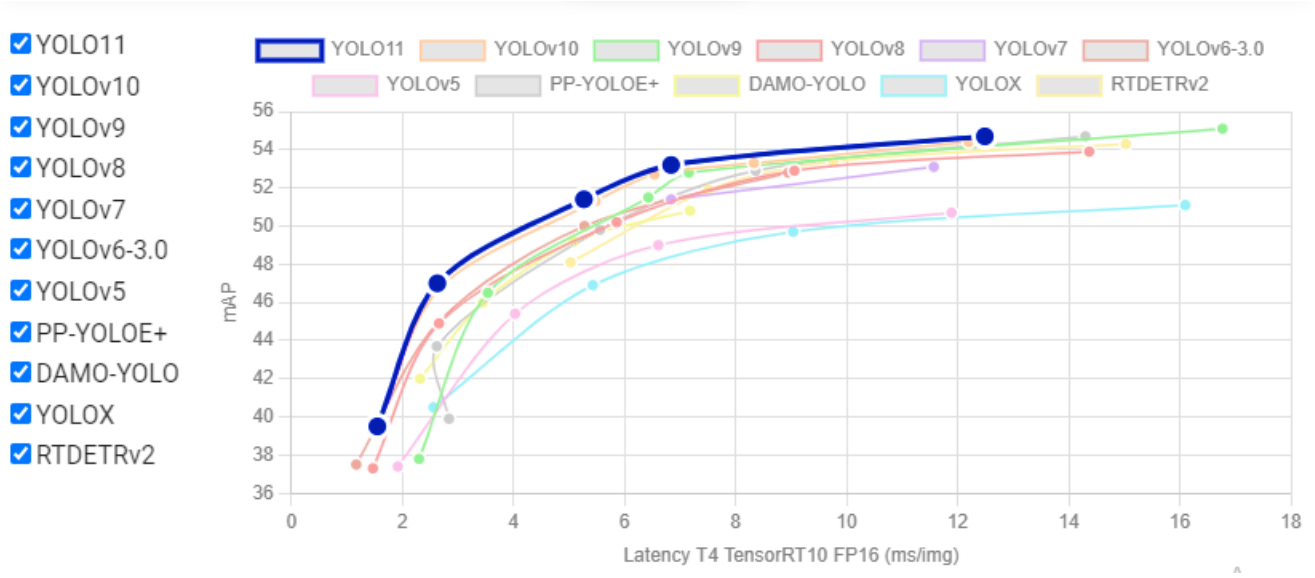


Рис. 2.2. Порівняння параметрів точності та швидкості різних моделей сімейства YOLO [14]

Моделі YOLO широко та успішно використовуються у сфері комп'ютерного зору, тому дослідники вдосконалюють існуючі моделі, додають нові модулі до методу, На сьогоднішній день вже є 11 версія YOLO.

В табл. 2.1 наведено порівняльний аналіз методів виявлення небезпечних ситуацій з використанням моделей PoseNet та YOLO.

*Таблиця 2.1*

**Порівняння моделей PoseNet та YOLO для виявлення небезпечних ситуацій**

| Критерій | Модель PoseNet | Модель YOLO |
|---|---|---|
| 1 | 2 | 3 |
| Швидкість та ефективність | Ефективна, але обмежена у складних сценаріях | Забезпечує швидше опрацювання в реальному часі, що дозволяє реагувати на загрози оперативно |

*Продовження таблиці 2.1*

| 1 | 2 | 3 |
|---|---|---|
| Багатооб'єктна ідентифікація | Орієнтована на розпізнавання поз однієї людини | Дозволяє ідентифікувати та відслідковувати кілька об'єктів одночасно, що підвищує точність у багатолюдних місцях |
| Гнучкість і розширення функціональності | Обмежена здатність розпізнавати інші важливі аспекти поведінки | Може розпізнавати широкий спектр об'єктів та дій, що робить систему більш універсальною. |
| Точність у складних умовах | Може мати обмежену ефективність у складних умовах (низька якість зображення, погане освітлення) | Розроблена для роботи в умовах зниженої видимості, забезпечуючи кращу точність. |

Таким чином, покращення існуючих методів виявлення небезпечних ситуацій шляхом використання та впровадження в КСВ таких моделей, як YOLO, може значно підвищити швидкість, точність, стійкість і універсальність системи відеонагляду. Це дозволить краще адаптувати систему до різноманітних умов і сценаріїв, підвищуючи її здатність забезпечувати безпеку в режимі реального часу.

З метою аналізу та вибору найоптимальнішого методу виявлення КСВ небезпечних ситуацій для людей за піднятими вверх руками досліджено моделі YOLOv8, YOLOv10 та YOLOv11.

## 2.2. Модель YOLOv8

YOLOv8 — це вдосконалена та передова модель із можливостями семантичної сегментації та виявлення, яка пропонує вищу точність і швидкість виявлення,

випущена компанією Ultralytics. Архітектура YOLOv8 відповідає тій самій архітектурі, що й YOLOv5, з кількома невеликими коригуваннями, такими як використання модуля c2f замість модуля CSPNet [14,15]. Архітектура YOLOv8 в основному складається з хребта, шиї та голови, як показано на рис.2.3 [15].

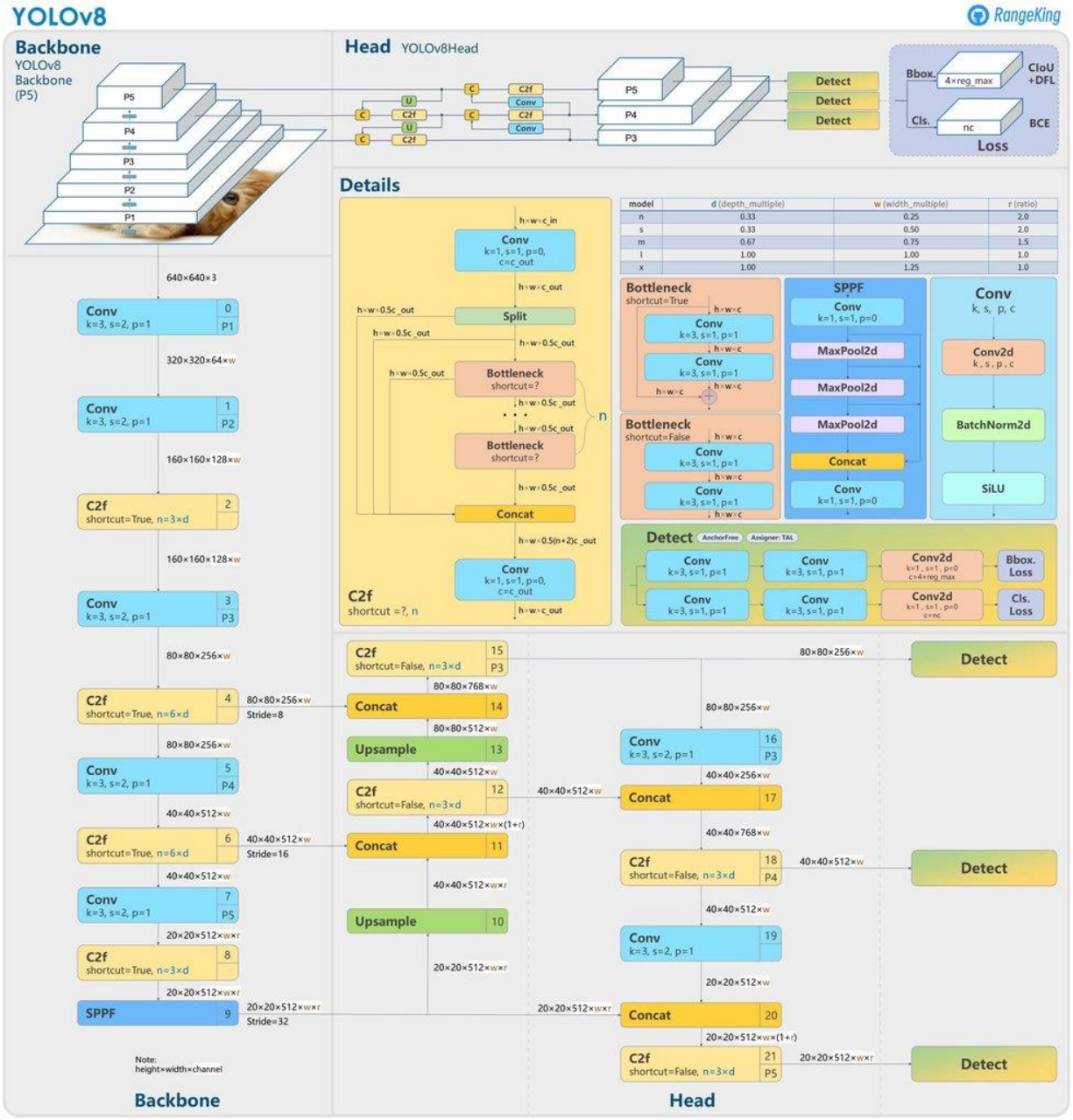


Рис. 2.3. Архітектура YOLOv8 [15]

YOLOv8 використовує функції втрат CIoU та DFL для втрат обмежувальної рамки та бінарну крос-ентропію для втрат класифікації. Ці втрати покращили ефективність виявлення об’єктів, особливо при роботі з меншими об’єктами.

YOLOv8 також високоефективний і гнучкий, підтримує численні формати експорту, і модель може працювати на графічних і центральних процесорах [14, 15].

Доступні моделі YOLOv8: YOLOv8nano(YOLOv8n), YOLOv8small(YOLOv8s), YOLOv8medium(YOLOv8m), YOLOv8medium(YOLOv8m) та YOLOv8 extra large. У кожній категорії є моделі для виявлення, класифікації та сегментації. Найшвидшою і найменшою є YOLOv8 nano, а YOLOv8 extra large (YOLOv8x) - є найточнішою, але і найповільнішою. Останнє сімейство моделей YOLOv8 також включає моделі оцінки пози, які можуть виявляти ключові точки людини з високою точністю.

## 2.3. Модель YOLOv10

YOLOv10 — це одна із найновіших версій моделі для виявлення об'єктів в реальному часі, розроблена дослідниками з Університету Цінхуа [17]. Вона є значним покращенням попередніх моделей YOLO, забезпечуючи високу точність і ефективність. Ключовими особливостями YOLOv10 є відмова від NMS, покращена архітектура, висока продуктивність та гнучкість [14, 17].

Процес виявлення YOLO складається з двох частин: основного обчислення моделі та постобробка NMS(noninear maximum compression - нелінійне пригнічення максимумів). Проте обидві частини все ще мають недоліки, що впливають на оптимальність співвідношення точність/затримка. YOLO використовує стратегію навчання "один-до-багатьох", де кожен реальний об’єкт відповідає декільком позитивним взірцям. Хоча цей підхід і забезпечує високу точність, він потребує використання NMS для вибору найкращого прогнозу. Це уповільнює швидкість прогнозування та робить продуктивність чутливою до налаштувань NMS, що заважає оптимальному "кінець-до-кінця" впровадженню YOLO. Вирішення цієї проблеми здійснено завдяки використанню підходів, які наслідують архітектуру DETR(Detection Transformers), а також дослідженню методів прямого "кінець-до-кінця" виявлення для детекторів на основі CNN. Проте ці підходи часто або додають

додаткові витрати на прогнозування, або забезпечують субоптимальну продуктивність для YOLO [14,17].

У процесі навчання YOLO зазвичай використовують стратегію "один-до-багатьох" для призначення міток, коли одному об’єкту відповідає кілька позитивних вибірок. Такий підхід забезпечує багатий набір навчальних сигналів, сприяючи оптимізації та досягненню високої продуктивності. Але вимагає застосування NMS під час інференсу, що уповільнює роботу моделі.

Для розв’язання цієї проблеми використано стратегію навчання YOLO без NMS із використанням подвійного призначення міток та узгодженої метрики відповідності. Це дозволяє моделі отримувати насичені та узгоджені навчальні сигнали, усуваючи потребу в NMS під час інференсу.

Замість стратегії "один-до-багатьох" стратегія "один-до-одного" передбачає відповідність лише одного прогнозу кожній еталонній мітці. Хоча це усуває потребу в NMS, такий підхід може спричинити слабке навчання через обмежені навчальні сигнали. Для подолання цього недоліка здійснено інтегрування обидвох стратегій. Для цього вводиться додаткова головна частина моделі, яка використовує стратегію "один-до-одного". Під час навчання обидві частини моделі оптимізуються спільно, що забезпечує збагачене навчання для всієї архітектури. Під час прогнозування використовується лише головна частина, що працює за принципом "один-до-одного", для усунення дублювань у прогнозах [14,17].

Для забезпечення узгодженості між двома стратегіями, використано єдину метрику для оцінки відповідності між прогнозами та еталонами. Ця метрика включає оцінку класифікації, рівень перекриття (IoU) та просторовий пріоритет. Узгодження між цими компонентами дозволяє оптимізувати обидві частини моделі в гармонійному напрямку.

Окрім постобробки, архітектура YOLO також має значний вплив на співвідношення продуктивності та ефективності. Тому запропоновано холістичний підхід до оптимізації компонентів моделі як з точки зору ефективності, так і точності.

YOLOv10 модель не потребує постобробки з використанням NMS, тому має меншу затримку та підвищену швидкість опрацювання. YOLOv10 використовує

вдосконалену версію CSPNet(Cross Stage Partial Network) для покращення потоку градієнтів і зменшення обчислювальних витрат. В шиї моделі є шари PAN(Path Aggregation Network), що ефективно об'єднують властивості різних масштабів. Також модель досягає значних покращень у точності і швидкості порівняно з попередніми версіями. Наприклад, YOLOv10s в 1,8 рази швидше RT-DETR-R18 при схожій середній точності (AP) на наборі даних COCO. YOLOv10l - має на 46% меншу затримку та на 25% менше параметрів, ніж YOLOv9s із такою ж продуктивністю[14].

Ці покращення роблять YOLOv10 одним із потужних інструментів для застосувань швидкого і точного розпізнавання об'єктів, що є важливим для виявлення небезпечних ситуацій КСВ.

Архітектура YOLOv10 базується на сильних сторонах попередніх моделей YOLO, одночасно впроваджуючи кілька ключових інновацій. Архітектура моделі складається з таких основних структурних компонентів моделі: хребет, шия, голова «один-до-багатьох», голова «один-до-одного» [17].

Хребет у YOLOv10, що відповідає за виділення функцій, використовує вдосконалену версію CSPNet (часткова мережа перехресних етапів) для покращення градієнтного потоку та зменшення надлишковості обчислень. Шия призначена для агрегування рис різних масштабів і передачі їх до голови. Він містить шари PAN (мережі агрегації шляхів) для ефективного багатомасштабного об’єднання функцій. Голова «один-до-багатьох» генерує кілька прогнозів для кожного об’єкта під час навчання, щоб надати розширені контрольні сигнали та підвищити точність навчання, а голова «один-до-одного» генерує єдиний найкращий прогноз для кожного об’єкта під час висновку, щоб усунути потребу в NMS, тим самим зменшуючи затримку та підвищуючи ефективність.

Архітектура YOLOv10 представлена на рис. 2.3 [17].

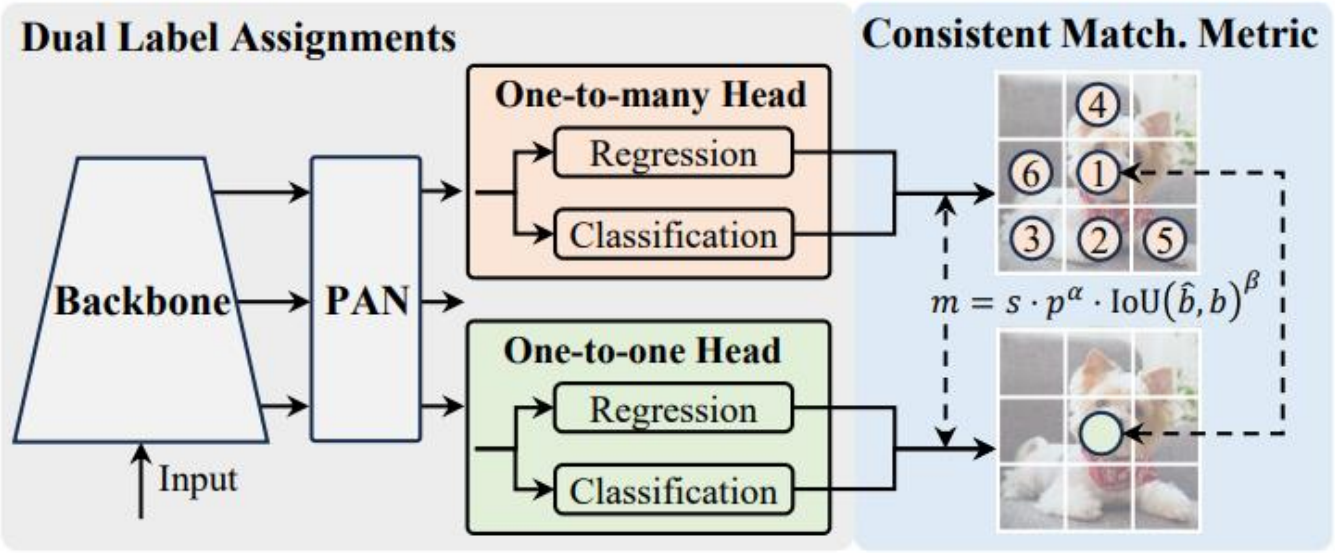


Рис. 2.3. Архітектура YOLOv10 [17]

Послідовні подвійні призначення для навчання без використання NMS YOLOv10 використовує подвійні призначення міток, поєднуючи стратегії «один-до-багатьох» і «один-до-одного» під час навчання, щоб забезпечити повний контроль і ефективне наскрізне розгортання. Послідовна метрика відповідності вирівнює спостереження між обома стратегіями, підвищуючи якість прогнозів під час логічного висновку.

Таким чином, YOLOv10 використовує послідовні подвійні призначення, щоб усунути потребу в NMS, зменшивши затримку, а комплексна оптимізація різних компонентів як з точки зору ефективності, так і з точки зору точності, включаючи полегшені класифікаційні голови, вибірку з розділенням просторових каналів і блочну конструкцію з ранжуванням, забезпечує цілісний дизайн моделі. Також вона має розширені можливості, оскільки включає згортки великого ядра та часткові модулі самоконтролю для підвищення продуктивності без значних обчислювальних витрат.

YOLOv10 доступна в різних варіантах (Nano, Small, Medium, Balanced, Large, Extra Large), що дозволяє вибрати оптимальний баланс між точністю та обчислювальними витратами в залежності від конкретних завдань. YOLOv10n - версія Nano для середовищ із надзвичайно обмеженими ресурсами, YOLOv10s -

невелика версія з балансуванням швидкості та точності, YOLOv10m - середня версія для загального використання, YOLOv10b - збалансована версія зі збільшеною шириною для більш високої точності, YOLOv10l - велика версія для підвищення точності за рахунок збільшення обчислювальних ресурсів, YOLOv10x - дуже велика версія для максимальної точності [14, 17].

YOLOv10 було ретельно протестовано на стандартних тестах, таких як COCO, що продемонструвало чудову продуктивність і ефективність. Ця модель забезпечує найсучасніші результати в різних варіантах, демонструючи підвищення швидкості та точності порівняно з попередніми версіями та іншими сучасними детекторами.

## 2.4. Модель YOLOv11

YOLOv11 є останньою ітерацією в серії Ultralytics YOLO детекторів об’єктів у реальному часі, яка була представлена на конференції YOLO - Vision 2024 (YV24). У цій новій версії суттєво вдосконалено як архітектуру, так і методику навчання, збільшуючи точність, швидкість та ефективність [19]. Це робить її універсальним вибором для широкого спектру різноманітних завдань комп’ютерного зору. Інноваційний дизайн YOLOv11 використовує вдосконалені методи вилучення функцій, що дозволяє охоплювати більш тонкі деталі, зберігаючи мінімальну кількість параметрів. Це призводить до підвищення точності та адаптивності у різних середовищах та широкий спектр підтримуваних завдань комп’ютерного зору, починаючи від виявлення об’єктів до класифікації.

Ultralytics YOLOv11 - це універсальна структура, розроблена для охоплення всього життєвого циклу моделей машинного навчання — від отримання даних і навчання моделі до перевірки, розгортання та відстеження у реальному світі. Кожен режим служить певній меті та створений для забезпечення гнучкості та ефективності, необхідних для різних завдань і випадків використання.

Конструкція YOLOv11 зосереджена на балансі потужності та практичності, спрямованому на вирішення конкретних завдань у різних галузях із підвищеною точністю та ефективністю. Спираючись на усталену архітектуру, YOLOv11

розширює та вдосконалює основу, закладену YOLOv8, запроваджуючи архітектурні інновації та оптимізуючи параметри для досягнення високої продуктивності виявлення, як показано на рис. 2.4 [14, 19].

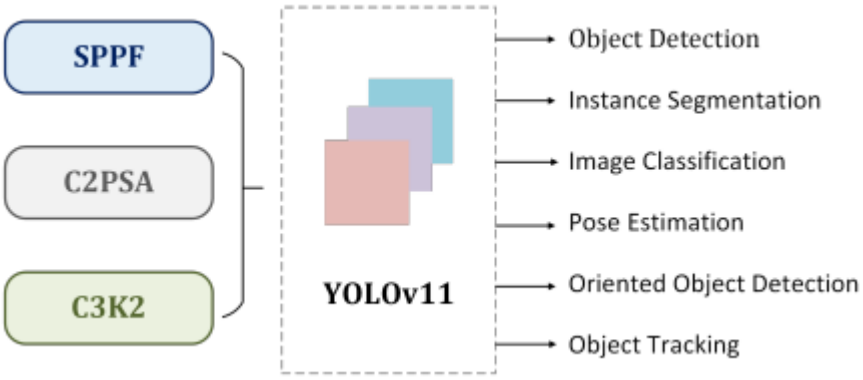


Рис. 2.4. Основні архітектурні компоненти YOLOv11[19]

YOLOv11 використовує вдосконалену архітектуру хребта та шиї, що покращує можливості вилучення функцій для більш точного виявлення об’єктів і виконання складних завдань. Ця модель представляє вдосконалені архітектурні конструкції та оптимізовані конвеєри навчання, забезпечуючи більш високу швидкість опрацювання та підтримуючи оптимальний баланс між точністю та продуктивністю. Завдяки вдосконаленню дизайну моделі YOLOv11m досягає вищої середньої точності (mAP) на наборі даних COCO, використовуючи на 22% менше параметрів, ніж YOLOv8m, що робить його обчислювально ефективним без шкоди для точності. YOLOv11 можна безперешкодно розгортати в різних середовищах, включаючи периферійні пристрої, хмарні платформи та системи, що підтримують графічні процесори NVIDIA, забезпечуючи максимальну гнучкість. Також YOLOv11 розроблено для вирішення різноманітних завдань комп’ютерного зору, чи то виявлення об’єктів, сегментація екземплярів, класифікація зображень, оцінка пози або орієнтоване виявлення об’єктів [14,19,20].

Архітектура YOLOv11 показана на рис.2.5 [20].

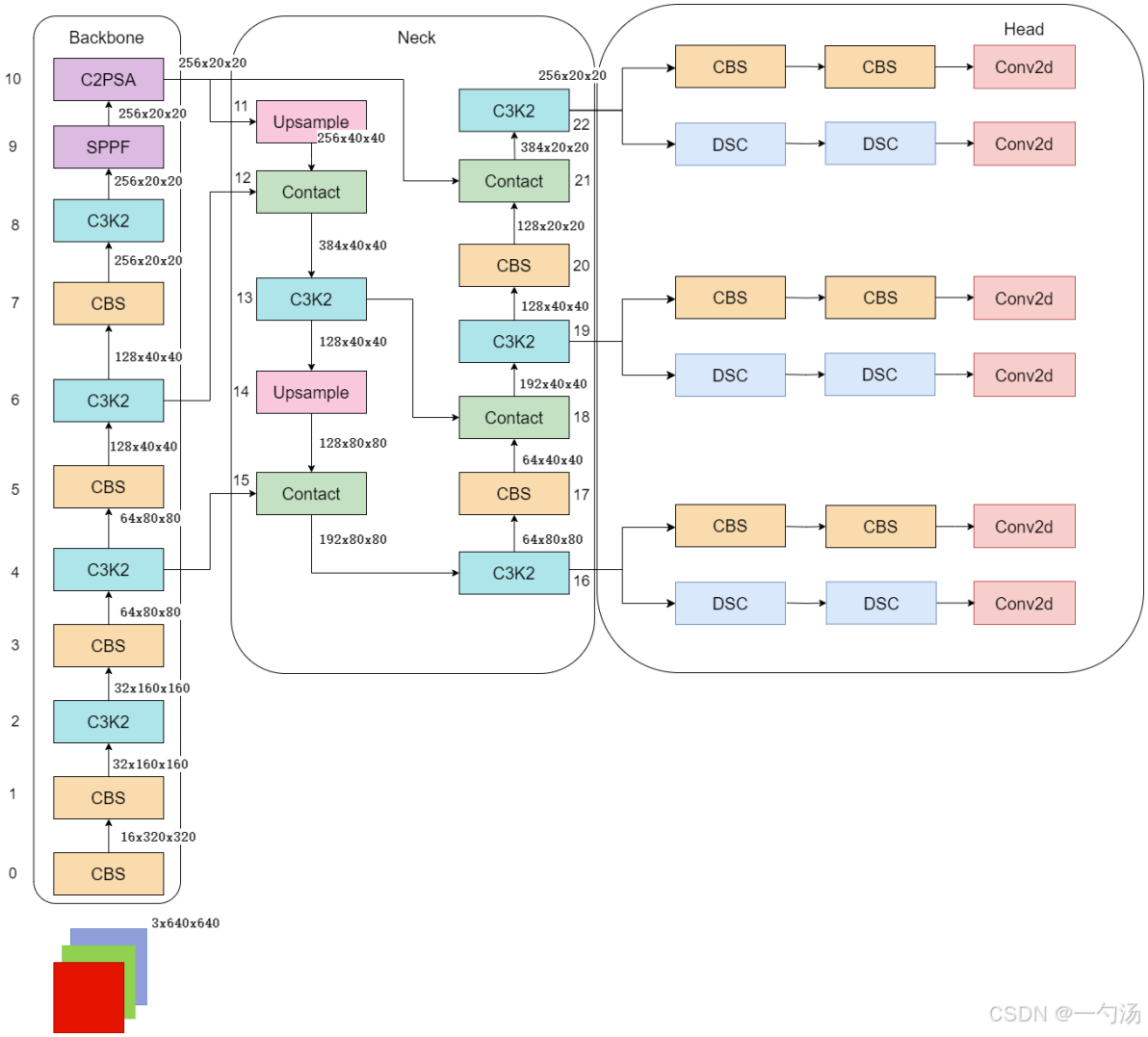


Рис. 2.5. Архітектура YOLOv11 [20]

Навчання моделі передбачає передавання даних і налаштування її параметрів. Режим навчання в Ultralytics YOLOv11 розроблений для ефективного та результативного навчання моделей виявлення об’єктів із повним використанням можливостей сучасного апаратного забезпечення. У цьому режимі модель навчається з використанням заданого набору даних і гіперпараметрів. Процес навчання передбачає оптимізацію параметрів моделі, щоб вона могла точно прогнозувати класи, розташування та величину об’єктів на зображенні.

Для перевірки моделі YOLOv11 після її навчання використовується режим Val (режим валідації або перевірки). У цьому режимі модель оцінюється на перевірочному наборі для вимірювання її точності та ефективності узагальнення на

“небаченій” вибірці. Цей режим можна використовувати для налаштування гіперпараметрів моделі для покращення її продуктивності. Режим тестування використовується для профілювання швидкості та точності різних форматів експорту для YOLOv11.

У машинному навчанні та комп’ютерному зорі процес визначення візуальних даних називається «висновком», «прогнозом» або «передбаченням» (англійською - inference). Режим висновку використовується для створення прогнозів за допомогою навченої моделі YOLOv11 на нових зображеннях або відео. У цьому режимі модель завантажується з файлу контрольної точки, і користувач може надати зображення або відео для виконання висновку. Модель передбачає класи та розташування об’єктів у вхідних зображеннях або відео [14,19,20].

Відстеження об’єктів у сфері відеоаналітики є критично важливим завданням, яке не тільки визначає розташування та клас об’єктів у кадрі, але й підтримує унікальний ідентифікатор для кожного виявленого об’єкта під час просування відео. Режим відстеження моделі YOLOv11 використовується для відстеження об’єктів у реальному часі. У цьому режимі модель завантажується з файлу контрольної точки, і користувач може надати живий відеопотік для здійснення відстеження об’єкта в реальному часі. Цей режим корисний для виявлення небезпених ситуацій КСВ.

Для експорту моделі YOLOv11 у формат, який можна використовувати для розгортання, передбачено режим експорту. У цьому режимі модель перетворюється у формат, який можна використовувати іншими програмними додатками або апаратними пристроями. Цей режим корисний під час розгортання моделі у виробничих середовищах [14,20].

Використання TensorRT для експорту моделей значно покращує продуктивність. Моделі YOLOv11 можуть досягнути 5-кратного прискорення графічного процесора за допомогою TensorRT і 3-кратного прискорення процесора за допомогою ONNX або OpenVINO. Це робить її придатною для використання в

реальному часі, особливо в системах відеонагляду з метою виявлення небезпечних ситуацій.

YOLOv11-pose - це потужна модель для оцінки пози, що є технікою комп'ютерного зору, і яка використовується для ідентифікації та визначення конкретних точок, відомих як ключові точки, на об'єктах або людях у зображеннях або відео. Ці ключові точки можуть представляти суглоби, орієнтири або інші характерні особливості. Для наборів даних оцінки пози формат Ultralytics YOLO передбачає позначення кожного зображення відповідним текстовим файлом [14]. Кожен рядок текстового файлу зберігає інформацію про екземпляр об’єкта:

- Індекс класу об'єктів.
- Координати центру об’єкта (нормалізовані x і y).
- Ширина і висота об'єкта (нормовані).
- Координати ключових точок об’єкта (нормалізовані pxn і pyn).

Для 2D поз ключовими точками є координати пікселів. Для 3D кожна ключова точка також має позначку видимості. В цій кваліфікаційній роботі проведено дослідження також і моделі YOLOv11-pose.

## 2.5. Висновки до розділу 2

Обгрунтовано використання YOLO, як ефективного засобу для виявлення небезпечних ситуацій у системах відеонагляду завдяки їх здатності працювати в реальному часі та високій точності виявлення об'єктів. У порівнянні з PoseNet, моделі YOLO забезпечують швидше опрацювання даних в реальному часі, кращу точність у складних умовах освітлення, більшу гнучкість та універсальність.

Проведено аналіз різних версій архітектури YOLO, зокрема YOLOv8, YOLOv10, YOLOv11 з метою вибору найефективнішої для задачі виявлення небезпечних ситуацій.

Підсумовано, що використання сучасних моделей YOLO в системах відеонагляду дозволяє суттєво підвищити ефективність виявлення небезпечних ситуацій завдяки їх високій швидкодії, точності та адаптивності до різних умов

роботи. Всі досліджені моделі підтримують експорт у різні формати (TensorRT, ONNX, OpenVINO), що забезпечує гнучкість при розгортанні та дозволяє досягти значного прискорення опрацювання (зокрема, для моделі YOLOv11 - до 5 разів на GPU та до 3 разів на CPU).

# РОЗДІЛ 3
# ПРАКТИЧНА ЧАСТИНА

## 3.1. Формування набору даних

Для проведення досліджень необхідним є сформувати репрезентативний набір даних для навчання моделей розпізнавання небезпечних ситуацій, зокрема, за позою "руки вгору". Тому значна увага була приділена і формуванню датасету, оскільки необхідно створити та розмітити (анотувати) датасет із різних демографічних груп, забезпечити різноманітність умов зйомки та сценаріїв та підготувати дані для навчання моделей. Отже, виконано такі етапи, як: зйомка взірців відео; розкадровка; класифікація зображень; розпізнавання пози «руки вгору»; підготовка та експорт даних у YOLO-сумісний формат.

На рис.3.1 показано частину набору даних, використаних для навчання моделі, зокрема, підготовлені зображення із позою «руки вгору» та з іншим розташуванням рук.

Рис. 3.1. Приклад зображень з набору даних

Як наслідок, всього було зібрано 2054 зображення та всі анотовано вручну. В результаті, маркований набір даних містить 728 зображення жінок, 798 зображення чоловіків і 528 - дітей, включаючи 245 зображень хлопчиків та 283 зображення дівчат. Статистичне та графічне представлення зібраного набору даних за статями та віковими категоріями представлено на рисунку 3.2.

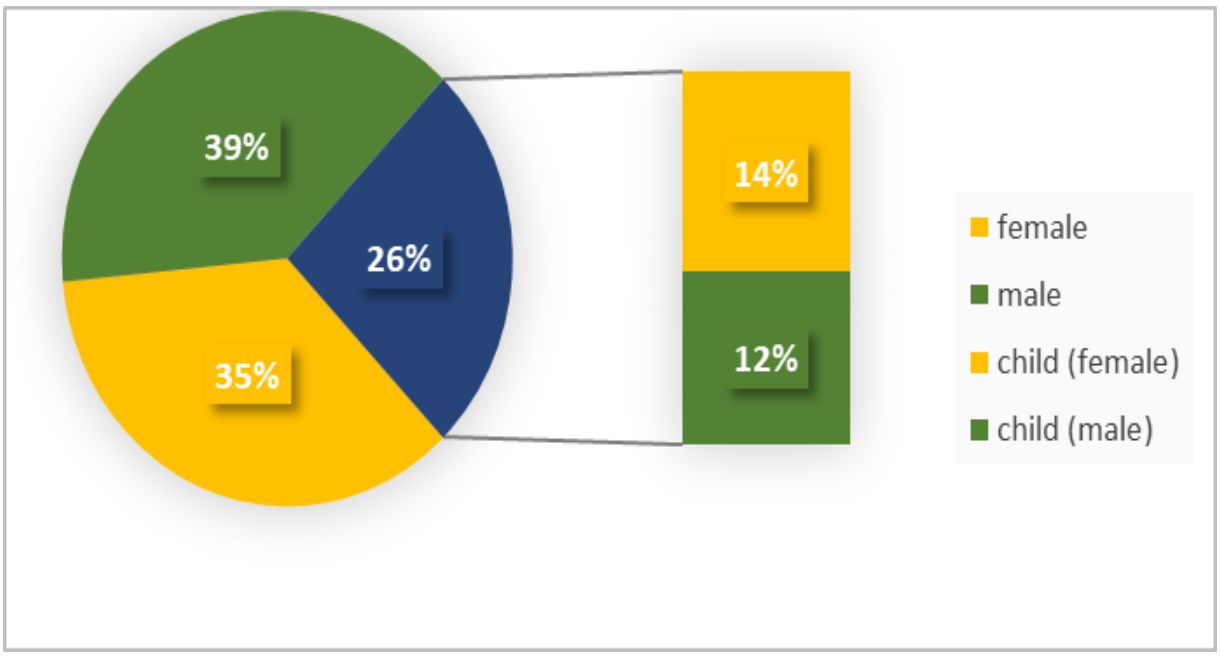


Рис. 3.2. Статистика набору даних

На рис. 3.3 показана невелика частина зображень з використаного сформованого та анотованого набору даних, на яких зображені люди з піднятими руками.

В подальшому сформований набір даних використовуватиметься для навчання моделі штучного інтелекту запропонованої в роботі архітектури.

Рис. 3.3. Частина анотованих зображень датасету

Анотування набору даних, відповідно до рекомендацій компанії Ultralytics, розробників YOLO, здійснено за допомогою Roboflow, як сумісного інструменту маркування. Ця платформа пропонує підтримку кількох форматів анотацій і дозволяє експортувати дані безпосередньо у YOLO-сумісний формат. Також можна використовувати популярні інструменти анотації, зокрема VOTT, LabelImg і CVAT, із відповідними етапами перетворення даних.

## 3.2. Отримання даних

Спершу дані були зібрані та зберігались у формі відео. Для тренування моделі зібрані відео було поділено на кадри за допомогою наступного коду із використанням бібліотеки cv2. Для уникнення надмірної насиченості одноманітними світлинами,

лише кожен четвертий кадр зберігається. Фрагмент коду програми представлений на рис.3.4.

```
    import cv2
    import os
    # Get list of all video files in the input folder
    video_files = [os.path.join(input_video, f) for f in
os.listdir(input_video) if f.endswith(".mp4") or f.endswith(".MOV")]
        video = cv2.VideoCapture(video_file)
        print(video_file)
        # Get frame rate and number of frames
        fps = int(video.get(cv2.CAP_PROP_FPS))
        frame_count = int(video.get(cv2.CAP_PROP_FRAME_COUNT))

        # Loop through each frame
        for i in range(0, frame_count,4):
            # Read frame
            ret, frame = video.read()
            # Save frame as an image file
            output_file = os.path.join(output_videoframes,
    f"{os.path.splitext(os.path.basename(video_file))[0]}_{i:04d}.jpg
");)
            cv2.imwrite(output_file, frame)
        # Release video object
        video.release()
```

Рис. 3.4. Фрагмент коду програми розкадровки

Цей код (рис. 3.4) може бути використаний, як для підготовки датасету з метою тренування моделі, так і для опрацювання відео під час прогнозування за допомогою моделі. Завдяки розширеній бібліотеці ultralytics, YOLO моделі мають вбудовану функцію predict для опрацювання відео під час виконання. Відео, що передається відеокамерою в режимі реального часу може бути розглянуто, як послідовний потік кадрів, відповідно, у запропонованому проєкті застосунку, реалізовано саме так.

## 3.3. Підключення датасету та YOLO для тренування та використання

Використання моделі YOLO потребує підключення бібліотеки ultralytics. Для тренування чи вдосконалення моделей YOLOv8 та YOLOv11, всі функції передбачені

цією бібліотекою, а для тренування YOLOv10 потрібно додатково клонувати репозиторій, в якому зберігається код, що необхідний для запуску саме цієї версії у зв'язку із тим, що YOLOv10 архітектура була розроблена командою науковців з іншої організації. У результаті, потрібно окремо завантажувати натреновану модель (див. рис. 3.5) [19].

```
wget -P {HOME}/weights -q https://github.com/THU-MIG/yolov10/releases/download/v1.1/yolov10s.pt
```

Рис. 3.5. Фрагмент коду завантаження YOLOv10

Дані, що використовуються для тренування, збережені на сервері Roboflow. Цей сервіс зберігає дані у різних форматах, адже різні версії моделей вимагають своєрідного форматування даних. Підключити датасет до YOLOv8 та YOLOv10 можна за допомогою наступного коду (рис.3.6):

```
from roboflow import Roboflow

rf = Roboflow(api_key=ROBOFLOW_API_KEY)
project = rf.workspace("diana-ijc4q").project("hup_1")
version = project.version(3)
dataset = version.download("yolov8")
```

Рис. 3.6. Фрагмент коду підключення датасету

Для підключення датасету, що можна використати у YOLOv11, змінюється лише останній рядок (рис.3.7):

```
dataset = version.download("yolov11")
```

Рис. 3.7. Фрагмент коду підключення датасету YOLOv11

Типово, для початку тренування чи вдосконалення моделі можна використовувати термінал. Для дотренування моделі YOLOv11 було використано наступний фрагмент коду (рис.3.8).

```
yolo task=detect mode=train model=yolo11s.pt data={dataset.location}/data.yaml
epochs=10 imgsz=640 plots=True
```

Рис. 3.8. Фрагмент коду дотренування YOLOv11

Таким чином, у результаті додаткового тренування зберігається оновлена модель, а також графіки, які описують перебіг тренування. Серед параметрів є такі [19]:

- task - завдання, яке виконуватиме YOLO модель (наприкдад, виявлення, класифікація);
- mode - режим виконання (наприкдад, тренування, валідація або тестування);
- model - шлях до файлу моделі (наприклад, "yolo11s.pt");
- data - шлях до файлу YAML, що визначає набір даних та його розташування;
- imgsz - розмір вхідного зображення у вигляді одного цілого числа для квадратного зображення або кортежу із висоти та ширини зображення;
- epochs - кількість епох для тренування або вдосконалення моделі.

Використання YOLO моделі у додатку чи для опрацювання даних, що не зберігаються на локальній обчислювальній машині, вимагає контейнеризації моделі для зручного та безпечного підключення, а також для масштабування використання. Цього можна досягнути за допомогою FastAPI. Application Programming Interface(API) — це набір правил і протоколів, що дозволяють програмам взаємодіяти одна з одною, що забезпечує безпеку, контроль над правами доступу, можливість масштабування системи, гнучкість підключень та модульність (підключення до системи без втручання у її внутрішній код). Для представленої системи створено API-

метод для розпізнавання об'єктів за допомогою моделі YOLOv11. Фрагмент коду наведено на рис.3.9.

```
from fastapi import FastAPI, File, UploadFile
import numpy as np
import cv2
from ultralytics import YOLO
app = FastAPI()
model = YOLO("yolov11s_hands_up.pt")

@app.post("/detect/")
async def detect_hands_up(img: UploadFile):

    image_bytes = await img.read()
    image = np.frombuffer(image_bytes, dtype=np.uint8)
    image = cv2.imdecode(image, cv2.IMREAD_COLOR)

    # Perform object detection with YOLOv8
    detections = model.predict(image)

    return {"detections": detections}
```

Рис.3.9. Фрагмент коду API для YOLOv11

У цьому коді почергово зчитується у форматі байтів, перетворюється на формат кольорового зображення із бібліотеки cv2 та опрацьовується моделлю YOLOv11s, натренованою для визначення пози “руки вверх”, один кадр та повертається список визначених об’єктів: у цьому випадку - поза “руки вверх”.

## 3.4. Тренування моделей YOLOv8, YOLOv10, YOLOv11

З метою вибору найоптимальнішої моделі для виявлення небезпечних ситуацій КСВ було натреновано моделі YOLOv8, YOLOv10 та YOLOv11. Тренування всіх моделей здійснювалося через Google Colaboratory за допомогою графічного процесора T4 GPU з операційною пам’яттю 16 Гб, а для порівняння швидкості

тренування моделі YOLOv11 - ще і на ноутбуці з характеристиками: центральний процесор Apple M3 Pro, операційна пам’ять 16 Гб.

Результати перебігу тренування моделей YOLOv8, YOLOv10, YOLOv11, показані на рис.3.10-3.12.

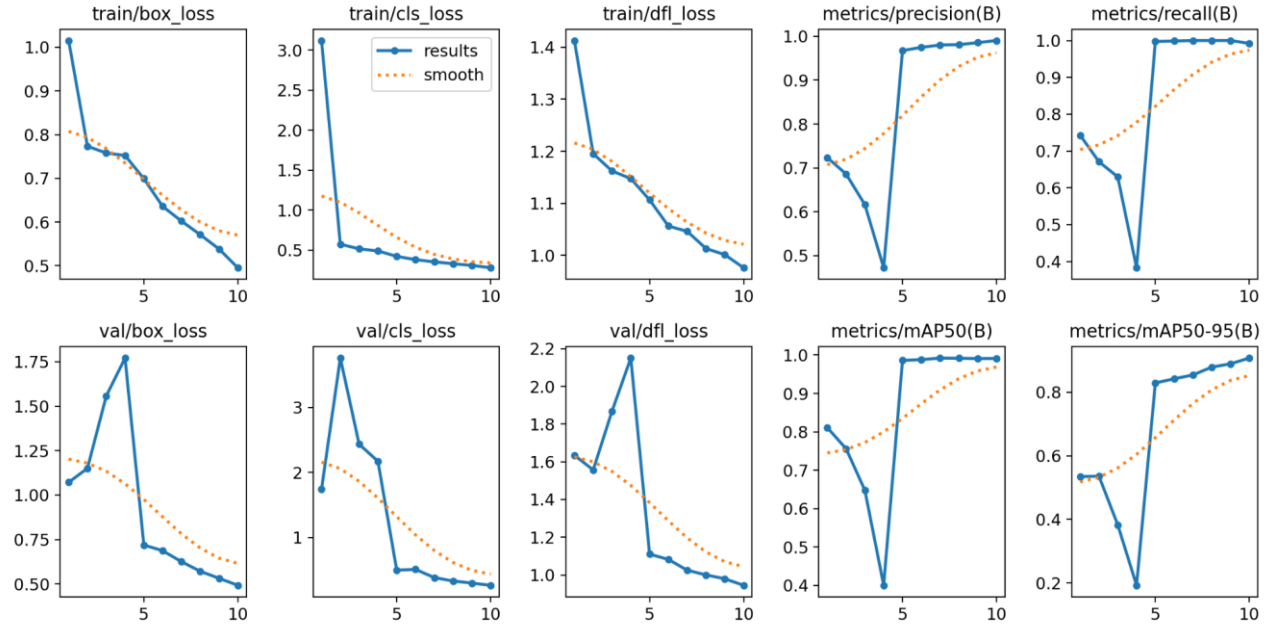


Рис. 3.10. Перебіг тренування моделі YOLOv8

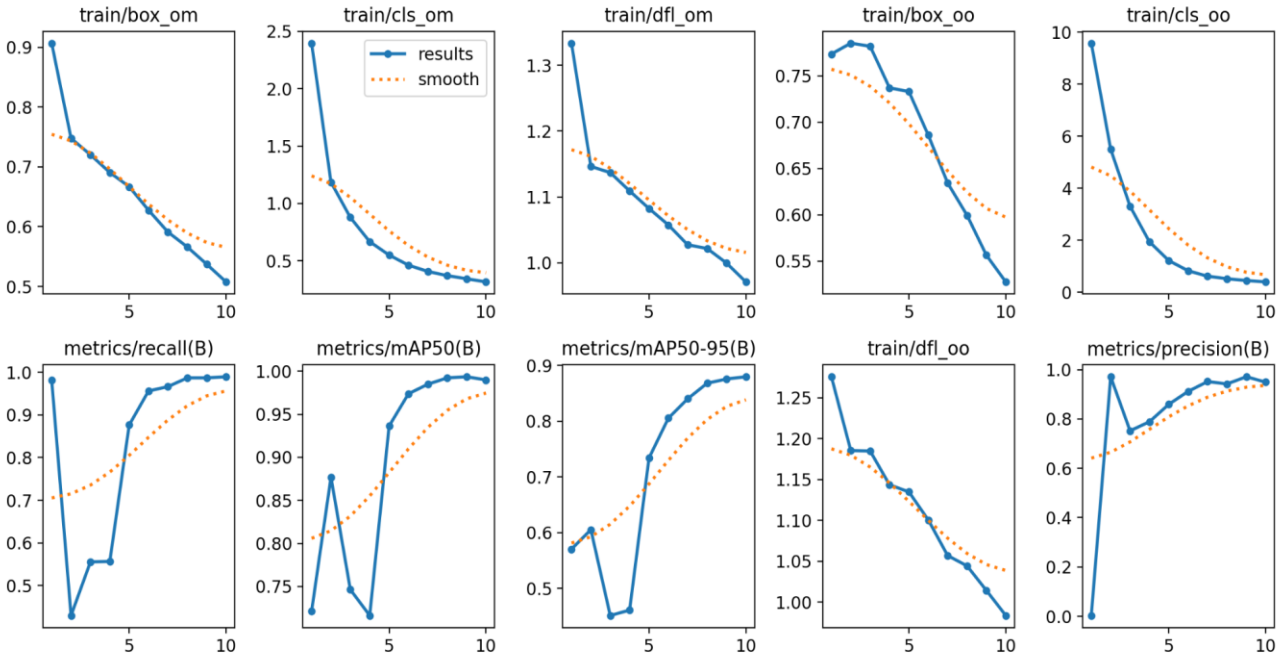


Рис. 3.11. Перебіг тренування моделі YOLOv10

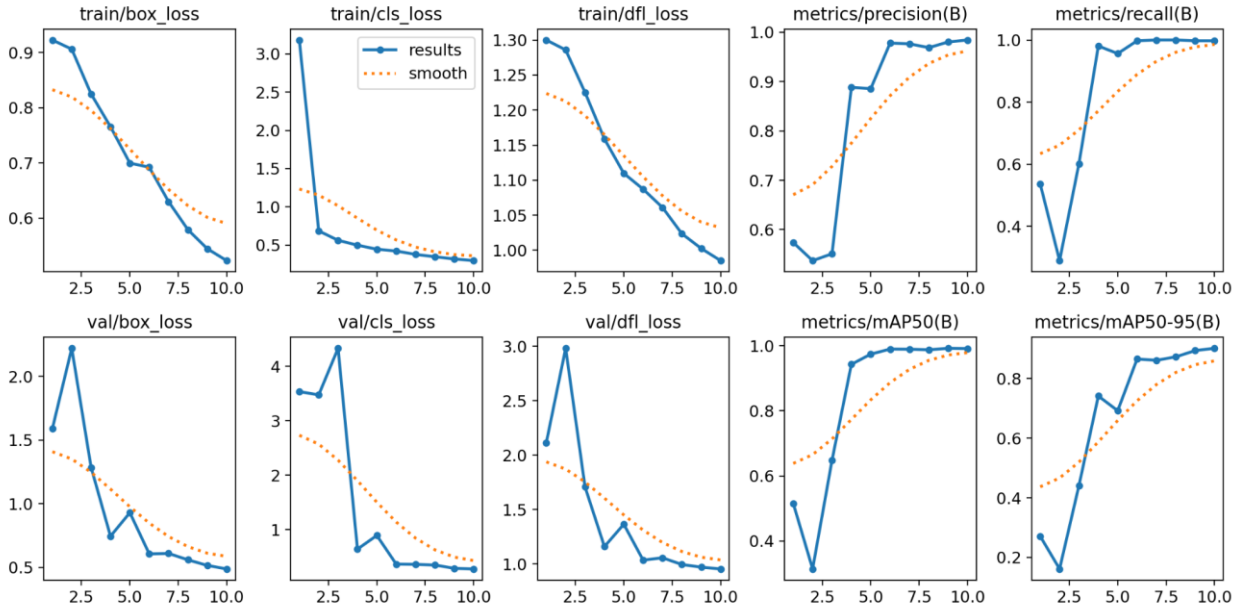


Рис. 3.12. Перебіг тренування моделі YOLOv11

Графіки демонструють ключові метрики та втрати, які відстежуються під час тренування моделей YOLOv8, YOLOv10, YOLOv11, та використовуються відстеження якості тренування та коригування гіперпараметрів моделі. Зазвичай (у YOLOv8 та YOLOv11) верхній рядок графіків позначає зміни на тренувальній вибірці зображень, нижній рядок показує графіки отримані при валідації результатів. YOLOv10 демонструє лише результати отримані на тренувальній вибірці, у зв’язку із відмінністю архітектури, а також із наявністю двох вихідних елементів. Дешифрування назв графіків на рисунках 3.10-3.12:

- box_loss - втрати, пов'язані з регресією координат обмежувальних рамок;
- cls_loss - втрати, пов'язані з класифікацією об'єктів у межах рамок;
- dfl_loss - втрати, пов’язані із точним передбаченням меж рамок;
- precision(B) - влучність моделі (Precision), тобто співвідношення правильно передбачених позитивних прикладів серед усіх передбачених позитивних;
- recall(B) - запам’ятовування моделі (Recall) — співвідношення правильно виявлених позитивних прикладів серед усіх реальних позитивних;
- mAP50(B) - середня точність mAP при IoU = 50%. Це основна метрика оцінки моделей об'єктного детектування, що поєднує і влучність, і запам’ятовування;

- mAP50-95(B) - середнє значення mAP для кількох порогових значень IoU (від 50% до 95%).

## 3.5. Результати виявлення небезпечних ситуацій

Експерименти було проведено із використанням моделей YOLOv8, YOLOv10, YOLOv11 та YOLOv11-pose, остання із яких не лише визначає, де знаходиться людина, а й визначає точки розміщення основних суглобів. Результати виявлення небезпечних ситуацій КСВ з використанням тренованих моделей YOLOv8, YOLOv10, YOLOv11 та YOLOv11-pose на тестувальному датасеті (які модель не “бачила” під час тренування), показані на рис. 3.13-3.16.


Рис.3.13. Результати виявлення небезпечних ситуацій КСВ з використанням моделі YOLOv8s


Рис.3.14. Результати виявлення небезпечних ситуацій КСВ з використанням моделі YOLOv10s


Рис.3.15. Результати виявлення небезпечних ситуацій КСВ з використанням моделі YOLOv11s


Рис.3.16. Результати виявлення небезпечних ситуацій КСВ з використанням моделі YOLOv11s Pose

Ultralytics YOLO моделі використовують кілька показників під час процесу валідації (перевірки) для оцінки ефективності моделі. До них належать: mAP, IOU, влучність (precision) і запам’ятовування (recall), час визначення та швидкість виконання.

mAP (mean average precision - середня точність) - оцінює точність виявлення об’єкта, IOU (Intersection over Union) - вимірює перекриття між передбачуваними та наземними обмежувальними прямокутниками, влучність (precision) вимірює відношення справжніх позитивних виявлень до загальної кількості всіх виявлених позитивних результатів, а запам’ятовування (recall) - співвідношення справжніх позитивних виявлень до загальної кількості фактичних позитивних результатів, час

визначення - це час, витрачений на кожне зображення в мілісекундах, швидкість виконання вказує на: час підготовки одного зображення, отримання результату моделі та вивід результату на зображення.

Для оцінки ефективності моделі було проведено валідацію (перевірку), результати якої представлені на рис. 3.17-3.21. Задля збереження часу перевірку здебільшого було запущено у середовищі Google Colaboratory із доступом до графічного процесора Tesla T4 GPU.

```
Model summary (fused): 168 layers, 11,125,971 parameters, 0 gradients, 28.4 GFLOPs
val: Scanning /content/datasets/HUP_1-3/valid/labels.cache... 402 images, 13 backgrounds, 0 corrupt: 100% 402/402 [00:00<?, ?it/s]
                 Class     Images  Instances      Box(P          R      mAP50  mAP50-95): 100% 26/26 [00:06<00:00,  3.83it/s]
                   all        402        389      0.991      0.992      0.991      0.908
Speed: 1.3ms preprocess, 9.2ms inference, 0.0ms loss, 2.5ms postprocess per image
```

Рис. 3.17. Перевірка ефективності YOLOv8s

```
YOLOv10n summary (fused): 285 layers, 2694806 parameters, 0 gradients, 8.2 GFLOPs
val: Scanning /content/datasets/HUP_1-3/valid/labels.cache... 402 images, 13 backgrounds, 0 corrupt: 100% 402/402 [00:00<?, ?it/s]
                 Class     Images  Instances      Box(P          R      mAP50  mAP50-95): 100% 26/26 [00:06<00:00,  4.29it/s]
                   all        402        389       0.95       0.99       0.99       0.88
Speed: 1.5ms preprocess, 6.1ms inference, 0.0ms loss, 0.3ms postprocess per image
```

Рис. 3.18. Перевірка ефективності YOLOv10n

```
YOLOv10s summary (fused): 293 layers, 8035734 parameters, 0 gradients, 24.4 GFLOPs
val: Scanning /content/datasets/HUP_1-3/valid/labels.cache... 402 images, 13 backgrounds, 0 corrupt: 100% 402/402 [00:00<?, ?it/s]
                 Class     Images  Instances      Box(P          R      mAP50  mAP50-95): 100% 26/26 [00:07<00:00,  3.36it/s]
                   all        402        389      0.976       0.99      0.992      0.883
Speed: 0.9ms preprocess, 11.3ms inference, 0.0ms loss, 0.4ms postprocess per image
```

Рис. 3.19. Перевірка ефективності YOLOv10s

```
YOLO11s summary (fused): 238 layers, 9,413,187 parameters, 0 gradients, 21.3 GFLOPs
val: Scanning /content/datasets/HUP_1-3/valid/labels.cache... 402 images, 13 backgrounds, 0 corrupt: 100% 402/402 [00:00<?, ?it/s]
                 Class     Images  Instances      Box(P          R      mAP50  mAP50-95): 100% 26/26 [00:06<00:00,  4.14it/s]
                   all        402        389      0.984      0.997      0.992      0.901
Speed: 0.6ms preprocess, 9.3ms inference, 0.0ms loss, 2.1ms postprocess per image
```

Рис. 3.20. Перевірка ефективності YOLOv11s у середовищі Google Colab

```
YOLO11s summary (fused): 238 layers, 9,413,187 parameters, 0 gradients, 21.3 GFLOPs
val: Scanning /Users/diana/Desktop/tntu/datasets/HUP_1-3/valid/labels.cache... 4
                 Class     Images  Instances      Box(P          R      mAP50  m
                   all        402        389      0.985      0.995      0.991      0.894
Speed: 0.7ms preprocess, 4922.2ms inference, 0.0ms loss, 0.4ms postprocess per image
```

Рис. 3.21. Перевірка ефективності YOLOv11s* на ЦП Apple M3 Pro

Порівняння матриць невідповідностей із позначенням правильних та передбачених результатів наведено на рис. 3.22-3.24.

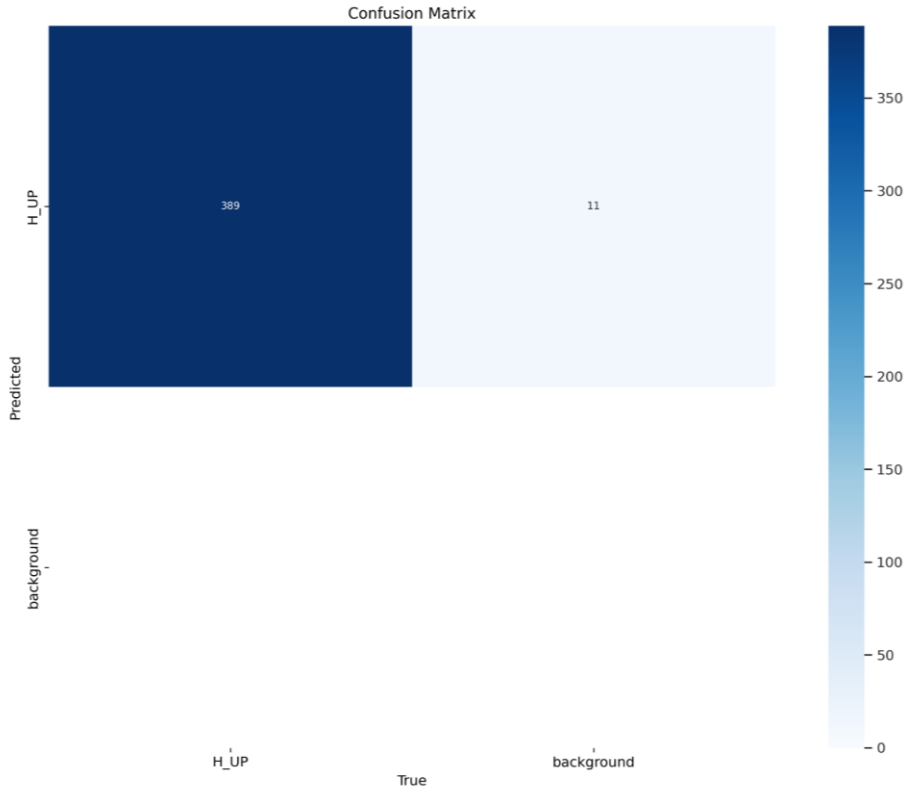


Рис. 3.22. Матриця невідповідностей YOLOv8

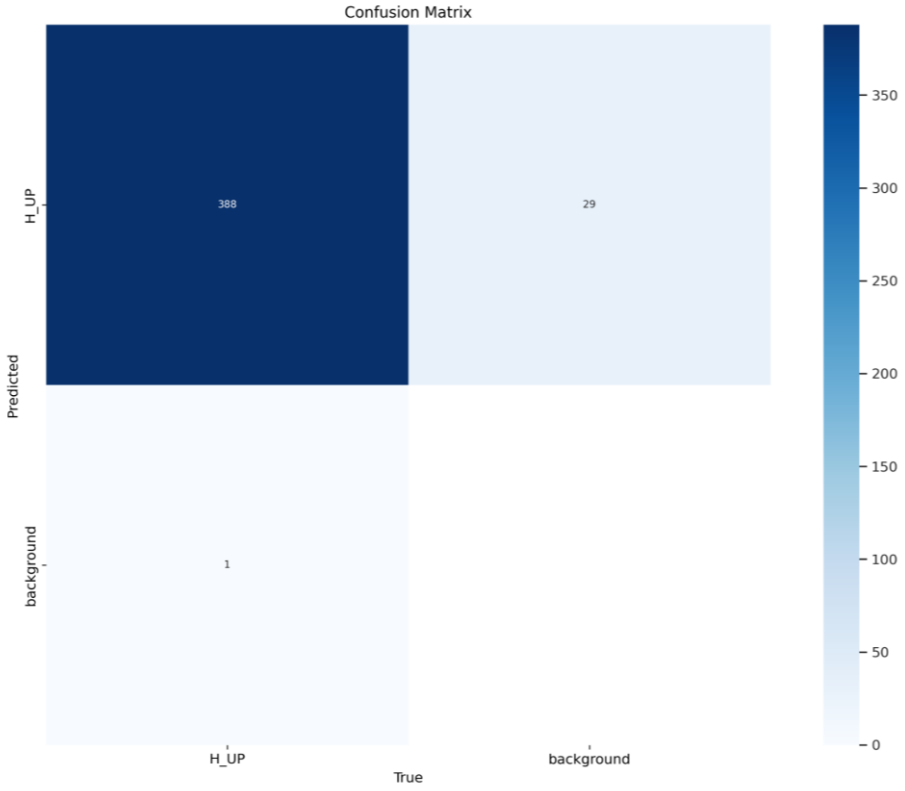


Рис. 3.23. Матриця невідповідностей YOLOv10

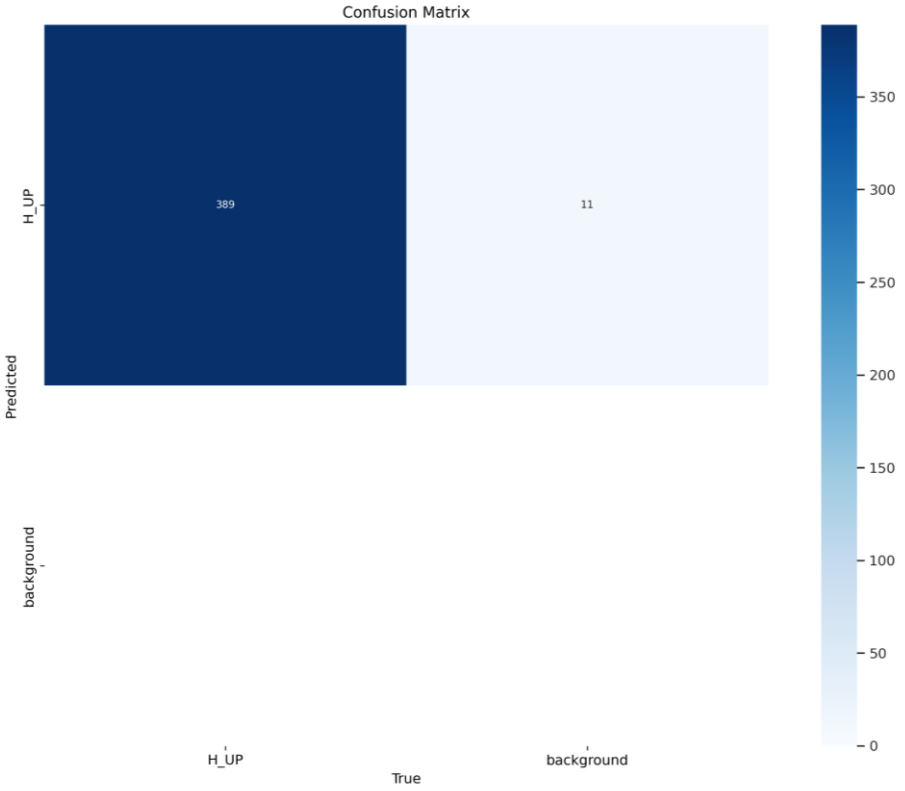


Рис. 3.24. Матриця невідповідностей YOLOv11

Для отримання результатів використовуються зображення із тестової вибірки, перетворені на квадратні, із розміром 640 на 640 пікселів. Порівняння проводиться по властивостях моделей YOLOv8s, YOLOv10n, YOLOv10s, YOLOv11s, YOLOv11s* та YOLOv11s-pose під час виконання прогнозування. Всі моделі натреновані у середовищі Google Colab, окрім YOLOv11s*, що була запущена та натренована на локальній машині.

Табл. 3.1 демонструє кількісну характеристику розглянутих моделей, а саме: середній час, за який проводиться попереднє опрацювання одного зображення, передбачення моделі для одного зображення та виведення результатів на одне зображення, а також розмір пам’яті, яку займає модель.

*Таблиця 3.1*

**Порівняльний аналіз опрацювання одного зображення моделями YOLOv8s, YOLOv10n, YOLOv10s, YOLOv11s, YOLOv11s* та YOLOv11s-pose**

| Модель | Попереднє опрацювання | Передбачення моделі | Виведення результатів | Розмір |
|---|---|---|---|---|
| YOLOv8s | 1.3 мс | 9.2 мс | 2.5 мс | 22.5 Мб |
| YOLOv10n | 1.5 мс | 6.1 мс | 0.3 мс | 11.2 Мб |
| YOLOv10s | 0.9 мс | 11.2 мс | 0.4 мс | 16.5 Мб |
| YOLOv11s | 0.6 мс | 9.3 мс | 2.1 мс | 19.5 Мб |
| YOLOv11s* | 0.7 мс | 4922 мс | 0.4 мс | 19.5 Мб |
| YOLOv11s-pose | 2.4 мс | 11.6 мс | 3.9 мс | 20.4 Мб |

Модель YOLOv11s-pose вимагає найбільшого часу опрацювання перед запуском, тоді як YOLOv11s демонструє найменший час попереднього опрацювання. YOLOv11s-pose також потребує найбільшого часу для аналізу зображення та виведення результатів, що пов'язано зі складнішою обробкою даних для задач пози та підвищеною складністю моделі. Додатково, YOLOv11s-pose займає більше місця на

диску пам’яті, у порівнянні із відповідною моделлю YOLOv11s. Це показує, що модель для визначення пози потребує більше ресурсів, а також додаткового опрацювання після отримання важливих точок, що приводить до висновку, що її використання не є ефективним.

Порівняння ефективності YOLO наведено в табл. 3.2. Серед порівнюваних характеристик наведені такі: точність (mAP50, mAP50-95), влучність, запам’ятовування та кількість шарів (складність архітектури) YOLOv8s, YOLOv10n, YOLOv10s, YOLOv11s, YOLOv11s* (YOLOv11s тренована у середовищі Google Colab, YOLOv11s* - на локальній машині) та YOLOv11s-pose.

*Таблиця 3.2*

**Порівняння характеристик ефективності моделей YOLOv8s, YOLOv10n, YOLOv10s, YOLOv11s, YOLOv11s* та YOLOv11s-pose**

| Модель | Кількість шарів | Влучність | Запам’ятову-вання | mAP50 | mAP50-95 |
|---|---|---|---|---|---|
| YOLOv8s | 168 | 0.991 | 0.992 | 0.991 | 0.908 |
| YOLOv10n | 285 | 0.95 | 0.99 | 0.99 | 0.88 |
| YOLOv10s | 293 | 0.976 | 0.99 | 0.992 | 0.883 |
| YOLOv11s | 238 | 0.984 | 0.997 | 0.992 | 0.901 |
| YOLOv11s* | 238 | 0.985 | 0.995 | 0.991 | 0.894 |

Всі протестовані моделі показали високу влучність (>95%) у виявленні пози "руки вгору" із розміром 11-22 Мб та швидкістю опрацювання 6-11 мс. Кілька пар моделей було обрано для глибшого аналізу різних варіацій моделей.

Модель YOLOv10n було обрано для аналізу у порівнянні із YOLOv10s, адже це менша і легша модифікація тієї ж версії YOLO. З табл. 3.1 видно, що YOLOv10n є найшвидшою за часом отримання висновків та виведення результатів. Також YOLOv10n є найлегшою за розміром і на 32% легша, аніж YOLOv10s. Із табл. 3.2 видно, що різниця у точності між YOLOv10n та YOLOv10s є 0.3%, що може бути

проігноровано. Ці спостереження наводять на висновок, що використання нано-моделей у порівнянні із малими моделями є привабливою альтернативою для IoT приладів, що не мають великого обсягу пам’яті чи обчислювальних ресурсів.

Для перевірки виконання моделі на іншому приладі, YOLOv11s* була запущена на ЦП ноутбука, що значно підвищило час, необхідний для отримання передбачення моделі з кількох мілісекунд до 4 секунд на один кадр. У результаті, стало зрозуміло, що час виконання критично високий і недостатній для опрацювання потокових відео. Додатково, точність mAP50-95 знизилась із 0.901 до 0.894 (0.5%). З цієї причини було запропоновано систему, в якій камера передає зображення у режимі реального часу на сервер, до якого підключено модуль із моделлю. Як наслідок, модель виконує передбачення віддалено, а результат передається і відображається через сервер.

YOLOv11s та YOLOv8s мають дуже схожі властивості: mAP50 = 0.992 та 0.991 і час передбачення моделі рівний 9.3 та 9.2 мс, відповідно. Час попереднього опрацювання є вдвічі меншим для YOLOv11s, та на 0.4 мс менше для постобробки, у результаті загальний час витрачений на одне зображення є меншим для YOLOv11s, хоч кількість шарів в архітектурі цієї версії моделі є на 70 штук більшою. Додатково, YOLOv8s є найважчою із розглянутих моделей, хоч різниця і не є значною. Сукупність усіх цих факторів дозволяє обрати YOLOv11s як основну модель для реалізації системи.

## 3.6. Висновки до розділу 3

Сформовано набір даних із 2054 зображень, включаючи 732 зображення жінок, 794 чоловіків і 528 дітей, для навчання моделей розпізнавання небезпечних ситуацій за позою "руки вгору" та підготовлено дані для навчання моделей.

Проведено навчання моделей на сформованому наборі даних на GPU T4 через Google Colab та локально на процесорі Apple M3 Pro.

Проведено комплексне порівняння різних версій архітектури YOLO (YOLOv8s, YOLOv10n, YOLOv10s, YOLOv11s та YOLOv11s-pose) для задачі виявлення небезпечних ситуацій за ключовими метриками: час опрацювання, точність

розпізнавання та вимоги до обчислювальних ресурсів.

Експериментально встановлено, що модель YOLOv11s демонструє оптимальне співвідношення між точністю та швидкодією, маючи найменший час попереднього опрацювання (0.6 мс), високі показники точності (mAP50 = 0.992) та помірні вимоги до пам'яті (19.5 Мб).

Виявлено, що нано-версія YOLOv10 (YOLOv10n) може бути ефективною альтернативою для пристроїв з обмеженими ресурсами, маючи найменший розмір (11.2 Мб) та демонструючи найшвидший час опрацювання результатів при прийнятній точності. Водночас встановлено критичну залежність швидкодії від апаратної платформи - на CPU час опрацювання зростає до 4922 мс порівняно з 6-11 мс на GPU, що обґрунтовує необхідність використання серверної архітектури з GPU для практичного впровадження системи відеонагляду.

Зауважено, що використання API через FastAPI забезпечує гнучкість розгортання та масштабування системи

На основі комплексного аналізу всіх характеристик, для практичної реалізації системи виявлення небезпечних ситуацій рекомендовано використовувати модель YOLOv11s з серверною архітектурою опрацювання даних, що забезпечить оптимальний баланс між точністю розпізнавання та швидкодією системи в реальних умовах експлуатації.

# РОЗДІЛ 4
# ОХОРОНА ПРАЦІ ТА БЕЗПЕКА В НАДЗВИЧАЙНИХ СИТУАЦІЯХ

## 4.1. Охорона праці

При дослідженні методів та засобів виявлення небезпечних ситуацій для людей комп'ютеризованою системою відеонагляду в основному вся робота виконувалася із використанням комп'ютерної техніки, тому важливим і необхідним є дотримання вимог охорони праці, техніки безпеки та протипожежної безпеки при роботі з ЕОМ.

Питання охорони праці регулюються відповідними законодавчими та нормативно-правовими актами, які, зокрема, визначають обов'язки роботодавця із забезпечення робітникам комфортних та безпечних умов для здійснення роботи. Ці обов'язки, а також права працівників передбачені частиною 2 ст.2 і частиною 1 ст.21 КЗпП, а також ст.13 Закону України «Про охорону праці», у яких визначаються основні положення з реалізації конституційного права працівників.

Існує цілий ряд вимог, які визначають специфіку заходів з охорони праці при роботі з персональним комп'ютером. Законодавчі та нормативно-правові акти, які за участі відповідних органів державної влади регулюють відносини між роботодавцем та робітником з питань безпеки, гігієни праці та виробничого середовища, а також встановлюють єдиний порядок організації охорони праці в Україні. На їх основі розроблені чисельні документи: правила, інструкції, норми, державні санітарні правила та ін., якими мають керуватись роботодавці та які регламентують певні питання щодо конструкції електронно-обчислювальної техніки, та особливостей їх розміщення.

На сьогодні основними документами, які регламентують питання охорони праці при використанні працівниками персональних комп'ютерів, можна вважати такі підзаконні акти:

- ДСанПіН 3.3.2.007-98 «Державні санітарні правила і норми роботи з візуальними дисплейними терміналами електронно-обчислювальних машин» [21].

- НПАОП 0.00-7.15-18 «Вимоги щодо безпеки та захисту здоров’я працівників під час роботи з екранними пристроями» [22].

У відповідності з цими документами, при роботі з програмним забезпеченням, в тому числі при проведені досліджень методів виявлення небезпечних ситуацій для людей, необхідно вжити всіх необхідних заходів з охорони праці.

Згідно НПАОП 0.00-7.15-18 електронно-обчислювальні засоби повинні відповідати вимогам чинних в Україні стандартів і пройти державну санітарно-епідеміологічну експертизу у Порядку проведення державної санітарно-епідеміологічної експертизи.

При організації робочих місць користувачів комп’ютеризованих систем необхідно забезпечити дотримання вимог щодо їх розташування, зокрема відстань робочого місця від стіни повинна складати не менше 1 м, а відстань між робочими місцями повинна становити не менше 1,7 м. Площа, яка виділяється на одне робоче місце, обладнане ПК становить – ≥6.0 м$^2$, а об’єм – ≥20 м$^3$ [21].

При виборі приміщень для розміщення робочих місць ПК необхідно враховувати ступінь відбиття світла на екранах дисплеїв, яке проходить через вікна і яке може викликати значне осліплення в тих, хто сидить перед ними, особливо влітку та в сонячні дні. Тому, ПК і оргтехніку необхідно розміщувати біля стін, які не знаходяться біля вікон або навпроти них [21].

Оскільки при незадовільному освітленні знижується продуктивність праці користувачів ПК і можливі негативні впливи на здоров’я такі як короткозорість, швидка втомленість, тому всі приміщення, які облаштовані робочими місцями з ПК, мають природне і штучне освітлення. Не допускається розташування робочих місць з ПК в підвальних приміщеннях [21].

Робочі місця з ПК при виконанні роботи, яка потребує значної розумової концентрації, зокрема при реалізації методів для виявлення небезпечних ситуацій, варто ізольовувати одне від одного перегородкою висотою 1,6 м [21]. Поверхня підлоги у приміщеннях повинна бути рівною та зручною для очищення та вологого прибирання.

Штучне освітлення у приміщеннях повинно бути виконано у вигляді комбінованої системи освітлення з використанням люмінесцентних джерел світла у світильниках загального освітлення, які розташовувати над робочими поверхнями у рівномірно-прямокутному порядку. Штучне освітлення повинно забезпечувати на робочих місцях з ПК освітленість 300 – 500 Лк [23].

Для запобігання засвітленню екранів ПК прямими світловими потоками лінії світильників розташовані з достатнім бічним зміщенням відносно рядів робочих місць, а також паралельно до світлових отворів. При цьому кожне вікно повинно мати світлорозсіюючі штори з коефіцієнтом відбивання 0,7 [23].

У приміщенні також необхідно забезпечити і природнє освітлення, при цьому на кожному вікні доцільно закріплити жалюзі з вертикальними ламелями, що регулюються для зменшення прямого попадання сонячного світла на екран комп’ютерів.

З метою запобігання нещасним випадкам та організації охорони праці на виробництві розробляються інструкції з охорони праці і техніки безпеки при використанні комп’ютерної техніки. Дія інструкції поширюється на всі структурні підрозділи підприємства [24].

До роботи на ПК допускаються особи, які пройшли спеціальне навчання, медичне обстеження, вступний інструктаж з охорони праці, інструктаж на робочому місці та інструктаж з пожежної безпеки.

З ергономічної точки зору при розташуванні елементів робочого місця необхідно враховувати такі фактори [21]: простір для розміщення користувача; можливість огляду елементів робочого місця; можливість огляду простору за межами робочого місця; можливість робити записи, розміщення документації і матеріалів, які використовує працівник.

При роботі над дослідженням методів та засобів виявлення небезпечних ситуацій для людей комп’ютеризованою системою враховано та дотримано чинних вимог з охорони праці і техніки безпеки, що дозволило забезпечити зручні умови для ефективної роботи.

## 4.2. Безпека в надзвичайних ситуаціях

На сьогоднішній день важливе значення має фукціонування підприємства під час війни в Україні, тому оцінка стійкості роботи підприємства до впливу вторинних вражаючих факторів є необхідною.

Стійкість роботи об'єкта – це здатність в умовах військового часу виготовляти продукцію в запланованому об'ємі і номенклатурі, а при одержанні слабких і частково середніх руйнувань відновлювати своє виробництво в мінімальні терміни. Ціль оцінки стійкості об'єкта полягає у виявленні слабких його елементів, щоб у подальшому провести інженерно-технічні заходи, спрямовані на підвищення стійкості об'єкта в цілому.

Оцінка стійкості роботи об'єкта – це всебічне вивчення підприємства з погляду спроможності його протистояти впливу вражаючих факторів, продовжувати роботу і відновлювати виробництво при одержанні слабких руйнувань [25].

Підприємства відрізняються одне від одного як за конструктивним рішенням, так і за технологічними процесами. Відмінності об'єктів полягають в будинках і спорудах, устаткуванні і технології виробництва, комунально-енергетичних мережах і території, на якій розташований об'єкт. Тому в усіх випадках оцінка стійкості кожного об'єкта має свої особливості і вимагає конкретного підходу до рішення цього питання. У даному випадку розглянемо загальні для всіх об'єктів питання оцінки їх стійкості до впливу вражаючих факторів зброї масового знищення.

Оцінка стійкості роботи об'єкта починається з вивчення району розташування. Об'єкт може знаходитися в місті, за межею його проектної забудови і на деякій віддалі від міста. Досліджується територія району, його структура, щільність і тип забудови, сусідні об'єкти і можливість виникнення на них вторинних чинників поразки. На об'єкті визначаються щільність забудови, розміщення основних будинків і споруджень, що впливають на характер руйнування, можливе утворення завалів і виникнення пожеж. Особлива увага приділяється ділянкам, де можливе виникнення небезпечних вторинних чинників ушкоджень. Беруться на облік усі будинки і споруди, робиться оцінка їх статичної стійкості. Вивчають кожен підрозділ і його

окремі елементи як по конструктивному рішенню, так і за матеріалами, що були використані в будівництві. Розглядаються умови розміщення внутрішнього технологічного устаткування і визначаються види руйнувань і ушкоджень, що можуть мати місце при різного роду обстрілах, бомбардуваннях та ядерному вибуху і заваленні огороджуваних конструкцій підприємства.

Особливо важливо визначити захист цінного й унікального устаткування, насиченість виробництва автоматикою і можливість продовження виробництва у випадку виходу з ладу контрольно-вимірювальної апаратури. Обстежуються комунально-енергетичні системи об'єкта і робиться оцінка стійкості споруджень і ліній, тобто визначаються параметри вражаючих факторів, при яких комунально-енергетичні мережі одержать ті або інші руйнування. Визначається забезпеченість працюючих захисними спорудженнями: встановлюється кількість сховищ, укриттів і оцінюються їхні захисні властивості. Вивчається система керування, зв'язку й оповіщення на основі вивчення стану захищених пунктів керування, вузлів і ліній зв'язку. Аналізується система матеріально-технічного постачання і виробничих зв'язків. Встановлюється об'єм запасів і можливих термінів продовження роботи без постачань; визначається відповідність їхньої кількості і номенклатури вимогам, запропонованим до виробництва у військовий час. Оцінюється стійкість складів сировини, комплектуючих виробів, готової продукції й інших матеріалів, а також сховища пальних матеріалів. Досліджується підготовка об'єкту до відновлення виробництва у випадку одержання слабких або середніх руйнувань [25, 26].

Аналіз виробничої діяльності об'єкта дозволяє виявити слабкі елементи, ділянки і підготувати план підвищення стійкості їх роботи і план відновлювальних робіт, забезпечити їх будівельно-монтажною і проектною документацією.

Оцінка стійкості роботи об'єкта організовується начальником ЦО (директором підприємства), його штабом і головними фахівцями: головним інженером, головним

механіком, головним технологом, головним енергетиком, тощо. До оцінки стійкості залучаються начальники служб та інші фахівці [25, 26].

## 4.3. Висновки до розділу 4

В даному розділі описані актуальні питання щодо охорони праці та забезпечення безпеки в надзвичайних ситуаціях. Була опрацьована інформація стосовно вимог з охорони праці і техніки безпеки, пожежної та електробезпеки. Також, розглянуто питання щодо оцінки стійкості роботи підприємства до впливу вторинних вражаючих факторів.

# ВИСНОВКИ

В результаті виконання кваліфікаційної роботи запропоновано методи та засоби підвищення ефективності виявлення небезпечних ситуацій для людини комп’ютеризованими системати відеонагляду, зокрема:

— На основі проведеного аналізу визначено необхідність подальшого вдосконалення існуючих методів та засобів виявлення небезпечних ситуацій для забезпечення підвищення швидкості та ефективності роботи в реальному часі, а також точності розпізнавання у складних умовах.

— Обгрунтовано використання YOLO, як ефективного засобу для виявлення небезпечних ситуацій у системах відеонагляду завдяки їх здатності працювати в реальному часі та високій точності виявлення об'єктів, що дозволить підвищити ефективність виявлення небезпечних ситуацій завдяки їх високій швидкодії, точності та адаптивності до різних умов роботи.

— Проведено аналіз різних версій архітектури YOLO, зокрема YOLOv8, YOLOv10, YOLOv11 з метою вибору найефективнішої для задачі виявлення небезпечних ситуацій.

— Сформовано набір даних із 2054 зображень, включаючи 732 зображення жінок, 794 чоловіків і 528 дітей, для навчання моделей розпізнавання небезпечних ситуацій за позою "руки вгору" та підготовлено дані для навчання моделей.

— Проведено комплексне порівняння різних версій архітектури YOLO (YOLOv8s, YOLOv10n, YOLOv10s, YOLOv11s та YOLOv11s-pose) для задачі виявлення небезпечних ситуацій за ключовими метриками: час опрацювання, точність розпізнавання та вимоги до обчислювальних ресурсів.

— Здійснено тренування моделей на GPU T4 через Google Colab та локально на процесорі Apple M3 Pro.

— Експериментально встановлено, що модель YOLOv11s демонструє оптимальне співвідношення між точністю та швидкодією, маючи найменший час попереднього опрацювання (0.6 мс), високі показники точності (mAP50 = 0.992) та помірні вимоги до пам'яті (19.5 Мб).

— Встановлено критичну залежність швидкодії від апаратної платформи - на CPU час опрацювання зростає до 4922 мс порівняно з 6-11 мс на GPU, що обґрунтовує необхідність використання серверної архітектури з GPU для практичного впровадження системи відеонагляду.

— Зауважено, що використання API через FastAPI забезпечує гнучкість розгортання та масштабування системи

— На основі комплексного аналізу всіх характеристик, для практичної реалізації системи виявлення небезпечних ситуацій рекомендовано використовувати модель YOLOv11s з серверною архітектурою опрацювання даних, що забезпечить оптимальний баланс між точністю розпізнавання та швидкодією системи в реальних умовах експлуатації.

# СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. D. Velychko, H. Osukhivska, Y. Palaniza, N. Lutsyk, Ł. Sobaszek. Artificial Intelligence Based Emergency Identification Computer System. Advances in Science and Technology Research Journal, 18(2), 2024. P.296-304.

2. Величко Д. Аналіз систем виявлення екстрених ситуацій. VI Міжнародна студентська науково-технічна конференція „Природничі та гуманітарні науки. Актуальні питання “, Т.: ТНТУ, 2023. С. 116. (Інформаційні технології).

3. Величко Д. Алгоритм роботи комп’ютеризованої системи відеонагляду з функцією ідентифікації екстрених ситуацій. Матеріали XI науково-технічної конференції „Інформаційні моделі, системи та технології “, 2023. С.138.

4. O. Elharrouss, N. Almaadeed, S. Al-Maadeed. A review of video surveillance systems. Journal of Visual Communication and Image Representation, Volume 77, 2021, P. 103116. https://doi.org/10.1016/j.jvcir.2021.103116. (Last accessed: 10.12.2024).

5. P. Kuppusamy, V.C. Bharathi. Human abnormal behavior detection using CNNs in crowded and uncrowded surveillance – A survey, Measurement: Sensors, Volume 24, 2022, 100510, https://doi.org/10.1016/j.measen.2022.100510. (Last accessed: 16.12.2024).

6. Jang Sein, Battulga Lkhagvadorj, Nasridinov Aziz. Detection of Dangerous Situations using Deep Learning Model with Relational Inference. J Multimed Inf Syst 2020;7(3). P. 205-214. https://doi.org/10.33851/JMIS.2020.7.3.205. (Last accessed: 06.12.2024).

7. Y. Zhu, S. Newsam. Motion-aware function for improved video anomaly detection. Proceedings of British Machine Vision Conference (BMVC2019), September 9-12, Cardiff, UK, 2019. P. 1-12.

8. M. Chebiyyam, R. D. Reddy, D. P. Dogra, H. Bhaskar, L. Mihaylova. Motion anomaly detection and trajectory analysis in visual surveillance. Multimedia Tools and Applications, vol. 77, P. 16223-16248, 2018. https://doi.org/10.1007/s11042-017-5196-6. (Last accessed: 16.12.2024).

9. M. Nakib, R. T. Khan, M. S. Hasan, J. Uddin. Crime scene prediction by detecting threatening objects using convolutional neural network. Proceedings of the International Conference on Computer, Communication, Chemical, Material and Electronic Engineering, Rajshahi, Bangladesh. P. 1-4, 2018.

10. A. Singh, D. Patil, S. N. Omkar. Eye in the Sky: Real-time Drone Surveillance System (DSS) for Violent Individuals Identification using ScatterNet Hybrid Deep Learning Network. arXiv preprint arXiv:1806.00746. (Last accessed: 16.12.2024).

11. J. Huang, V. Rathod, C. Sun, M. Zhu, A. Korattikara, A. Fathi, I. Fischer, Z. Wojna, Y. Song, S. Guadarrama, K. Murphy. Speed/accuracy trade-offs for modern convolutional object detectors. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, USA, 2017. P. 7310-7319.

12. A. Prati. An intelligent surveillance system for dangerous situation detection in home environments. Unipr, 2016. https://www.academia.edu/21940139/An_Intelligent_Surveillance_System_for_Dangerous_Situation_Detection_in_Home_Environments. (Last accessed: 18.12.2024).

13. A. Yadav, N. Thaker, D. Makwana, N. Waingankar, P. Upadhyay. Intruder Detection System: A Literature Review. Proceedings of the 4th International Conference on Advances in Science & Technology (ICAST2021), 2021. http://dx.doi.org/10.2139/ssrn.3866777. (Last accessed: 18.12.2024).

14. Ultralytics Docs. https://docs.ultralytics.com/ (Last accessed: 16.12.2024).

15. Mastering All YOLO Models from YOLOv1 to YOLOv9: Papers Explained (2024). https://learnopencv.com/mastering-all-yolo-models/ (Last accessed: 18.12.2024).

16. R. Khanam, M. Hussain. What is YOLOV5: A deep look into the internal features of the popular object detector. arXiv:2407.20892v1 [cs.CV] 30 Jul 2024. https://arxiv.org/pdf/2407.20892 (Last accessed: 17.12.2024).

17. Ao Wang, Hui Chen, Lihao Liu, Kai Chen, Zijia Lin, Jungong Han, Guiguang Ding, YOLOv10: Real-Time End-to-End Object Detection. arXiv:2405.14458v2 [cs.CV] 30 Oct 2024. https://arxiv.org/abs/2405.14458 (Last accessed: 17.12.2024).

18. A. Bochkovskiy, C.-Y. Wang, H.-Y. M. Liao. Yolov4: Optimal speed and accuracy of object detection. arXiv preprint arXiv:2004.10934, 2020. (Last accessed: 18.12.2024).

19. R. Khanam, M. Hussain. YOLOV11: An overview of the key architectural enhancements. arXiv:2410.17725v1 [cs.CV] 23 Oct 2024. https://arxiv.org/pdf/2410.17725 (Last accessed: 16.12.2024).

20. Qunmasj-Vision-Studio/Dynamite-Duele rs-Project201. https://github.com/ Qunmasj-Vision-Studio/Dynamite-Duelers-Project201(Last accessed: 16.12.2024).

21. ДСанПіН 3.3.2.007-98. Державні санітарні правила і норми роботи з візуальними дисплейними терміналами електронно-обчислювальних машин.

22. НПАОП 0.00-7.15-18. Вимоги щодо безпеки та захисту здоров'я працівників під час роботи з екранними пристроями.

23. ДБН В.2.5-28-2018. Природне і штучне освітлення.

24. НПАОП 0.00-4.15-98. Положення про розробку інструкцій з охорони праці.

25. Методичний посібник для здобувачів освітнього ступеня «магістр» всіх спеціальностей денної та заочної (дистанційної) форм навчання «Безпека в надзвичайних ситуаціях» / В.С. Стручок –Тернопіль: ФОП Паляниця В. А. 156 с.

26. Навчальний посібник «Техноекологія та цивільна безпека. Частина «Цивільна безпека»» / автор-укладач В.С. Стручок– Тернопіль: ФОП Паляниця В. А. 156 с.

27. Луцик Н.С., Луцків А.М., Осухівська Г.М., Тиш Є.В. Програма та методичні рекомендації з проходження практики за тематикою кваліфікаційної роботи для студентів спеціальності 123 «Комп'ютерна інженерія» другого (магістерського) рівня вищої освіти усіх форм навчання. Тернопіль: ТНТУ. 2024. 45 с.

28. Луцик Н.С., Луцків А.М., Осухівська Г.М., Тиш Є.В. Методичні рекомендації до виконання кваліфікаційної роботи магістра для студентів спеціальності 123 «Комп'ютерна інженерія» другого (магістерського) рівня вищої освіти усіх форм навчання. Тернопіль. 2024. 44 с.

29. Варавін А.В., Лещишин Ю.З., Чайковський А.В. Методичні вказівки до

виконання курсового проєкту з дисципліни «Дослідження і проєктування комп’ютерних систем та мереж» для здобувачів другого (магістерського) рівня вищої освіти спеціальності 123 «Комп’ютерна інженерія» усіх форм навчання. Тернопіль: ТНТУ, 2024. 32 с.

# ДОДАТКИ

Advances in Science and Technology Research Journal

*Advances in Science and Technology Research Journal* 2024, 18(2), 296–304
https://doi.org/10.12913/22998624/184343
ISSN 2299-8624, License CC-BY 4.0

Received: 2024.01.09
Accepted: 2024.02.20
Published: 2024.03.14

# Artificial Intelligence Based Emergency Identification Computer System

Diana Velychko[1], Halyna Osukhivska[1], Yuri Palaniza[2], Nadiia Lutsyk[1], Łukasz Sobaszek[3*]

[1] Computer Systems and Networks Department, Faculty of Computer Information Systems and Software Engineering, Ternopil Ivan Puluj National Technical University, Ternopil, Ukraine

[2] Department of Radio Engineering Systems, Faculty of Applied Information Technologies and Electrical Engineering, Ternopil Ivan Puluj National Technical University, Ternopil, Ukraine

[3] Department of Information Technology, Faculty of Mathematics and Information Technology, Lublin University of Technology, Lublin, Poland

* Corresponding author's e-email: l.sobaszek@pollub.pl

## ABSTRACT

The use of Artificial Intelligence is currently being observed in many areas of life. In addition to assisting in intellectual work, solving complex computational problems, or analyzing various types of data, the aforementioned techniques can also be applied in the process of providing security to people. The paper proposes an emergency identification system based on Artificial Intelligence that aims to provide timely detection and notification of dangerous situations. The proposed solution consider the position of a person "hands up" as an emergency situation that will indicate a potential danger for a person. Because people in the face of potential danger are mostly forced to raise their hands up and this pose attracts attention, emphasizes the emotional reaction to certain events and is usually used as a sign of risk or as a means of subjugation. The system should recognize the pose of a person, detect it, and consequently inform about the threat. In this paper, an AI based emergency identification system was proposed to detect the human pose "hands up" for emergency identification using the PoseNet Machine Learning Model. The assumption consists that the utilization only of 6 key points made allows reducing the computing resources of the system since the conclusion is made taking into account a smaller amount of data. For the study, a dataset of 1510 images was created for training an Artificial Intelligence model, and the decisions were verified. Supervised Machine Learning methods are used to classify the definition of an emergency. Alternative methods: Support Vector Machine, Logistic Regression, Naïve Bayes Classifier, Discriminant Analysis Classifier, and K-nearest Neighbours Classifier based on the accuracy were evaluated. Overall, the paper presents a comprehensive and innovative approach to emergency identification for quick response to them using the proposed system.

**Keywords:** artificial intelligence, deep learning, detection, emergency system, computer system.

## INTRODUCTION

In today's world, the increasing number of terrorist attacks, assaults, thefts, and other emergencies that pose a threat to human security has become a serious challenge for law enforcement agencies, security services, businesses and citizens. Existing systems are not always able to provide timely detection and notification of a dangerous situation. A computerised system that identifies emergencies can be an effective tool for preventing and minimising the consequences of emergencies. The use of artificial intelligence will allow real-time recognition of emergencies and automatic notification.

Papers [1–4] describe the systems that allow the identification of various dangerous situations. The authors of study [1] propose a hazard detection technology based on the analysis of multimodal data from several sensors on a smartphone and

smartwatch, as well as contextual data from several people and places over time. The proposed Multilog Data Analysis (MLDA) helps to detect dangerous situations that are abnormal by analysing the contextual state of people, objects, and places.

The system described in [2] utilizes non-invasive and non-intrusive sensors, RFID tags, and GPS to collect and analyze real-time physiological signals in order to determine when a child is in danger. The system makes assumptions about the state of danger based on the verification of specific biometric responses to certain situations, using a self-learning algorithm developed for this architecture. In the paper [3], the authors propose a model that uses artificial intelligence to detect objects (objects) and human actions that could provoke a dangerous situation, as well as the relationship between them, in CCTV images. There are also systems that are used to determine the emergence of human conditions at home.

The authors of [4] proposed a system that allows the detection of dangerous situations, such as a person falling, based on posture detection. The various security automation systems and technologies are described in the paper [5]. The authors analysed different systems, such as Intelligent acoustic and vibration recognition/alert systems for security breaching detection, proximity danger identification, and perimeter protection, home security system using Arduino UNO, Low Power Accelerometer Based Intrusion and Tamper Detector, IoT based Intruder Detection System Using GSM, Identification of Damaging Activities for Perimeter Security, Smart IoT Security System for Smart-Home Using Motion Detection and Facial Recognition, Ambient Intelligence and IoT Based Decision Support System for Intruder Detection, Theft Detection System using PIR Sensor, Ambient Intelligence and IoT Based Decision Support System for Intruder Detection, Theft Detection System using PIR Sensor, etc.

In this article, we propose to use a different idea to identify emergency situations, namely, by the specific behaviour of people in the room, the "hands up" posture, since people in potential danger are mostly forced to raise their hands up as a sign of risk or as a means of submission. This pose attracts attention, emphasises an emotional reaction to certain events, and is usually used when there is a threat. In the context of a crime, the "hands up" pose is a serious warning signal and can cause fear and anxiety among people at the scene, which may indicate robbery, intimidation or detention, as well as that a person is being coerced. Therefore, it is advisable for the system to detect an emergency situation based on the "hands up" pose. In this paper, we propose a system that detects the "hands up" pose and, based on this, informs about an emergency. The system recognises the "hands up" pose using artificial intelligence.

Artificial intelligence is already commonly used for human pose recognition. Several researchers have studied the problem of human pose recognition and have determined the most effective algorithms for this purpose. Their research is described in the literature [6–10]. These studies mainly consider different algorithms and verify the accuracy of the determination of key joints in the human body. However, these studies do not take into account the particular posture that a person adopts for a given set of data, i.e., for a given set of joint coordinates. For some purposes, it is not necessary to have information about a person's posture at any given time, but it is necessary for the model to identify and report a particular posture that may signal danger, i.e., the "hands up" posture.

Therefore, the task is to develop a system that recognizes a specific pose – "hands up". Such a system can be used in different premises for different industries, such as banks, shops, schools, and other institutions. In addition, the use of such a system can reduce security costs, as it replaces the human factor in the process of monitoring the video stream.

## System architecture and algorithm

The generalized architecture of the AI-based emergency identification system is shown in Figure 1. The main task performed by the system is to identify a dangerous situation for a person, namely, to determine the person's "hands up" pose.

According to the proposed architecture, the first step is to acquire an image from the video stream of the video surveillance system, for example. Next, the video is recorded and transmitted, i.e., "captured", converted into digital signals, processed, adjusted, and transmitted to the server. The video information is displayed and access to the module is provided on a smartphone, tablet, or PC via a browser, and the emergency situation is identified using artificial intelligence. It is proposed to store data (i.e., received video recordings and emergency history) in the cloud or on physical

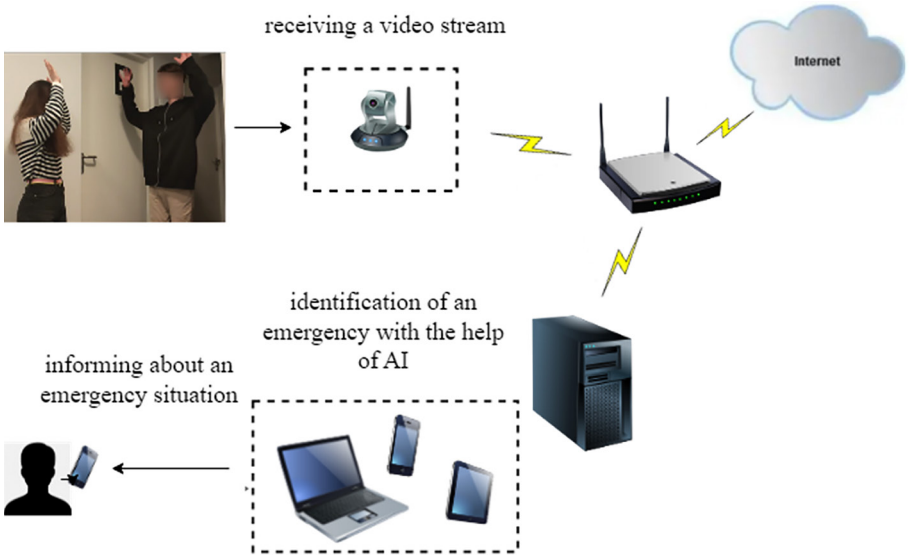


**Fig. 1.** General architecture of the system

server storage. It is proposed to use a Telegram bot to send a message about an emergency.

This architecture allows you to access video content at any time, implement the function of identifying an emergency (hands up), and provide the ability to inform about an emergency. A pose detection algorithm is used to detect an emergency. This component of the system analyses the received video data and determines the pose of a person's "hands up". Pose recognition uses pose and orientation to predict and track the location of a person or object. Accordingly, pose detection allows applications to estimate the spatial position of a body ("pose") in an image or video. Pose detection is performed by finding key points on a person or object.

Assessing body posture is a challenging task because the appearance of the body is dynamically changing, and the angle of view and external context also have an impact. In the field of human posture estimation, there are two main approaches: bottom-up and top-down methods [7]. Bottom-up methods involve first estimating each individual body joint and then grouping them to determine the overall pose. Top-down methods first detect a person in the image using a person detector and then estimate the body joints within the bounding boxes of the detected person [8]. On the other hand, top-down methods start with running a person detector to identify the person in the image and then estimate the body joints within the detected bounding boxes [6]. The ultimate goal of human posture estimation is to predict the positions of the different body parts and joints in photographs or videos. Accurate estimation of body position is important for identifying human actions since movement patterns are often influenced by body posture [6].

In addition, the posture estimation can be performed in both 2D and 3D. 2D human pose estimation involves using visual data, such as photographs and videos, to approximate the spatial positioning or location of key points on the human body [9–10]. Such models exist for human pose estimation: kinematic, planar, and volumetric. After analyzing different models for human pose estimation, it was found that the kinematic model, also known as the skeleton-based model, is the most suitable for detecting unusual human behaviour such as the "hands up" posture. Additionally, most techniques use an N-joints rigid kinematic model, which represents the human body as an object with joints and limbs that provide information on body kinematic structure and shape [6, 9]. When assessing a person's pose, the key points will be joints such as elbows, knees, wrists, etc.

In this paper propose to detect the human pose "hands up" for emergency identification using the PoseNet machine learning model. PoseNet is a Deep Learning TensorFlow model that determines a person's pose by estimating body parts defined as key points (17 in this model), ranging from eyes, and ears, to ankles and knees, including wrists, elbows, shoulders, and nose [11]. These points are connected to form the skeletal structure of the body. All the points and the pose itself are also assessed for authenticity [9, 10].

*Advances in Science and Technology Research Journal* 2024, 18(2), 296–304

The AI-based emergency recognition function ("hands up" pose) is implemented using the following algorithm: storyboarding the video received from the camera, recognising key points (joints), determining the coordinates of six informative points, normalising the data, and recognising the "hands up" pose.

The flowchart of the artificial intelligence based emergency identification computer system is shown in Figure 2. The video stream received is first pre-processed, and then divided into separate frames (storyboarded), then in each frame, using Deep Neural Networks, the presence of a person is determined and key points (joints) are recognised. If there is no person in the frame, the next frame is analysed. The next step is to identify informative points and normalise them, and then, based on the location of these points, the trained model decides whether the person is in a "hands up" pose. If the system detects this pose, an emergency notification is sent, if not, the next frame is analysed.

The main informative points used to recognise the "arms up" pose are: "left_shoulder", "right_shoulder", "left_elbow", "right_elbow", "left_wrist", "right_wrist". In other words, the system needs 6 key points (classes) to make a decision instead of 17 and recognises the "arms up" pose according to their location. This solution reduces the computing resources of the system since the conclusion is made taking into account a smaller amount of data (i.e., instead of 17 points in one image frame, only 6 are analysed), and thus the computational complexity of the task is significantly lower.

## DATASET AND METHODS

Since no data set of people with a given pose was found on the Internet, for the testing of the system, therefore, 1510 images were formed, of which 930 are in the "hands up" pose, 580 are poses with other hand positions. They were obtained from videos in which people of different ages and genders, in different rooms and at different distances from the camera, make hand movements, including raising their hands up, imitating the pose that the system should recognise as an emergency.

Figure 3 shows an image from the generated dataset used for training with key points identified using the PoseNet model. The proposed system, using PoseNet, recognises people in images and then recognises key points (joints). The next step is to obtain the coordinates of 6 informative points and normalise them for further analysis. These points are: "left_shoulder", "right_shoulder", "left_elbow", "right_elbow", "left_wrist", and "right_wrist". The data is normalised in the following way: the point is chosen in the middle

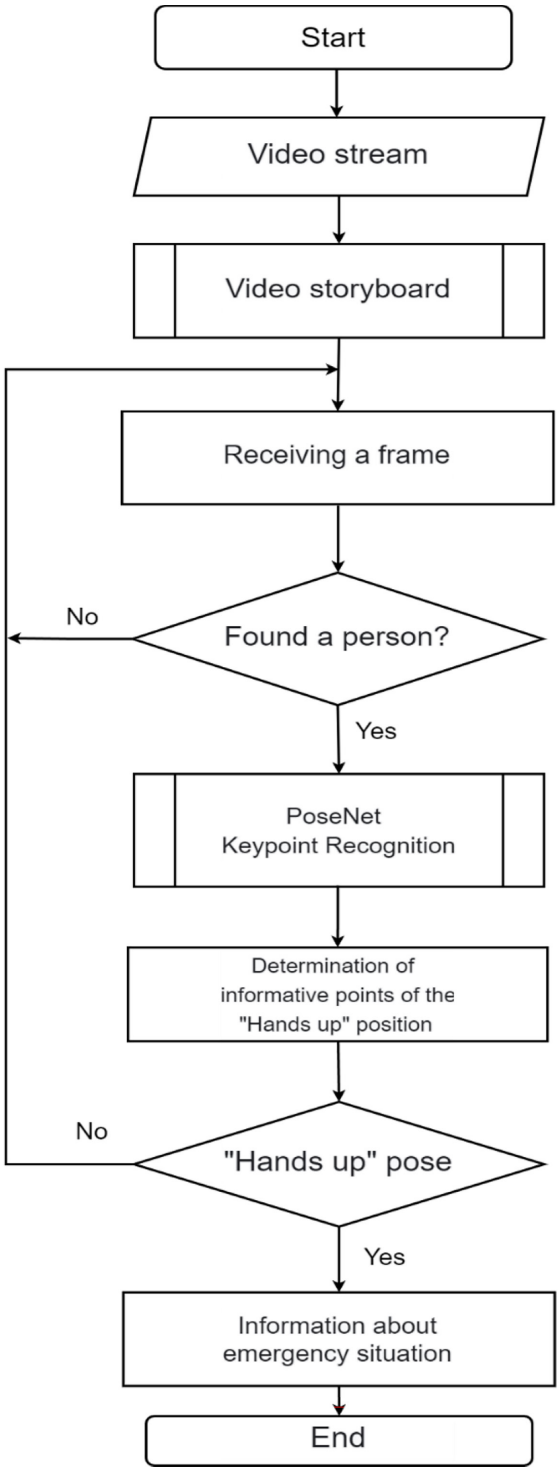


**Fig. 2.** The flowchart of the artificial intelligence based emergency identification computer system

*Advances in Science and Technology Research Journal* 2024, 18(2), 296–304

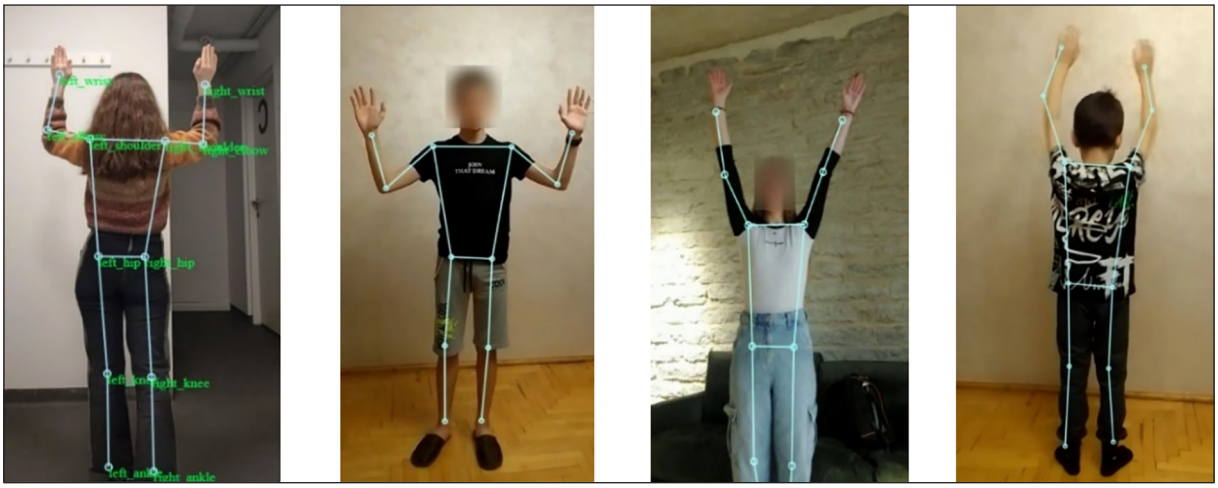

**Fig. 3.** Example of images with key points identified

between the shoulders, and the distance from the middle point to all other points is calculated to find the furthest points in two dimensions ($x$ and $y$). Then the furthest points are fit into the square in such a way that they lie on the sides of the square. Thus, the distances between the furthest points in both dimensions are used as a scaling factor for all coordinates. As a result, all points fit into a square with side lengths equal to 1, so that the value of any coordinate is in the range between 0 and 1. Figure 4 shows the visual representation of the selected points and a point for normalisation – a midpoint.

As a result of normalisation, the coordinates are found in the range from 0 to 1, so the height of the person, the distance from the camera, and the placement of the person in a video or a picture do not affect the prediction result. The graphs of normalised coordinates, when hands are in the "hands up" position and in other positions, are presented in Figure 5. Basically, the input consists of 12 variables ($x$ and $y$ coordinates of those 6 points), and the output is class 0 if the posture is predicted to be "hands up", and 1 if the posture is considered as any other.

## RESULTS AND DISCUSSION

Thus, the generated dataset used for the study consists of 1510 images: 930 are "hands up" ("0") and 580 are other poses, i.e., not "hands up" ("1"). For further classification of the achieved coordinates, the methods of machine learning are utilised. This paper considered an approach that takes into account that there are two classes of data (the location of the hands to determine the pose): the "hands up" pose (an emergency identifier), and any other pose. Therefore, the task of binary classification is essentially solved. For this purpose, we used the support vector method, i.e., SVM (Support Vector Machine), and for comparison, logistic regression, Naïve Bayes classifier, Discriminant Analysis classifier, and K-nearest

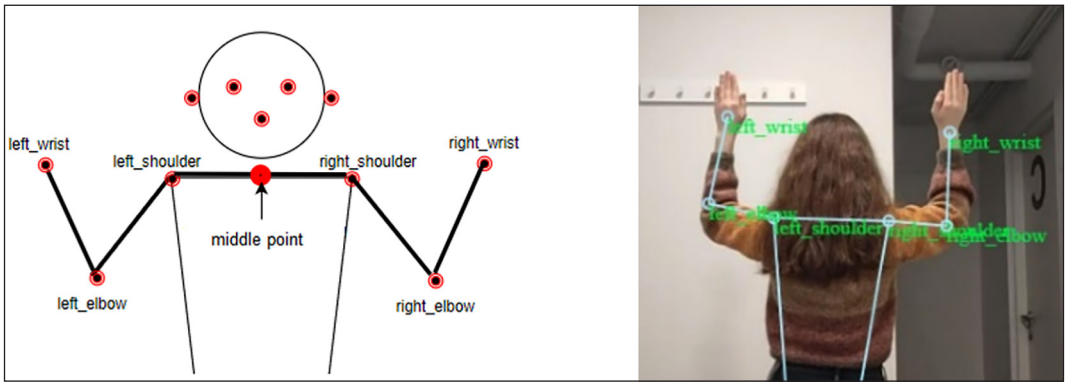


**Fig. 4.** Visualisation of 6 chosen points and a normalisation point

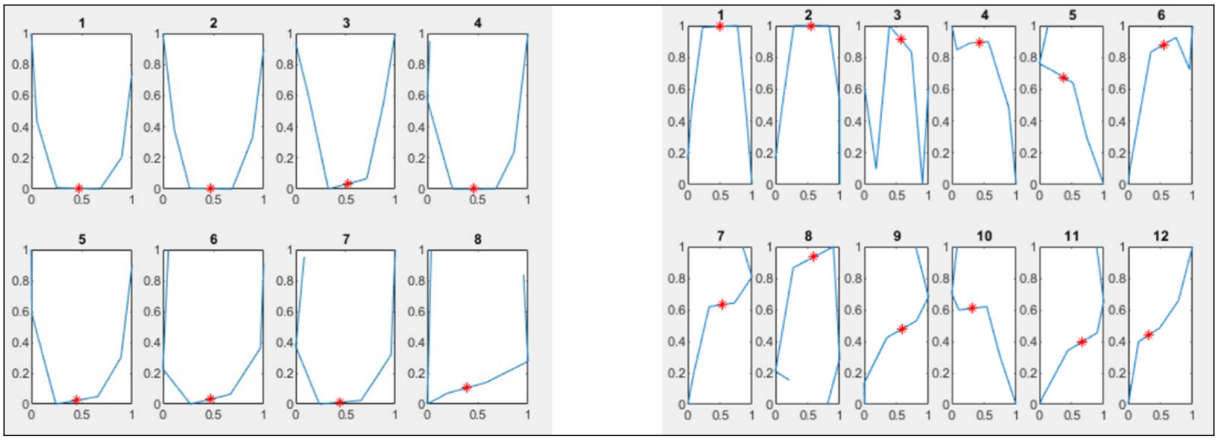

**Fig. 5.** Graphs of normalised coordinates when hands are in (a) position "hands up", (b) other positions

Neighbours classifier. The methods are used for the classification of postures based on 6 normalised coordinates. The implementation of these methods was done in MATLAB.

The classifier is first "trained" on objects from the training set (from the generated dataset), which are predefined with class labels. Then, the trained algorithm predicts the class label from the test sample for each object [12]. The SVM algorithm uses a set of functions that are defined as a kernel (a function that is provided to the machine learning algorithm). In this case, we used Gaussian kernels. By maximising the distance between points and finding the best hyperplane, SVM divides the data into different categories. Given a dataset where each element belongs to one of two categories ("hands" up and other), the SVM is trained and then the classifier is tested.

Figure 6 shows the mismatch matrix of SVM with Gaussian kernels and ROC graph. The matrix displays the total amount of data in each cell, with rows corresponding to the correct class and columns to the predicted class [13]. The diagonal corresponds to the correctly classified classes (930 for the "hands up" pose, 580 for the other). The line at the bottom of the figure shows the percentage of all data that belongs to the correctly and incorrectly classified class. The column in the right corner of the figure shows the percentage of all data belonging to each class that is correctly or incorrectly classified, i.e., accuracy [13, 14]. A ROC curve is a graph showing the performance of a classification model at all classification thresholds [15, 16, 17]. The "steepness" of ROC curves is particularly significant because it is desirable to maximise true positive rates while minimising false positive rates [13].

For comparison, machine learning models based on the following classification methods were built: Logistic Regression; Naive Bayes;

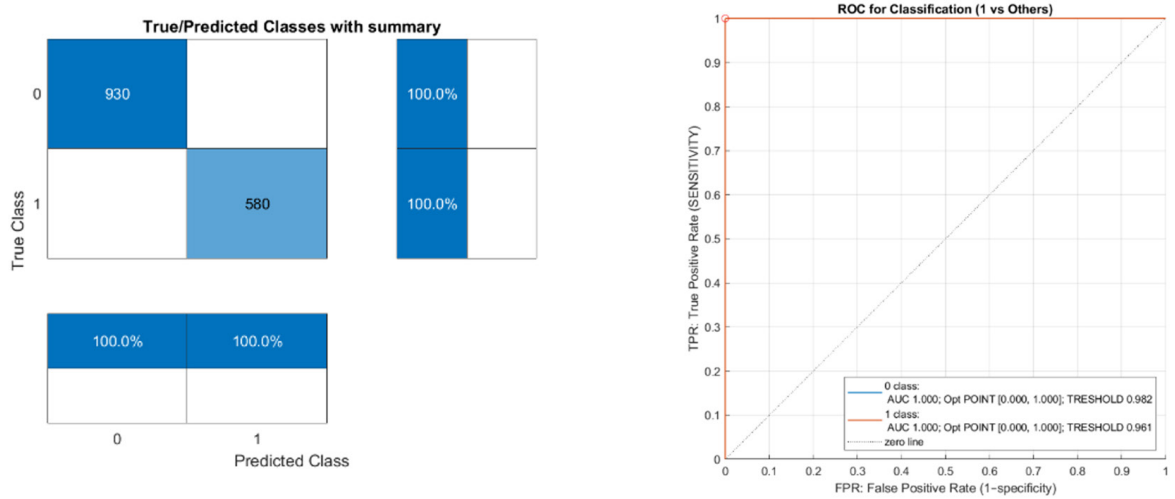


**Fig. 6.** The confusion matrix and ROC for SVM classifiers using a Gaussian Kernel

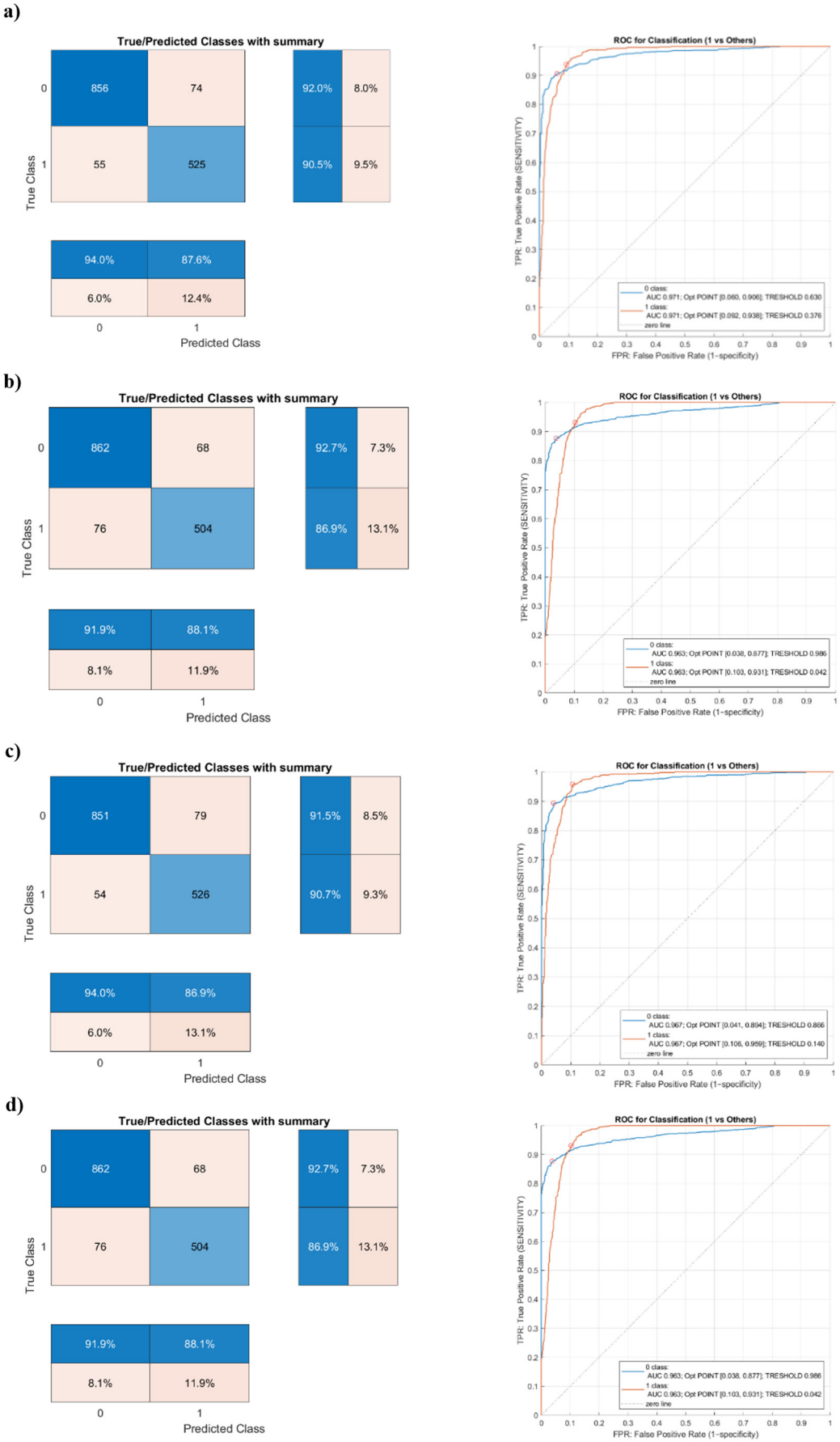


**Figure 7.** Confusion matrices and ROC for classification by: (a) logistic regression; (b) naive bayes; (c) discriminant analysis; (d) K-Nearest Neighbors

**Table 1.** The results of human pose recognition

| Name of classification method | "Hands up" pose | | Other poses | |
|---|---|---|---|---|
| | Total: 930 | | Total: 580 | |
| | Correctly classified | Incorrectly classified | Correctly classified | Incorrectly classified |
| SVM classifiers using a Gaussian Kernel | 930 | 0 | 580 | 0 |
| Logistic regression | 856 | 74 | 525 | 55 |
| Naive bayes | 862 | 68 | 504 | 76 |
| Discriminant analysis | 851 | 79 | 526 | 54 |
| K-Nearest neighbors | 862 | 68 | 504 | 76 |

**Table 2.** The comparison of the prediction results

| Name of classifier | Recall | Precision |
|---|---|---|
| SVM classifiers using a Gaussian Kernel | 100 % | 100 % |
| Logistic regression | 94.0 % | 92.0 % |
| Naive bayes | 91.9 % | 92.7 % |
| Discriminant analysis | 94.0 % | 91.5 % |
| K-Nearest neighbors | 91.9 % | 92.7 % |

Discriminant Analysis; and K-Nearest Neighbours, and their accuracy were evaluated. In general, all investing using the studied methods is done in a similar way. The resulting Confusion matrices and ROC for classification by logistic regression, Naive Bayes, Discriminant Analysis, and K-Nearest Neighbours are shown in Figure 7. The results of human pose recognition by different machine learning methods are presented in Table 1. The comparison of the prediction results by 5 models is presented in Table 2.

Analysing the results, we can conclude that SVM Classifier is ideal for this type of problem, as our dataset shows a prediction accuracy of 100%. The other methods showed slightly lower results, in particular, the accuracy of the K-Nearest Neighbours method for classifying the "hands up" pose in the emergency detection system was the same and amounted to 92.7%, the logistic regression method was 92%, and Discriminant Analysis was 91.5%. Nevertheless, the classification results obtained are high in each case. Therefore, machine learning methods can be effectively used for classification in order to recognise emergencies in the proposed system

## CONCLUSIONS

The article proposes an architecture and algorithm for the operation of an artificial intelligence-based system for identifying emergencies, which is capable of detecting atypical (emergency) situations by determining the "hands up" posture of a person in a room. It is justified to identify an emergency situation using the PoseNet model by six key points, which reduces the computing resources of the system. Supervised machine learning methods are used to classify the definition of an emergency: SVM (Support Vector Machine), Logistic Regression, Naïve Bayes classifier, Discriminant Analysis classifier, and K-nearest Neighbours classifier, and their accuracy were evaluated. The SVM method with Gaussian kernels showed the best result. Further research is needed to optimize the system's performance and evaluate its effectiveness in real-world scenarios. The proposed computer system can facilitate the more efficient work of security services. It can automatically detect emergencies and report them. This reduces the need for constant video stream monitoring and increases the accuracy of identifying dangerous situations, ensuring prompt response to potential dangers and reducing risks from them. Such a system can be used in organizations, enterprises, and institutions of various purposes, such as banks, schools, shopping and entertainment establishments, the service sector, etc., as well as a „smart home" system component. In addition, using such a system can reduce security costs, as it replaces the human factor in the monitoring video stream process.

## REFERENCES

1. Park S.Y., Kwon E., Byon S., Shin W., Jung E., Lee Y. Danger detection technology based on multimodal and multilog data for public safety services. Etri Journal 2021; 44(2): 300–312. https://doi.org/10.4218/etrij.2020-0372
2. Lopes N.V., Santos H., Azevedo A.I. Detection of dangerous situations using a smart internet of things system. In: Rocha A., Correia A., Costanzo S., Reis L. (eds) New contributions in information systems and technologies. Advances in Intelligent Systems and Computing, vol. 354. Springer, Cham, 2015. https://doi.org/10.1007/978-3-319-16528-8_36
3. Jang S., Battulga L., Nasridinov A. Detection of dangerous situations using deep learning model with relational inference. Journal of Multimedia Information System 2020; 7(3): 205–214. https://doi.org/10.33851/jmis.2020.7.3.205
4. Prati A. An intelligent surveillance system for dangerous situation detection in home environments. Unipr, 2016. https://www.academia.edu/21940139/An_Intelligent_Surveillance_System_for_Dangerous_Situation_Detection_in_Home_Environments
5. Yadav, A., Thaker, N., Makwana, D., Waingankar, N., Upadhyay, P. Intruder Detection System: A Literature Review 2021; Proceedings of the 4th International Conference on Advances in Science & Technology (ICAST2021), Available at SSRN: http://dx.doi.org/10.2139/ssrn.3866777
6. Dang Q., Yin J., Wang B., Zheng W. Deep learning based 2D human pose estimation: A survey. Tsinghua Science & Technology 2019; 24(6): 663–676. https://doi.org/10.26599/tst.2018.9010100
7. Gong W., Zhang X., Gonzàlez J., Sobral A., Bouwmans T., Tu C., Zahzah E. Human pose estimation from monocular images: a comprehensive survey. Sensors 2016; 16(12): 1966. https://doi.org/10.3390/s16121966
8. Odemakinde E. Human Pose Estimation with Deep Learning – Ultimate Overview in 2023. Viso.ai, 2023, https://viso.ai/deep-learning/pose-estimation-ultimate-overview/
9. Toshpulatov M., Lee W., Lee S., Roudsari A.H. Human Pose, Hand., Mesh Estimation using Deep Learning: a Survey. The Journal of Supercomputing 2022; 78(6): 7616–7654. https://doi.org/10.1007/s11227-021-04184-7
10. University of Toronto Machine Intelligence Team. Sparse R-CNN: towards more efficient object Detection models. Medium, 2021, https://utorontomist.medium.com/sparse-r-cnn-towards-more-efficient-object-detection-models-fb244178998f
11. Pandey S. Posenet model in ML. OpenGenus IQ: Computing Expertise & Legacy, 2022, https://iq.opengenus.org/posenet-model/
12. Hastie, T., Tibshirani, R., Friedman, J.H. Data Mining, Inference, and Prediction, Second Edition. Springer, New York; 2009, https://doi.org/10.1007/978-0-387-84858-7
13. Fawcett T. An introduction to ROC analysis. Pattern Recognition Letters 2006; 27(8): 861–874. https://doi.org/10.1016/j.patrec.2005.10.010
14. Machine Learning | Google for Developers. (n.d.). Google for Developers. https://developers.google.com/machine-learning/crash-course
15. Alyamani A., Yasniy O. Classification of EEG signal by methods of machine learning. Applied Computer Science 2020; 16(4): 56–63. doi:10.23743/acs-2020-29
16. Shabliy N., Lupenko S., Lutsyk N., Yasniy O., Malyshevska O. Keystroke Dynamics Analysis using Machine Learning Methods. Applied Computer Science 2021; 17(4): 75–83. https://doi.org/10.23743/acs-2021-30.
17. Yasniy O., Lutsyk N., Demchyk V., Osukhivska H., Malyshevska O. The prediction of structural properties of Ni-Ti shape memory alloy by the supervised machine learning methods. ITTAP 2023: 73–78. https://ceur-ws.org/Vol-3628/short1.pdf

**Міністерство освіти і науки України,**
**Тернопільський національний технічний університет**
**імені Івана Пулюя**
**Маріборський університет (Словенія)**
**Технічний університет в Кошице (Словаччина)**
**Каунаський технологічний університет (Литва)**
**Львівський національний університет**
**імені Івана Франка,**
**Гірничо-металургійна академія ім. Станіслава Сташиця (Польща)**
**Луцький національний технічний університет,**
**Чернівецький національний університет**
**імені Юрія Федьковича,**
**Вроцлавський економічний університет (Польща)**
**Університет технологій та економіки**
**імені Хелени Ходковської (Польща)**
**Донбаська державна машинобудівна академія**

*Студентське наукове товариство*

# VI МІЖНАРОДНА

## студентська науково - технічна конференція

# "ПРИРОДНИЧІ ТА ГУМАНІТАРНІ НАУКИ.

# АКТУАЛЬНІ ПИТАННЯ"

**27-28 квітня 2023 р.**

***(збірник тез конференції)***

***Тернопіль 2023***

УДК 654.9

Величко Д. - ст. гр. СІ-41
*Тернопільський національний технічний університет імені Івана Пулюя*

# АНАЛІЗ СИСТЕМ ВИЯВЛЕННЯ ЕКСТРЕНИХ СИТУАЦІЙ

Науковий керівник: Ph.D., доцент Луцик Н.С.

Velychko D.
*Ternopil Ivan Puluj National Technical University*

# ANALYSIS OF DANGEROUS SITUATION DETECTION SYSTEMS

Supervisor: Ph.D., Assoc. Prof. Lutsyk N.S.

Ключові слова: екстрена ситуація, система відеонагляду, штучний інтелект
Keywords: dangerous situation, video surveillance system, artificial intelligence

На сьогоднішній день вкрай важливим завданням є безпека людей. На жаль, бувають обставини, за яких необхідно терміново повідомити про екстрену ситуацію, яка загрожує здоров'ю, життю чи майну людей. Для виявлення та інформування про такі випадки існують різні технічні засоби, наприклад, «тривожна кнопка», системи охоронної сигналізації, системи відеонагляду і т.ін. Але такі системи не завжди можуть забезпечити своєчасне виявлення та інформування про небезпечну ситуацію.

«Тривожна кнопка» - прихована від сторонніх очей клавіша, призначена для миттєвого сповіщення про небезпеку. Цей метод оповіщення ефективно використовується, але недоліком її є те, що для повідомлення про екстрену ситуацію потрібно непомітно натиснути на неї, що не завжди є можливим.

Охоронна сигналізація - це система, яка розрахована на попередження несанкціонованого доступу до приміщення і використовується для захисту об'єкта від вторгнення, викрадання майна або інших небезпечних подій. Сигналізація спрацьовує, за умови, коли давачі реагують на певний сигнал (рух, звук, температуру, концентрацію газів та ін.), але, коли загрожує небезпека безпосередньо людям в приміщенні від інших осіб, то така система відреагувати не здатна. В таких випадках добре себе зарекомендували системи відеонагляду.

Система відеонагляду - це система, яка використовує відеокамери для відстеження подій, що відбуваються. Сучасні системи відеонагляду, використовуючи технології штучного інтелекту, мають можливість попередити, а в подальшому і допомогти виявити зловмисників. Відеокамери можуть розпізнати різні об’єкти, а також здійснювати ідентифікацію особи, порівнюючи її з своєю базою. Такі системи здійснюють передавання відеозображення в реальному часі на пульт спостереження. Але про екстрену ситуацію, викликану погрозою чи нападом безпосередньо на людину, такі системи не інформують миттєво, натомість все залежить від реагування працівника охорони, який слідкує за моніторами.

Тому актуальним завданням є вдосконалення системи відеонагляду функцією ідентифікації екстрених ситуацій з миттєвим оповіщенням про небезпеку. Ідентифікацію небезпечних ситуацій можна реалізувати, використовуючи штучний інтелект, а саме, розпізнаючи позу людини, яка вказує на небезпеку. Такою позою, яку система сприйматиме як екстрену ситуацію, запропоновано вважати позу «руки вгору».

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**

**ТЕРНОПІЛЬСЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ ІМЕНІ ІВАНА ПУЛЮЯ**

**М А Т Е Р І А Л И**

**XI НАУКОВО-ТЕХНІЧНОЇ КОНФЕРЕНЦІЇ**

# «ІНФОРМАЦІЙНІ МОДЕЛІ, СИСТЕМИ ТА ТЕХНОЛОГІЇ»

**13-14 грудня 2023 року**

**ТЕРНОПІЛЬ**
**2023**

**УДК 654.9:004.8**
**Діана Величко**
Тернопільський національний технічний університет імені Івана Пулюя

## АЛГОРИТМ РОБОТИ КОМП'ЮТЕРИЗОВАНОЇ СИСТЕМИ ВІДЕОНАГЛЯДУ З ФУНКЦІЄЮ ІДЕНТИФІКАЦІЇ ЕКСТРЕНИХ СИТУАЦІЙ

**Diana Velychko**

## ALGORITHM OF OPERATION OF A COMPUTERIZED VIDEO SURVEILLANCE SYSTEM WITH THE FUNCTION OF IDENTIFYING EMERGENCY SITUATIONS

Сучасні технології надають можливість створення комп'ютеризованих систем відеонагляду, які здатні виявляти нетипові (екстрені) ситуації. Ключовим елементом таких систем є штучний інтелект, який забезпечує їх ефективне функціонування. В запропонованій системі екстрена ситуація ідентифікується шляхом визначення пози «руки вгору» в людини, яка знаходиться в приміщенні [1]. Визначення такої сигнальної пози здійснюється за допомогою моделі PoseNet [2].

PoseNet - це модель глибокого навчання TensorFlow, яка визначає позу людини, оцінюючи частини тіла, визначені як 17 ключових точок, починаючи від очей і вух до колін і щиколоток, включаючи зап'ястя, лікті, плечі та ніс [2]. Ці точки з'єднані між собою, утворюючи скелетну структуру тіла. В запропонованій комп'ютеризованій системі основними інформативними точками для розпізнавання пози "руки вгору" є: "ліве плече", "праве плече", "лівий лікоть", "правий лікоть", "ліве зап'ястя", "праве зап'ястя". Тобто, система приймає рішення за 6 ключовими точками замість 17, і розпізнає позу "руки вгору" відповідно до їх розташування. В такому випадку алгоритм роботи комп'ютеризованої системи відеонагляду з функцією ідентифікації екстрених ситуацій наступний: отриманий відеопотік попередньо опрацьовується, потім розбивається на окремі кадри (розкадровується), після чого в кожному кадрі за допомогою глибоких нейронних мереж визначається присутність людини. Якщо в кадрі немає людини, аналізується наступний кадр. Якщо людина виявлена, тоді здійснюється розпізнавання ключових точок, далі визначаються координати шести інформативних точок, проводиться нормалізація даних, і, виходячи з розташування цих точок, навчена модель вирішує, чи знаходиться людина в позі "руки вгору". Якщо система виявляє цю позу, тоді надсилається повідомлення про екстрену ситуацію, якщо ні - аналізується наступний кадр.

Таким чином, за запропонованим алгоритмом система може ідентифікувати нетипову (екстрену) ситуацію для людини в приміщенні, а також така його реалізація дозволяє зменшити обчислювальні ресурси, оскільки висновок про екстрену ситуацію робиться з урахуванням меншої кількості даних (тобто замість 17 точок на зображені людини в кожному кадрі аналізується лише 6), а отже, обчислювальна складність значно нижча.

**Література**

1. Величко Д.В. Аналіз систем виявлення екстрених ситуацій. Природничі та гуманітарні науки. Актуальні питання: Матеріали VI Міжнародної студентської науково - технічної конференції (27-28 квітня 2023 р.). Тернопіль: Тернопільський національний технічний університет ім. І.Пулюя, 2023. С. 116.

2. Posenet Model in ML. URL: https://iq.opengenus.org/posenet-model/ (дата звернення: 28.11.2023).