

Міністерство освіти і науки України  
Харківський національний університет радіоелектроніки

Факультет комп'ютерної інженерії та управління  
(повна назва)

Кафедра Комп'ютерних інтелектуальних технологій та систем  
(повна назва)

**АТЕСТАЦІЙНА РОБОТА**  
**Пояснювальна записка**

Рівень вищої освіти Другий (магістерський) рівень

Розпізнавання поведінки людини  
на основі байєсівської системи  
комп'ютерного зору  
(тема)

Виконав:

студент II курсу, групи КіТм-19-1  
Охотников О.С.  
(прізвище, ініціали)

Спеціальність 123 – Комп'ютерна інженерія  
(код і повна назва спеціальності)

Тип програми освітньо-професійна  
(освітньо-професійна або освітньо-наукова)

Освітня програма Комп'ютерні інтелектуальні технології  
(повна назва освітньої програми)

Керівник: проф. Руденко О.Г.  
(посада, прізвище, ініціали)

Допускається до захисту

Зав. кафедри КІТС

Руденко О.Г.  
(прізвище, ініціали)

2020 р.

Харківський національний університет радіоелектроніки

Факультет \_\_\_\_\_ комп'ютерної інженерії та управління  
Кафедра \_\_\_\_\_ комп'ютерних інтелектуальних технологій та систем  
Рівень вищої освіти \_\_\_\_\_ другий (магістерський)  
Спеціальність (напрямок) \_\_\_\_\_ 123 – Комп'ютерна інженерія  
(код і назва)  
Освітня програма \_\_\_\_\_ Комп'ютерні інтелектуальні технології  
(повна назва)

ЗАТВЕРДЖУЮ:

Зав. кафедри \_\_\_\_\_  
(підпис)

“ \_\_\_\_\_ ” \_\_\_\_\_ 2020 р.

**ЗАВДАННЯ**

**НА АТЕСТАЦІЙНУ РОБОТУ**

студентові \_\_\_\_\_ Охотникову Олександр Сергійовичу  
(прізвище, ім'я, по батькові)

1. Тема роботи Розпізнавання поведінки людини на основі байєсівської системи комп'ютерного зору

затверджена наказом по університету від “ 11 ” листопада 2020 р. № 1582 Ст

2. Термін подання студентом роботи до екзаменаційної комісії \_\_\_\_\_ 10 грудня 2020 р.

3. Вхідні дані до роботи \_\_\_\_\_

1. Система комп'ютерного зору

2. Прихована марківська модель (ПММ)

3. Пов'язана прихована марковська модель (ППММ)

4. Машинний зір

4. Перелік питань, що потрібно опрацювати в роботі \_\_\_\_\_

1. Аналіз існуючих підходів обробки поведінки людини

2. Розробка системи виявлення і супроводу рухомих об'єктів за ознакою їх руху

3. Сегментація та відстеження

4. Розробка моделі поведінки

5. Розробка архітектури агента

6. Отримання експериментальних даних

7. Порівняння архітектур ПММ та ППММ

8. Висновки

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій (слайдів) \_\_\_\_\_

Демонстраційні матеріали. Плакати – 10 арк. ф. А4

6. Консультанти розділів роботи (заповнюється за наявності консультантів згідно з наказом, зазначеним у п. 1 )

Найменування розділу	Консультант (посада, прізвище, ім'я, по батькові)	Позначка консультанта про виконання розділу	
		підпис	дата

### КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Термін виконання етапів роботи	Примітка
1	Аналіз проблеми та огляд існуючих рішень	29.09.20—15.10.20	
2	Вибір технології розробки та інструментальних засобів	15.10.20—20.10.20	
3	Розробка системи виявлення і супроводу	20.10.20—10.11.20	
4	Відлагодження програмних модулів	10.11.20—25.11.20	
5	Отримання експериментальних даних	25.11.20—01.12.20	
6	Оформлення матеріалів атестаційної роботи	01.12.20—15.12.20	

Дата видачі завдання 11 листопада 2020 р.

Студент \_\_\_\_\_  
(підпис)

Керівник роботи \_\_\_\_\_  
(підпис)

проф. Руденко О.Г.  
(посада, прізвище, ініціали)

## РЕФЕРАТ

Пояснювальна записка атестаційної роботи: 86 с., 50 рис., 2 табл., 1 дод., 48 джерел.

### ВІЗУАЛЬНЕ СПОСТЕРЕЖЕННЯ, ВИЯВЛЕННЯ ЛЮДЕЙ, СТЕЖЕННЯ, РОЗПІЗНАВАННЯ ЛЮДСЬКОЇ ПОВЕДІНКИ, ПРИХОВАНИЙ МАРКІВСЬКА МОДЕЛЬ

Метою атестаційної роботи є досліджена система комп'ютерного зору і машинного навчання в реальному часі для моделювання і розпізнавання поведінки людини в задачі візуального спостереження

У ході виконання атестаційної роботи буде розроблена система, яка особливо зацікавлена в виявленні, коли відбуваються взаємодії між людьми, і класифікації типу взаємодії. Приклад цікавої поведінки взаємодії включають в себе слідування за іншою людиною, зміна свого шляху, щоб зустрітися з іншим, і так далі

## ABSTRACT

86 pages, 50 figures, 2 tables, 1 appendices, 48 sources.

VISUAL SURVEILLANCE, PEOPLE DETECTION, TRACKING,  
HUMAN BEHAVIOR RECOGNITION, HIDDEN MARKOV MODELS

The major goal of this thesis is to study a system of computer vision and machine learning in real time for modeling and recognizing human behavior in the task of visual observation

In order to certification work, a system will be developed that is particularly interested in identifying when interactions between people occur and classifying the type of interaction. Examples of interesting interaction behaviors include following another person, changing your path to meet another, and so on

# ЗМІСТ

\_Точ59148401

ЗМІСТ .....	10
ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ .....	12
ВСТУП.....	13
1 АНАЛІЗ ІСНУЮЧИХ ПІДХОДІВ ОБРОБКИ ПОВЕДІНКИ.....	14
1.1. Задачі машинного зору та області його застосування .....	14
1.1.1 Машинне зір в даний час .....	14
1.1.2. Основні завдання машинного зору .....	15
1.1.3. Методи обробки зображення .....	17
1.1.4. Компоненти системи комп'ютерного зору .....	18
1.1.5. Принципи функціонування систем машинного зору .....	19
1.1.6. Система виявлення і супроводу рухомих об'єктів за ознакою їх руху .....	21
1.2. Модулі обробки поведінки .....	22
1.3. Інший підхід для моделювання міжособистісних взаємодій .....	23
1.4. Баєсівський підхід до моделювання .....	23
1.5 Прихована марківська модель: стани та зв'язки.....	25
1.5.1 Класична ПММ.....	26
1.5.2 Пов'язані ПММ .....	27
1.5.3 Оцінка параметрів .....	28
1.5.6 Оцінка параметрів максимальної правдоподібності .....	30
2 СЕГМЕНТАЦІЯ ТА ВІДСТЕЖЕННЯ .....	32
2.1 Об'єктне представлення .....	33
2.2 Вибір функцій для відстеження.....	38
2.3 Виявлення об'єктів .....	41
2.3.1 Точкові Детектори.....	41
2.3.2 Віднімання фону .....	44

2.3.3 Сегментація.....	48
2.3.4 Стеження за об'єктом .....	53
2.3.5 Відстеження Точок .....	56
<b>3 МОДЕЛЬ ПОВЕДІНКИ .....</b>	<b>64</b>
3.1 Огляд системи.....	64
3.2 Моделі поведінки .....	65
3.2.1 Візуальне розуміння за допомогою графічних моделей: ПММ і ППММ .....	67
3.3 Синтетичний Поведінковий Агент.....	70
3.3.1. Архітектура Агента .....	71
<b>4 ЕКСПЕРИМЕНТАЛЬНІ ДАНІ.....</b>	<b>73</b>
4.1 Поведінка Агентів .....	73
4.2 Порівняння архітектур ПММ і ППММ з даними синтетичного агента .....	76
4.2 Поведінка Пішоходів.....	80
4.2.1 Збір та попередня обробка даних .....	81
4.2.2 Моделі поведінки та результати .....	82
4.2.3 Чутливість параметрів.....	84
<b>ВИСНОВОК .....</b>	<b>85</b>
<b>ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ .....</b>	<b>86</b>
<b>ГЮОК.XXXXXXX.XXX ВД .....</b>	<b>90</b>

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І  
ТЕРМІНІВ

ПММ – Прихована Марківська Модель

ППММ – Пов'язані Приховані Марковські Моделі

BLOB – Бінарно Великий Об'єкт (англ., Binary Large Object)

PINs – Імовірнісні Мережі Виводу

DAG – Статистичні Спрямовані Ациклічні Графи

## ВСТУП

Інтерес до комп'ютерного зору виник одним з перших в області штучного інтелекту поряд з такими завданнями, як автоматичне доказ теорем і інтелектуальні ігри. Навіть Архітектура першої штучної нейронної мережі перцептрона — була запропонована Френком Розенблаттом, виходячи з аналогії з сітківкою ока, а її дослідження проводилося на прикладі завдання розпізнавання зображень символів.

Прикладов використання ком'ютерного зору можна привести багато. Від систем системи управління автомобілями та медичні системи аналізу зображень до системи контролю точності в будівництві та реконструкцій. Але найскладніша галузь використання ком'ютерного зору це моделювання та розпізнавання поведінки людини.

У цій роботі буде досліджена система комп'ютерного зору і машинного навчання в реальному часі для моделювання і розпізнавання поведінки людини в задачі візуального спостереження. Система особливо зацікавлена в виявленні, коли відбуваються взаємодії між людьми, і класифікації типу взаємодії. Приклад цікавої поведінки взаємодії включають в себе слідування за іншою людиною, зміна свого шляху, щоб зустрітися з іншим, і так далі.

Система поєднує в собі інформацію зверху вниз і знизу вгору в замкнутому контурі зворотного зв'язку, причому і те, і інше компоненти, що використовують статистичний байєсівський підхід. Буде порівняно дві різні архітектури навчання на основі станів, а саме ПММ та ППММ, для моделювання поведінки та взаємодій.

Нарешті, для вирішення проблеми обмежених навчальних даних використовується синтетична навчальна система " Alife—style" для розробки гнучких апріорних моделей розпізнавання людських взаємодій. Задемонструється здатність використовувати ці апріорні моделі для точної класифікації реальної людської поведінки та взаємодій без додаткової настройки або навчання.

## 1 АНАЛІЗ ІСНУЮЧИХ ПІДХОДІВ ОБРОБКИ ПОВЕДІНКИ

За останнє десятиліття в спільнотах комп'ютерного зору та машинного навчання зріс інтерес до проблеми аналізу людської поведінки у відео. Такі системи зазвичай складаються з комп'ютерного зору, низького або середнього рівня системи для виявлення і сегментації рухомого об'єкта — людини або автомобіля. Та модуль інтерпретації вищого рівня, який класифікує рух на "атомарні" типи поведінки, такі як, наприклад, вказівний жест або поворот автомобіля.

### 1.1. Задачі машинного зору та області його застосування

Машинний зір — це застосування комп'ютерного зору для промисловості та виробництва. Областю машинного зору є, як інженерне обладнання і цифрові пристрої вводу/виводу та комп'ютерні мережі, призначені для контролю виробничого обладнання, такі як роботи—маніпулятори або апарати для виведення бракованої продукції.

Дослідження комп'ютерного зору включає не тільки програмне забезпечення, але також апаратне середовище та методи отримання зображень, необхідних для його застосування. Таким чином, він відрізняється від комп'ютерного зору, який, як видається, у більшості книг на цю тему є сферою можливого проектування програмного забезпечення, не надто уважно ставлячись до інтегрованої системи зору (хоча сучасні книги про комп'ютерний зір зазвичай кажуть досить багато про "неприємні реалії" зору, такі як усунення шуму та аналіз прикусу).

#### 1.1.1 Машинне зір в даний час

В даний час існує чітка межа між монокулярним і бінокулярним комп'ютерним зором. До першої області відносяться дослідження і розробки в області комп'ютерного зору, пов'язані з інформацією, що надходить від однієї

камери або від кожної камери окремо. До другої області відносяться дослідження і розробки, що мають справу з інформацією, одночасно надходить від двох і більше камер. Кілька камер в таких системах використовуються для вимірювання глибини спостереження. Ці системи називаються стереосистемами.

До теперішнього моменту теорія комп'ютерного зору повністю склалася як самостійний розділ кібернетики, що спирається на наукову і практичну базу знань. Щорічно з даної тематики видаються сотні книг і монографій, проводяться десятки конференцій і симпозіумів, випускається різне програмне та апаратно— програмне забезпечення. Існує ряд науково—громадських організацій, що підтримують і які висвітлюють дослідження в області сучасних технологій, в тому числі технології комп'ютерного зору.

### 1.1.2. Основні завдання машинного зору

В цілому, в завдання систем машинного зору входить отримання цифрового зображення, обробка зображення з метою виділення значущої інформації на зображенні і математичний аналіз отриманих даних для вирішення поставлених завдань.

Однак машинний зір дозволяє вирішувати безліч завдань, які умовно можна розділити на чотири групи (Рисунок 1.1).

Розпізнавання положення. Мета машинного зору в даному застосуванні це — визначення просторового розташування (розташування об'єкта щодо зовнішньої системи координат) або статичного положення об'єкта (в якому становищі знаходиться об'єкт щодо системи координат з початком відліку в межах самого об'єкту) і передача інформації про стан і орієнтації об'єкта в систему управління або контролер.

Прикладом такого додатка може служити вантажно—розвантажувальний робот, перед яким стоїть завдання переміщення об'єктів різної форми з бункера. Інтелектуальна задача машинного зору полягає, наприклад, у визначенні оптимальної базової системи координат і її центру для локалізації центру ваги деталі. Отримана інформація дозволяє роботу захопити деталь належним чином і

перемістити її в належне місце.



Рисунок 1.1 — Завдання машинного зору

**Вимірювання.** У додатках даного типу основне завдання відеокамери полягає в вимірі різних фізичних параметрів об'єкта.

Прикладом фізичних параметрів може служити лінійний розмір, діаметр, кривизна, площа, висота і кількість. Приклад реалізації даного завдання — вимір різних діаметрів горлечка скляної пляшки.

**Інспекція.** У додатках, пов'язаних з інспекцією, мета машинного зору — підтвердити певні властивості, наприклад, наявність або відсутність етикетки на пляшці, болтів для проведення операції зборки, шоколадних цукерок в коробці або наявність різних дефектів.

**Ідентифікація.** В задачах ідентифікації основне призначення відеокамери — зчитування різних кодів (штрих—кодів, 2D—кодів і т. П.) З метою їх розпізнавання засобами камери або системним контролером, а також визначення різних буквено—цифрових позначень. Крім того до завдань даної групи можна віднести системи, які виконують завдання безпеки, такі як ідентифікація особистості і техніки, детектори руху.

### 1.1.3. Методи обробки зображення

У системах машинного зору, для вирішення перерахованих завдань, використовуються різні технології і методи. Нижче перераховані основні методи обробки зображення:

Лічильник пікселів: підраховує кількість світлих або темних пікселів і на основі результату робить необхідні висновки про зображення.

Виділення пов'язаних областей: Зв'язкова область зображення — це, з одного боку, тип об'єкта, все ще дуже близько пов'язаний з растровим зображенням, і в той же час — це вже якась самостійна семантична одиниця, що дозволяє вести подальший геометричний, логічний, топологічний і будь—який інший аналіз зображення

Бінаризація: перетворює зображення в сірих тонах в бінарне (білі і чорні пікселі).

Гістограма і гістограмного обробка: Гістограма характеризує частоту зустрічальності на зображенні пікселів однакової яскравості.

Сегментація: використовується для пошуку та/або підрахунку деталей. Сегментацією зображення називається розбиття зображення на несхожі за деякою ознакою області. Передбачається, що області відповідають реальним об'єктам, або їх частин, а кордони областей відповідають кордонам об'єктів:

- читання штрих—кодів: декодування 1D і 2D кодів, розроблених для зчитуючи або сканування машинами;
- оптичне розпізнавання символів: автоматизоване читання тексту, наприклад, серійних номерів;
- вимірювання: вимірювання розмірів об'єктів в дюймах або міліметрах;
- зіставлення шаблонів: пошук, підбір, і / або підрахунок конкретних моделей;
- інваріантні алгоритми зіставлення точкових особливостей на зображеннях: виявлення і зіставлення точкових особливостей на зображеннях;
- методи ідентифікація особистості по райдужній оболонці ока;

- різні методи відновлення форми об'єкта по зображеннях.

У більшості випадків, системи машинного зору використовують послідовне поєднання цих методів обробки для виконання повного інспектування. Наприклад, система, яка зчитує штрих—код може також перевірити поверхню на наявність подряпин або пошкодження і виміряти довжину і ширину оброблюваних компонентів.

#### 1.1.4. Компоненти системи комп'ютерного зору

Типова система машинного зору складається з однієї або декількох цифрових або аналогових камер (чорно—білі або кольорові) з відповідною оптикою для отримання зображень, підсвічування і об'єкта (Рисунок 1.2), обладнання введення/виведення або канали зв'язку для доповіді про отримані результати. Крім того, важлива і програмна складова систем машинного зору, а саме програмне забезпечення для підготовки зображень до обробки (для аналогових камер це оцифровщик зображень), специфічні додатки програмного забезпечення для обробки зображень і виявлення відповідних властивостей.

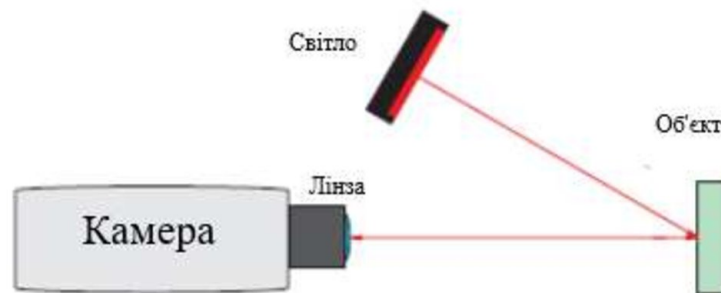


Рисунок 1.2 — Склад типової системи машинного зору

Матриця чутливих елементів, що входять до складу відеокамери, призначена для отримання цифрового зображення. До складу матриці чутливого елемента входить безліч аналого—цифрових перетворювачів, призначених для перетворення інформації про світловий інтенсивності в цифрове значення.

Об'єктив дозволяє камері фокусуватися на певній відстані і отримувати

чітке зображення об'єкта. У разі, коли об'єкт знаходиться поза фокусної відстані, зображення виходить розмитим (розмитим, з нечіткими краями), що погіршує можливість обробки відеоряду. На відміну від звичайних цифрових фотоапаратів з об'єктивами, що підтримують функції автофокусування, в машинному зорі застосовується оптика з фіксованою фокусною відстанню або налаштуванням фокусу. Існують різні типи об'єктивів для самих різних завдань (стандартні, телескопічні, з широким кутом огляду, зі збільшенням та інші), і вибір правильного типу оптики — важливий етап при проектуванні системи машинного зору.

Підсвічування — ще один важливий елемент в машинному зорі. Завдяки використанню різних типів освітлення можна розширити коло завдань, що вирішуються машинним зором. Існує різні типи підсвіток, але найбільш популярним є світлодіодна — в зв'язку з її високою яскравістю. При цьому сучасний рівень розвитку світлодіодної техніки забезпечує великий термін служби пристрою і мале енергоспоживання.

#### 1.1.5. Принципи функціонування систем машинного зору

Послідовність дій, виконуваних системою машинного зору, можна представити як на рисунку 1.3.

Зображення, отримане з камери, потрапляє в загарбник кадрів або в пам'ять комп'ютера. Загарбник кадрів — це пристрій, який перетворює вихідні дані з камери в цифровий формат (як правило, це двовимірний масиву чисел) і розміщує зображення в пам'яті комп'ютера, так щоб воно могло бути оброблено за допомогою програмного забезпечення для машинного зору.

Програмне забезпечення, як правило, робить кілька кроків для обробки зображень. Часто зображення для початку обробляється з метою зменшення шуму або конвертації безлічі відтінків сірого в просте поєднання чорного і білого (бинаризації). Після початкової обробки програма буде вважати, проводити вимірювання і/або визначати об'єкти, розміри, дефекти та інші характеристики

зображення. В якості останнього кроку, програма пропускає або забраковується деталь відповідно до заданих критеріях. Якщо деталь йде з браком, програмне забезпечення подає сигнал механічному пристрою для відхилення деталі; інший варіант розвитку подій, система може зупинити виробничу лінію і попередити людини працівника для вирішення цієї проблеми, і що призвело до невдачі.

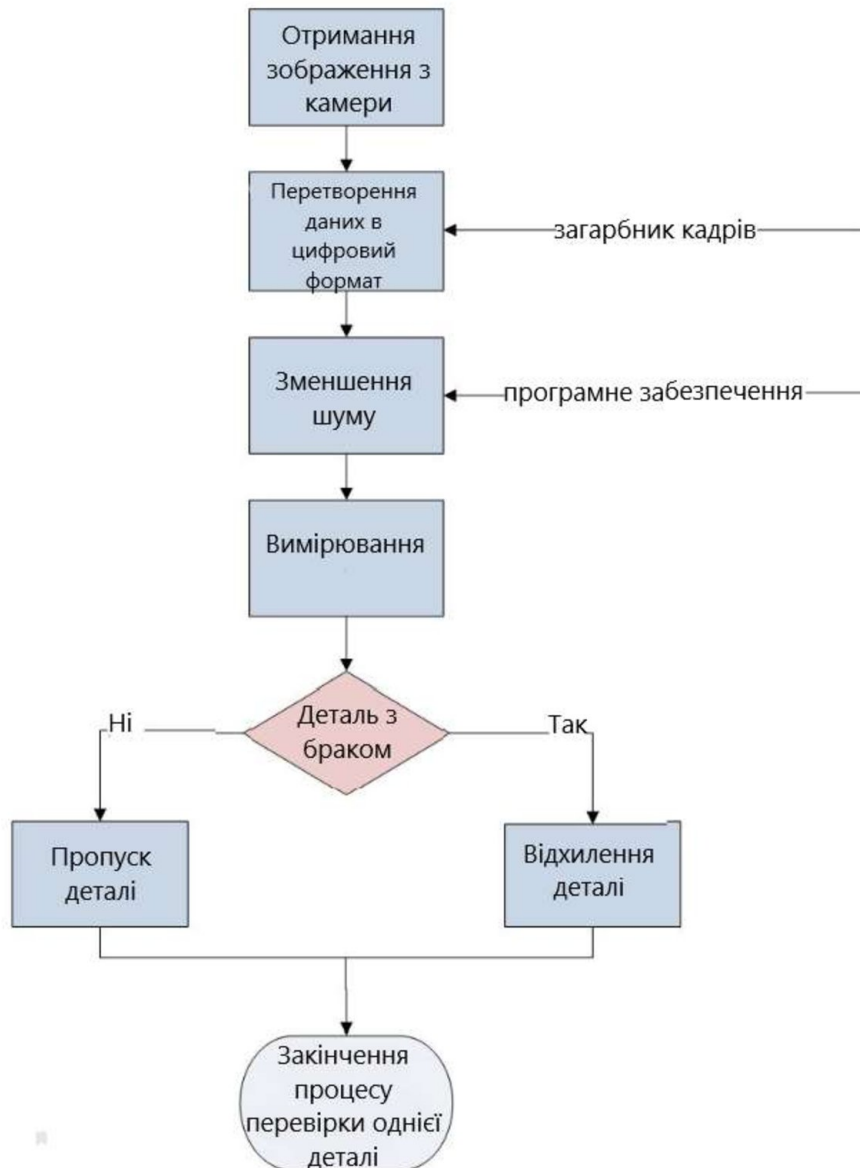


Рисунок 1.3 — Послідовність дій системи машинного зору

Хоча більшість систем машинного зору покладатися на «чорно—білі» камери, використання кольорових камер стає все більш поширеним явищем. Крім того, все частіше системи машинного зору використовують цифрові камери прямого підключення, а не камери з окремим загарбником кадрів, що скорочує

витрати і спрощує систему.

#### 1.1.6. Система виявлення і супроводу рухомих об'єктів за ознакою їх руху

Технологія визначення руху в поле зору камери, або motion detection, стала однією з перших комерційних технологій на споживчому ринку Web—камер. Принцип роботи технології дуже простий, оскільки передбачається, що камера нерухома, а отже, нерухомий і фон. Рухаються тільки об'єкти. Хоча існують і такі технології, які дозволяють детектувати рухомі об'єкти на рухомому тлі.

Розроблена в ІІТ (Інститут інформаційних технологій) система призначена для автоматичного виділення і супроводу малорозмірних об'єктів за ознакою їх руху на зображеннях, отриманих з рухомою відеокамери.

Система, реалізована на базі персонального комп'ютера, здійснює введення з відеокамери, візуалізацію і обробку цифрових зображень рухомих сцен в реальному часі. Система проводить оцінку і компенсацію загального зсуву зображення сцени, що відбувається за рахунок власного відносного руху приймача зображення, а також виділення рухомих малорозмірних об'єктів за ознакою їх руху і одночасний супровід руху декількох виявлених об'єктів. На рисунку 1.4 показані: зліва — кадр відеопослідовності з виділеними рухомими об'єктами; праворуч — стадії виділення рухомих об'єктів.

У систем виявлення і супроводу рухомих об'єктів є різні функції, такі як:

- виявлення нових об'єктів;
- сцени спостереження;
- виявлення зниклих об'єктів сцени спостереження;
- реєстрація нової сцени спостереження;
- контроль відсутності зрушень камери щодо сцени спостереження.

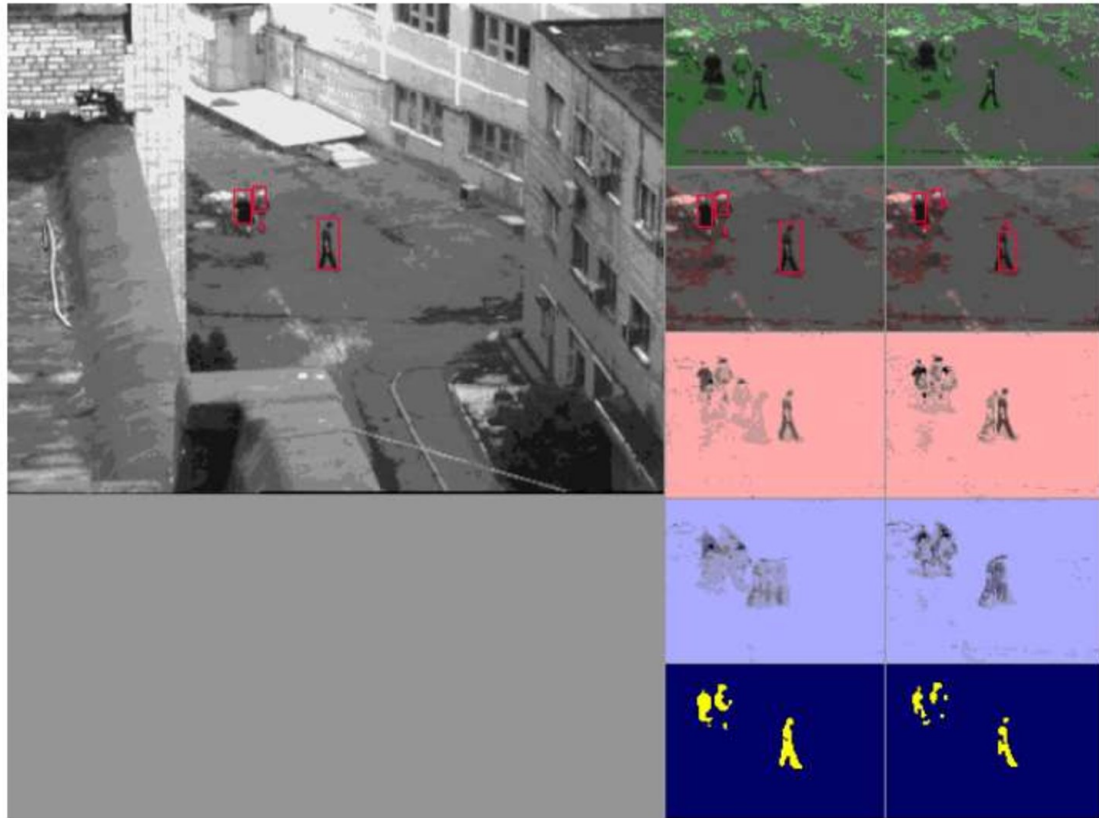


Рисунок 1.4 — Виділення рухомих об'єктів сцени спостереження

## 1.2. Модулі обробки поведінки

Тим не менш, було відносно мало спроб зрозуміти людську поведінку, яке має значну протяжність у часі, особливо коли воно пов'язане із взаємодією між людьми. Цей рівень інтерпретації є метою

даної статті з наміром побудувати системи, які можуть впоратися зі складністю багатолюдських пішохідних і дорожніх сцен.

Це обчислювальні завдання, поєднує в собі елементи штучного інтелекту, машинного навчання і комп'ютерного зору. Завдання представляє складні проблеми в обох сферах:

- комп'ютерний зір потребує в режимі реального часу, точне і надійне виявлення і відстеження об'єктів, що представляють інтерес в невимушеній обстановці;
- машинне навчання і штучний інтелект необхідний для моделі поведінки

взаємодіючих агентів потрібні, щоб інтерпретувати набір сприйманих дій і виявлення можливого аномального поведінки або потенційно небезпечних ситуаціях.

Крім того, всі модулі обробки повинні бути інтегровані узгодженим чином.

### 1.3. Інший підхід для моделювання міжособистісних взаємодій

У цій роботі буде запропоновано модель міжособистісних взаємодій, яка полягає у використанні методів контрольованого статистичного машинного навчання. Ця модель буде навчати систему розпізнавання нормальної поведінки однієї людини і звичайних міжособистісних взаємодій. Основна проблема статистичного підходу, особливо при моделюванні рідкісних або аномальних моделей поведінки, полягає в обмеженому числі прикладів такої поведінки для навчання моделей. Таким чином, основна увага в нашій роботі приділяється ефективній байєсівській інтеграції обох апіорних знань моделі з доказами з даних (шляхом налаштування параметрів конкретної ситуації).

Мета цієї роботи полягає в тому, щоб мати можливість успішно застосовувати систему до будь—якої нормальної ситуації взаємодії кількох людей без додаткової підготовки.

Інша потенційна проблема виникає, коли в системі представляється абсолютно новий патерн поведінки, після того як система була навчена на декількох різних ділянках, раніше неспостережувана поведінка буде (за визначенням) рідкісним і незвичайним. Щоб пояснити таку нову поведінку, система повинна вміти розпізнавати нову поведінку і будувати моделі з нього.

### 1.4. Байєсівський підхід до моделювання

Байєсівські підходи, на відміну від пошуку по решітці і його модифікації — випадкового пошуку, відстежує результати минулих оцінок. Оцінюючи гіперпараметри, які здаються більш багатообіцяючими з минулих результатів,

байєсовські методи можуть знаходити найкраще налаштування моделі, ніж випадковий пошук, за меншу кількість ітерацій (рисунки 1.5).

Застосована до гіперпараметричної оптимізації байєсівська оптимізація будує стохастичну модель функції відображення зі значень гіперпараметра в цільову функцію, застосовану на безлічі перевірок. Шляхом ітеративного застосування перспективної конфігурації гіперпараметрів, заснованої на поточній моделі, а потім її оновлення, байєсівська оптимізація прагне зібрати якомога більше інформації про цю функцію і, зокрема, місце оптимуму.

В даному випадку, крім критерію зупинки, необхідно визначити функцію, що приймає значення гіперпараметрів і виводить якусь оцінку, яку ми хочемо максимізувати або мінімізувати, а також критерій, званий функцією вибору, для оцінки того які гіперпараметри варто вибирати наступними.

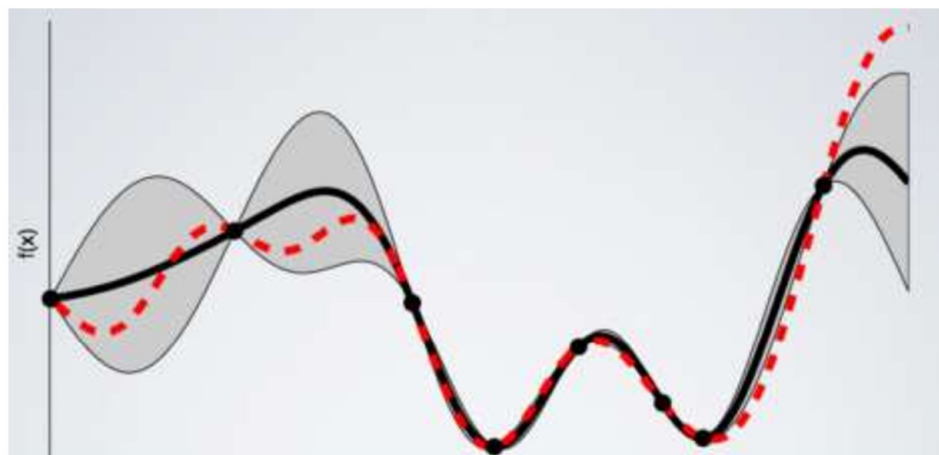


Рисунок 1.5 – Графік апроксимації (чорний) функції (червоний), яка описує ідеальний кортеж гіперпараметрів

На практиці байєсівська оптимізація показала кращі результати з меншою кількістю обчислень в порівнянні з пошуком по решітці і випадковим пошуком через можливість судження про якість експериментів ще до їх виконання.

Переваги і недоліки:

- більш швидка збіжність
- якість оптимізаційного пошуку безпосередньо залежить від якості побудованої нами моделі

Тому, у цій роботі було дотримано байєсівського підходу до моделювання,

який включає в себе як попередні знання, так і фактичні дані, вважаючи, що байєсівський підхід забезпечує найкращу основу для роботи з невеликими наборами даних і новими моделями поведінки. Графічні моделі [10], такі як приховані Марковські моделі (ПММ)[11] та пов'язані приховані Марківські моделі(ППММ) [12], [13], [14]. Ці моделі здаються найбільш придатними для моделювання та класифікації людської поведінки, тому що вони пропонують динамічне викривлення часу, добре зрозумілий алгоритм навчання і чітку байєсівську семантику як для індивідуальних (ПММ), так і для взаємодіючих або пов'язаних ПММ генеративних процесів.

Щоб вказати пріоритети в системі, було розроблена структура для побудови і навчальні моделі зацікавленості поведінки з використанням синтетичних агентів [15]. Моделювання за допомогою агентів дає синтетичні дані, які використовуються для навчання попередніх моделей. Ці попередні моделі потім рекурсивно використовуються в байєсівській структурі для отримання реальних поведінкових даних. Цей підхід забезпечує досить простий і гнучкий метод проектування пріорів, який не вимагає сильних аналітичних припущень про форму пріорів.

У експериментах було виявлено, що, комбінуючи такі синтетичні Пріори з обмеженими реальними дані можна легко досягти дуже високої точності розпізнавання різних взаємодій між людьми. Таким чином, система стійка до випадків, в яких є тільки кілька прикладів певної поведінки.

### 1.5 Прихована марківська модель: стани та зв'язки

Для моделювання і розпізнавання поведінки людини в задачі візуального спостереження можна моделювати в просторі станів, а не в просторі спостережень, тобто моделювати взаємодії після першого переведення спостережень в загальну область. Пов'язані приховані Марковські моделі є такими моделями простору станів. Вони утворюють природне продовження стандартних прихованих марковських моделей. У даній роботі ми виконуємо

оцінку параметрів пов'язаних ПММ в рамках байєсівської парадигми, використовуючи вибірку Гіббса і в рамках максимальної правдоподібності, використовуючи алгоритм максимізації очікувань. Відмінності в продуктивності між оцінювачами демонструються на змодельованих даних, а також на біомедичних даних. Показано, що запропонований метод дає значущі результати при порівнянні двох різних сигналів, таких як дихання і ЕЕГ.

### 1.5.1 Класична ПММ

Стандартна ПММ зі  $K$  станами і довжиною  $N$  визначається набором параметрів,  $\theta = \{A, X, \pi\}$ , де  $\pi$  початкова ймовірність стану параметра  $set^1$ ,  $A$  – набір ймовірностей спостереження, обумовлених станом, і  $X$  набір ймовірностей переходу стану. Графічно вона представлена у вигляді розгорнутої моделі суміші, стани якої в часі  $t$  обумовлені станами в часі  $t-1$  (рисунок 1.6) також відомим як властивість Маркова першого порядку.

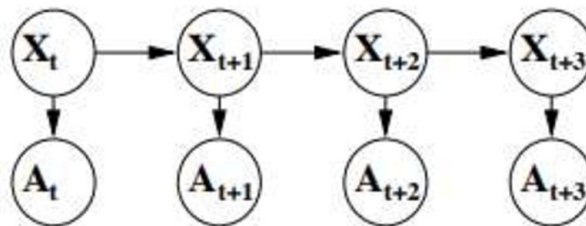


Рисунок 1.6 – Стандартний (однотанцюговий) ПММ

Оцінка часто виконується з використанням структури максимальної правдоподібності [17], де приховані стани ітеративно переоцінюються як очікувані значення і, використовуючи їх, переоцінюються параметри моделі. ПММ можуть бути розширені з точки зору розмірів  $A$  і  $X$  або шляхом зміни розподілів  $A$  і  $X$ . Однак в даній роботі будуть системи з різними топологіями і, зокрема, з розширеннями більш ніж на одну змінну простору станів.

### 1.5.2 Пов'язані ПММ

Одним з можливих розширень багатовимірних систем є пов'язана прихована Марковська модель, як показано на рисунку 1.7. Тут поточний стан залежить від станів його власного ланцюга ( $X$ ) і сусіднього ланцюга ( $Y$ ) на попередньому часовому кроці. Ці моделі були використані в комп'ютерному зору[3] і цифрового зв'язку[19].

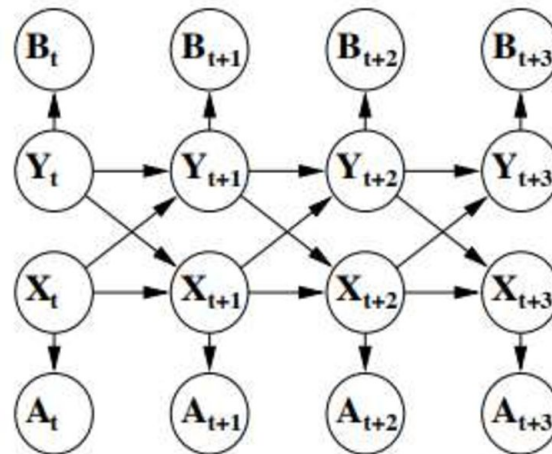


Рисунок 1.7 – Пов'язані ПММ 1

Модель, хоча і являє собою спрямований ациклічний граф (САГ), містить петлю, що ускладнює оцінку параметрів ПММ. Одним з рішень цієї проблеми є злиття вузлів циклу і формування єдиного вузла більш високої розмірності. Хоча це може бути життєздатним підходом для простої моделі, показаної на рисунку 1.7, де змінні стану будуть об'єднані  $X_t, Y_t, X_{t+1}, Y_{t+1}$ , неможливо об'єднати змінні стану для складних топологій, показаних на рисунку 1.8, через результуючий вибух розмірності вузлів[20]. Інший підхід, підхід зрізаних множин[20], також недоцільний, оскільки число змінних зрізаних множин збільшується з довжиною ланцюга  $N$ .

Існує 2 ланцюжки з 2, можливо, різними моделями спостереження. Якщо моделі спостереження  $A, B$  не ідентичні, ланцюжки не можуть бути просто перетворені в єдину ПММ шляхом формування декартового добутку просторів спостереження

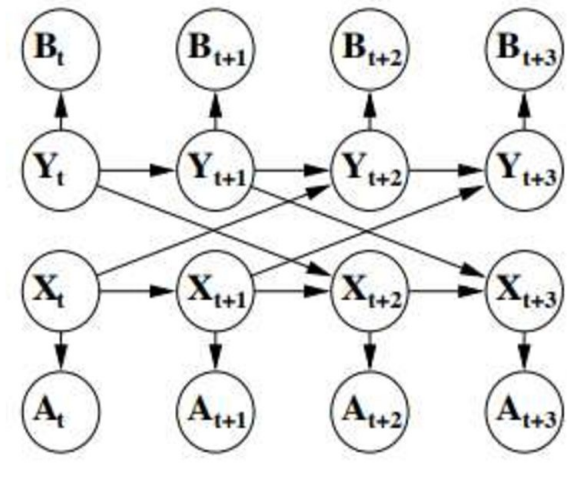


Рисунок 1.8 – Пов'язані ПММ 2

У даній роботі ми виводимо повні прямі-зворотні рівняння для оцінки максимальної правдоподібності. При це стає очевидним, що повний підхід, без "явного" злиття вузлів 2, можливий тільки для моделей "ЛАГ-один". Зокрема, згладжувач максимальної правдоподібності (тобто поширення переконань як вперед, так і назад у часі) не може бути отримано без кластеризації вузлів, в той час як фільтрація (поширення переконань тільки вперед у часі) може бути отримана і для більш високих лаг. Методи вибірки, однак, не покладаються на поширення переконань. Тому злиття вузлів не потрібно.

### 1.5.3 Оцінка параметрів

Було використано наступну нотацію:

$O = \{A_1^T, B_1^T\}$  це послідовність спостережень, де  $A_1^T = \{a_1, \dots, a_T\}$  чи є спостереження першим ланцюжком; аналогічно ми визначаємо  $B_1^T$  для другого ланцюга

$S = \{X_1^T, Y_1^T\}$  є послідовністю станів, де  $X_1^T = \{x_1, \dots, x_T\}$  і  $x \in 1 \dots M$  є станами першого ланцюга; аналогічно ми визначаємо  $Y_1^T$ .

$P(X_{t+1}|X_1^T, Y_1^T)$  є ймовірностями переходу станів першого ланцюга.

$P(Y_{t+1}|X_1^T, Y_1^T)$  є ймовірностями переходу станів другого ланцюга.

$P(X_1)$  і  $P(Y_1)$  є апріорними ймовірностями першого і другого ланцюга відповідно.

$P(A_t|X_t)$  і  $P(B_t|X_t)$  є щільності спостереження ланцюгів, які тут передбачаються багатовимірними Гаусові з середнім векторів  $\mu_x, \mu_y$  і коваріаційні матриці  $\Sigma_x, \Sigma_y$ .

$K_x$  і  $K_y$  такі розмірності простору стану.

Потім функція правдоподібності ППММ задається як

$$\mathcal{L}(\theta) = P(X_1)P(Y_1) \prod_{t=1}^T P(A_t|X_t)P(B_t|Y_t) \\ P(X_{t+1}|X_t, Y_t)P(Y_{t+1}|X_t, Y_t)$$

де вектор  $\theta$  містить параметри ймовірностей переходу, апіорні ймовірності та параметри щільності спостереження.

#### 1.5.4 Оцінка параметрів байесівської вибірки

Щоб отримати повні умови, необхідні для вибірки Гіббса, ми вибираємо пов'язані Пріори для кожного коефіцієнта:

- $\mu_x \sim N(\varepsilon_x, k_x)$ ,
- $\mu_y \sim N(\varepsilon_y, k_y)$ ,
- $\Sigma_x^{-1} \sim Whshart(R_x, v_x)$ ,
- $\Sigma_y^{-1} \sim Whshart(R_y, v_y)$ ,
- $P(X_{t+1}|X_1^T, Y_1^T)$  і  $P(X_1) \sim Dirichlet(a_{x_1}, \dots, a_{x_{K_x}})$ ,
- $P(Y_{t+1}|X_1^T, Y_1^T)$  і  $P(Y_1) \sim Dirichlet(a_{y_1}, \dots, a_{y_{K_y}})$ ,

де  $\mu_x \sim N(\varepsilon_x, k_x)$ , позначає те  $\mu_x$ , що взято з нормального розподілу з середнім  $\varepsilon_x$  і коваріаційної матрицею  $k_x$ , які є гіперпараметрами розподілу. Гіперпріори були такі, що всі пріори були розпливчастими або плоскими [21]. З огляду на ці пріори, повні умовні позначення є сполученими розподілами Діріхле, Гаусса або Уїшарта. Результати вибірки, представлені в даній роботі, використовують вибірку Гіббса, реалізовану на програмній мові BUGS [22].

### 1.5.6 Оцінка параметрів максимальної правдоподібності

В оцінці максимальної правдоподібності (МП) явно нестандартною частиною є поширення переконань при обчисленні ймовірностей прихованих станів. Рівняння максимізації для моделей спостереження ідентичні рівнянням для ПММ.

Для стандартної ПММ у нас є:

$$\begin{aligned}\alpha_{t+1} &\equiv P(O_1^{t+1}, S_{t+1}) = P(O_{t+1}|S_{t+1}) \int P(O_1^t, S_t) P(S_{t+1}|S_t) dS_t \\ &= P(O_{t+1}|S_{t+1}) \int P(S_{t+1}, S_t|O_1^t) dS_t \\ &= P(O_{t+1}|S_{t+1}) P(S_{t+1}|O_1^t),\end{aligned}\tag{1.1}$$

і умовною незалежністю

$$\alpha_{t+1} \equiv P(O_1^{t+1}, S_{t+1})\tag{1.2}$$

Для ПММ визначаємо змінну  $a$  для кожного ланцюжка:  $a_t^A$  і  $a_t^B$ . Біля  $t=1$ ,

$$\alpha_1^A = P(A_1|X_1)P(X_1),\tag{1.3}$$

$$\alpha_1^B = P(B_1|Y_1)P(Y_1).\tag{1.4}$$

При  $t=2$

$$\begin{aligned}\alpha_2^A &= P(A_2|X_2) \int \int P(A_1, X_1) P(B_1, Y_1) P(X_2|X_1, Y_1) dX_1 dY_1 \\ &= P(A_2|X_2) \int \int P(A_1, B_1, X_1, Y_1) P(X_2|X_1, Y_1) dX_1 dY_1 \\ &= P(A_2|X_2) P(A_1, B_1, X_2)\end{aligned}\tag{1.5}$$

оскільки  $A_2$  не залежить від  $A_1$  даного  $X_2$ , і тому

$$\alpha_2^A = P(A_1^2, B_1, X_2).\tag{1.6}$$

Повторення кроків для другого ланцюжка і для довільної  $t$  дає

$$\alpha_{t+1}^A = P(A_{t+1}|X_{t+1}) \int \int \alpha_t^A \alpha_t^B P(X_{t+1}|X_t, Y_t) dX_t dY_t\tag{1.7}$$

$$\alpha_{t+1}^B = P(B_{t+1}|Y_{t+1}) \int \int \alpha_t^A \alpha_t^B P(Y_{t+1}|X_t, Y_t) dX_t dY_t\tag{1.8}$$

Таким чином, існують два " незв'язаних " рівняння прогнозування, і кластеризація вузлів не потрібна.

Зворотні рекурсії для стандартних ПММ визначаються наступним чином

$$\beta_{t+1} \equiv P(O_{t+1}^T|S_t).\tag{1.9}$$

Ми можемо вирішити для них індуктивно наступним чином:

$$\begin{aligned}
\beta_T &= 1, \\
\beta_{T-1} &= \int P(O_T|S_T)P(S_T|S_{T-1})dS_T \\
&= P(O_T|S_{T-1}), \\
\beta_{T-2} &= \int P(O_T|S_{T-1})P(S_{T-1}|S_{T-2})P(O_{T-1}|S_{T-1})dS_T \\
&= \int \beta_{T-1}P(S_{T-1}|S_{T-2})P(O_{T-1}|S_{T-1})dS_T \\
&= \int P(O_{T-1}^T, S_{T-1}|S_{T-2}), \\
&= P(O_{T-1}^T|S_{T-2}).
\end{aligned} \tag{1.10}$$

Повторюючи кроки для будь-якого  $t$  ми отримуємо:

$$\beta_t = \int \beta_{t+1}P(S_{t+1}|S_t)P(O_{t+1}^T|S_{t+1})dS_{t+1} \tag{1.11}$$

Цикл в ПММ запобігає розділенню зворотних рекурсій на дві незв'язані рекурсії. Таким чином, необхідно виконати операцію кластеризації всіх вузлів в циклі, а саме  $X_t, Y_t, X_{t+1}, Y_{t+1}$ , що призводить до наступної рекурсії:

$$\begin{aligned}
\beta_{t+1} &\equiv P(O_{t+1}^T|S_t) \\
&= \int \int P(A_{t+1}|X_{t+1}, Y_{t+1})P(B_{t+1}|X_{t+1}, Y_{t+1}) \\
&\quad P(A_{t+2}^T, B_{t+2}^T|X_{t+1}, Y_{t+1})P(X_{t+1}, Y_{t+1}|X_t, Y_t)dX_{t+1}dY_{t+1} \\
&= \int \int P(A_{t+1}, B_{t+1}|X_{t+1}, Y_{t+1}) \beta_t P(X_{t+1}, Y_{t+1}|X_t, Y_t)dX_{t+1}dY_{t+1} \\
&= \int \int P(A_{t+1}^T, B_{t+1}^T|X_{t+1}, Y_{t+1})P(X_{t+1}, Y_{t+1}|X_t, Y_t)dX_{t+1}dY_{t+1}.
\end{aligned} \tag{1.12}$$

Це також демонструє, що швидко стає неможливим досліджувати різні топології зв'язку, оскільки вони будуть експоненціально збільшувати розмірність кластеризованого контура.

## 2 СЕГМЕНТАЦІЯ ТА ВІДСТЕЖЕННЯ

Відстеження об'єктів-важливе завдання в області комп'ютерного зору. Поширення потужних комп'ютерів, доступність високоякісних і недорогих відеокамер і зростаюча потреба в автоматизованому відеоаналізі викликали великий інтерес до алгоритмів стеження за об'єктами. У відеоаналізі є три ключові етапи:

- виявлення цікавих рухомих об'єктів;
- відстеження таких об'єктів від кадру до кадру;
- аналіз слідів об'єктів для розпізнавання їх поведінки.

Тому використання стеження за об'єктами актуально в задачах:

- розпізнавання на основі руху, тобто ідентифікація людини по ході, автоматичне виявлення об'єктів і т. д.;
- автоматизоване спостереження, тобто спостереження за місцем події з метою виявлення підозрілих дій або малоймовірних подій;
- індексація відео, тобто автоматичне анотування і пошук відео в мультимедійних база даних;
- взаємодія людини і комп'ютера, тобто розпізнавання жестів, відстеження погляду очей для введення даних в комп'ютери і т. д.;
- моніторинг трафіку, тобто збір статистики трафіку в режимі реального часу для управління транспортним потоком
- навігація автомобіля, тобто планування шляху на основі відео і можливості обходу перешкод.

У своїй найпростішій формі стеження можна визначити як завдання оцінки траєкторії об'єкта в площині зображення при його русі навколо сцени. Іншими словами, трекер присвоює послідовні мітки відстежуваним об'єктам в різних кадрах відео. Крім того, в залежності від області відстеження трекер може також надавати об'єктно-орієнтовану інформацію, таку як орієнтація, площа або форма об'єкта.

Відстеження об'єктів може бути складним із-за:

- втрата інформації, викликана проекцією 3D-світу на 2D-зображення;
- шум у зображеннях;
- рух складного об'єкта;
- нежорстка або артикульована природа об'єктів;
- часткові і повні оклюзії об'єктів;
- складні форми об'єктів;
- освітлення сцени змінюється;
- вимоги до обробки в реальному часі.

Можна спростити відстеження, наклавши обмеження на рух і/або зовнішній вигляд об'єкт. Наприклад, майже всі алгоритми стеження припускають, що рух об'єкта плавне, без різких змін. Можна далі обмежити рух об'єкта постійною швидкістю або постійним прискоренням, ґрунтуючись на апріорній інформації. Попередній знання про кількість і розмір об'єктів або про зовнішній вигляд і формі об'єктів також можуть бути використані для спрощення завдання.

Були запропоновані численні підходи до відстеження об'єктів. Вони в першу чергу відрізняються один від одного тим, як вони підходять до наступних питань: яке представлення об'єкта підходить для відстеження? Які функції зображення слід використовувати? Як слід моделювати рух, зовнішній вигляд і форму об'єкта? Відповіді на ці питання залежать від контексту середовища, в якому здійснюється відстеження, і кінцевого використання, для якого запитується інформація відстеження. Було запропоновано велику кількість методів відстеження, які намагаються відповісти на ці питання для безліч сценаріїв.

## 2.1 Об'єктне представлення

У сценарії відстеження об'єкт може бути визначений як все, що представляє інтерес для подальшого аналізу. Наприклад, човни в морі, риби в акваріумі, транспортні засоби на дорозі, літаки в повітрі, люди, що йдуть по дорозі, або бульбашки у воді-це набір об'єктів, які можуть бути важливими для відстеження в певній галузі. Об'єкти можуть бути представлені їх формами і зовнішнім виглядом.

Форми які використовуються у відстеженні:

Точка. Об'єкт представлений точкою, тобто центроїдом (рисунок 3.1(б)) або набором точок (рисунок 2.1(а)). У загальному випадку точкове представлення підходить для відстеження об'єктів, які займають невеликі області на зображення.

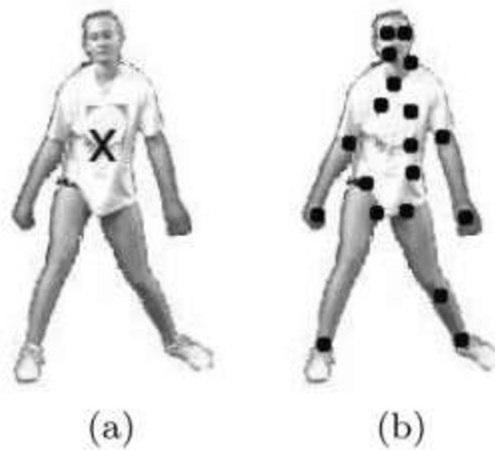


Рисунок 2.1 – Об'єктне представлення точки(а) та набір точок(б)

Примітивні. Геометричні фігури. Форма об'єкта представлена прямокутником, еліпсом (рисунок 2.2(а) і рисунок 2.2(б)) та ін. Рух об'єкта для таких уявлень зазвичай моделюється трансляційним, афінним або проєктивним (гомографічним) перетворенням. Хоча примітивні геометричні фігури більше підходять для представлення простих жорстких об'єктів, вони також використовуються для відстеження нежорстких об'єктів.

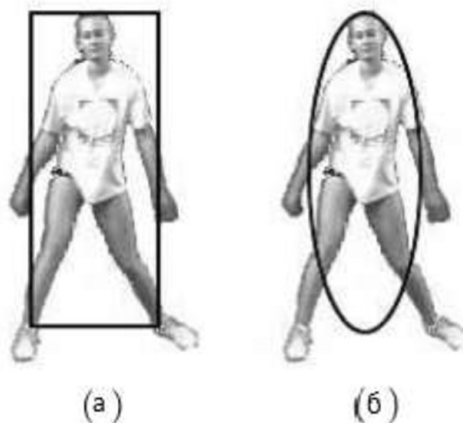


Рисунок 2.2 – Об'єктне представлення прямокутника(а) та еліпса(б)

Силует і контур об'єкта. Контурне представлення визначає кордон об'єкта

(рисунок 2.3(а) та 2.3(б)). Область всередині контуру називається силуетом об'єкта(Рисунок 3.4). Силуетні і Контурні уявлення підходять для Відстеження складних нежорстких форм.

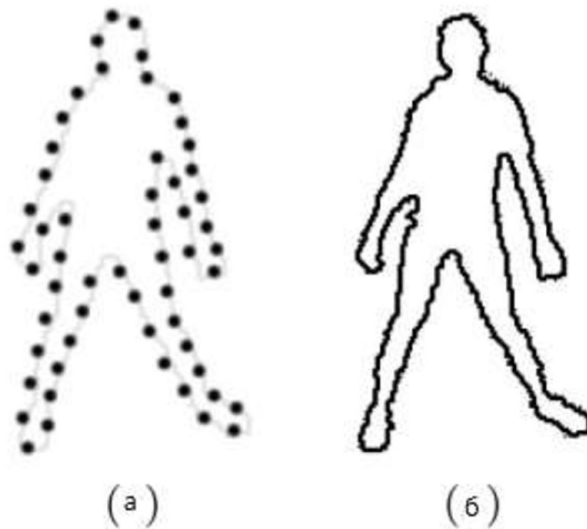


Рисунок 2.3 – Об'єктне представлення повний контур об'єкта (а) та контрольні точки на контурі об'єкта (б)



Рисунок 2.4 – Об'єктне представлення силуета об'єкта

Зчленовані моделі форм. Зчленовані об'єкти складаються з частин тіла, які утримуються разом за допомогою суглобів. Наприклад, людське тіло-це зчленований об'єкт з тулубом, ногами, руками, головою і ступнями, З'єднаними суглобами. Відносини між деталями регулюються кінематичними моделями руху, наприклад, кутом з'єднання і т. д. Щоб уявити зчленований об'єкт, можна змодельовати складові частини за допомогою циліндрів або еліпсів, як показано

на рисунку 2.5.

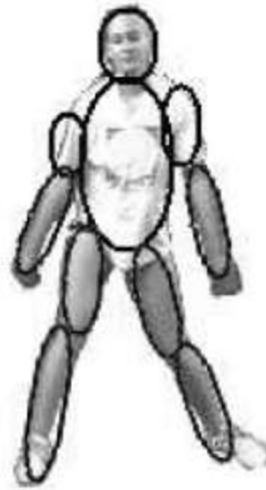


Рисунок 2.5- Об'єктне представлення зчленованого об'єкта

Скелетні моделі. Скелет об'єкта може бути витягнутий шляхом застосування перетворення медіальної осі до силуету об'єкта [23]. Ця модель зазвичай використовується як представлення форми для розпізнавання об'єктів. Скелетне уявлення може бути використано для моделювання як зчленованих, так і жорстких об'єктів (рисунок 2.6).

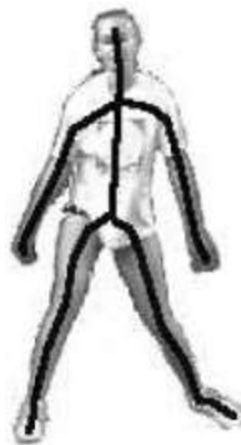


Рисунок 2.6- Об'єктне представлення скелета об'єкта

Існує кілька способів представлення ознак зовнішнього вигляду об'єктів. Звернути увагу, що подання форми також можуть бути об'єднані з уявленнями зовнішнього вигляду для відстеження. Деякі загальні уявлення зовнішнього вигляду в контексті відстеження об'єктів:

- щільності ймовірності появи об'єкта;

- шаблон;
- активні моделі зовнішнього вигляду;
- багатовидові моделі зовнішнього вигляду.

Щільності ймовірності появи об'єкта. Оцінки щільності ймовірності появи об'єкта можуть бути або параметричними, наприклад Гауссовськими і суміш гауссіан [24], або непараметричних, таких як Вікна парцена та гістограми. Щільність ймовірності ознак зовнішнього вигляду об'єкта (колір, текстура) вичислюється за областями зображення, заданими моделями форми (внутрішня область еліпса або контур).

Шаблон. Шаблони формуються з використанням простих геометричних фігур або силуетів[25]. Перевага шаблону в тому, що він несе в собі як просторову, так і зовнішню інформацію. Шаблони, однак, кодують тільки зовнішній вигляд об'єкта, створений з одного подання. Таким чином, вони підходять тільки для стеження за об'єктами, пози яких в процесі стеження істотно не змінюються.

Активні моделі зовнішнього вигляду. Активні моделі зовнішнього вигляду генеруються шляхом одночасного моделювання форми і зовнішнього вигляду об'єкта. У загальному випадку форма об'єкта визначається набором орієнтирів. Подібно контурному поданням, орієнтири можуть розташовуватися на кордоні об'єкта або, як варіант, всередині області об'єкта. Для кожного орієнтира зберігається вектор зовнішнього виду у вигляді кольору, текстури або величини градієнта. Активний зовнішній вигляд моделі вимагає навчання, коли і форма, і пов'язаний з нею зовнішній вигляд вивчаються з набору зразків з використанням. Наприклад, аналізу головних компонентів.

Багатовидові моделі зовнішнього вигляду. Ці моделі кодують різні уявлення об'єкта. Один з підходів до представлення різних уявлень об'єктів полягає у створенні підпростору з заданих уявлень. Підпросторові підходи, аналіз головних компонентів і аналіз незалежних компонентів, використовувалися як для представлення форми, так і для представлення зовнішнього вигляду

Інший підхід до вивчення різних уявлень об'єкта полягає в навчанні набору

класифікаторів, наприклад, машин опорних векторів або байєсівських мереж. Одним з обмежень багатовидових моделей зовнішнього вигляду є те, що зовнішній вид у всіх видах потрібно заздалегідь.

Загалом, існує міцний зв'язок між уявленнями об'єктів і алгоритмами відстеження. Представлення об'єктів зазвичай вибираються відповідно до області програми. Для відстеження об'єктів, які на зображенні здаються дуже маленькими, зазвичай підходить точкове подання. Наприклад, Винмен [26]. використовують точкове подання для відстеження насіння в рухомій послідовності тарілок. Точно так же Шафік і Шах [27] використовують точкове представлення для відстеження далеких птахів. Для об'єктів, форми яких можуть бути апроксимовані прямокутниками або еліпсами, примітивна геометрична форма уявлення більш доречні. Інші використовували представлення еліптичної форми і використовуйте колірну гістограму, обчислену з еліптичної області, для моделювання зовнішнього вигляду. У 1998 році Блек і Джемсон використовували власні вектори для представлення зовнішнього виду. Власні вектори були згенеровані з прямокутних шаблонів об'єктів. Задля відстеження об'єктів зі складними формами, наприклад, людей, підходить контурне або силуетне уявлення. Харітаоглу та ін. [28] використовували силуети для відстеження об'єктів в додатку спостереження.

## 2.2 Вибір функцій для відстеження

Вибір правильних функцій відіграє вирішальну роль у відстеженні. Загалом, найбільш бажаним властивістю візуального об'єкта є його унікальність, що дозволяє легко розрізнити об'єкти в просторі об'єктів. Вибір об'єкта тісно пов'язаний з поданням об'єкта. Наприклад, колір використовується як об'єкт для представлення зовнішнього вигляду на основі гістограми, в той час як для представлення на основі контуру в якості об'єкта зазвичай використовуються краї об'єктів. Загалом, багато алгоритмів відстеження використовують комбінацію цих функцій.

Деталі загальних візуальних особливостей полягають в наступному:

- колір;
- стик;
- оптичний потік;
- текстура.

Колір. На видимий колір об'єкта впливають в першу чергу два фізичних фактори:

- спектральний розподіл потужності освітлювача;
- поверхневі відбивні властивості об'єкта. При обробці зображень колірний простір RGB (червоний, зелений, синій) зазвичай використовується для представлення кольору.

Однак простір RGB не є перцептивно однорідним колірним простором, тобто відмінності між кольорами в просторі RGB не відповідають колірним відмінностям, сприйманим людиною [29]. Крім того, розміри RGB сильно корелюють. Навпаки,  $L^*u^*v^*$  і  $L^*a^*b^*$  є перцептивно однорідні колірні простори, в той час як HSV (відтінок, насиченість, значення) є приблизно однорідним колірним простором. Однак ці кольорові простори чутливі до шуму. Таким чином, немає останнього слова про те, який колірний простір є більш ефективним, тому в трекінгу використовувалися різні колірні простору.

Краї. Межі об'єктів зазвичай викликають сильні зміни інтенсивності зображення. Виявлення країв використовується для ідентифікації цих змін. Важливою властивістю кромek є те, що вони менш чутливі до змін освітленості в порівнянні з колірними об'єктами. Алгоритм, що відстежують межі об'єктів, зазвичай використовують ребра в якості репрезентативного об'єкта. Через свою простоту і точність найбільш популярним підходом до виявлення країв є детектор Canny Edge detector [30].

Оптичний потік. Оптичний потік-це щільне поле векторів зміщення, яке визначає переміщення кожного пікселя в області. Він обчислюється за допомогою обмеження яскравості, яке передбачає сталість яскравості відповідних пікселів в послідовних кадрах. Оптичний потік зазвичай використовується як функція в додатках сегментації і відстеження руху на основі руху. Популярні методи

обчислення щільного оптичного потоку включають методи Хорна і Шунка , Лукаса і Канаде , Блека і Анандана , а також Шеліскі і Куглана .

Текстура. Текстура-це міра зміни інтенсивності поверхні, яка кількісно визначає такі властивості, як гладкість і регулярність. У порівнянні з кольором, текстура вимагає крок обробки для створення дескрипторів. Існують різні дескриптори текстур: Матриці збігів сірого рівня (МЗСР) (двовимірна гістограма, яка показує збіги інтенсивностей в заданому напрямку і відстані), Текsturні заходи Лоу [31] (двадцять п'ять 2D-фільтрів, що генеруються з п'яти 1D- фільтрів, відповідних рівню, краю, плямі, хвилі і пульсації), вейвлети (ортогональний Банк фільтрів) і керовані піраміди. Подібно крайовим елементам, текстурні елементи менш чутливі до змін освітлення у порівнянні з кольором.

В основному функції вибираються користувачем вручну в залежності від домену програми. Однак проблема автоматичного вибору ознак отримала значну увагу в завданнях розпізнавання образів. Методи автоматичного вибору об'єктів можна розділити на методи фільтрації і методи обгортки. Методи фільтрації намагаються вибрати об'єкти на основі загального критерію, наприклад, об'єкти повинні бути некорельованими. Методи обгортки вибирають функції, засновані на корисності функцій в конкретній проблемній області, наприклад, продуктивність класифікації з використанням підмножини ознак. Аналіз основних компонент (РСА) є прикладом методів фільтрації для зменшення функції. РСА включає в себе перетворення ряду (можливо) корельованих змінних в (менше) число некорельованих змінних, званих головними компонентами. Обгортка методом вибору дискримінаційних ознак для відстеження певного класу об'єктів впроваджується алгоритмом Adaboost [31]. Adaboost-це метод пошуку сильного класифікатора, заснований на комбінації помірно неточних слабких класифікаторів. При наявності великого набору ознак для кожної ознаки можна навчити один Класифікатор. Adaboost, виявить зважену комбінацію класифікаторів (що представляють ознаки), які максимізують класифікаційну продуктивність алгоритму: чим вище вага ознаки, тим більше він дискримінаційний. Для відстеження можна використовувати перші  $n$  найбільш

зважених функцій.

Серед усіх функцій колір є однією з найбільш широко використовуваних функцій для відстеження. Команісі та ін. [32] використовували кольорову гістограму для представлення зовнішнього вигляду об'єкта. Незважаючи на свою популярність, більшість колірних смуг чутливі до зміни освітленості. Отож, у сценаріях, де цей ефект неминучий, для моделювання зовнішнього вигляду об'єкта використовуються інші функції..

## 2.3 Виявлення об'єктів

Кожен метод відстеження вимагає механізму виявлення об'єкта або в кожному кадрі, або при першій появі об'єкта на відео. Загальний підхід до виявлення об'єктів полягає в використанні інформації в одному кадрі. Однак деякі методи виявлення об'єктів використовують тимчасову інформацію, обчислену з послідовності кадрів, щоб зменшити кількість помилкових виявлень. Ця тимчасова інформація зазвичай представлена у вигляді різниці кадрів, яка виділяє зміни в області в послідовних кадрах. Враховуючи області об'єкта то завдання трекера полягає в тому, щоб виконати відповідність об'єкта від одного кадру до іншого, щоб згенерувати треки.

### 2.3.1 Точкові Детектори

Точкові Детектори використовуються для пошуку точок інтересу на зображеннях, що мають виразну текстуру у відповідних місцях. Точки інтересу вже давно використовуються в контексті проблем руху, стерео і відстеження.

Щоб знайти точки, що цікавлять, оператор Моравека [33] обчислює зміну інтенсивності зображення в патчі 4x4 в горизонтальному, вертикальному, діагональному і антидіагональному напрямках і вибирає мінімум з чотирьох варіацій в якості репрезентативних значень для вікна. Точка оголошується цікавою, якщо зміна інтенсивності є локальним максимумом в 12x12 патч.



Рисунок 2.7 – Процентні точки, виявлені при застосуванні методу Харріса

Детектор Харріса обчислює похідні зображення першого порядку  $I_x, I_y$  в напрямках  $x$  і  $y$ , щоб виділити спрямовані варіації інтенсивності, потім для кожного пікселя в невеликій околиці обчислюється друга матриця моментів, яка кодує ці варіації:

$$M = \begin{pmatrix} \sum I_x^2 & \sum I_x, I_y \\ \sum I_x, I_y & \sum I_y^2 \end{pmatrix} \quad (2.1)$$

Точка інтересу ідентифікується за допомогою детермінанта і сліду  $M$ , який вимірює варіацію в локальній околиці  $R = \det(M) - k * \text{tr}(M)^2$ , де  $k$ -постійна. Точки інтересу відзначені пороговим значенням  $R$  після застосування придушення не максимумів (рисунок 2.7). Та ж сама матриця моментів  $M$ , наведена в рівнянні (13), використовується на етапі виявлення цікавить точки методу стеження KLT. Довірча точка інтересу  $R$  обчислюється з використанням мінімального власного значення  $M$ ,  $\lambda_{\min}$ . Відбираються кандидати на процентні точки шляхом порогового значення  $R$ . серед точок-кандидатів KLT усуває кандидатів, які просторово близькі один до одного (рисунок 2.8).



Рисунок 2.8 – Процентні точки, виявлені при застосуванні методу KLT

Кількісно і Харріс, і KLT підкреслюють варіації інтенсивності, використовуючи дуже схожі заходи. Наприклад,  $R$  в Харрісі пов'язаний з характеристичним поліномом, використовуваним для знаходження власних значень  $M$ :  $\lambda^2 + \det(M) - \lambda * \text{tr}(M) = 0$ , в той час як KLT обчислює власні значення безпосередньо. На практиці обидва ці методи знаходять майже однакові точки інтересу. Єдиною відмінністю є додатковий критерій KLT, який забезпечує просторову відстань між виявленими точками інтересу.

У теорії, матриця  $M$  є інваріантною до ротації і перекладу. Однак ця матриця не інваріантний до афінних або проєктивних перетворень. Щоб ввести надійне виявлення точок інтересу при різних перетвореннях, Лоу [31] ввів метод SIFT (Scale Invariant Feature Transform), який складається з чотирьох кроків. По-перше, масштабний простір будується шляхом згортки зображення з Гауссовими фільтрами в різних масштабах. Згорнуті зображення використовуються для створення зображень різниці гаусів (Difference-Of-Gaussians (DoG)). Потім з мінімумів і максимумів вибираються точки інтересу-кандидати. Наступний крок оновлює місце розташування кожного кандидата шляхом інтерполяції значень кольору з використанням сусідніх пікселів. На третьому етапі усуваються кандидати з низьким контрастом, а також кандидати по краях. Нарешті, залишившимся точкам, призначаються орієнтації на основі піків в гістограмах напрямків градієнта в невеликій околиці навколо точки-кандидата. Детектор просіювання генерує більшу кількість точок інтересу в

порівнянні з іншими детекторами точок інтересу. Це пов'язано з тим, що накопичуються точки інтересу в різних масштабах і різних дозволах (піраміда). Емпірично було показано в роботі Миколайчика і Шміда [34], що SIFT перевершує більшість точкових детекторів і більш стійкий до деформацій зображення (Рисунок 2.9).



Рисунок 2.9 – Процентні точки, виявлені при застосуванні методу SIFT

### 2.3.2 Віднімання фону

Виявлення об'єкта може бути досягнуто шляхом побудови уявлення сцени, званого фоновою моделлю, а потім знаходження відхилень від моделі для кожного вхідного кадру. Будь-яка значна зміна області зображення від фонової моделі означає Рухомий об'єкт. Пікселі, складові змінюються області, позначаються для подальшої обробка. Зазвичай алгоритм зв'язних компонентів застосовується для отримання зв'язкових областей, відповідних об'єктам. Цей процес називається відніманням фону.

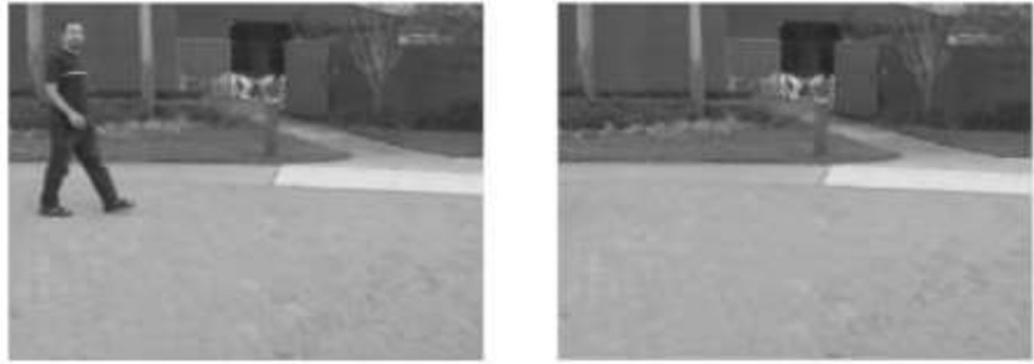


Рисунок 2.10 – Суміш гауссовського моделювання для віднімання фону

Віднімання фону стало популярним після роботи Рена та ін. [35]. Щоб вивчити поступові зміни в часі, Рен і ін. пропонують моделювати колір кожного пікселя  $I(x, y)$  стаціонарного фону з одним 3D ( $Y, U$  і  $V$  колірним простором) гауссів,  $I(x, y) \sim N(\mu(x, y), \Sigma(x, y))$ . Параметр моделі, середнє значення  $\mu(x, y)$  і коваріація  $\Sigma(x, y)$ , витягуються з кольорові спостереження в декількох послідовних кадрах. Як тільки фонові модель отримана, для кожного пікселя  $(x, y)$  у вхідному кадрі ймовірність того, що його колір буде отриманий з  $N(\mu(x, y), \Sigma(x, y))$ , обчислюється, і пікселі, які відхиляються від фонові моделі, позначаються як пікселі переднього плану. Однак один Гаусівська не є гарною моделлю для зовнішніх сцен, оскільки в певному місці можна спостерігати кілька кольорів через повторюваного руху об'єкта, тіней або відображення. Істотне поліпшення у фоновому режимі моделювання досягається за рахунок використання мультимодальних статистичних моделей для опису кожного пікселя кольору фону. У цьому методі піксель у поточному кадрі перевіряється за фонові моделлю, порівнюючи його з кожним гаусовим в моделі, поки не буде знайдений відповідний гауссів піксель. Якщо збіг знайдено, то середнє значення і дисперсія збігається, в іншому випадку створюється новий Гаус з середнім значенням. У суміш вводиться рівний поточному кольору пікселя і деяка початкова дисперсія. Кожен піксель класифікується залежно від того, чи представляє узгоджений розподіл фоновий процес. Рухомі області, які виявляються за допомогою цього підходу, поряд з фоновими моделями показані на рисунку 2.10 та рисунку 2.11.



Рисунок 2.11 – Суміш Гауссовського моделювання для віднімання фону

Інший підхід полягає в тому, щоб включити регіональну (просторову) інформацію про сцену замість використання тільки колірної інформації. Елгаман і Девіс [36] використовують непараметричну оцінку щільності ядра для моделювання фону на піксель. У процесі віднімання поточний піксель зіставляється не тільки з відповідним пікселем фонові моделі, але і з сусідніми розташуваннями пікселів. Таким чином, цей метод може обробляти тремтіння камери або невеликі рухи у фоновому режимі. Лі і Люн [37] об'єднали текстурні і колірні функції, щоб виконати віднімання фону по блоках розміром 5x5 пікселів. Оскільки текстура не сильно змінюється при зміні освітлення, метод менш чутливий до освітлення. Тояма та ін. [38] пропонують трирівневий алгоритм для вирішення завдання віднімання фону. На додаток до віднімання на рівні пікселів автори використовують інформацію про регіон і рівні кадру. На піксельному рівні автори пропонують використовувати фільтрацію Вінера для отримання ймовірнісних прогнозів очікуваний колір фону. На рівні регіону заповнюються області переднього плану, що складаються з однорідного кольору. На рівні кадру, якщо більшість пікселів в кадрі раптово змінюються, передбачається, що засновані на пікселях моделі колірного фону більше не дійсні. У цей момент або замінюється раніше збережена піксельна фонові модель, або модель повторно ініціалізується.

Альтернативний підхід до віднімання фону полягає в поданні варіацій інтенсивності пікселя в послідовності зображень у вигляді дискретних станів,

відповідних подіям в навколишньому середовищі. Наприклад, для відстеження автомобілів на шосе пікселі зображення можуть перебувати у фоновому стані, стані переднього плану (автомобіля) або в тіньовому стані. використовують приховані марковські моделі для класифікації невеликих блоків зображення як належать одному з цих трьох станів. У контексті виявлення подій включення і виключення світла у кімнаті Стенгер [39] використовував ПММ для віднімання фону. Перевага використання ПММ полягає в тому, що деякі події, які важко правильно змодельовати за допомогою неконтрольованих підходів фонового моделювання, можуть бути вивчені за допомогою навчальної вибірки.

Замість того щоб моделювати зміну окремих пікселів, Олівер та ін [41] пропонують цілісний підхід з використанням декомпозиції власного простору. Для введення кадрів до  $k$ ,  $I^i: i = 1 \dots k$ , розміром  $n * m$ , фонові матриця  $V$  розміром  $k * l$  формується каскадом  $m$  рядків у кожному кадрі одна за одною, де  $l = (n * m)$ , а розкладання на власні значення застосовується до коваріації  $B, C = V^T V$ . Потім фон представлений найбільш описовими власними векторами  $\eta, u_i$ , де  $i < \eta < k$ , які охоплюють всі можливі ілюмінації в полі зору (FOV від англ. Field of View). Таким чином, цей підхід менш чутливий до ілюмінації. Об'єкти переднього плану виявляються шляхом проектування поточного зображення у власне простір і знаходження різниці між реконструйованими і реальними зображеннями. Детальніше показано виявлені області об'єктів з використанням підходу власного простору на рисунку 3.12.



Рисунок 2.12 – Декомпозиція власного простору на основі віднімання фону

Одним з обмежень вищезазначених підходів є те, що вони вимагають

статичного фону. Ці методи здатні працювати з мінливим у часі фоном (наприклад, хвилі на воді, рухомі хмари і ескалатори). Ці методи моделюють області зображення як процеси авторегресії ковзного середнього (ARMA), які забезпечують спосіб вивчення і прогнозування патернів руху в сцені. Процес ARMA – це модель часових рядів. Він складається з сум авторегресійних і ковзних середніх компонент, де авторегресійний процес може бути описаний як зважена сума його попередніх значень і помилки потипу білий шум.

Таким чином, більшість сучасних методів стеження за фіксованими камерами. Це пов'язано з тим, що сучасні методи віднімання мають можливості моделювання змінюється освітленості, шуму і періодичного руху фонових областей і, отже, можуть точно виявляти об'єкти в різних обставинах. Більш того, ці методи є обчислювально ефективними. На практиці передісторія віднімання в багатьох випадках забезпечує неповні області об'єкта, тобто об'єкти можуть бути розбиті на кілька областей або всередині об'єкта можуть бути отвори, оскільки немає ніяких гарантій, що об'єкти будуть відрізнятися від фонових об'єктів. Найбільш важливим обмеженням віднімання фону є вимога стаціонарних камер. Рух камери зазвичай спотворює фонові моделі. Дані методи може бути застосований до відео, отриманого мобільними камерами, шляхом регенерації фонових моделей для невеликих тимчасових вікон, наприклад, трьох кадрів, з нуля або компенсуючи рух сенсора, наприклад, створюючи фонові мозаїки. Однак обидва ці рішення вимагають припущень про плоских сценах і малому русі в послідовних кадрах.

### 2.3.3 Сегментація

Метою алгоритмів сегментації зображень є поділ зображення на перцептивно подібні області. Кожен алгоритм сегментації вирішує дві проблеми: критерії хорошого розбиття і метод досягнення ефективного розбиття.



Рисунок 2.13 – Початкове зображення для сегментації

Для вирішення проблеми сегментації зображень Comaniciu і Meer пропонує підхід середнього зсуву для пошуку кластерів у спільному просторовому + колірному просторі  $[l, u, v, x, y]$ , де  $[l, u, v]$  представляє колір, а  $[x, y]$  представляє просторове місце розташування. При заданому зображенні алгоритм ініціалізується великою кількістю гіпотетичних кластерних центрів, випадково обраних з даних. Потім кожен центр кластера переміщується в середнє значення даних, що лежать всередині багатовимірного еліпсоїда з центром в центрі кластер. Вектор, який визначається старим і новим центрами кластерів, називається вектор середнього зсуву. Вектор середнього зсуву обчислюється ітеративно до тих пір, поки центри кластерів не змінять свого становища. Зверніть увагу, що під час ітерацій із середнім зсувом деякі кластери можуть бути об'єднані. На рисунку 3.13 показано початкове зображення, а на рисунку 3.14 показано сегментацію з використанням підходу середнього зсуву, згенерованого з використанням вихідного коду, доступного в `MeanShiftSegmentSrc`.

Зміна кластеризації є масштабованим для різних інших додатків, таких як виявлення краю, зображення регуляризації, і відстеження. Зміна на основі сегментації вимагає тонкої настройки різних параметрів для отримання краще сегментації, наприклад, вибір кольору і просторового ядра, пропускну здатності, а поріг мінімального розміру регіону значно ефект в результаті сегментація.

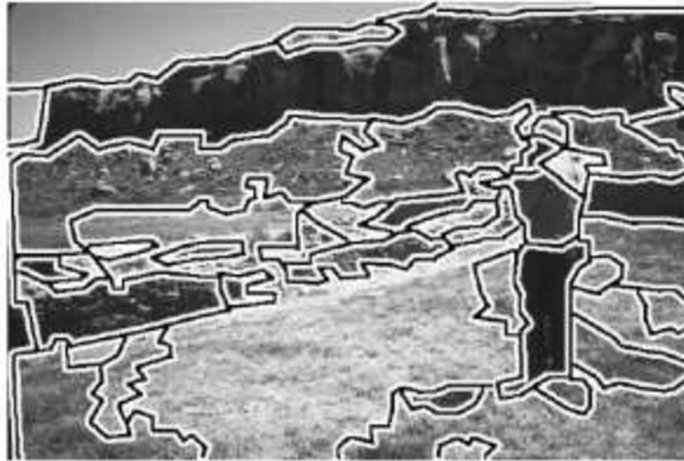


Рисунок 2.14 – Сегментація зображення з використанням сегментації із середнім зсувом

Сегментація зображення також може бути сформульована як задача розбиття графа, де вершини (пікселі)  $V = \{u, v, \dots\}$  графа (зображення)  $G$  розбиті на  $N$  непересічних підграфів (областей)  $A_i$ ,  $\bigcup_{i=1}^N A_i = V$ ,  $A_i \cap A_j = \emptyset, i \neq j$ , обрізаючи зважені ребра графа. Загальна вага обрізаних ребер між двома підграфами називається розрізом. Вага зазвичай обчислюється за кольором, яскравості або подібності текстури між вузлами. Ву і Ліхі використовують критерій мінімального розрізу, де мета полягає в тому, щоб знайти розділи, які мінімізують розріз. У їхньому підході ваги визначаються на основі подібності кольорів. Одним з обмежень мінімального розрізу є його зміщення в бік надмірної сегментації зображення. Цей ефект обумовлений збільшенням вартості розрізу при збільшенні числа ребер, що проходять через два секціонованих сегмента.

Ши і Малік [41] пропонують нормалізований розріз для подолання проблеми свержсегментації. В їх підході розріз залежить не тільки від суми ваг ребер в розрізі, але і від відношення сумарних ваг зв'язків вузлів в кожному розбитті ко всім вузлам графа. Для сегментації на основі зображень ваги між вузлами визначаються твором колірної схожості і просторової близькості. Як тільки ваги між кожною парою вузлів обчислені, вагова матриця  $W$  і діагональна матриця  $D$ , де  $D_{i,i} = \sum_{j=1}^N W_{i,j}$ , побудовані. Сегментація виконується спочатку шляхом обчислення власних векторів і власних значень узагальненої власної системи  $(D - W)u = \lambda Du$ , то другий за величиною власний вектор

використовується для поділу зображення на два сегмента. Для кожного нового сегмента цей процес рекурсивно виконується до тих пір, поки не буде досягнутий поріг. На рисунку 2.15 показано результати сегментації, отримані за допомогою підходу нормалізованих розрізів

У нормалізованій сегментації на основі розрізів рішення узагальненої власної системи для великих зображень може бути дорогим з точки зору обробки і вимог до пам'яті. Однак цей метод вимагає меншої кількості вручну вибраних параметрів у порівнянні з сегментацією *meanshift*. Нормалізовані розрізи також використовувалися в контексті відстеження контурів об'єктів .



Рисунок 2.15 – Сегментація зображення з використанням нормалізованих розрізів

**Активні Конттури.** В активній контурній структурі сегментація об'єкта досягається шляхом розвитку замкнутого контуру до межі об'єкта таким чином, що контур щільно охоплює область об'єкта. Еволюція контуру управляється енергетичним функціоналом, який визначає пристосованість контуру до гіпотетичної області об'єкта. Енергетичний функціонал для еволюції контуру має наступну загальну форму:

$$E(\Gamma) = \int_0^1 E_{int}(v) + E_{im}(v) + E_{ext}(v) ds \quad (2.2)$$

де  $s$ -довжина дуги контуру  $\Gamma$ ,  $E_{int}$  включає обмеження регуляризації,  $E_{im}$  включає енергію, засновану на зовнішньому вигляді, а  $E_{ext}$  задає додаткові обмеження.  $E_{int}$  зазвичай включає в себе член кривизни, член безперервності першого порядку ( $\nabla v$ ) або другого порядку ( $\nabla^2 v$ ), щоб знайти найкоротший контур. Енергія на основі зображень,  $E_{im}$ , може бути обчислена локально або глобально. Локальна інформація зазвичай знаходиться у вигляді градієнта зображення і оцінюється навколо контур. Навпаки, глобальні об'єкти обчислюються всередині і за межами області об'єкта. Глобальні функції включають в себе колір.

Різні дослідники використовували різні енергетичні терміни в рівнянні (2.2). Каселіс [42] пропонує виключити  $E_{ext}$  і використовуйте тільки градієнт зображення в якості енергії зображення  $E_{im} = g(|\nabla I|)$ , де  $G$ -сигмовидна функція. У порівнянні з градієнтом функція градієнта визначає контур об'єкта як геодезичну криву в рімановому просторі. Однак градієнти зображення дають дуже локальну інформацію і чутливі до локальних мінімумів. Щоб подолати цю проблему, дослідники ввели енергетичні терміни зображення на основі регіону. У 1996 році Чжу і Юйле [41] запропонували використовувати інформацію про регіон замість градієнта зображення. Однак використання регіональних термінів в енергетичному функціонал не призводить до гарної локалізації контуру об'єкта. Останнім часом стали популярні методи, що поєднують в собі як регіональну, так і градієнтну енергію зображення. Параджіос і Деріче [43] пропонують використовувати опуклу комбінацію градієнтних і регіональних енергій,  $E_{image} = \lambda E_{boundary} + (1 - \lambda) E_{region}$ . Зокрема, автори змодельюють поява в Ерегіоні суміші гаусів. Еволюція контуру спочатку виконується глобально, а потім локально шляхом зміни  $\alpha$  від 0 до 1 на кожній ітерації.

Важливою проблемою в контурних методах є ініціалізація контуру. У підходах, заснованих на градієнті зображення, контур зазвичай поміщається за

межі області об'єкта і стискається до тих пір, поки не зустрінеться межа об'єкта. Це обмеження послаблюється в регіональних методах таким чином, що контур може бути ініціалізовано або всередині, або зовні об'єкта, так що контур може або розширюватися, або стискатися відповідно, щоб відповідати межі об'єкта. Однак ці підходи вимагають попереднього об'єкта або фонове знання. Використовуючи кілька кадрів або опорний кадр, ініціалізація може бути виконана без побудови пріоритетів областей. Наприклад, в роботі Параджіос і Деріче автори використовують віднімання фону для ініціалізації контуру.

Крім вибору енергетичного функціоналу та ініціалізації, ще одним важливим питанням є вибір правильного контурного представлення. Контур об'єкта, може бути представлений або явно (контрольні точки,  $v$ ), або неявно (набори рівнів,  $\varphi$ ). В явному представленні відносини між контрольними точками визначаються сплайновими рівняннями. У поданні наборів рівнів контур представлений на просторовій сітці, яка кодує знакові відстані сіток від контуру з протилежними знаками для об'єкта і фонові області. Контур неявно визначається як нульові перетину в сітці набору рівнів. Еволюція контуру управляється зміною значень сітки відповідно до енергії, обчисленої за допомогою рівняння (14), оцінюваної в кожній позиції сітки. Зміни в значеннях сітки призводять до нових перетинів нулів, отже, до нових позиціям контуру. Вихідний код для універсальних наборів рівнів, які можуть бути використані для різних додатків шляхом вказівки швидкості еволюції контуру, наприклад, сегментація, відстеження, тепловий потік і т.д. доступні на LevelSetSrc. Найбільш важливою перевагою неявного подання перед явним є його гнучкість у вирішенні змін топології (поділ і злиття).

#### 2.3.4 Стеження за об'єктом

Мета стеження об'єктів полягає в тому, щоб генерувати траєкторію об'єкта з плином часу, визначаючи його положення в кожному кадрі відео. Стеження за об'єктом також може надати повну область на зображенні, яка зайнята об'єктом в кожен момент часу. Завдання виявлення об'єкта та встановлення відповідності

між екземплярами об'єкта під фреймах можуть виконуватися як окремо, так і спільно. У першому випадку можливі області об'єкта в кожному кадрі виходять за допомогою алгоритму виявлення об'єкта, а потім трекер співвідносить об'єкти між кадрами. В останньому випадку область об'єкта і відповідність спільно оцінюються шляхом ітеративного оновлення інформації про місцезнаходження об'єкта і області, отриманої з попередніх кадрів. При будь-якому підході відстеження об'єкти подаються з використанням моделей форми та/або зовнішнього вигляду, обмежує тип руху або деформації, яким він може зазнавати. Наприклад, якщо об'єкт представлений у вигляді точки, то можна використовувати тільки поступальний модель. У випадку, коли для об'єкта використовується геометричне представлення форми, таке як еліпс, підходять параметричні моделі руху, такі як афінні або проєктивні перетворення. Ці уявлення можуть апроксимувати рух жорстких об'єктів у сцена. Для нежорсткого об'єкта силует або контур є найбільш описовим поданням, і для визначення їх руху можуть використовуватися як параметричні, так і непараметричні моделі.

У світлі вищевикладеного ми наводимо таксономію методів відстеження на рисунку 2.16



Рисунок 2.16 – Таксономія методів відстеження.

Відстеження Точок. Об'єкти, виявлені в послідовних кадрах, представлені

точками, і асоціація точок заснована на попередньому стані об'єкта, який може включати в себе положення і рух об'єкта. Цей підхід вимагає зовнішнього механізму для виявлення об'єктів в кожному кадрі. Приклад відповідності об'єктів показаний на рисунку 3.15.

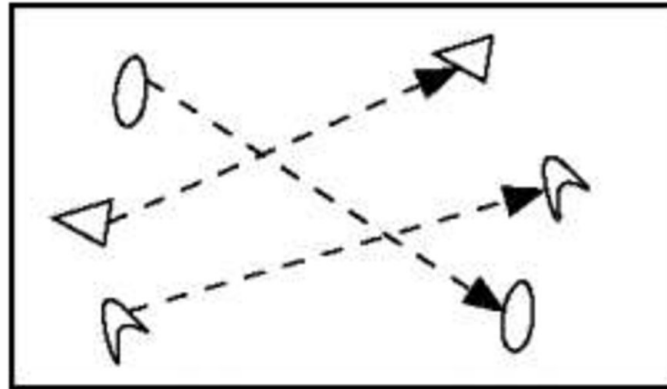


Рисунок 2.17 – Багатоточкова відстеження

Зернове відстеження. Зерно відноситься до форми і зовнішнього вигляду об'єкта. Наприклад, зерно може бути прямокутним шаблоном або еліптичною формою з відповідною гистограмою. Об'єкти відстежуються шляхом обчислення руху ядра в послідовних кадрах рисунок 3.16. Цей рух зазвичай має форму параметричного перетворення, такого як переміщення, обертання і афінне.

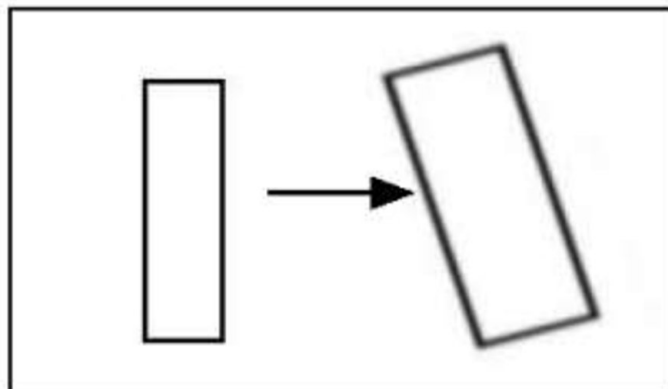


Рисунок 2.18 – Параметричне перетворення прямокутного патча

Відстеження Силуету. Відстеження виконується шляхом оцінки області об'єкта в кожному кадрі. Методи відстеження силуету використовують інформацію, закодовану всередині області об'єкта. Ця інформація може бути

представлена у вигляді моделей щільності зовнішнього вигляду і форми, які зазвичай представлені у вигляді крайових карт. Враховуючи об'єктні моделі, силуети відстежуються або відповідністю форми, або еволюцією контуру (рис. 3.17). Ці методи по суті можна розглядати як сегментацію об'єктів, що застосовується у часовій області з використанням апріорів, згенерованих з попередніх кадрів.

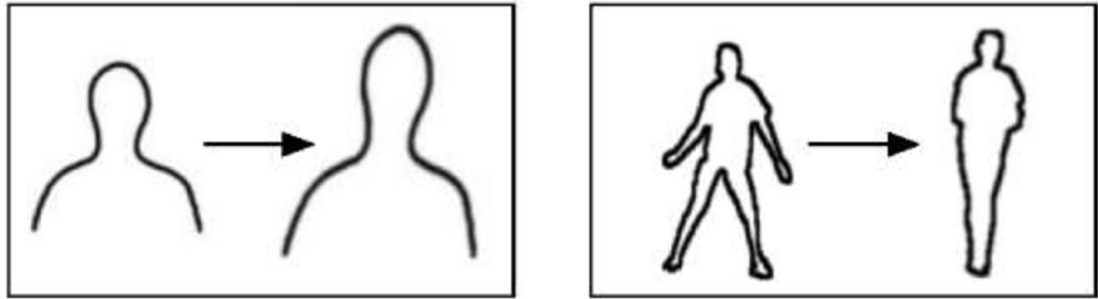


Рисунок 2.19 – Два приклади еволюції контурів.

### 2.3.5 Відстеження Точок

Відстеження може бути сформульовано як відповідність виявлених об'єктів, представлених точками на кадрах. Відповідність точок-складна проблема, особливо при наявності оклюзій, неправильних визначень, входів і виходів об'єктів. В цілому методи точкового відповідності можна розділити на дві широкі категорії: детерміністичні і статистичні. Детерміновані методи використовують якісну евристику руху, щоб обмежити проблему відповідності. З другої сторони, імовірнісні методи явно приймають вимірювання об'єкта і приймають невизначеності до уваги встановити листування.

Детерміновані методи відповідності точок визначають вартість зв'язування кожного об'єкта в кадрі  $t - 1$  з одним об'єктом в кадрі  $t$  за допомогою набору обмежень руху. Мінімізація вартості відповідності формулюється як завдання комбінаторної оптимізації. Рішення, що складається з взаємно однозначних відповідностей (рис. 3.18 серед усіх можливих асоціацій та рис. 3.19 може бути отримано методами оптимального присвоєння, наприклад угорським алгоритмом або методами жадібного пошуку.

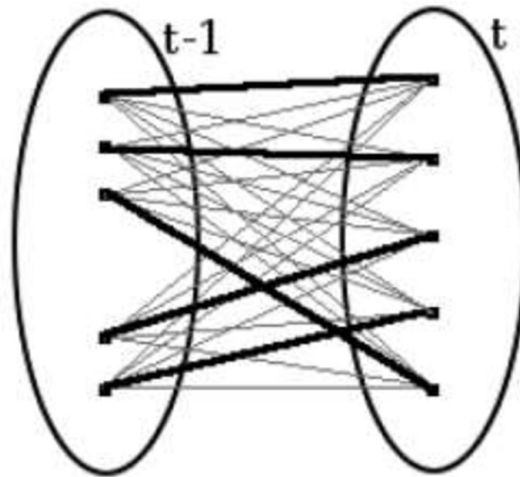


Рисунок 2.20 – Унікальний набір асоціацій

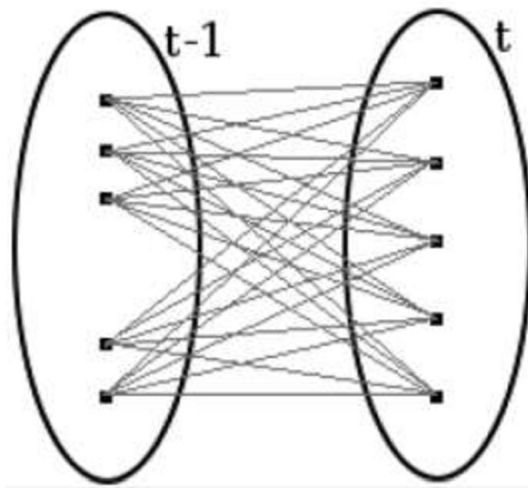


Рисунок 2.21 – Всі можливі асоціації точки (об'єкта) в кадрі  $t - 1$  з точками (об'єктами) в кадрі  $t$

Вартість кореспонденції зазвичай визначається за допомогою поєднання наступних обмежень:

- близькість передбачає, що місце розташування об'єкта не буде помітно змінюватися від одного кадру до іншого (рисунок 2.22);
- максимальна швидкість визначає верхню межу швидкості об'єкта і обмежує можливі відповідності кругової околиці навколо об'єкта(рисунок 2.23);
- мала зміна швидкості (плавний рух) передбачає, що напрямок і швидкість об'єкта не змінюються кардинально (рисунок 2.24);
- загальний рух обмежує швидкість об'єктів у невеликій околиці, щоб бути схожими (рисунок 2.25). Це обмеження підходить для об'єктів, представлених

декількома точками;

- жорсткість передбачає, що об'єкти в тривимірному світі є жорсткими, тому відстань між будь-якими двома точками на реальному об'єкті залишиться незмінним (рисунок 2.26);

- проксимальна однорідність-це поєднання близькості і малих обмежень на зміну швидкості.

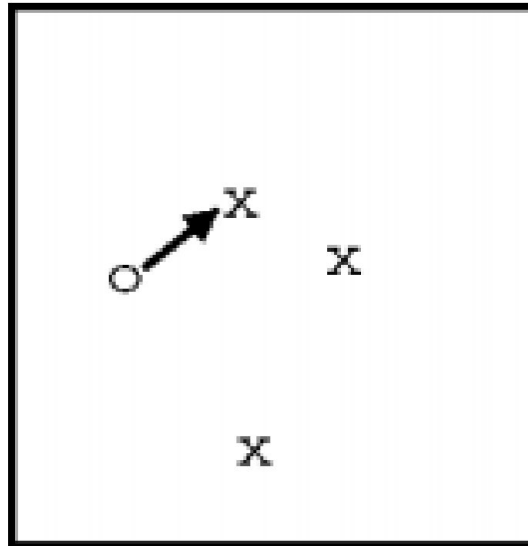


Рисунок 2.22 – Обмеження руху – близькість

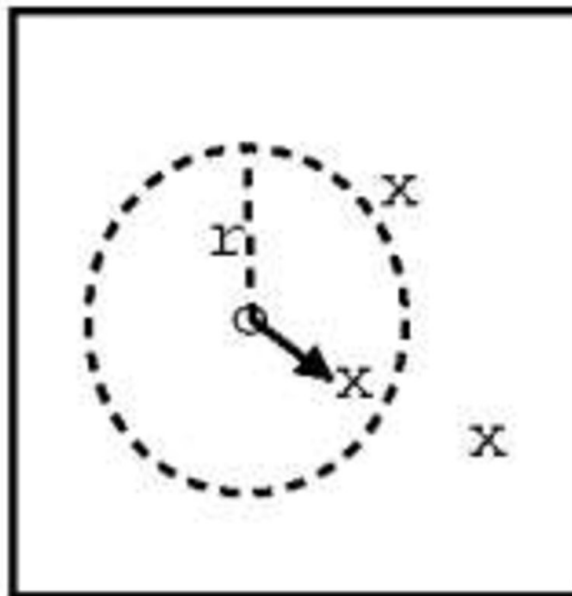


Рисунок 2.23 – Обмеження руху – максимальна швидкість

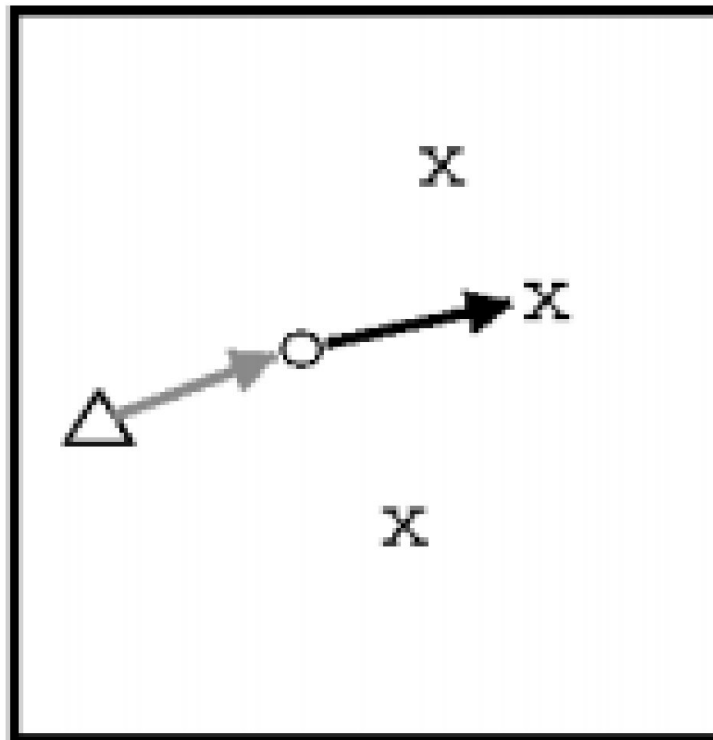


Рисунок 2.24 – Обмеження руху – мала зміна швидкості

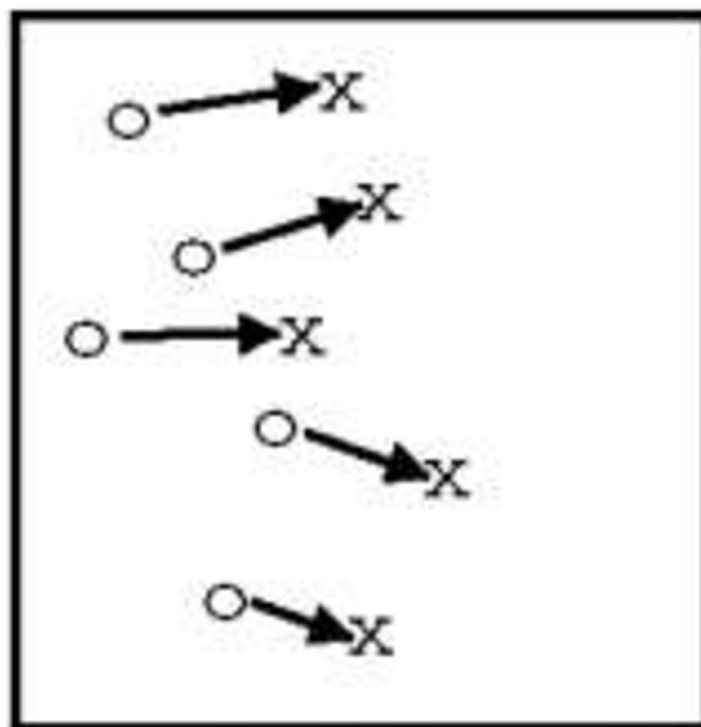


Рисунок 2.25 – Обмеження руху – загальний рух

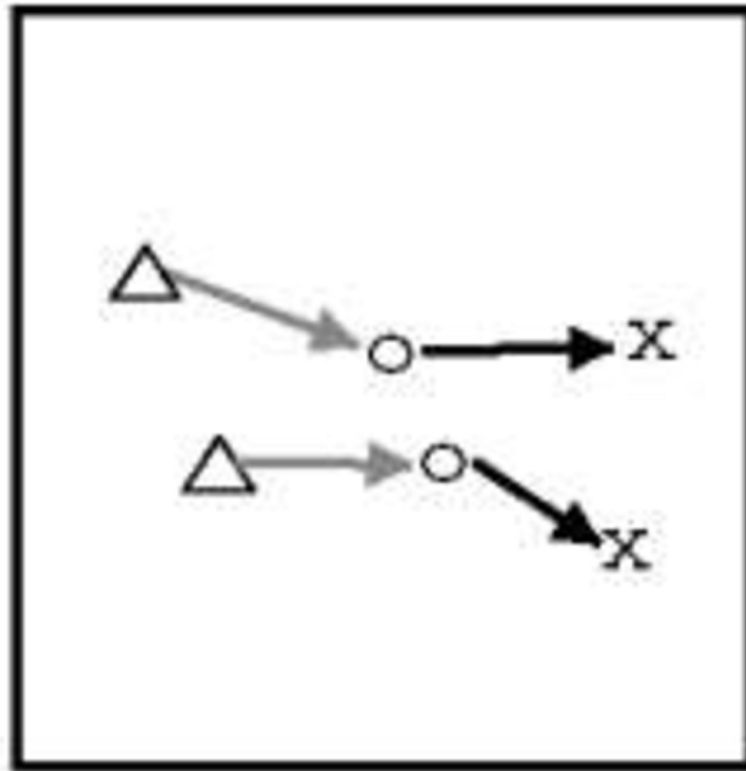


Рисунок 2.26 – Обмеження руху – обмеження жорсткості. Позначає положення об'єкта в кадрі  $t - 2$ ,  $\circ$  позначає положення об'єкта в кадрі  $t - 1$  і, нарешті,  $X$  позначає положення об'єкта в кадрі  $t$ .

Слід, однак, зазначити, що ці обмеження не є специфічними для детермінованих методів, і вони також можуть бути використані в контексті відстеження точок з використанням статистичних метод. Тут ми наводимо приклад різних методів, запропонованих в літературі в цій категорії. Сетхі і Джайн [44] вирішують відповідність жадібним підходом, заснованим на обмеженнях близькості і жорсткості. Їх алгоритм розглядає два послідовних кадру і ініціалізується критерієм найближчого сусіда. Кореспонденція обмінюються ітеративно, щоб мінімізувати витрати. Також аналізується модифікована версія того ж алгоритму, який обчислює відповідності в зворотному напрямку (від останнього кадру до першого кадру) на додаток до прямого напрямку. Цей метод не може обробляти оклюзії, входи або виходи. Саларі і Сетхі [45] вирішують ці проблеми, спочатку встановлюючи відповідність для виявлених точок, а потім розширюючи відстеження зниклих об'єктів шляхом

додавання ряду гіпотетичних точок. Початкові відповідності отримані шляхом обчислення оптичного потоку в перших двох кадрах. Метод не розглядає вхід і вихід об'єктів. Якщо кількість виявлених точок зменшується, передбачається оклюзія або неправильне виявлення. Оклюзія оброблена шляхом встановлення відповідності для виявлених об'єктів в поточному кадрі. Задля інших об'єктів положення передбачається на основі припущення про постійну швидкості. В роботі Интила [46], для зіставлення центроїдів об'єктів, об'єкти виявляються за допомогою віднімання фону. Автори явно обробляють зміну кількості об'єктів, досліджуючи певні області зображення, наприклад двері, щоб виявити входи / виходи перед обчисленням відповідності.

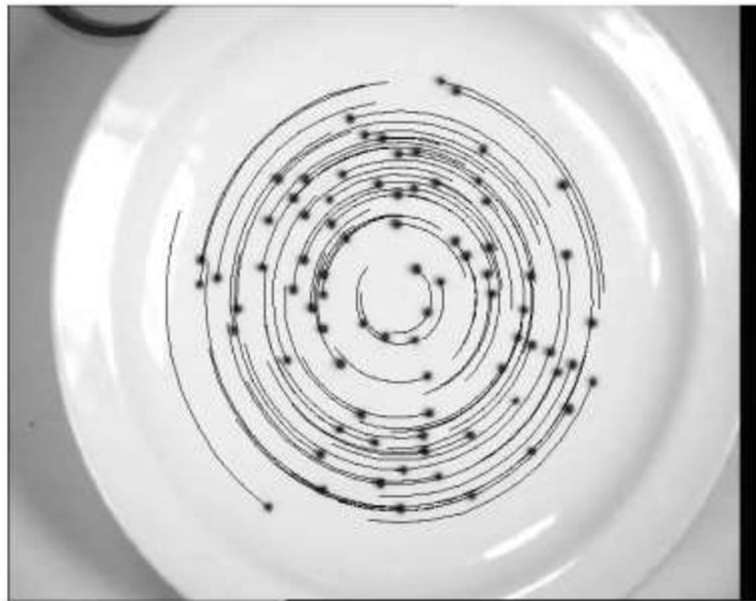


Рисунок 2.27 – Послідовності оберткових тарілок колірна сегментація використовувалася для виявлення чорних крапок на білій тарілці

Винман розширюють роботу Саларі і Сетхі, вводячи загальне обмеження руху для відповідності. Загальне обмеження руху забезпечує сильне обмеження для когерентного відстеження точок, що лежать на одному і тому ж об'єкті; однак воно не підходить для точок, що лежать на ізольованих об'єктах, рухомих в різних напрямках. Алгоритм ініціалізується шляхом генерації початкових треків з використанням двопрхідного алгоритму, а функція витрат мінімізується угорською мовою. алгоритм призначення в двох послідовних кадрах. Цей підхід

може обробляти помилки оклюзії і неправильного виявлення, однак передбачається, що кількість об'єктів однаково в усієї послідовності, тобто немає ніяких входів або виходів об'єктів. Рисунок 2.27 для відстеження результатів.

Шафік і Шах[27] пропонують багатокладовий підхід для збереження тимчасової когерентності швидкості і положення (рисунок 2.28). Вони формулюють відповідність як теоретико-графове завдання. Відповідність декількох кадрів пов'язано з знаходженням найкращого унікального шляху  $P_i = \{x^0, \dots, x^k\}$  для кожної точки (верхній індекс представляє номер кадру). Для неправильно виявлених або оклюзованих об'єктів шлях складатиметься з пропущених позицій у відповідних кадрах. Орієнтований граф, який генерується з використанням точок в  $k$  кадрах, перетворюється в дводольний граф шляхом розбиття кожного вузла (об'єкта) на два (+ і -) вузли і уявлення спрямованих ребер як неорієнтованих ребер від + до - вузлів.

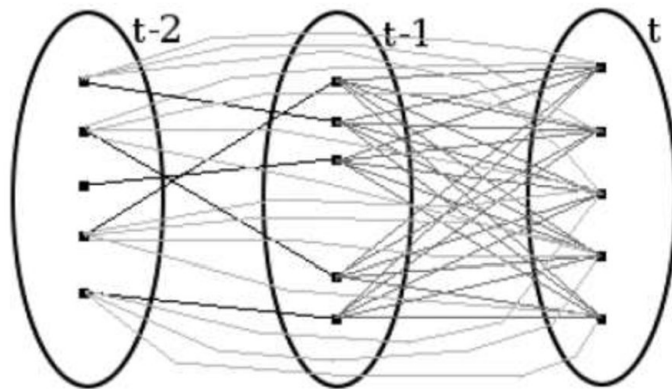


Рисунок 2.28 – Багатокладові відповідності.

Потім відповідність встановлюється жадібним алгоритмом. Вони використовують вікно кадрів під час точкової відповідності для обробки оклюзій, тривалість яких коротше тимчасового вікна, використововуваного для виконання зіставлення. Рисунок 2.29 для отримання результатів по цьому алгоритму відстеження птахів

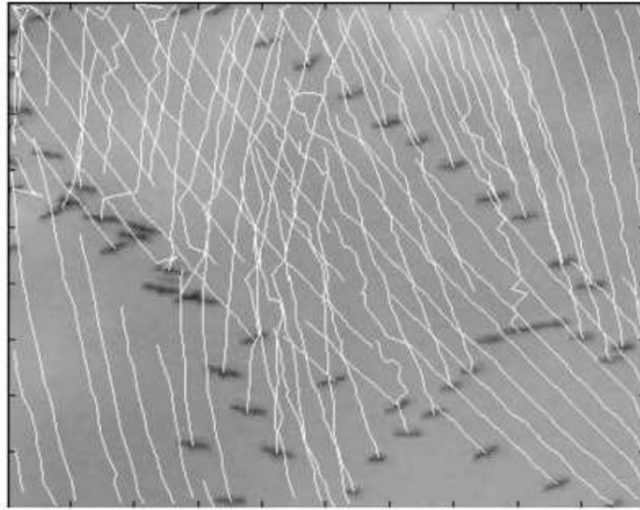


Рисунок 2.29 – Результат двоточкового алгоритма відповідності: відстеження птахів за допомогою алгоритму, запропонованого Shafique і Shah; Птахи виявляються за допомогою фонового віднімання.

## 3 МОДЕЛЬ ПОВЕДІНКИ

### 3.1 Огляд системи

У системі використовується статичне відео камери відеоспостереження з широким полем огляду, що спостерігає динамічну зовнішню сцену. Система комп'ютерного зору в реальному часі сегментує рухомі об'єкти з вивченої сцени. Метод опису сцени дозволяє вивчати і точно скидати з рахунків варіації освітлення, погоди і т. д.

Для кожного рухомого об'єкта створюється зовнішній опис, що дозволяє відстежувати його через тимчасові оклюзії і зустрічі з декількома об'єктами. Фільтр Калмана відстежує місце розташування об'єктів, грубу форму, колірну структуру і швидкість. Цей темпорально впорядкований потік даних потім використовується для отримання поведінкового опису кожного об'єкта і виявлення взаємодій між об'єктами.

На рисунку 3.1. показано контур обробки та основні функціональні блоки нашої кінцевої системи:

- модуль введення комп'ютерного зору в реальному часі виявляє і відстежує рухомі об'єкти в і для кожного рухомого об'єкта виводить вектор ознак, що описує його рух і напрямок, а також його просторове відношення до всіх довколишніх рухомих об'єктів;

- ці вектори ознак є вхідними даними для стохастичних моделей поведінки, заснованих на станах. І ПММ, і ППММ, з різними структурами в залежності від складності поведінки, потім використовуються для класифікації сприйманого поведінки.

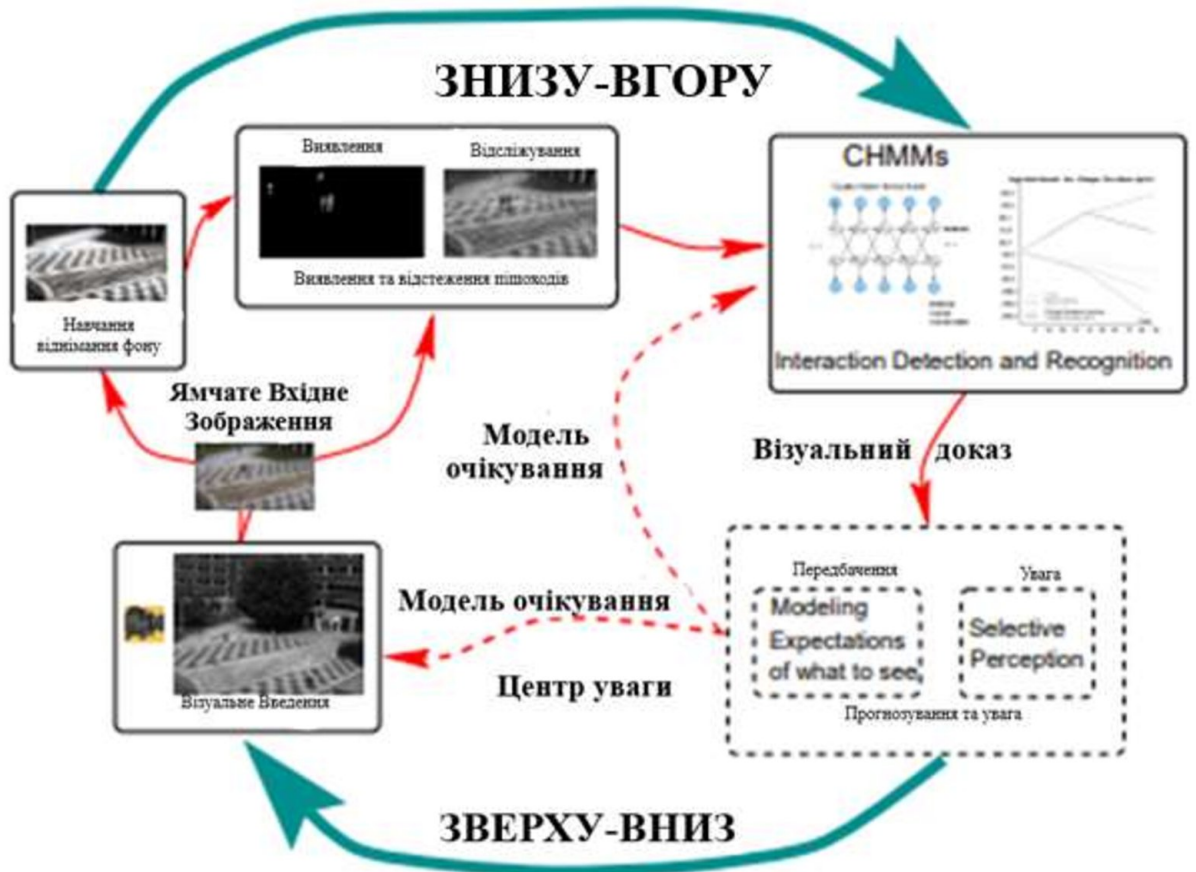


Рисунок 3.1. Цикл обробки зверху вниз і знизу вгору

Отже, потоки інформації зверху вниз, так і знизу вгору будуть постійно керовані та об'єднані для кожного рухомого об'єкта в межах сцени. Отже, наш байєсовський підхід пропонує математичну основу для поєднання спостережень (знизу вгору) зі складними поведінковими пріоритетами (зверху вниз), щоб забезпечити очікування, якими буде повернуто система сприйняття.

### 3.2 Моделі поведінки

У цьому розділі ми розробляємо нашу основу для побудови і застосування моделей індивідуального поведінки та міжособистісних взаємодій. Для того щоб побудувати ефективні комп'ютерні моделі людської поведінки, нам необхідно вирішити питання про те, як знання можуть бути зіставлені з обчисленнями для динамічного забезпечення послідовних інтерпретацій.

З обчислювальної точки зору існують дві ключові проблеми при обробці безперервного потоку даних ознак, що надходять з потоку вхідного відео:

- управління обчислювальним навантаженням, що накладається покадровим дослідженням всіх агентів і їх взаємодія. Наприклад, число можливих взаємодій між будь—якими двома агентами з множини  $N$  агентів дорівнює  $N * (N - 1)/2$ . При наївному управлінні це навантаження може легко стати великим навіть для помірною  $N$ ;

- навіть коли покадрове навантаження мала і подання оскільки миттєва поведінка кожного агента є компактною, все ще існує проблема управління всією цією інформацією з плином часу.

Статистичні спрямовані ациклічні графи (DAG) або імовірнісні мережі виводу (PINs) [8] може забезпечити обчислювально ефективно вирішення цих завдань. ПММ та їх розширення, такі як ППММ, можна розглядати як приватний, простий випадок тимчасового PIN або DAG коду. PIN складаються з набору випадкових величин, представлених у вигляді вузлів, а також спрямованих ребер або зв'язків між ними. Вони визначають математичну форму спільного або умовного PDF між випадковими величинами. Вони являють собою простий графічний спосіб подання причинно—наслідкові залежності між змінними. Відсутність спрямованих зв'язків між вузлами має на увазі умовну незалежність. Крім того, існує сімейство перетворень, що виконуються над графічною структурою, яка має прямий переклад в термінах математичних операцій, що застосовуються до базового PDF—файлу. Нарешті, вони є модульними, тобто можна висловити спільний глобальний PDF як продукт локального умовного PDF.

PIN являють собою кілька важливих переваг, які мають відношення до нашої проблеми:

- вони можуть обробляти неповні дані, а також невизначеність;
- вони навчені і легко уникнути надмірного використання; вони кодують причинність природним чином;
- існують алгоритми як для прогнозування, так і для ймовірнісного

виведення;

- вони пропонують основу для об'єднання попередніх знань і даних.;
- вони модульні і розпаралелюються.

Досліджувана поведінка генерується пішоходами, що йдуть у відкритій зовнішньому середовищі. Наша мета полягає в тому, щоб розробити загальний, композиційний аналіз спостережуваного поведінки в термінах станів і переходів між станами в часі таким чином, щоб

- стани відповідали нашим загальноприйнятим уявленням про людську поведінку і
- вони були безпосередньо застосовні до широкого кола об'єктів і ситуацій перегляду.

На рисунку 3.2 показано типове зображення для пішохідного сценарію.



Рисунок 3.2 — Типове зображення пішохідної площі

### 3.2.1 Візуальне розуміння за допомогою графічних моделей: ПММ і ППММ

Приховані марковські моделі (ПММ) — це популярна імовірнісна

структура для моделювання процесів, що мають структуру в часі. Вони мають чітку байєсівську семантику, ефективні алгоритми оцінки стану і параметрів і автоматично виконують динамічне спотворення часу. ПММ — це, по суті, квантування простору конфігурації системи на невелике число дискретних станів разом з ймовірностями переходів між станами. Одна кінцева дискретна змінна індексує поточний стан системи. Будь — яка інформація про історію процес, необхідний для майбутніх висновків, повинен бути відображений в поточному значенні цієї змінної стану. Графічно ПММ часто зображуються "розгорнутими в часі" у вигляді РІНов, наприклад на рисунку 3.3.

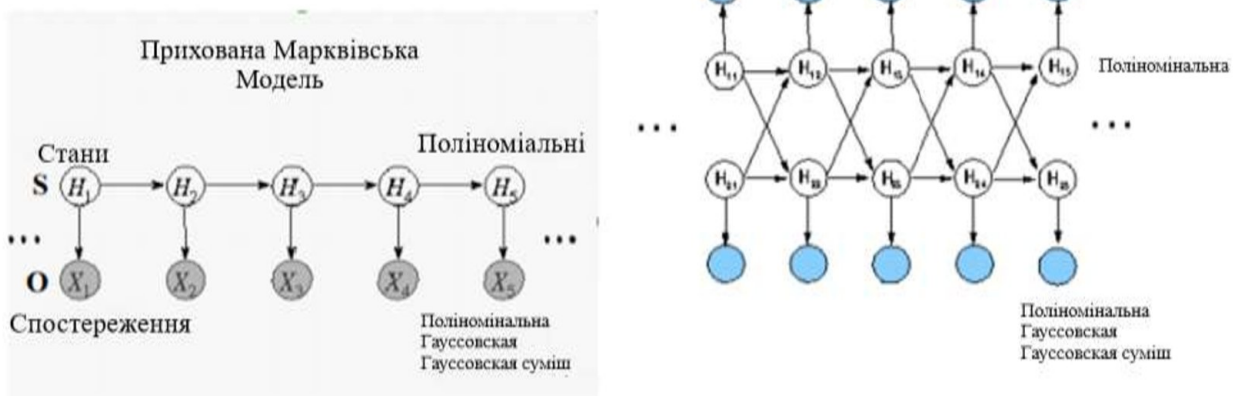


Рисунок 3.3 — Графічне представлення ПММ і ППММ, розгорнутих у часі

Однак багато цікавих систем складаються з безлічі взаємодіючих процесів і тому заслуговують композиційного представлення двох або більше змінних. Це типово для систем, що мають структуру як у часі, так і в просторі. З однією змінною стану марковські моделі погано підходять для цих завдань. Для моделювання цих взаємодій необхідна більш складна архітектура.

Розширення базової марковської моделі зазвичай збільшують обсяг пам'яті системи (durational modeling), забезпечуючи їй композиційний стан у часі. Нас цікавлять системи, які мають композиційний стан у просторі, наприклад, більше однієї одночасної змінної стану. Відомо, що точне рішення розширень базової ПММ до 3 і більше ланцюгів є найскладніше. У цих випадках необхідні методи апроксимації [47],[48].

Тому було використано дві пов'язані приховані марковські моделі

(ППММ) для моделювання двох взаємодіючих процесів, в даному випадку вони відповідають окремим людям. В цій архітектурі ланцюжка станів пов'язані через матриці умовних ймовірностей, що моделюють причинні (тимчасові) залежності між їх прихованими змінними станів. Графічне представлення ППММ показано на рисунку 3.3. На цьому рисунку видно, що для кожного ланцюжка стан в момент часу  $t$  залежить від стану в момент часу  $t_1$  в обох ланцюжках. Вплив одного ланцюга на іншу відбувається через причинно—наслідковий зв'язок.

У даній роботі ми порівнюємо продуктивність ПММ і ППММ для максимального апостеріорного значення (МАЗ) оцінка стану. Ми обчислюємо найбільш вірогідну послідовність станів  $S$  в рамках моделі заданою послідовністю спостережень  $O = \{i_1, \dots, i_n\}$ . Ця найбільш ймовірна послідовність виходить шляхом  $S = \operatorname{argmax}_S P(S|O)$ .

У разі послідовності ПММ задніх станів ймовірність  $P(S|O)$  задається формулою :

$$P(S|O) = P_{s_1} p_{s_1}(o_1) \prod_{t=2}^T p_{s_1}(o_1) P_{s_t|s_{t-1}} \quad (3.1)$$

Де  $S = \{a_1, \dots, a_N\}$  — безліч дискретних станів,  $s_t \in S$  відповідає стану в момент часу  $t$   $P_{i|j} = P_{s_t=a_i|s_{t-1}=a_j}$  це ймовірність переходу зі стану в стан (тобто ймовірність знаходження в стані  $a_i$  в момент часу  $t$  за умови, що система перебувала в стані  $a_j$  в момент часу  $t_1$ ). Надалі вони будуть обозначатися як  $P_i = P_{s_1=a_i} = P_{s_1}$ . Нарешті  $p_i(o_t) = p_{s_t=a_i}(o_t) = p_{s_t}(o_t)$  є вихідні ймовірності для кожного з них, (ймовірність спостереження  $o_t$  заданого стану  $a_i$  в момент часу).

У випадку ППММ нам потрібно ввести ще один набір ймовірностей,  $P_{s_t|s_{t-1}^m}$  який відповідають ймовірності стану  $s_t$  в момент часу  $t$  в одному ланцюзі за умови, що інший ланцюг  $m$  позначається далі супер  $s_{t-1}^m$  під час  $t - 1$ . Ці нові ймовірності виражають причинний вплив (зчеплення) одного ланцюга з іншим. Ймовірність заднього стану для ППММ дорівнює:

$$P(S|O) = \frac{P_{s_1}(o_1)P_{s_1^m}p_{s_1^m}(d_1)}{P(O)} * \prod_{t=2}^T P_{s_t|s_{t-1}} P_{s_t^m|s_{t-1}^m} P_{s_t^m|s_{t-1}} P_{s_t|s_{t-1}^m} p_{s_t}(o_t) p_{s_t^m}(o_t^m) \quad (3.2)$$

Де  $s_t, s_t^m, o_t, o_t^m$  позначимо стану і спостереження для кожної з ланцюгів Маркова, складових ППММ. Ми направляємо читача до МАЗ[13] для більш детального опису оцінки карти в ППММ.

Повертаючись до нашої проблеми моделювання людської поведінки, дві людини (кожен з яких моделюється як генеративний процес) можуть взаємодіяти, не повністю визначаючи поведінку один одного. Напроти, кожен з них володіє власною внутрішньою динамікою і знаходиться під впливом (слабким або сильним) інший. Ймовірності  $P_{s_t|s_{t-1}^m}$  та  $P_{s_t^m|s_{t-1}}$  описують цей вид взаємодій і ППММ призначені для їх моделювання настільки ефективним способом наскільки це можливо.

### 3.3 Синтетичний Поведінковий Агент

Був розроблений фреймворк для створення синтетичних агентів, що імітують поведінку людини у віртуальному середовищі [15], [16]. Агентам можуть бути призначені різні моделі поведінки, і вони також можуть взаємодіяти один з одним. В даний час вони можуть генерувати 5 різних взаємодіючих поведінок і різні види індивідуальної поведінки (без взаємодії). Параметр цієї віртуальної середовища моделюються на основі реальної пішохідної сцени, з якої отримаєм вимірювання типового пішохідного руху.

Одним з основних мотивів побудови таких синтетичних агентів є здатність генерувати синтетичні дані, які дозволяють визначити, яка Архітектура марковської моделі найкраще підходить для розпізнавання нової поведінки (оскільки важко зібрати реальні приклади рідкісних видів поведінки). Конструюючи моделі синтетичних агентів таким чином, щоб вони володіли найкращими властивостями узагальнення і інваріантності, можна отримати гнучкі апріорні моделі, що переносяться на реальна поведінка людини практично без необхідності додаткового навчання. Використання синтетичних агенти для

створення надійних моделей поведінки з дуже небагатьох реальних прикладів поведінки мають особливе значення в задачі візуального спостереження, де, як правило, поведінка представляє найбільший інтерес бувають і найрідкісніші.

### 3.3.1. Архітектура Агента

Дана динамічна мультиагентная система складається з деякого числа агентів, які виконують певну поведінку з набору можливих поведінок. Система запускається в момент часу 0, дискретно рухаючись вперед до моменту часу T або до тих пір, поки агенти не зникнуть зі сцени.

Агенти можуть слідувати трьома різними шляхами з двома можливими напрямками. Вони рухаються зі випадковими швидкостями в межах інтервалу. Вони з'являються у випадкові моменти часу. Вони можуть сповільнюватися, прискорюватися, зупинятися або змінювати напрямок незалежно від інших агентів на сцена. Коли задовольняються певні передумови, відбувається специфічна взаємодія між двома агентами. Кожен агент володіє досконалим знанням світу, включаючи положення інших агентів.

Далі буде описано двухагентную систему, яка використовується для генерації апріорних моделей і синтетичних даних взаємодій агентів. Кожен агент приймає свої власні рішення в залежності від типу взаємодії, його місця розташування та місця розташування іншого агента на сцені. Не існує жодного сценарію поведінки або апріорного знання про те, якого роду взаємодія, якщо таке буде мати місце. Поведінка агентів визначається сприйманої контекстуальною інформацією:

- поточною позицією;
- відносним положенням іншого агента;
- швидкості;
- шляхи, в яких вони знаходяться;
- напрямки ходьби і т. д.

А також за своїм власним репертуаром можливих поведінок і запускають подій. Наприклад, якщо один агент вирішить "слідувати" за іншим агентом, він буде рухатися по своєму власному шляху, поступово збільшуючи свою швидкість, поки

не досягне іншого агента, який також буде йти тим же шляхом. Як тільки агент буде досягнутий, вони адаптують свої взаємні швидкості, щоб триматися разом і продовжувати рухатися разом до тих пір, поки не покинуть сцену.

Для кожного агента вимірюється положення, орієнтація і швидкість, і з цих даних будується вектор ознак, який складається з:  $d_{12}$ — похідної відносної відстані між двома агентами  $a_{1,2} = \text{sign}(\langle v_1, v_2 \rangle)$  або ступінь вирівнювання агентів, і  $v_i = \sqrt{x^2 + y^2}, i = 1, 2$  — величина їх швидкостей. Зверніть увагу, що такий вектор ознак інваріантний до абсолютного положення і напрямку агентів і конкретному середовищі, в якій вони знаходяться.

Правильне проектування інтерактивної поведінки вимагає від агентів знання про положенні один одного, а також синхронізації між послідовними індивідуальними поведінками, активованими в кожному з агентів.

Ці взаємодії можуть відбуватися в будь—який момент часу і в будь—якому місці, за умови тільки, що передумови для цих взаємодій задоволені. Швидкість, з якою вони ходять, тривалість їх розмов, зміна напрямку, початок і закінчення дій сильно розрізняються. Ця висока дисперсія в кількісних аспектах взаємодій надає стійкість вивченим моделям, які, як правило, фіксують тільки інваріантні частини взаємодій. Інваріантність відображає природу їх взаємодій і навколишнього середовища.

## 4 ЕКСПЕРИМЕНТАЛЬНІ ДАНІ

Мета експеримента полягає в тому, щоб мати систему, яка буде точно інтерпретувати поведінку і взаємодію практично в будь-якій пішохідній сцені практично без навчання. Тому однією з найважливіших проблем є створення моделей, що відображають наші попередні знання про людську поведінку. Вибір Пріоров — одне з найбільш спірних і відкритих питань в байєсівському умовиводі. Цю проблему було вирішено за допомогою пакету моделювання синтетичних агентів, який дозволяє нам будувати гнучкі апріорні моделі поведінки.

### 4.1 Поведінка Агентів

Поведінкова система агента структурована ієрархічно. Існують примітивні або прості форми поведінки та складні інтерактивні форми поведінки, що імітують людські взаємодії.

В експериментах описаних було розглянуто п'ять різних взаємодіючих поведінок:

- сліднують, досягають і йдуть разом (взаємодія 1);
- підходять, зустрічаються і йдуть далі порізно (взаємодія 2);
- підходять, зустрічаються і продовжують разом (взаємодія 3);
- змінюють напрямок, щоб зустрітися, наближуються, зустрічаються і продовжують разом (взаємодія 4);
- змінюють напрямок, щоб зустрітися, наближуються, зустрічаються і йдуть далі окремо (взаємодія 5).

Два агенти опиняються на одному шляху, йдучи в одному напрямку(рисунок 4.1). Агент позаду вирішує, що він хоче дістатися до інший. Тому він прискорюється, щоб досягти іншого агента. Коли це відбувається він сповільнюється так що вони продовжують йти разом з тією ж швидкістю.

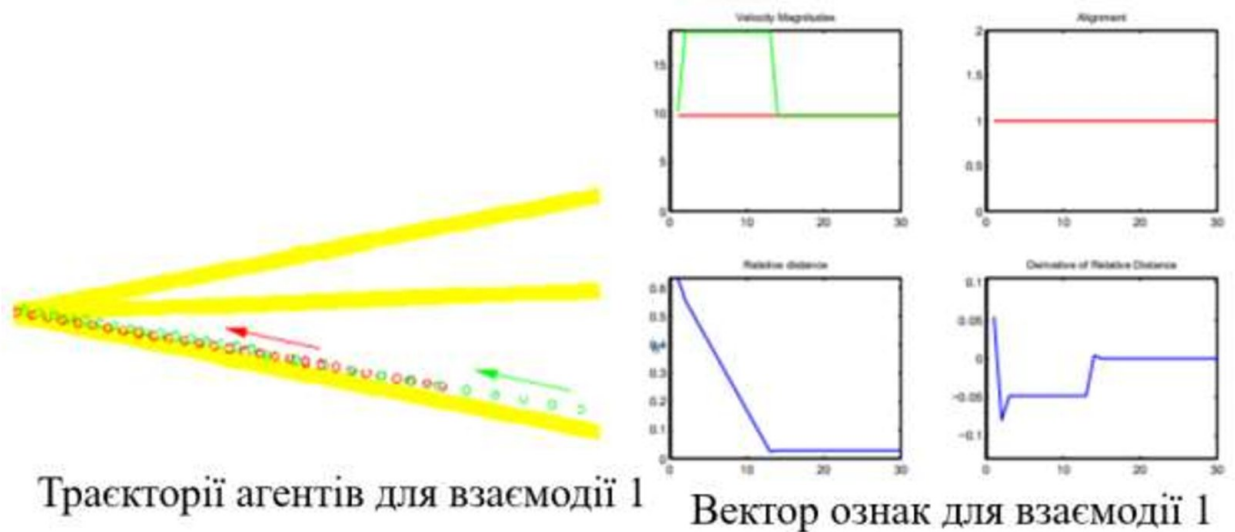


Рисунок 4.1 — Приклад траєкторій і вектор ознак для взаємодій: «Слідують, досягають і йдуть разом»

Агенти йдуть по одному і тому ж шляху, але в протилежному напрямку(рисунок 4.2). Коли вони досить близькі, якщо вони розуміють, що «знають» друг друга, вони сповільнюються і, нарешті, зупиняються, щоб поговорити. Після розмови вони йдуть окремо, знову стаючи незалежними.

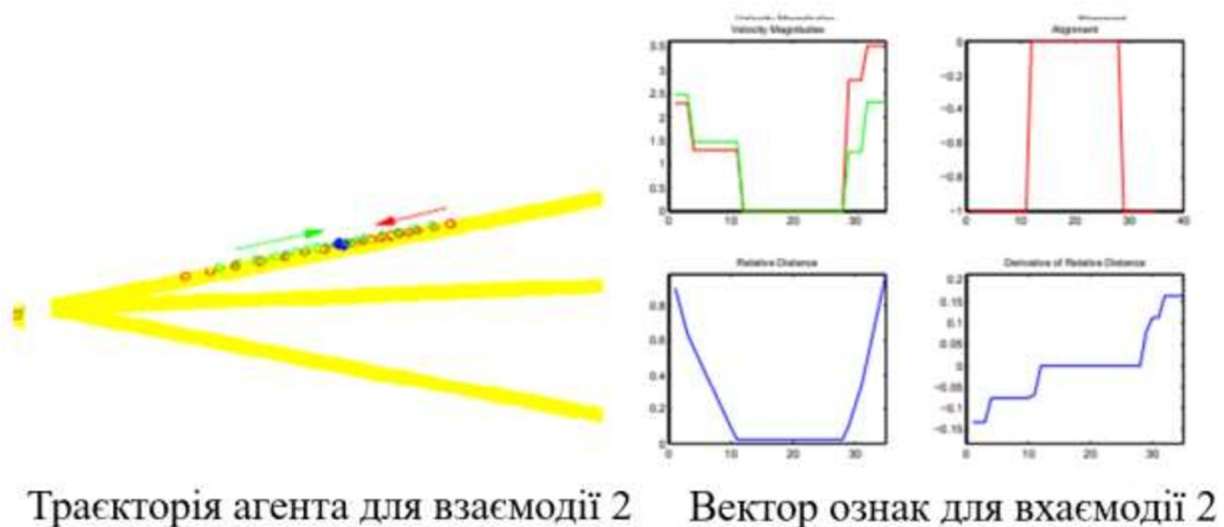


Рисунок 4.2 — Приклад траєкторій і вектор ознак для взаємодій: «Підхід + зустріч + продовження окремо»

В цьому випадку агенти поведуться як в взаємодії 2, але тепер після розмови вони вирішують продовжити разом (рисунок 4.3). Тому один агент змінює своє напрямком, щоб слідувати за іншим.

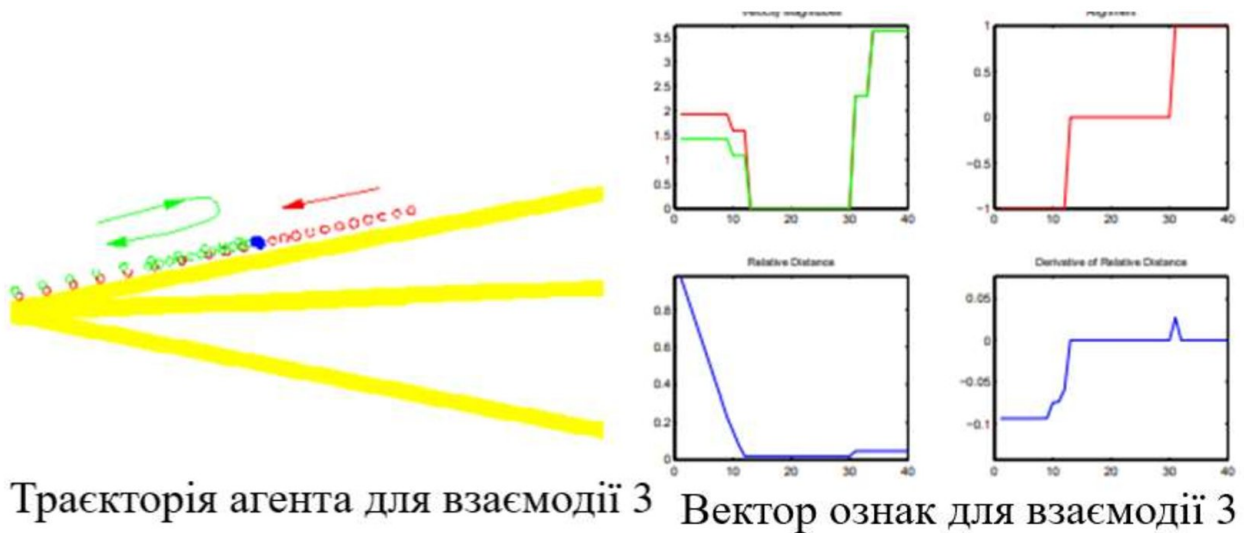


Рисунок 4.3 — Приклад траєкторій і вектор ознак для взаємодій: «Підхід + зустріч + продовження разом»

Агенти починають по різних шляхах. Коли вони знаходяться досить близько, вони можуть бачити один одного і вирішують взаємодіяти (рисунок 4.3). Один агент чекає, поки інший добереться до нього. Інший змінює напрямком, щоб попрямувати до очікує агента. Потім вони зустрічаються, базикають деякий час і вирішують продовжити разом.

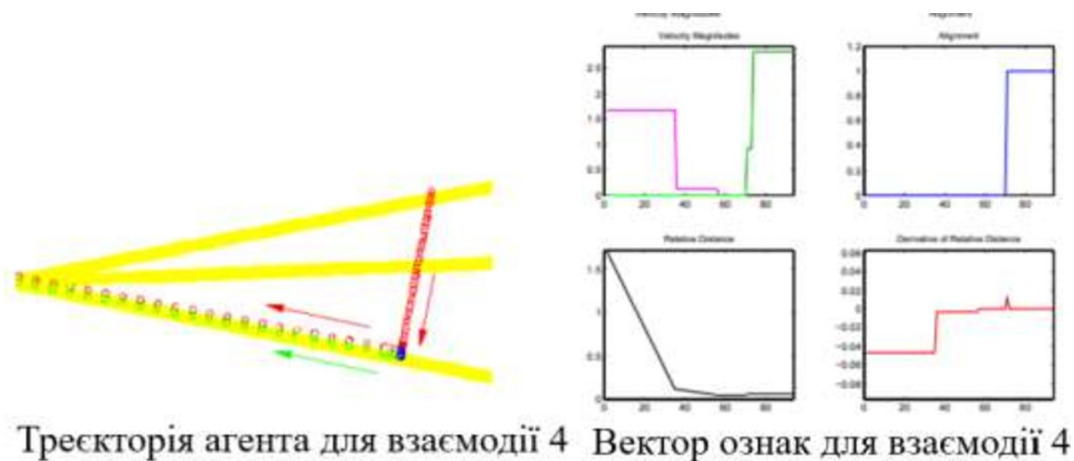


Рисунок 4.4 — Приклад траєкторій і вектор ознак для взаємодій: «Змінити напрямком + підійти + зустрітися + продовжити разом»

Взаємодія 5 така ж, як і взаємодії 4, за винятком того, що, коли вони вирішують йти далі після розмови, вони поділяються, стаючи незалежними(рисунок 4.5).

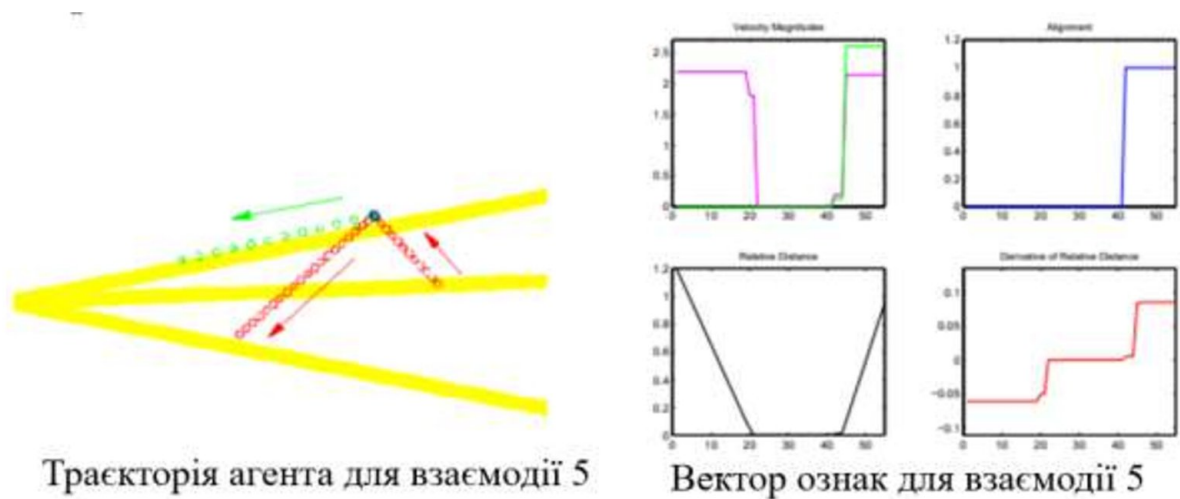


Рисунок 4.5 — Приклад траєкторій і вектор ознак для взаємодій: «Змінити напрямок + підійти + зустрітися + продовжити окремо»

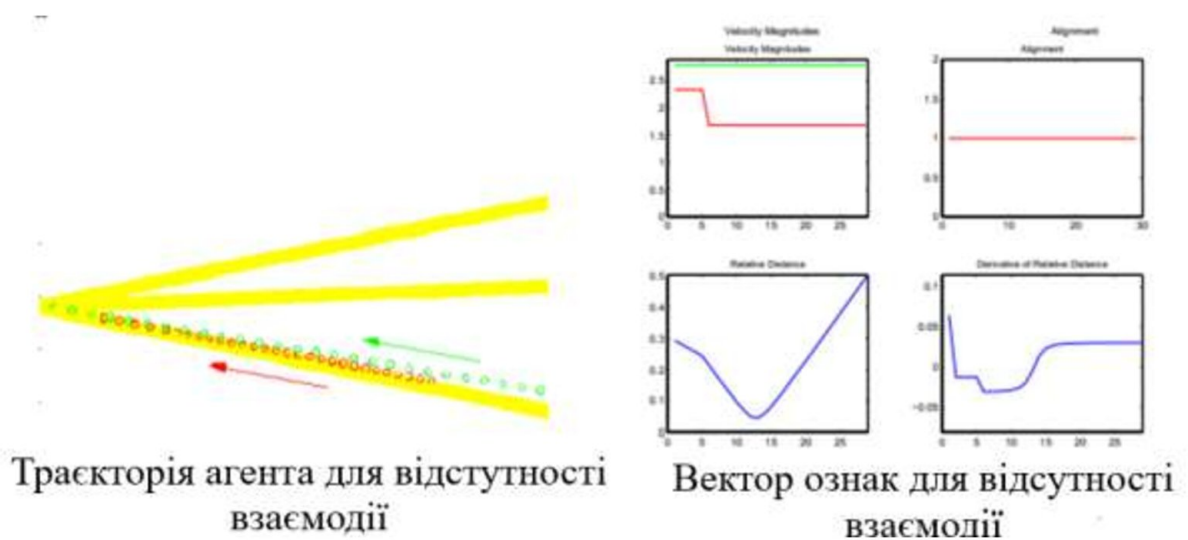


Рисунок 4.6 — Приклад траєкторій і вектор ознак для взаємодій: «Відсутність взаємодіючої поведінки»

4.2 Порівняння архітектур ПММ і ППММ з даними синтетичного агента

Було побудовано моделі взаємодії п'яти раніше описаних синтетичних агентів з обома ПММ і ППММ. Також було використано 2 або 3 стани на

ланцюжок у випадку ППММ і від 3 до 5 станів у випадку ПММ (залежно від складності різних взаємодій) (рисунок 4.7).

Кожна з цих архітектур відповідає різній фізичній гіпотезі: ППММ кодують просторовий зв'язок у часі між двома агентами (наприклад, нестационарний процес), тоді як ПММ моделюють дані як ізольований стаціонарний процес. Було використано від 11 до 75 послідовностей для навчання кожної з моделей, в залежності від їх складності, так що не було надмірного перенасичення. Оптимальна кількість навчальних станів для кожної взаємодії, а також оптимальні параметри моделі були отримані шляхом 10%—ної перехресної валідації. У всіх випадках моделі були налаштовані з повною топологією з'єднання між станами, так що алгоритм навчання відповідав за визначення відповідної структури станів для навчальних даних. Вектор ознак був 6—мірним у випадку ПММ, тоді як у випадку ППММ кожен агент моделювався різним ланцюжком, кожен з яких мав 3—мірний вектор ознак.

Щоб порівняти продуктивність двох раніше описаних архітектур, ми використовували кращі навчені моделі для класифікації 20 невидимих нових послідовностей. Щоб знайти найбільш ймовірну модель, був використаний алгоритм Вітербі для ПММ. А N—heads алгоритм динамічного програмування прямого і зворотного поширення для ППММ.

Таблиця 4.1 — Точність для ПММ і ППММ на синтетичних даних

Точність синтетичних тестових даних (%)		
	ПММ	ППММ
Нема взаємодії	68.7	90.9
Взаємодія 1	87.5	100
Взаємодія 2	85.4	100
Взаємодія 3	91.6	100
Взаємодія 4	77	100
Взаємодія 5	97.9	100

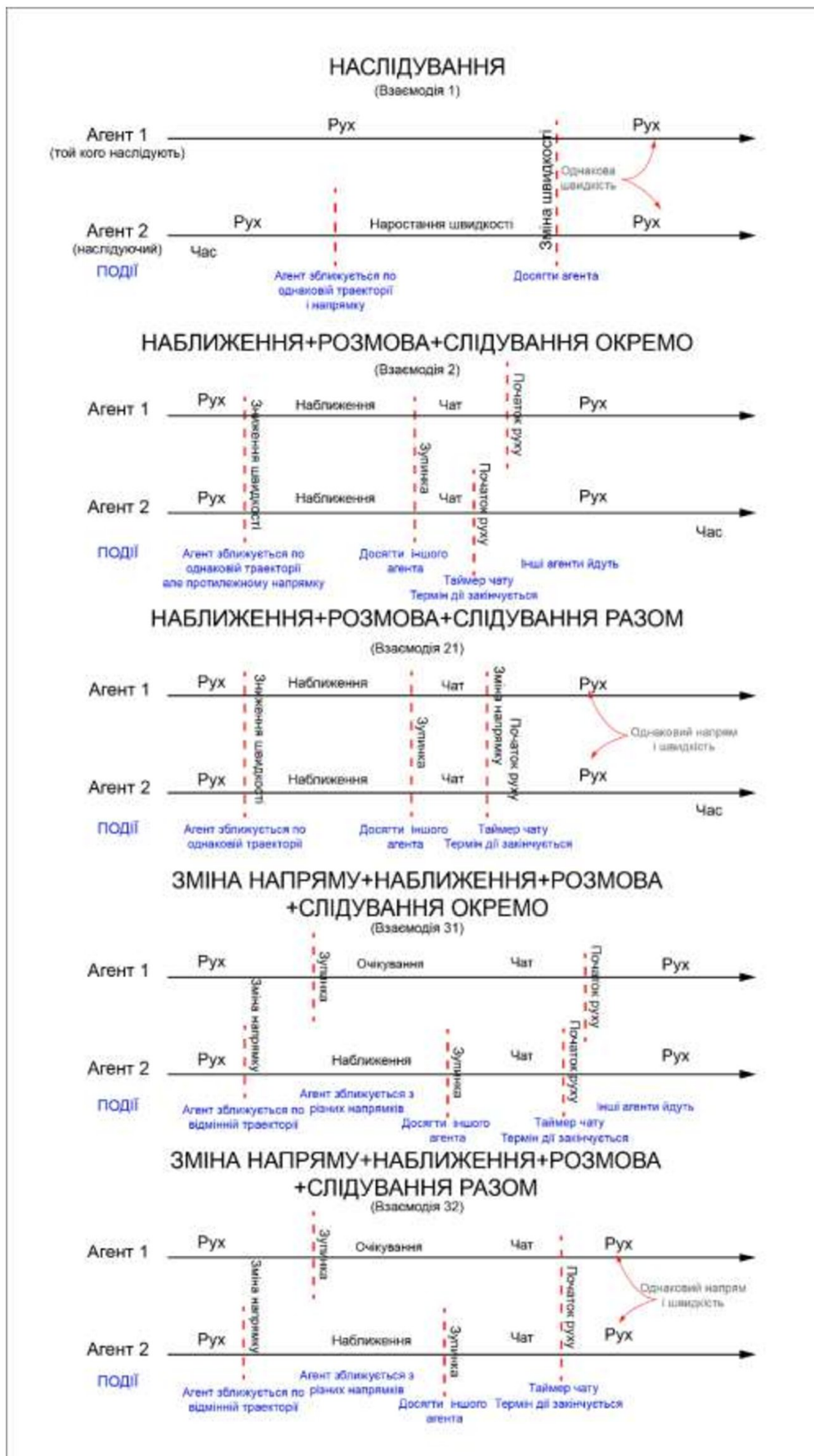


Рисунок 4.7 — Хронологія п'яти складних поведінок з точки зору подій і простих поведінок

Складність у часі та просторі є важливою проблемою при моделюванні динамічних часових рядів. Число ступенів свободи (ймовірності від стану до стану+вихідні середні+вихідні коваріації) в найбільшій моделі з найкращими оцінками склало 85 для ПММ і 54 для ППММ. Також був проведений аналіз точності моделей і архітектур щодо кількості послідовностей, що використовуються для навчання. Ефективність в термінах дані навчання особливо важливі в онлайн системі навчання в реальному часі. Таких, якими в кінцевому рахунку буде ця, в яких збір чистих позначених даних може бути утруднений.

Міжпродуктові ПММ, що виникають в результаті включення обох генеративних процесів в одне і те ж ж простір станів спільного продукту, зазвичай вимагають набагато більшої кількості послідовностей для навчання за більшого числа параметрів. Але це, здається, призводить до стелі точності близько 80% для будь-якої кількості оцінюваних тренувань, в той час як для ППММ було досягнуто приблизно 100% точності тільки з невеликою кількістю тренувань. Із цього результату випливає, що архітектура ППММ з двома пов'язаними генеративними процесами є більше підходить для завдання моделювання поведінки взаємодіючих агентів, ніж генеративний процес, закодований одним ПММ.

У системі візуального спостереження частота помилкових тривог часто так само важлива, як і точність класифікації. В ідеальній автоматичній системі спостереження все цілеспрямоване поведінка має бути виявлено з близькою до нуля частотою помилкових тривог, щоб ми могли розумно попередити людину — оператора про необхідність їх подальшого вивчення. Щоб проаналізувати цей аспект продуктивності нашої системи, ми розрахували ROC-криву системи. Рисунок 6.4 показує, що цілком можливо досягти дуже низька частота помилкових тривог при збереженні хорошої точності класифікації.

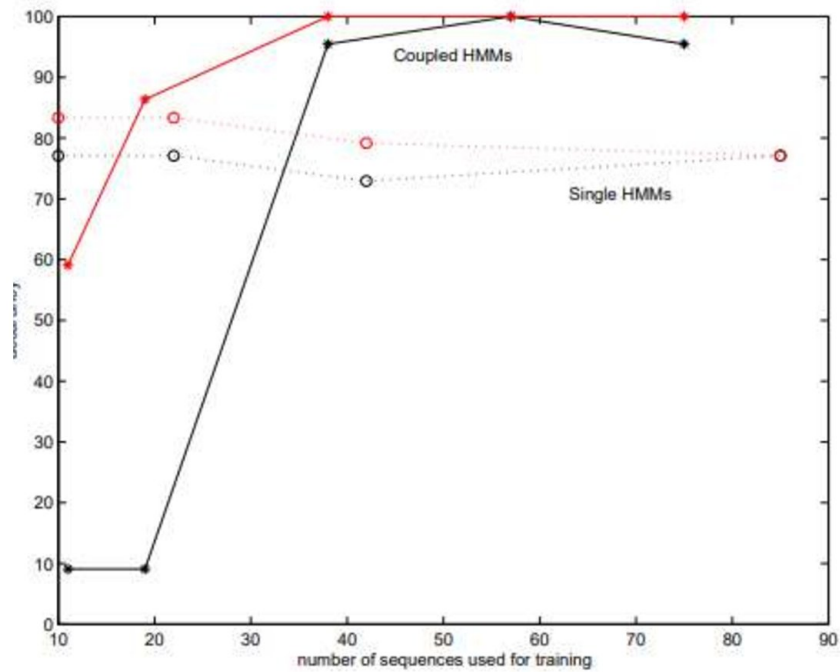


Рисунок 4.8 – Точність ППММ (суцільна лінія) і ПММ (пунктирна лінія) для одного конкретного взаємодія

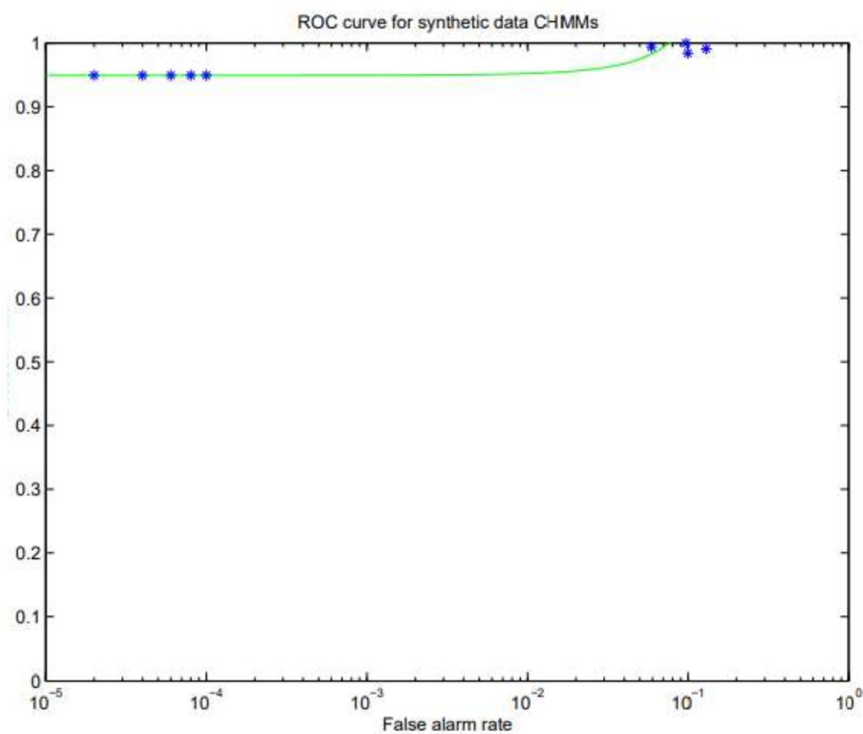


Рисунок 4.9 – Крива ROC на синтетичних даних.

## 4.2 Поведінка Пішоходів

Мета роботи – розробити основу для виявлення, класифікації та вивчення загальних моделей поведінки в ситуації візуального спостереження. Важливо,

щоб моделі були універсальними, застосовними до багатьох різних ситуацій, а не налаштовувалися на конкретне відео або сайт. Це було одним з головних мотивів для розробки віртуального середовища агента для моделювання поведінки. Якщо синтетичні агенти досить "схожі" в своїй поведінці на людей, то ті ж моделі, які були навчені за допомогою синтетичних даних, повинні бути безпосередньо застосовні до людських даних. У цьому розділі описуються експерименти, які були проведені, аналізуючи реальні пішохідні дані з використанням як синтетичних, так і специфічних для об'єкта моделей (моделей, навчених на даних з контрольованого об'єкт).

#### 4.2.1 Збір та попередня обробка даних

Використовуючи систему виявлення та відстеження людей, описану в розділі 2, було отримано 2D – об'єкти «blob» для кожної людини за кілька годин відео. Було виявлено та оброблено до 20 прикладів наступного та різних типів поведінки при зустрічі.

Вектор ознак  $x$ , що надходить з модуля обробки комп'ютерного зору, складався з 2D  $(x; y)$  Центроїд (середнє положення) краплі кожної людини, стан фільтра Калмана для кожного моменту часу, що складається  $\hat{x}, \hat{\dot{x}}, \hat{y}, \hat{\dot{y}}$ , де  $\hat{\cdot}$ : представляє оцінку фільтра і компоненти  $(r, g, b)$  середнього значення Гауссова ТТ для кожного згустку в колірному просторі. Частота кадрів зорової системи становила близько 20–30 Гц на комп'ютері SGI R10000 O2. Коли було відфільтровано дані низькочастотним фільтром з частотою 3 Гц і вираховали для кожної пари найближчих людей вектор ознак, що складається з:  $d_{12}$ , похідна від відносної відстані між двома людьми,  $|v_i|, i = 1, 2$ , норма вектора швидкості для кожної людини,  $a = \text{sign}(\langle v_1, v_2 \rangle)$ , або ступінь вирівнювання траєкторій кожної людини. Типові траєкторії і вектори ознак для поведінки "підхід, зустріч і продовження окремо" (взаємодія 2) Показані на рисунку 4.10. Це той же тип поведінки, що і "взаємодія 2", показаний на малюнку 4.2 для синтетичних агент. Зверніть увагу на схожість векторів ознак в обох випадках.

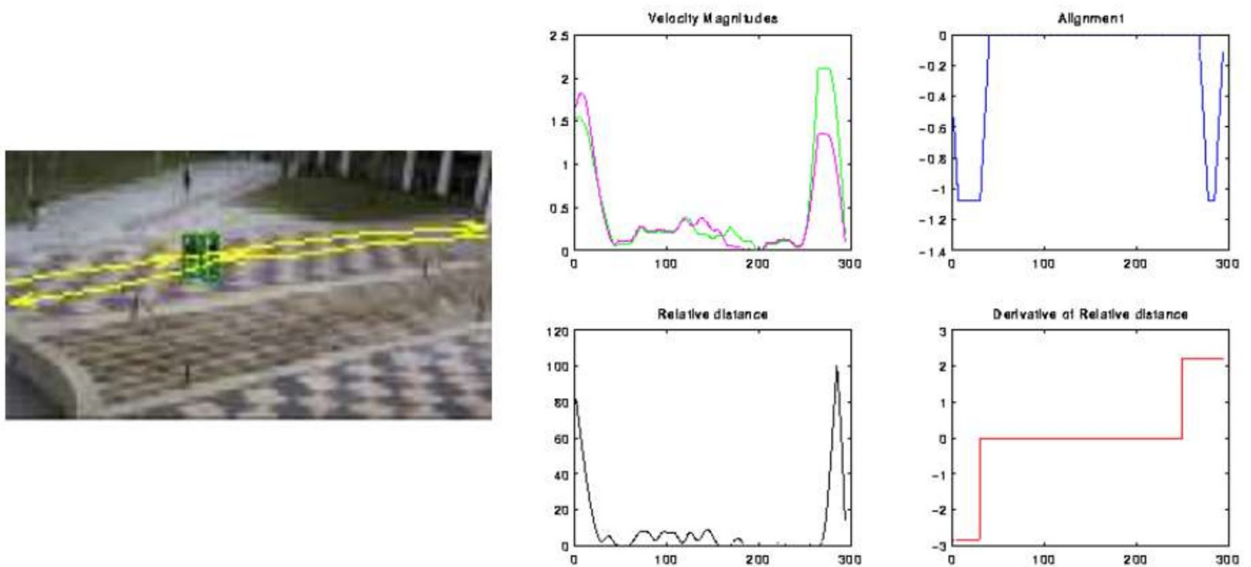


Рисунок 4.10 — Приклад траєкторії і вектор ознак для взаємодії 2, або «підхід, зустріч і продовження окремо» поведінка.

#### 4.2.2 Моделі поведінки та результати

Алгоритм ППММ використовувався для моделювання трьох різних моделей поведінки: «зустрітися і продовжити разом» (взаємодія 3); «зустрітися і розділити» (взаємодія 2) і «слідувати» (взаємодія 1). Крім того, був також проведено тест на виявлення взаємодії порівняно з відсутністю взаємодії. ПММ показали набагато гірші результати, ніж ППММ, і тому не будемо розглядати результати.

Було використано моделі, навчені двом типам даних:

- моделі тільки для попередніх (синтетичних даних): тобто моделі поведінки, вивчені в середовищі синтетичних агентів, а потім безпосередньо застосовуються до реальних даних без додаткового навчання або налаштування параметрів;

- задні (синтетичні плюс реальні дані) моделі: нові моделі поведінки, навчені з використанням в якості відправної точки синтетичних кращих моделей. Ми використовували 8 прикладів кожної взаємодії даних з конкретного сайту.

Точність розпізнавання як для цих "попередніх", так і для "задніх" ППММ підсумована в таблиці 4.2 примітно, що тільки за допомогою 8 навчальних

прикладів точність розпізнавання на реальних даних може бути підвищена до 100%. Ці результати демонструють здатність здійснювати надзвичайно швидке відновлення наших моделей поведінки в порівнянні з вихідними попередніми моделями.

Таблиця 4.2 — Точність як для неналаштованих, апріорних моделей, так і для конкретних об'єктів ППММ, перевірених на реальних пішохідних даних

Точність на реальних пішохідних тестових даних (%)		
	Попередній ППММ	Задній ППММ
Нема взаємодії	90.9	90.9
Взаємодія 1	93.7	100
Взаємодія 2	100	100
Взаємодія 3	100	100

Нарешті, крива ROC для задніх ППММ показана на рисунку 6.11.

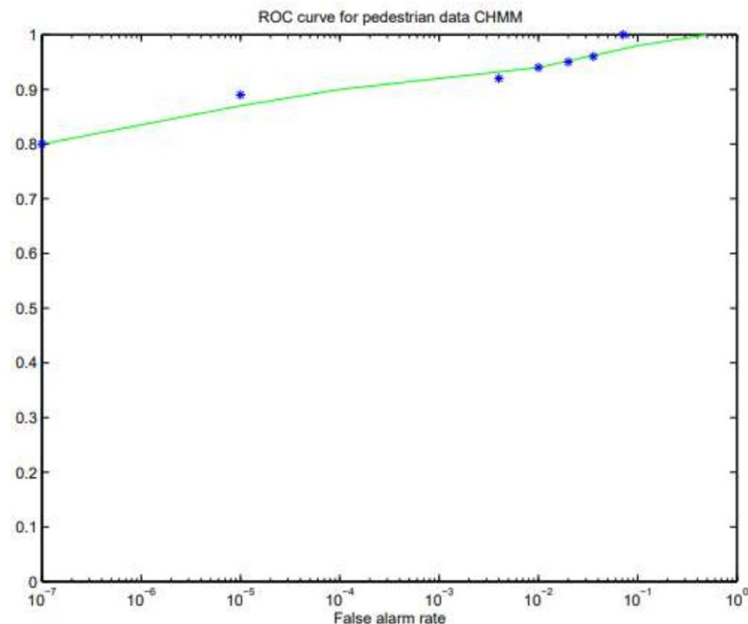


Рисунок 4.11 — Крива ROC для реальних пішохідних даних

Одним з найцікавіших результатів цих експериментів є висока точність, отримана при тестуванні апріорних моделей, отриманих при моделюванні синтетичних агентів. Той факт, що апріорні моделі так добре переносяться на реальні дані, що демонструє надійність підходу. Це показує, що за допомогою цієї

системи навчання синтетичних агентів можна розробляти моделі багатьох різних типів поведінки. Уникаючи, таким чином, проблеми обмеженого обсягу навчальних даних і застосовувати ці моделі до реальної поведінки людини без додаткового налаштування параметрів або навчання.

#### 4.2.3 Чутливість параметрів

Для оцінки чутливості та точності класифікації до змін параметрів моделі було навчано набір моделей, в яких було змінено різні параметри динаміки агентів на коефіцієнти 2.5 і 5. Продуктивність цих змінених моделей виявилася практично однаковою у всіх випадках, за винятком "взаємодії 1" взаємодія, яка, здається, чутлива до відносної швидкості руху людей.

## ВИСНОВОК

У цій статті було описано систему комп'ютерного зору і математичну модель для розпізнавання різних видів людської поведінки і взаємодій в задачі візуального спостереження. Ця система об'єднує спадну і висхідну інформацію в замкнутому контурі зворотного зв'язку, причому обидва компоненти використовують статистичний байєсівський підхід.

Для моделювання поведінки та взаємодій були запропоновані та зіставлені дві різні архітектури статистичного навчання на основі станів, а саме ПММ та ППММ. Перевага формулювання ППММ було продемонстровано як з точки зору ефективності навчання, так і з точки зору точності класифікації. Задля розробки гнучких та інтерпретованих апріорних моделей поведінки була створена система навчання синтетичних агентів, і продемонструвано здатність використовувати ці апріорні моделі для точної класифікації реальної поведінки без будь—яких додаткових зусиль. налаштування або тренування. Цей факт особливо важливий, враховуючи обмежений обсяг наявних даних про навчання.

## ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Oliver N. M., Rosario B., Pentland A. P. A Bayesian computer vision system for modeling human interactions //IEEE transactions on pattern analysis and machine intelligence. – 2000. – Т. 22. – №. 8. – С. 831-843.
2. Darrell T., Pentland A. Active gesture recognition using partially observable Markov decision processes //Proceedings of 13th International Conference on Pattern Recognition. – IEEE, 1996. – Т. 3. – С. 984-988.
3. Bobick A. F. Computers Seeing Action //BMVC. – 1996. – С. 1-10.
4. Pentland A., Liu A. Modeling and prediction of human behavior //Neural computation. – 1999. – Т. 11. – №. 1. – С. 229-242.
5. Buxton H. Advanced visual surveillance using Bayesian networks. – 1997. – С. 1-40.
6. Nagel H. H. From image sequences towards conceptual descriptions //Image and vision computing. – 1988. – Т. 6. – №. 2. – С. 59-74.
7. Huang T. Et al. Automatic symbolic traffic scene analysis using belief networks //AAAI. – 1994. – Т. 94. – С. 966-972.
8. Buntine W. A guide to the literature on learning probabilistic networks from data //IEEE Transactions on knowledge and data engineering. – 1996. – Т. 8. – №. 2. – С. 195-210.
9. Fernyhough J., Cohn A. G., Hogg D. C. Constructing qualitative event models automatically from video input //Image and Vision Computing. – 2000. – Т. 18. – №. 2. – С. 81-103.
10. Buntine W. L. Operations for learning with graphical models //Journal of artificial intelligence research. – 1994. – Т. 2. – С. 159-225.
11. Rabiner L. R. A tutorial on hidden Markov models and selected applications in speech recognition //Proceedings of the IEEE. – 1989. – Т. 77. – №. 2. – С. 257-286.
12. Brand M., Oliver N., Pentland A. Coupled hidden Markov models for complex action recognition //Proceedings of IEEE computer society conference on computer vision and pattern recognition. – IEEE, 1997. – С. 994-999.
13. Brand M. Coupled hidden Markov models for modeling interacting

processes. – 1997. – C. 35-55.

14. Oliver N., Rosario B., Pentland A. Graphical models for recognizing human interactions //Advances in Neural Information Processing Systems. – 1999. – C. 924-930.

15. Rosario B., Oliver N., Pentland A. A synthetic agent system for bayesian modeling of human interactions // Proceedings of the third annual conference on Autonomous Agents. – 1999. – T. 1999. – C. 342-343.

16. Yilmaz A., Javed O., Shah M. Object tracking: A survey //Acm computing surveys (CSUR). – 2006. – T. 38. – №. 4. – C. 13-70.

17. Roweis S., Ghahramani Z. A unifying review of linear Gaussian models //Neural computation. – 1999. – T. 11. – №. 2. – C. 305-345.

18. Brand M. An entropic estimator for structure discovery //Advances in Neural Information Processing Systems. – 1999. – C. 723-729.

19. Frey B. J., Brendan J. F., Frey B. J. Graphical models for machine learning and digital communication. – MIT press, 1998. – C. 550-595.

20. Pearl J. Probabilistic reasoning in intelligent systems: networks of plausible inference. – Elsevier, 2014 – C. 90-131.

21. Lee P. M. Bayesian Statistics: An Introduction Edward Arnold. – 1989. C. 220-21.

22. Gilks W. R., Thomas A., Spiegelhalter D. J. A language and program for complex Bayesian modelling //Journal of the Royal Statistical Society: Series D (The Statistician). – 1994. – T. 43. – №. 1. – C. 169-177.

23. Ballard D. H., Brown C. M. Computer vision, article, 4 pages prentice-hall //Englewood Cliffs, New Jersey, believed to be published more than one year prior to the filing date of the present application. – 1982. – C. 60-80.

24. Paragios, N. and deriche, R. 2000. Geodesic active contours and level sets for the detection and tracking of moving objects. IEEE Trans. Patt. Analy. Mach. Intell. 22, 3. – C. 340-397.

25. Fieguth, P. and Terzopoulos, D. 1997. Color-based tracking of heads and other mobile objects at video frame rates. In IEEE Conference on Computer Vision and

Pattern Recognition (CVPR). – C. 600-635.

26. Veenman, C., Reinders, M., and Backer, E. 2001. Resolving motion correspondence for densely moving points. *IEEE Trans. Patt. Analy. Mach. Intell.* 23, 1. – C. 54-72.

27. Shafique, K. and Shah, M. 2003. A non-iterative greedy algorithm for multi-frame point correspondence. In *IEEE International Conference on Computer Vision (ICCV)*. – C. 102-112.

28. Haritaoglu, I., Harwood, D., and Davis, L. 2000. W4: real-time surveillance of people and their activities. *IEEE Trans. Patt. Analy. Mach. Intell.* 22, 8, – C. 809-830.

29. Paschos, G. 2001. Perceptually uniform color spaces for color texture analysis: an empirical evaluation. *IEEE Trans. Image Process.* 10, – C. 932-972.

30. Bowyer, K., Kranenburg, C., and Dougherty, S. 2001. Edge detector evaluation using empirical roc curve. *Comput. Vision Image Understand.* 10, – C. 77-103.

31. Lowe, D. 2004. Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vision* 60, 2, – C. 91-110.

32. Comaniciu, D. and Meer, P. 2002. Mean shift: A robust approach toward feature space analysis. *Trans. Patt. Analy. Mach. Intell.* 24, 5, – C. 603-619.

33. Moravec, H. 1979. Visual mapping by a robot rover. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*. – C. 598-600.

34. Mikolajczyk, K. and Schmid, C. 2002. An affine invariant interest point detector. In *European Conference on Computer Vision (ECCV)*. Vol. 1. – C. 128-142.

35. Wren, C., Azarbayejani, A., and Pentland, A. 1997. Real-time tracking of the human body. *IEEE Trans. Patt. Analy. Mach. Intell.* 19, 7, – C. 780-785.

36. Elgammal, A., Duraiswami, R., Harwood, D., and Davis, L. 2002. Background and foreground modeling using nonparametric kernel density estimation for visual surveillance. *Proceedings of IEEE* 90, 7, – C. 1151-1163.

37. Yilmaz, A., Li, X., and Shah, M. 2004. Contour based object tracking with occlusion handling in video acquired using mobile cameras. *IEEE Trans. Patt. Analy.*

Mach. Intell. 26, 11, – C. 1531-1540.

38. Toyama, K., J. Krumm, B. B., and Meyers, B. 1999. Wallflower: Principles and practices of background maintenance. In IEEE International Conference on Computer Vision (ICCV). – C. 255-261.

39. Stenger, B., Ramesh, V., Paragios, N., Coetzee, F., and Buhmann, J. 2001. Topology free hidden markov models: Application to background modeling. In IEEE International Conference on Computer Vision (ICCV). – C. 294-301.

40. Oliver, N., Rosario, B., and Pentland, A. 2000. A bayesian computer vision system for modeling human interactions. IEEE Trans. Patt. Analy. Mach. Intell. 22, 8, – C. 831-943.

41. Xu, N. and Ahuja, N. 2002. Object contour tracking using graph cuts based active contours. In IEEE International Conference on Image Processing (ICIP). – C. 420-423.

42. Caselles, V., Kimmel, R., and Sapiro, G. 1995. Geodesic active contours. In IEEE International Conference on Computer Vision (ICCV). – C. 694-699.

43. Paragios, N. and Deriche, R. 2000. Geodesic active contours and level sets for the detection and tracking of moving objects. IEEE Trans. Patt. Analy. Mach. Intell. 22, 3, – C. 112-124.

44. Sethi, I. and Jain, R. 1987. Finding trajectories of feature points in a monocular image sequence. IEEE Trans. Patt. Analy. Mach. Intell. 9, 1, – C. 56-73.

45. Salari, V. and Sethi, I. K. 1990. Feature point correspondence in the presence of occlusion. IEEE Trans. Patt. Analy. Mach. Intell. 12, 1, – C. 87-95.

46. Intille, S., Davis, J., and Bobick, A. 1997. Real-time closed-world tracking. In IEEE Conference on Computer Vision and Pattern Recognition (CVPR). – C. 697-703.

47. Saul L., Jordan M. Boltzmann chains and hidden Markov models //Advances in neural information processing systems. – 1994. – T. 7. – C. 435-442.

48. Ghahramani Z., Jordan M. I. Factorial hidden Markov models //Machine learning. – 1997. – T. 29. – №. 2. – C. 245-273.