

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

До захисту допускається  
Завідувач кафедри  
\_\_\_\_\_ І.С. Скарга-Бандурова  
« \_\_\_\_ » \_\_\_\_\_ 20\_\_ р.

**ДИПЛОМНИЙ ПРОЕКТ (РОБОТА) БАКАЛАВРА**  
**ПОЯСНЮВАЛЬНА ЗАПИСКА**

НА ТЕМУ:

**Методи структурного розпізнавання візуальних об'єктів**

---

---

---

Освітній рівень “бакалавр”  
Спеціальність 123 “Комп’ютерна інженерія”

Науковий керівник роботи:

\_\_\_\_\_

(підпис)

О.І.Рязанцев

(ініціали, прізвище)

Консультант з охорони праці:

\_\_\_\_\_

(підпис)

Я.О.Критська

(ініціали, прізвище)

Здобувач вищої освіти:

\_\_\_\_\_

(підпис)

Е.В.Приходько

(ініціали, прізвище)

Група:

КІ-163

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
СХІДНОУКРАЇНСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ  
ІМЕНІ ВОЛОДИМИРА ДАЛЯ

Факультет Інформаційних технологій та електроніки

Кафедра Комп'ютерних наук та інженерії

Освітній рівень Бакалавр

Спеціальність 123 "Комп'ютерна інженерія"

(шифр і назва)

**ЗАТВЕРДЖУЮ:**

Т.в.о. завідувача кафедри \_\_\_\_\_

С.О. Сафонова

« \_\_\_\_\_ » \_\_\_\_\_ 20 \_\_\_\_\_ р.

**З А В Д А Н Н Я  
НА ДИПЛОМНИЙ ПРОЕКТ (РОБОТУ) БАКАЛАВРА**

Приходько Едуарду Віталійовичу

(прізвище, ім'я, по батькові)

1. Тема роботи Методи структурного розпізнавання візуальних об'єктів

керівник проекту (роботи) Рязанцев Олександр Іванович, д.т.н., проф.

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

затверджені наказом вищого навчального закладу від «30» 04 2020 р. № 77/15.15

2. Строк подання студентом роботи 10.06.2020

3. Вихідні дані до роботи Матеріали переддипломної практики,

тестові зображення, література у галузі розробки методів розпізнавання

зображень, теоретичні відомості про структурні методи аналізу розпізнавання

зображень, середовище розробки Microsoft Visual Studio 2012

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно

розробити) Методи структурного розпізнавання зображень, застосування

статистичних ознак в структурному розпізнаванні зображень, охорона праці,

висновки

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

Електронні плакати

## 6. Консультанти розділів проекту (роботи)

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

7. Дата видачі завдання 30.04.2020

Керівник

\_\_\_\_\_ (підпис)

Завдання прийняв до виконання

\_\_\_\_\_ (підпис)

## КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломного проекту (роботи)	Строк виконання етапів проекту ( роботи )	Примітка
1	Розробка технічного завдання	01.05.2020-07.05.2020	
2	Аналіз завдання, огляд літератури	08.05.2020-10.05.2020	
3	Аналіз програмних засобів	10.05.2020-13.05.2020	
4	Розробка частини проекту "Охорона праці"	13.05.2020-15.05.2020	
5	Реалізація методу	15.05.2020-01.06.2020	
6	Оформлення пояснювальної записки та презентації	2.06.2020-09.06.2020	

**Здобувач вищої освіти**

\_\_\_\_\_ ( підпис )

**Е.В.Приходько**

\_\_\_\_\_ (прізвище та ініціали)

**Науковий керівник**

\_\_\_\_\_ ( підпис )

**О.І.Рязанцев**

\_\_\_\_\_ (прізвище та ініціали)

## РЕФЕРАТ

Пояснювальна записка дипломної роботи бакалавра містить: 69 с., 20 рис., 1 табл., 27 бібліографічних джерел посилань, 2 додатки.

Дана робота присвячена моделюванню та обчисленню статистичних ознак для структурного розпізнавання візуальних об'єктів. Проведення моделювання засновано на ефективному сучасному методі структурного аналізу SURF, який вирішує завдання пошуку та ідентифікації особливих точок зображення та створення дескрипторів, інваріантних до зсувів, масштабу і обертання.

Створена програмна модель методу в середовищі MS Visual Studio 2012 з використанням бібліотеки OpenCV. З використанням вбудованих програмних засобів при здійсненні геометричних перетворень сформовані множини характерних ознак, що відображають властивості реальних відео-об'єктів.

**Ключові слова:** характерні ознаки, розпізнавання, інваріантність, дескриптор, метод surf

## ЗМІСТ

ПЕРЕЛІК СКОРОЧЕНЬ ТА УМОВНИХ ПОЗНАЧЕНЬ.....	5
ВСТУП.....	6
1 МЕТОДИ СТРУКТУРНОГО РОЗПІЗНАВАННЯ ЗОБРАЖЕНЬ .....	8
1.1 Використання методу SURF для формування характерних ознак .....	8
1.2 Моделі структурного розпізнавання .....	18
1.3 Постановка задачі.....	25
2 ЗАСТОСУВАННЯ СТАТИСТИЧНИХ ОЗНАК В СТРУКТУРНОМУ РОЗПІЗНАВАННІ ЗОБРАЖЕНЬ .....	26
2.1 Формування набору класифікаційних ознак.....	26
2.2 Результати моделювання .....	30
3 ОХОРОНА ПРАЦІ .....	40
3.1 Аналіз потенційно небезпечних і шкідливих виробничих факторів , що впливають на персонал .....	40
3.2 Заходи щодо техніки безпеки .....	41
3.3 Заходи, що забезпечують виробничу санітарію і гігієну праці .....	45
3.4 Рекомендації по пожежній профілактиці .....	49
ВИСНОВКИ .....	53
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ.....	54
Додаток А Лістинг програми.....	57
Додаток Б Комп'ютерна презентація .....	65

## ПЕРЕЛІК СКОРОЧЕНЬ ТА УМОВНИХ ПОЗНАЧЕНЬ

SURF – (Speeded Up Robust Features) – прискорене виявлення стійких ознак

OpenCV – (Open Source Computer Vision Library) – бібліотека комп'ютерного зору з відкритим кодом

GDI – (Graphics Device Interface) - інтерфейс для представлення графічних об'єктів

НШВФ - небезпечні і шкідливі виробничі фактори

РО – розпізнавання образів

ХО – характерні ознаки

ШІ – штучний інтелект

## ВСТУП

Із завданням розпізнавання образів живі системи, у тому числі і людина, стикаються постійно з моменту своєї появи. Зокрема, інформація, що надходить з органів почуттів, обробляється мозком, який у свою чергу сортує інформацію, забезпечує прийняття рішення, а далі за допомогою електрохімічних імпульсів передає необхідний сигнал далі, наприклад, органам руху, які реалізують необхідні дії. Потім відбувається зміна навколишнього оточення, і вищевказані явища відбуваються заново. І якщо розібратися, то кожен етап супроводжується розпізнаванням.

З розвитком обчислювальної техніки стало можливим вирішити ряд завдань, що виникають в процесі життєдіяльності, полегшити, прискорити, підвищити якість результату.

Розвиток сучасних технологій штучного інтелекту викликає потребу в розробці нових, більш ефективних підходів до аналізу багатовимірної інформації, наприклад, при розпізнаванні зображень. Рівень інтелектуалізації сучасних систем комп'ютерного зору (СКЗ) визначається якістю вирішення проблем візуального аналізу при виділенні і розпізнаванні зображень об'єктів. Цей напрямок у сучасній науці є одним з найбільш швидко розвиваються і пов'язане зі створенням штучних інтелектуальних систем для «розуміння зображень» [6].

У теорії розпізнавання існує багато алгоритмів знаходження ознак об'єкту (зображення), інваріантних до таких геометричних перетворень, як масштабування, поворот, зсув, а також до зміни яскравості. Ще більше алгоритмів існує для використання цих ознак для віднесення об'єкту до одного з класів. У дипломній роботі будуть розглянуті деякі з цих підходів до розпізнавання. Ці методи основані на голосуванні як окремо взятої ознаки об'єкту, так і всього об'єкту в цілому за певний клас із заданої бази відеоінформації. Створені програмні моделі для розрахунку ймовірностей

віднесення об'єкта або його ознак за кожен з класів, проведено для конкретних баз відеоданих, представлених списком характерних ознак (ХО).

Серед структурних підходів до класифікації, пов'язаних із зіставленням описів об'єктів у вигляді кінцевих множин ХО, з точки зору критерію повноти або точності структурного відповідності найбільш ефективними можна вважати методи розмітки та задоволення обмежень [1].



# 1 МЕТОДИ СТРУКТУРНОГО РОЗПІЗНАВАННЯ ЗОБРАЖЕНЬ

## 1.1 Використання методу SURF для формування характерних ознак

Метод SURF позитивно зарекомендував себе в задачах пошуку об'єктів на зображеннях, 3D реконструкції, при порівнянні зображень. Цей метод можна застосовувати надалі в інформаційних технологіях обробки інформації при вирішенні прикладних задач комп'ютерного зору, пов'язаних із розпізнаванням документів, грошових знаків, підписів, людських осіб, різних архітектурних будівель або інших специфічних візуальних об'єктів. Основними функціями системи буде розпізнавання зображення на основі методу SURF. З визначеної тематики на підставі наявних зразків програма буде знаходити на зображенні характерні точки і порівнювати їх із задалегідь знайденими точками зображень – зразками. Після проведеного аналізу користувачеві буде повідомлено про знайдені подібності, або їх відсутності.

SURF вирішує два завдання - пошук особливих точок зображення і створення їх дескрипторів, інваріантних до масштабу і обертанню. Це означає, що опис ключової точки буде однаково, навіть якщо зразок змінить розмір і буде повернутий. Крім того, сам пошук ключових точок теж повинен володіти інваріантністю. Так, що б повернений об'єкт сцени мав той же набір ключових точок, що і зразок[5].

На рисунку 1.1 - особливі точки зображення монстра, знайдені за допомогою матриці Гессе. Діаметр кола показує масштаб особливої точки. Зелена лінія - напрям градієнта яскравості.

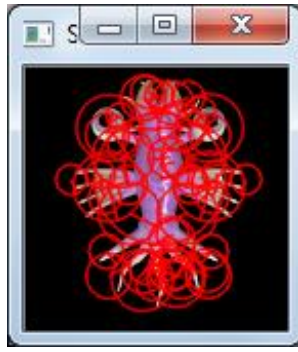


Рисунок 1.1 – Приклад формування характерних ознак

Гессіан інваріантний щодо обертання. Але не інваріантний масштабу. Тому SURF використовує різномасштабні фільтри для знаходження гессіанів.

Для кожної ключової точки обчислюється напрям максимальної зміни яскравості (градієнт) і масштаб, взятий з масштабного коефіцієнта матриці Гессе. Градієнт в точці обчислюється за допомогою фільтрів Хаара [11].

Після знаходження ключових точок SURF формує їх дескриптори. Дескриптор являє собою набір з 64 (або 128) чисел для кожної ключової точки. Ці числа відображають флуктуації градієнта навколо ключової точки (що розуміється під флуктуацією – розглянемо нижче). Оскільки ключова точка являє собою максимум гессіан, то це гарантує, що в околиці точки повинні бути ділянки з різними градієнтами. Таким чином, забезпечується відмінність дескрипторів для різних ключових точок.

Флуктуації градієнта околиці ключової точки вважаються відносно напрямлення градієнта навколо точки в цілому (по всій околиці ключової точки). Таким чином, досягається інваріантність дескриптора щодо обертання. Розмір же області, на якій вважається дескриптор, визначається масштабом матриці Гессе, що забезпечує інваріантність щодо масштабу.

Флуктуації градієнта також обчислюються за допомогою фільтра Хаара. Для ефективного обчислення фільтрів Гессе і Хаара використовується інтегральне представлення зображень. Інтегральне подання є матрицею, розмірність якої збігається з розмірністю вихідного зображення, а елементи обчислюються за формулою:

$$I(x, y) = \sum_{i=0, j=0}^{i \leq x, j \leq y} I(i, j), \quad (1.1)$$

де  $I(i, j)$  - яскравість пікселів вихідного зображення.

Маючи інтегральну матрицю, можна дуже швидко обчислити суму яскравостей пікселів для довільних прямокутних областей зображення (рис. 1.2), за формулою:

$$S = A + C - B - D, \quad (1.2)$$

де  $S$  – шуканий нами прямокутник.

Виявлення особливих точок в SURF засновано на обчисленні детермінанта матриці Гессе (гессіан) [12].

Матриця Гессе для двовимірної функції та її детермінант визначається за формулами:

$$H(f(x, y)) = \begin{bmatrix} \frac{\partial^2 f}{\partial x^2} & \frac{\partial^2 f}{\partial x \partial y} \\ \frac{\partial^2 f}{\partial x \partial y} & \frac{\partial^2 f}{\partial y^2} \end{bmatrix}, \quad (1.3)$$

$$\det(H) = \frac{\partial^2 f}{\partial x^2} \frac{\partial^2 f}{\partial y^2} - \left( \frac{\partial^2 f}{\partial x \partial y} \right)^2. \quad (1.4)$$

Обчислення матриці Гессе зводиться до знаходження лапласіану гауссіан, а її елементи обчислюються як згортка (сума добутків) пікселів зображення на фільтри, зображені на рисунку 1.2. На ньому зображені дискретизовані фільтри для знаходження чотирьох елементів матриці Гессе (четвертий - збігається з третім, оскільки матриця Гессе симетрична). Фільтри мають просторовий масштаб 9x9 пікселів [15]. Темні ділянки відповідають негативним значенням фільтра, світлі – позитивним.

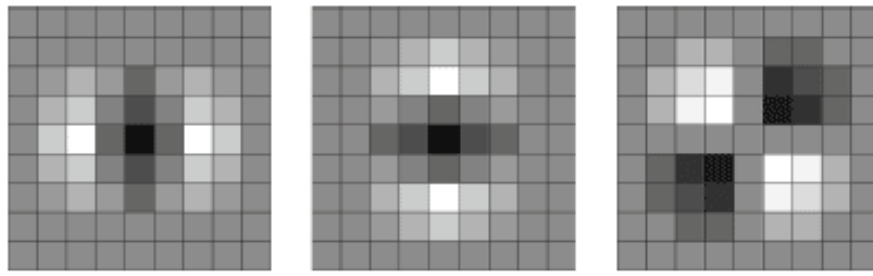


Рисунок 1.2 – Дискретизовані фільтри для знаходження чотирьох елементів матриці Гессе

Однак SURF не використовує лапласіан гауссіану в тому вигляді, який зображений на малюнку. По-перше, дискретизований лапласіан гауссіан має досить великий розкид значення детермінанта, при обертанні зразка. Особливо детермінант «просідає» в районі повороту на 45 градусів. А по-друге, і це головне, фільтр для лапласіану гауссіан має безперервний характер. Майже всі пікселі фільтра мають різні величини яскравості [9]. А це не дозволяє ефективно використовувати такий потужний механізм розрахунку, як інтегральну матрицю зображення.

Тому SURF використовує бінарізовану апроксимацію лапласіану гауссіан (автори назвали його Fast-Hessian).

На рисунку 1.3 зображені фільтри, які використовуються для знаходження матриці Гессе в SURF. Білі області відповідають значенням плюс 1, чорні мінус 2 (на третьому фільтрі мінус 1), сірі - нульові. Просторовий масштаб – 9x9 пікселів.

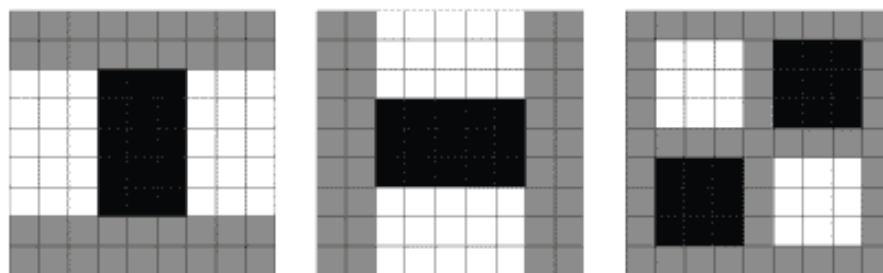


Рисунок 1.3 – Фільтри для знаходження матриці Гессе

Цей фільтр більш стійкий до обертання, і його можна ефективно обчислити за допомогою інтегральної матриці.

Таким чином, в SURF, гессіан обчислюється за формулою:

$$\det(H_{\text{approx}}) = D_{xx}D_{yy} - (0.9D_{xy})^2, \quad (1.5)$$

де  $D_{xx}$ ,  $D_{yy}$ ,  $D_{xy}$  – згортки по фільтрам, зображеним на рисунку 1.6. Коефіцієнт 0.9 має теоретичне обґрунтування, і коригує наближений характер обчислень.

Для знаходження особливих точок, SURF пробігається по пікселям зображення і шукає максимум гессіан. У методі задається граничне значення гессіан. Якщо обчислене значення для пікселя вище порога - піксель розглядається як кандидат на ключову точку[8].

З міркувань симетрії і дискретизації, розмір фільтра Fast-Hessian не може приймати довільні значення. Допустимі розміри цього фільтра такі (починаючи з мінімального): 9, 15, 21, 27 і так далі, з кроком 6. Однак, на практиці, поступово збільшувати розмір фільтра на 6 - не вигідно, тому що для великих масштабів крок 6 виявляється занадто дрібним, а фільтри - надлишковими. Тому (і з деяких інших причин), SURF розбиває всі множини масштабів на так звані октави. Кожна октава покриває певний інтервал масштабів, і має свій характерний розмір фільтра[9].

При цьому якби на октаву припадав тільки один фільтр, це було б занадто грубим наближенням. Крім того, ми б не могли знайти локальний максимум гессіан, серед різних масштабів, у різних октавах. Адже одна і та ж точка може мати кілька локальних максимумів гессіан, в різних масштабах. Це добре видно на рисунку (рис. 1.4):

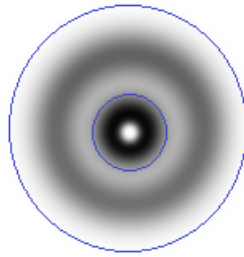


Рисунок 1.4 – Дві ключові точки різного масштабу в одній точці зображення

Якщо ми будемо шукати максимум серед усіх гесіанов, за всіма масштабами, то ми б знайшли тільки один з максимумів, в той час як їх може бути кілька. Один - в одному масштабі, інший - в іншому.

Для знаходження локального максимуму гесіан, використовується так званий метод сусідніх точок 3x3x3 (рис. 1.5).

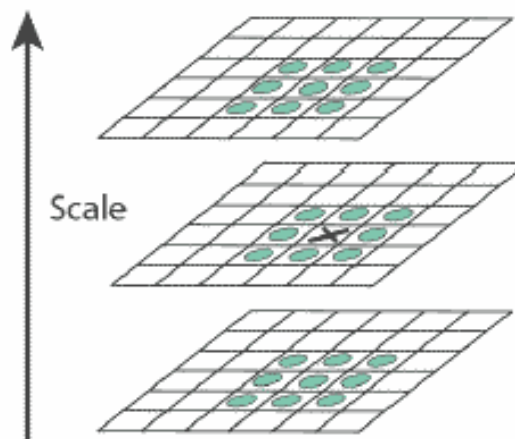


Рисунок 1.5– Зображення локального максимуму та його сусідів

Піксель, позначений хрестиком вважається локальним максимумом, якщо його гесіан більше ніж у будь-якого його сусіда в його масштабі, а також більше будь-якого із сусідів масштабом менше і масштабом більше (всього 26 сусідів).

Виходячи з такого визначення локального максимуму, зрозуміло, що октава повинна містити не менше трьох фільтрів, інакше ми не зможемо

визначити факт знаходження локального максимуму гессіан всередині октави.

Відзначимо ще такий момент. Фільтри октави вважаються не для всіх пікселів поспіль. Перша октава вважається для кожного другого пікселя зображення. Друга - для кожного четвертого, третя - для кожного восьмого і так далі. Сенс зрозумілий - дві точки з відстанню 2 не можуть містити більше одного максимуму масштабу 2, 3 або більш високих масштабів. Тому немає сенсу перебирати всі точки зображення, для знаходження максимуму масштабу 3, наприклад.

Подвоєння кроку пікселів для октав дозволяє економити при розрахунку фільтрів. Як ви напевно вже помітили, розміри фільтрів в октавах повторюються. Так, наприклад, фільтр розміром 27 присутній в трьох октавах. Так от, при обчисленнях, цей фільтр буде вважатися тільки для першої октави. Друга і третя - просто використовують розрахунки першої октави. А подвоєння кроку пікселів гарантує, що точки в яких потрібно вважати гессіан, вже були прораховані попередньої октавою.

Тому, незважаючи на те, що октава містить чотири фільтра, насправді кожна октава (крім першої) вважає тільки два характерних для неї розміру, два інших - завжди можна взяти з попередніх октав. Перша ж октава змушена вважати всі чотири своїх фільтра.

Отже, після знаходження максимального гессіан методом сусідніх точок  $3 \times 3 \times 3$ , ми знайшли піксель в якому цей максимум досягається. Однак, оскільки, октава перебирає не всі точки зображення, то істинний максимум може не збігатися з знайденим пікселом, а лежати десь поруч, в сусідніх пікселях.

Для знаходження точки істинного максимуму, використовується інтерполювання знайдених гессіанов куба  $3 \times 3 \times 3$  квадратичною функцією. Далі, обчислюється похідна (методом кінцевих різниць сусідніх точок). Якщо вона близька до нуля - ми в точці істинного максимуму. Якщо похідна велика - зрушуємося в бік її зменшення, і повторюємо ітерацію, до тих пір поки

похідна не стане менше заданого порогу. Якщо в процесі ітерацій ми відходимо від початкової точки занадто далеко, то це вважається помилковим максимумом, і точка більше не вважається особливою[5].

Для інваріантності обчислення дескрипторів особливої точки, які будуть розглянуті нижче, потрібно визначити переважаючу орієнтацію перепадів яскравості в особливій точці. Це поняття близьке до поняття градієнта, але SURF використовує трохи інший алгоритм знаходження вектора орієнтації.

Спочатку, обчислюються точкові градієнти в пікселях, сусідніх з особливою точкою. Для розгляду беруться пікселі в окружності радіуса  $6s$  навколо особливої точки. Де  $s$  - масштаб особливої точки. Для першої октави беруться точки з околиці радіусом 12.

Для обчислення градієнта, використовується фільтр Хаара. Розмір фільтра береться рівним  $4s$ , де  $s$  - масштаб особливої точки. Вид фільтрів Хаара, чорні області мають значення  $-1$ , білі  $+1$ , показаний на зображенні 1.8:



Рисунок 1.6 - Фільтри Хаара

Фільтри Хаара дають точкове значення перепаду яскравості по осі  $X$  і  $Y$  відповідно. Оскільки фільтри Хаара мають прямокутну форму, їх значення легко вважаються за допомогою інтегральної матриці. Для розрахунку одного фільтра довільного розміру потрібно всього 6 операцій.

Значення вейвлета Хаара  $dX$  і  $dY$  для кожної точки множаться на вагу і запам'ятовуються в масиві. Вага визначається як значення гауссіана з центром в особливій точці і сигмою рівній  $2s$ . Зважування на гауссіане необхідно для відсікання випадкових перешкод на далеких від особливої



точки відстанях.

Далі, всі знайдені значення  $dX$  і  $dY$ , умовно наносяться у вигляді точок на площину, як показано на рисунку 1.7:

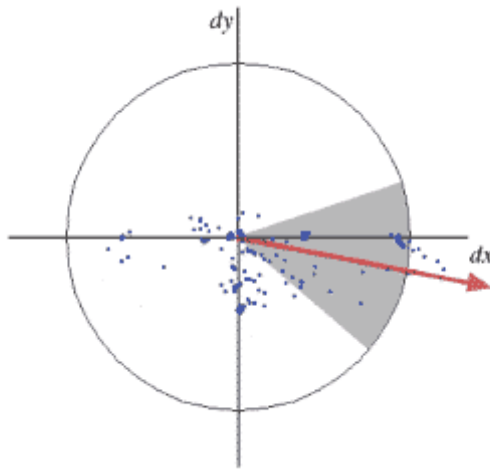


Рисунок 1.7- Знайдені градієнти у вигляді точок в просторі  $dX$   $dY$

Далі, береться кутове вікно (показано сірим на малюнку) розміром  $\pi / 3$ , і обертається навколо центру координат. Вибирається таке положення вікна, при якому довжина сумарного вектора для потрапили у вікно точок - максимальна. Обчислений таким чином вектор нормується і приймається як пріоритетний напрямок в області особливої точки.

Не завжди потрібно інваріантність дескрипторів щодо обертання. Метод SURF має модифікацію, в якій орієнтація особливих точок не розраховується. Така модифікація дозволяє надійно ідентифікувати точки, повернені не більше ніж на  $\pm 15$  градусів.

Дескриптор являють собою масив з 64 (в розширеній версії 128) чисел, що дозволяють ідентифікувати особливу точку. Дескриптори однієї і тієї ж особливої точки на зразку і на сцені повинні приблизно збігатися. Метод розрахунку дескриптора такий, що він не залежить від обертання і масштабу[2].

Для обчислення дескриптора, навколо особливої точки формується прямокутна область (рис. 1.8), що має розмір  $20s$ , де  $s$  - масштаб в якому була

знайдена особлива точка. Для першої октави, область має розмір 40x40 пікселів. Квадрат орієнтується уздовж пріоритетного напрямку, обчисленого для особливої точки.

Дескриптор вважається як опис градієнта для 16 квадрантів навколо особливої точки.

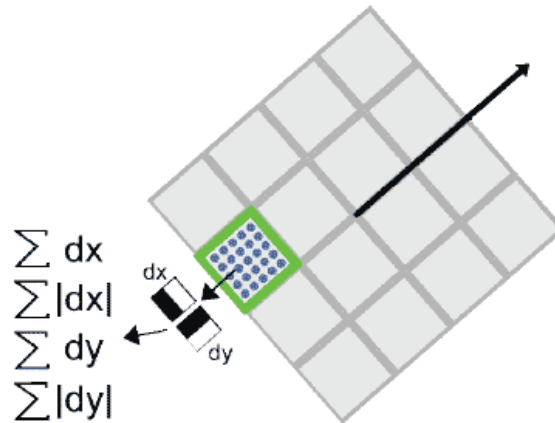


Рисунок 1.8 – Прямокутна область, навколо особливої точки

Далі, квадрат розбивається на 16 дрібніших квадрантів, як показано на малюнку. У кожному квадраті береться регулярна сітка 5x5 і для точки сітки шукається градієнт, за допомогою фільтра Хаара. Розмір фільтра Хаара береться рівним  $2s$ , і для першої октави становить 4x4.

Слід зазначити, що при розрахунку фільтра Хаара, зображення не повертається, фільтр вважається в звичайних координатах зображення. А ось отримані координати градієнта ( $dX$ ,  $dY$ ) повертаються на кут, відповідний орієнтації квадрата.

Для обчислення дескриптора особливої точки, потрібно обчислити 25 фільтрів Хаара, в кожному з 16 квадрантів. Разом, 400 фільтрів Хаара. Враховуючи, що на фільтр потрібно 6 операцій, виходить, що дескриптор обійдеться мінімум в 2400 операцій.

Після знаходження 25 точкових градієнта квадранта, обчислюються чотири величини, які власне і є компонентами дескриптора:

$$\Sigma dX, \Sigma |dX|, \Sigma dY, \Sigma |dY|. \quad (1.6)$$

Дві з них є просто сумарний градієнт по квадранту, а дві інших - сума модулів точкових градієнтів[3].

## 1.2 Моделі структурного розпізнавання

Узагальнимо модель заходів подібності для зіставлення множин ХО при розпізнаванні шляхом голосування. В задачах розпізнавання об'єктів, описуваних наборами ХО, вихідна інформація представляється у вигляді кінцевої множини дескрипторів. По суті, такий структурний опис є не вектором, а множиною, тому порядок проходження елементів залежить від процесу обробки. Тому оцінка подібності (близькості) повинна розглядатися як міра на множинах, а завдання віднесення об'єкта до класу вирішується шляхом оптимізації близькості множин ХО об'єкта та еталона [10].

Найбільш близькі до задоволення необхідних властивостей заходи, пов'язані з множинними операціями перетину і симетричної різниці, які можуть бути обчислені шляхом підрахунку відповідностей схожих елементів. Підрахунок еквівалентних ХО апроксимує оцінку подібності об'єктів.

Вдало побудована міра повинна не тільки знизити вплив похибок через незначних відхилень при дії адитивних перешкод, але і шляхом відкидання помилкових відповідностей дескрипторів забезпечити надійне розпізнавання об'єктів при істотних змінах через дії просторових перешкод [6].

При вирішенні проблем аналізу двовимірних сигналів і оцінки біометричної інформації розрізняють три основних типи заходів подоби:

- асоціативні, виражають відносини числа співпадаючих ознак до загального їх числа (коефіцієнти зв'язку, парні функції);
- кореляційні («косинусні» заходи);

– відстані в метричному просторі.

Вибір заходи подібності залежить в першу чергу від мети застосування. Всі заходи, по суті, пов'язані між собою, тому зрештою оцінюють ймовірність події ідентичності порівнюваних об'єктів в сенсі володіння загальними ознаками або атрибутами. Заходи подібності можна розділити на метрики і міри схожості. Міра подібності (позначимо) повинна бути позитивною, симетричною, максимальною при порівнянні об'єкта з самим собою [13]. Формально ці властивості виглядають так:

$$r(i, j) \geq 0, \text{ б) } r(i, j) = r(j, i), \text{ в) } r(i, i) = \max_j r(i, j).$$

Враховуючи складність завдань структурного розпізнавання, пов'язану з близькістю дескрипторів для різних об'єктів, основна увага при побудові заходів зосередимо на тому, щоб на першому етапі шляхом логічної обробки оцінити «збіг» окремих ХО, а потім підрахунком прецедентів (голосів) прийняти остаточне рішення. При цьому на рівні ХО перевагу віддамо використанню метрик.

На рівні зіставлення множин ХО застосовні всі обговорювані типи заходів. У загальному вигляді модель заходи подібності можна представити як функцію від симетричної матриці значень метрики для пар елементів порівнюваних множин [4].

$$r_{\Lambda}(\Lambda^1, \Lambda^2) = \Upsilon[\{\rho(\lambda_i^1, \lambda_k^2)\}]$$

Метрики відносяться до основних способів формування альтернатив класів у задачах розпізнавання, де оцінки приналежності обчислюються однозначним чином. Крім того, застосування метрик при пошуках в об'ємних базах відео даних дозволяє істотно скоротити обчислення, спираючись на внутрішню структуру бази [13]. Таким чином, опис у вигляді множини ХО

може розглядатися як елемент або метричного простору, або простору з мірою відмінності.

На рівні зіставлення множин ХО застосовні всі обговорювані типи заходів. У загальному вигляді модель заходи подібності можна представити як функцію від симетричної матриці  $\{\rho(\lambda_i^1, \lambda_k^2)\}$  значень метрики для пар елементів  $\lambda_i^1, \lambda_k^2$  порівнюваних множин  $\Lambda^1, \Lambda^2$  [4].

$$r_{\Lambda}(\Lambda^1, \Lambda^2) = \Upsilon[\{\rho(\lambda_i^1, \lambda_k^2)\}]$$

Прості в обчислювальному плані асоціативні міри зводяться до підрахунку числа «потрібних» елементів, що дозволяє оцінити ступінь неповноти відповідності. Крім того, застосування асоціативних мір поширюється на випадок аналізу комбінацій ознак.

Метрика - це математична модель подібності об'єктів і її вибір у багатьох випадках не можна здійснити однозначно. Останнім часом найчастіше метрику вибирають на основі аналізу навчальної вибірки. Лінійним простором називають сукупність елементів, для яких встановлені операції додавання і множення на число з виконанням необхідних властивостей. Відстань  $\|\lambda\| = \rho(\lambda, 0)$  від елемента  $\lambda$  до нуля називають нормою або довжиною елемента [17].

Нехай  $\lambda, \lambda^0$  – вектори ХО. Найбільше застосування при зіставленні ХО і фрагментів отримали наступні метрики:

– метрика «ізолюваних точок» [14] заснована на «співпадінні» елементів  $\lambda, \lambda^0$ ;

$$\rho(\lambda, \lambda^0) = \begin{cases} 0, & \lambda = \lambda^0, \\ 1, & \lambda \neq \lambda^0, \end{cases} \quad (1.7)$$

– відстань Евкліда

$$\rho(\lambda, \lambda^0) = [\sum_{i=1}^n (\lambda_i - \lambda_i^0)^2]^{1/2}; \quad (1.8)$$

– манхеттенська відстань ( $R_1^n$ )

$$\rho(\lambda, \lambda^0) = \sum_{i=1}^n |\lambda_i - \lambda_i^0|; \quad (1.9)$$

– відстань Махаланобіса

$$\rho(\lambda, \lambda^0) = [(\lambda - \lambda^0) \Sigma^{-1} (\lambda - \lambda^0)]^{1/2}, \quad (1.10)$$

де  $\Sigma$  – матриця коваріації випадкових векторів  $\lambda, \lambda^0$ ;

– для випадку статистично незалежних ознак, якщо  $\Sigma = \text{diag}(\sigma_1^2, \dots, \sigma_n^2)$ ,

(1.12) набуває вигляду

$$\rho(\lambda, \lambda^0) = \sum_{i=1}^n \frac{(\lambda_i - \lambda_i^0)^2}{\sigma_i^2}; \quad (1.11)$$

– метрика  $\chi^2$

$$\rho(\lambda, \lambda^0) = \frac{1}{2} \sum_{i=1}^n \frac{(\lambda_i - \lambda_i^0)^2}{(\lambda_i + \lambda_i^0)}; \quad (1.12)$$

– метрика Хеммінга, підраховує кількість розбіжностей

$$\rho(\lambda, \lambda^0) = \sum_{i=1}^n L(\lambda_i, \lambda_i^0), \text{ где } L(\lambda_i, \lambda_i^0) = \begin{cases} 0, & \lambda_i = \lambda_i^0, \\ 1, & \lambda_i \neq \lambda_i^0. \end{cases} \quad (1.13)$$

У більшості випадків вектори ознак  $\lambda, \lambda^0$  з метою спрощення аналізу передбачаються нормованими на відрізку  $[0,1]$ . Кожна з розглянутих метрик має свої переваги і недоліки в порівнянні з іншими. Наприклад, метрика (1.9) простіше в обчислювальному плані, ніж (1.8), однак, меншою мірою відображає відмінності ознак. Метрики (1.10), (1.11) враховують статистичні властивості ознак і є узагальненням метрики (1.8).

Для оцінки близькості дескрипторів ХО використовуємо поняття  $\delta$ -околиці елемента  $\lambda_0$

$$O_\delta(\lambda_0) = \{\lambda: \rho(\lambda, \lambda_0) \leq \delta\} \quad (1.14)$$

як множини елементів, відстань від яких до  $\lambda_0$  не перевищує деякого порога  $\delta$  (елемент належить «шару» радіуса  $\delta$  з центром в  $\lambda_0$ ). Правило (1.14) визначає для кожного з множин  $\Lambda^1, \Lambda^2$  сімейство куль і встановлює бінарне відношення нерозрізненості дескрипторів [16]. Належність кулі означає еквівалентність  $\lambda, \lambda_0$ . Параметр  $\delta$  визначає ступінь точності при встановленні нерозрізненості, зі збільшенням  $\delta$  росте розмір конгломератів еквівалентних точок. Зауважимо, що  $\delta \in [0,1]$  в силу  $\rho \in [0,1]$ . Окіл  $O_\delta(\lambda)$  задаються еквівалентні підмножини як між  $\Lambda^1, \Lambda^2$ , так і всередині кожного з них. Належність околиці перевіряється предикатом, який дорівнює 0 або 1 залежно від істинності нерівності. Фактично задає відповідність  $\lambda$  и  $\lambda_0$ , здійснюючи елімінування зв'язків [18].

$$L[\rho(\lambda, \lambda_0), \delta] = \begin{cases} 0, & \rho(\lambda, \lambda_0) \leq \delta \\ 1, & \text{otherwise} \end{cases}, \quad (1.15)$$

Предикат у вигляді заперечення  $L[.]$  називають в теорії інтелекту

предикатом «впізнання» [16]. Інший спосіб аналізу - порогова обробка виду

$$L^\delta[\rho(\lambda, \lambda_0), \delta] = \begin{cases} \rho(\lambda, \lambda_0), & \rho(\lambda, \lambda_0) \leq \delta \\ 1, & \text{otherwise} \end{cases}, \quad (1.16)$$

де  $\rho(\lambda, \lambda_0)$  відображає величину близькості ХО в межах порога. Області формування значень для  $L$  та  $L^\delta$  співпадають, але в  $L^\delta$  зберігаються значення  $\rho(\lambda, \lambda_0)$ , які можна використовувати в якості ваг. Розглядаючи елемент як куля, метрику ізольованих точок (1.7) можна перетворити до вигляду

$$\rho(\lambda_1, \lambda_2) = \begin{cases} 0, & \lambda_2 \in O_\delta(\lambda_1) \vee \lambda_1 \in O_\delta(\lambda_2), \\ 1, & \text{otherwise,} \end{cases} \quad (1.17)$$

тобто елементи  $\lambda_1, \lambda_2$  вважаються однаковими у випадку, якщо один з них потрапляє в околицю другого. Іншою умовою може бути непорожнє перетин околиць  $O_\delta(\lambda_1), O_\delta(\lambda_2)$ , тобто  $O_\delta(\lambda_1) \cap O_\delta(\lambda_2) \neq \emptyset$ . Співвідношення типу (1.17) часто застосовують як практичну апроксимацію для класичної метрики. Вирази (1.15), (1.17) при використанні в метриках для множин підраховують число незбіжних елементів.

Елементи вважаються еквівалентними, якщо виконуються умови  $\rho(\lambda_1, \lambda_2) = 0$  або  $\rho(\lambda_1, \lambda_2) \leq \delta$ . Однак, у практиці реальних завдань комп'ютерного зору встановлення еквівалентності досить складно, тому при певному  $\delta$  значне число дескрипторів може виявитися «еквівалентними». Тому поріг  $\delta$  вибирають експериментально, спираючись на аналіз фіксованої бази даних [7].

Розглянемо особливості застосування асоціативних мір оцінки близькості множин  $\Lambda^1, \Lambda^2$ . Визначимо для множин  $\Lambda^1, \Lambda^2$  на основі  $O_\delta(\lambda)$



операції перетину  $C = \Lambda^1 \cap \Lambda^2$  і різниці  $A_1 = \Lambda^1 \setminus \Lambda^2$ ,  $A_2 = \Lambda^2 \setminus \Lambda^1$ . Позначимо потужності  $c = \mu(C)$ ,  $a = \mu(A_1)$ ,  $b = \mu(A_2)$ . Проаналізуємо низку мір на основі множин  $C, A_1, A_2$ :

- Жаккара  $K_1 = c/(a + b + c)$ ; Сьеренсена  $K_2 = 2c/(a + b)$ ;
- Дейка  $K_3 = 2c/(a + b + 2c)$ ;
- Кульчинського  $K_4 = (a + b)/(2ab)$ ;
- незгоди Танимото  $K_5 = (a + b)/(a + b + c)$ ;
- Соукала та Снита  $K_6 = c/(c + 2(a + b))$ ;
- Дейка  $K_7 = (c - \min(a, b))/(c + \min(a, b))$  [19].

Асоціативні міри володіють такими перевагами, як універсальність і швидкодія обчислень. Крім того, вони, як правило, безрозмірні і обмежені відрізком  $[0,1]$ . При цьому кожна із мір має свої особливості застосування. Наприклад, запобіжний  $K_7$  задана відрізком  $[-1,1]$ , а  $K_2$  не нормована, тому що перетин може перевищувати величину симетричної різниці. Значення окремих мір, наприклад  $K_2, K_4$ , не визначені при збігу множин, тому що при цьому  $a = b = 0$ , тому міри потрібно довизначити для цих випадків. Зауважимо, що заходи  $K_1, K_5$  можна виразити один через одного. Значення  $(a + b)$  є потужність симетричної різниці  $\Lambda^1, \Lambda^2$ , широко використовуваної в інтелектуальному аналізі даних, зокрема, пов'язаному з проблемами грануляції інформації [18]. Значення  $(a + b + 2c)$  відповідає сумі  $\mu(1) + \mu(2)$ . Одним з критеріїв вибору конкретної міри є відносна важливість подій збігу або розбіжності елементів. Наприклад, функція  $K_3$  надає вдвічі більшу вагу елементам, що збігаються, а функція  $K_6$  – що не збігаються [19].

### 1.3 Постанова задачі

Метою даної роботи є моделювання та обчислення статистичних ознак для структурного розпізнавання візуальних об'єктів. Для реалізації нам необхідно:

- 1) провести аналіз методу структурного розпізнавання SURF;
- 2) провести аналіз можливостей бібліотеки OpenCV;
- 3) розробити програму, яка буде моделювати та обчислювати статистичні ознаки, та порівнювати зображення;
- 4) провести аналіз та використати методи для порівняння статистичних ознак.

## 2 ЗАСТОСУВАННЯ СТАТИСТИЧНИХ ОЗНАК В СТРУКТУРНОМУ РОЗПІЗНАВАННІ ЗОБРАЖЕНЬ

### 2.1 Формування набору класифікаційних ознак

Растрові зображення земної поверхні являють собою сукупність об'єктів і фону, що мають різні яскравість, колір, текстурні, топологічні, геометричні і т.п. характеристики. Так як відомо, що основними дешифруючі ми ознаками є тон, розміри і форма об'єкта [20], в якості ознак для виявлення об'єктів обрані відповідно, колір, площа і параметри форми об'єктів. Виходячи з цього, запропонований набір класифікаційних ознак має вигляд (2.1):

$$(I, S_{ob}, D_{ob}, K_{ob}), \quad (2.1)$$

де  $I$  – кінцева множина значень інтенсивності кольору об'єкта;  $S_{ob}$  - площа об'єкта;  $D_{ob}$  - еквівалентний діаметр об'єкта;  $K_{ob}$  - коефіцієнт заповнення об'єкта.

Наведена в [21] модель кольорового цифрового зображення дозволяє представити кінцеву множину значень інтенсивності  $I$  пікселів у вигляді (2.2):

$$I = \{I_R, I_G, I_B\}, \quad (2.2)$$

де  $I_R$  – кінцева множина значень інтенсивності каналу  $R$  об'єкта;  $I_G$ , - кінцева множина значень інтенсивності каналу  $G$  об'єкта;  $I_B$  – кінцева множина значень інтенсивності каналу  $B$  об'єкта. Площа об'єкта цифрового зображення визначається у відповідності з виразом:

$$S_{ob} = \sum_i \sum_j S(i, j), \quad (2.3)$$

Площа об'єкта враховує як пікселі, що знаходяться всередині контуру, так і пікселі, складові контур. Еквівалентний діаметр об'єкта зображення дорівнює діаметру кола, в яку вписаний об'єкт(2.4):

$$D_{obl} = \text{sqrt}(4 \times S_o / \pi), \quad (2.4)$$

де  $S_o$  – площа кола, в яку вписаний об'єкт. Коефіцієнт заповнення дорівнює відношенню площі об'єкта до площі обмежує її прямокутника (2.5):

$$K_{obl} = S_{obl} / S_{pr}, \quad (2.5)$$

де  $S_{pr}$  – площа прямокутника, що містить об'єкт зображення. У процесі отримання зображення виникає ряд факторів, що впливають на його якість. До основних таких факторів належать метеорологічні умови отримання зображення і навігаційно-технічні умови отримання [20]. На значення інтенсивності кольору впливають погодні умови і освітлення при зйомці.

Для видалення помилкових об'єктів в роботі запропоновано використовувати ознаки форми об'єктів на зображенні. Для вибору ознак форми було проведено ряд експериментів, суть яких полягала в наступному. Було взято зображення з об'єктами різних класів. Для кожного об'єкта між класами і всередині класів були розраховані значення ознак засобами програмного пакету Matlab. Після чого були проведені розрахунки інформативних ознак форми відповідно с формулами (2.6) - (2.10).

Після розрахунку математичного очікування і дисперсії ознак визначаються ознаки з мінімальним коливанням значення усередині класу і найбільшим розкидом значень від класу до класу.

Інформативність ознак визначається виразом (2.6) [22]:

$$F = D_d / D_m, \quad (2.6)$$

де,

$$D_m = \frac{\sum_{i=1}^k (M_m - M_i)^2}{k-1}, \quad (2.7)$$

де,

$$M_m = \frac{\sum_{i=1}^k M_i}{k}, \quad (2.8)$$

де,

$$D_d = \frac{\sum_{i=1}^k (M_d - D_i)^2}{k-1}, \quad (2.9)$$

де,

$$M_d = \frac{\sum_{i=1}^k D_i}{k}, \quad (2.10)$$

Вираз (2.6) має наступний фізичний зміст. Чим більше для даної ознаки дисперсія величин  $M_i$ , тим ширше діапазон коливання ознаки при описі різних об'єктів, отже, тим вище цінність даної ознаки для їх розрізнення. Чим менше дисперсія величин  $D_i$ , тим менше коливання ознаки для одного і того ж об'єкта, тобто тим ближче будуть розташовуватися в просторі ознак елементи одного кластера. У граничному випадку величина  $F$  прагне до нуля. Така оцінка ознаки не залежить від значень інших ознак і від зміни розмірності простору ознак, тобто будь-яка ознака, що претендує на включення в підсумковий набір, може бути оцінений один раз і зрівняний з іншими ознаками без перерахунку їх оцінок.

Завдання класифікації зображень для об'ємних баз відеоданих може вимагати істотних витрат часу і використання значних обчислювальних потужностей. Для прискорення процедури класифікації шляхом скорочення структурного опису пропонується застосувати на попередньому етапі

процедуру фільтрації множин ХО еталонів на основі критерію унікальності (повторюваності) їх значень.

Одним із варіантів зниження розмірності простору ознак є видалення з опису схожих один з одним ХО в межах кожного окремого еталона [3] або в межах досліджуваної бази зображень. У результаті в описі залишаються найбільш унікальні (що не повторюються) ХО, або найменш унікальні ХО, на основі чого буде прийматися рішення про клас аналізованого зображення.

Схожість окремих ознак еталона та об'єкта визначалася на основі евклідової відстані між векторами їх дескрипторів

$$r(H_1, H_2) = \sqrt{\sum_{i=1}^{64} (h_{1i} - h_{2i})^2} . \quad (2.11)$$

Ознаки вважалися схожими, якщо відстань між ними була меншою від порогового значення, яке вибиралося як процентне відношення від максимально можливого значення відхилення між векторами SURF в евклідовій метриці. Дослідження проводилися для порогових значень  $\delta = 0,16 - 1\%$  ,  $0,24 - 1,5\%$  ,  $0,48 - 3\%$  ,  $0,64 - 4\%$  ,  $0,8 - 5\%$  ,  $1,6 - 10\%$ . Для кожного порогового значення для всіх еталонних зображень будувалася гістограма повторюваності, що показує, наскільки унікальною є кожна ХО зображення. Кількість стовпців гістограми відповідає кількості N ХО еталона. Висота L кожної шпальти гістограми для ХО N обчислювалася за формулою:

$$L_{\delta}(H) = \sum_{i=1}^N K_{\delta}[r(H, H_i)] , \quad (2.12)$$

$$K_{\delta}[r(H, H_i)] = \begin{cases} 1, & r(H, H_i) \leq \delta \\ 0, & r(H, H_i) > \delta \end{cases} . \quad (2.13)$$

При пороговому значенні  $\delta < 0,16$  для більшої частини ХО (більше 80 %) не визначалося жодної схожої ознаки, при  $\delta \geq 1,6$  для кожної ознаки всі інші ознаки стають схожими з нею, що в обох випадках унеможливило знаходження найменш повторюваних унікальних ознак [19]. Гістограми повторюваності з порогами 0,64 і 0,8, як показало моделювання, мають близький вигляд і можуть використовуватися для визначення множини унікальних ознак. Приклад зображення з виділеними ХО і гістограми повторюваності для порога 0,8 показані відповідно на рисунок 2.1.. Стовпці, відповідних десяти найбільш унікальним ХО, виділені світлим кольором.



Рисунок 2.1. Приклад еталонного зображення з виділеними ХО та його порівняння з іншим схожим зображенням

## 2.2 Результати моделювання

Існуючі структурні методи аналізу об'єктів на зображеннях реальних візуальних сцен дозволяють з високою ефективністю здійснити класифікацію об'єктів в умовах впливу просторових і адитивних перешкод. У процесі застосування даних методів здійснюється вирішення двох основних завдань. Для отримання структурного представлення об'єкта проводиться виділення структурних елементів у вигляді характерних ознак які відображають локально-просторові властивості об'єктів в окремих точках і формують

структурний опис. Друге завдання пов'язане із встановленням подібності структурних описів об'єкта і еталонів і пошуком оптимального значення міри подібності.

Найбільш поширеними з сучасних технологій формування та зіставлення структурних описів є SURF і SIFT. Їх застосування забезпечує високу ймовірність правильної класифікації в складних умовах наявності перешкод і помилкових об'єктів, однак вимагає істотних обчислювальних витрат на встановлення подібності [5].

Найбільш поширені методи розпізнавання образів засновані на геометричному і статистичному підходах. Широке застосування знайшов статистичний підхід, в основі якого лежить класифікація або кластеризація. У певному  $n$ -вимірному просторі ознак заданий вектор, необхідно визначити до якого класу він відноситься. Клас задається в тому ж просторі ознак групою компактно розташованих векторів, або заданих аналітично. В ідеальному випадку розкид значень векторів усередині класу значно менше, ніж відстані між класами, і клас можна окреслити опуклою формою. У реальному житті це далеко не так, класи можуть бути складної форми, мати значний розкид усередині себе і перетинатися з іншими класами. Тому завдання віднесення даного вектора до деякого класу значно ускладнюється.

Метою моєї програми, було моделювання та обчислення статистичних ознак для структурного розпізнавання візуальних об'єктів. Моделювання цих ознак було виконано за допомоги бібліотек OpenCV та методу SURF. В програмі ці ознаки зображенні у вигляді різнокольорових кілець. Приклади деяких порівнянь зображені на рисунках 2.2 та 2.3, також на рисунку 2.4 зображена частина обчислення ознак:





Рисунок 2.2 – Порівняння рисунка рисунку з самим собою

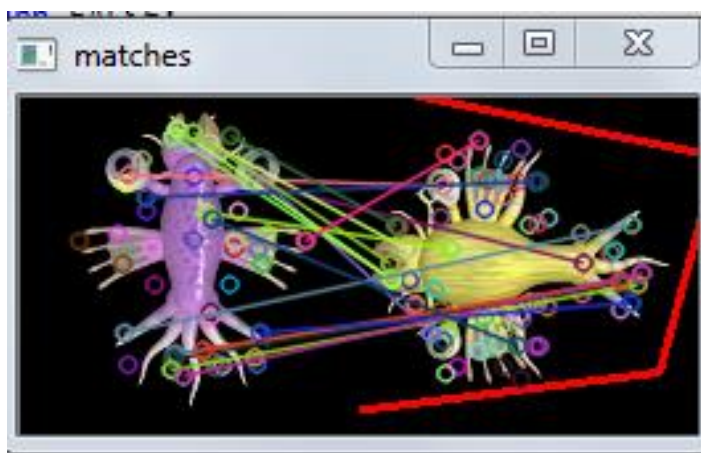


Рисунок 2.3 – Порівняння того ж малюнку з іншим

```

C:\Users\Home\documents\visual studio 2012\Projects\Test2\Debug\Test2.exe
Mat ozh [45]= 0.0698736
Mat ozh [46]= 0.00888508
Mat ozh [47]= 0.00305833
Mat ozh [48]= 0.0102187
Mat ozh [49]= 0.0105881
Mat ozh [50]= 0.0045903
Mat ozh [51]= 0.0105824
Mat ozh [52]= 0.00893123
Mat ozh [53]= 0.00372387
Mat ozh [54]= 0.0861912
Mat ozh [55]= 0.010181
Mat ozh [56]= 0.00623909
Mat ozh [57]= 0.0129124
Mat ozh [58]= 0.00600852
Mat ozh [59]= 0.00780898
Mat ozh [60]= 0.00609132
Mat ozh [61]= 0.0115884
Mat ozh [62]= 0.00603079
Mat ozh [63]= 0.0100995
Sum = 64
OBJECT FOUND!
Good matches found: 26
Algorithm duration: 356.031

```

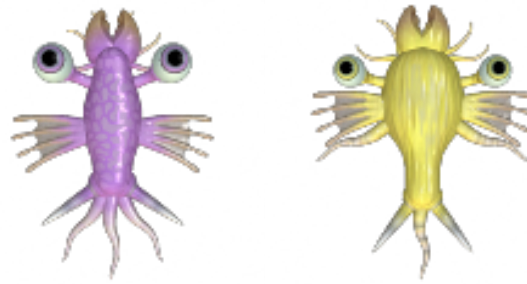
Рисунок 2.4 – Консольна частина програми

Як я писав раніше, для порівняння рисунків я використовував OpenCV. Ця бібліотека дуже популярна і має багато можливостей пов'язаних з комп'ютерним зором, обробкою зображень та числових алгоритмів загального напрямку. При написанні програми я використовував декілька дуже корисних функцій, які полегчують роботу з методом SURF. В даній бібліотеці є багато різних методів для пошуку ключових точок на рисунках. Я використовував SurfFeatureDetector – як і написано в назві, він оснований на методі SURF. Але на відміну від повного написання цього методу в програмному коді, використовуючи бібліотеку OpenCV – це займе лише дві строки програмного коду.

Після того, як були знайдені ключові точки, потрібно виділити з них дескриптори. Для цього я використав метод SurfDescriptorExtractor. У цьому методі, характерна точка дескриптора представлена у вигляді фіксованої розмірності вектора основного типу. Наступним кроком у моїй програмі було порівняння цих самих дескрипторів. В бібліотеці OpenCV є декілька методів для порівняння – це BFMatcher, FlannBasedMatcher та DescriptorMatcher. Серед них я обрав FlannBasedMatcher. В програмі я використав кожен метод, для того щоб обрати який краще, але через те що в мене не така велика кількість ключових точок, помітної різниці між ними не було.

Це основні та найцікавіші методи що я використовував у своїй програмі.

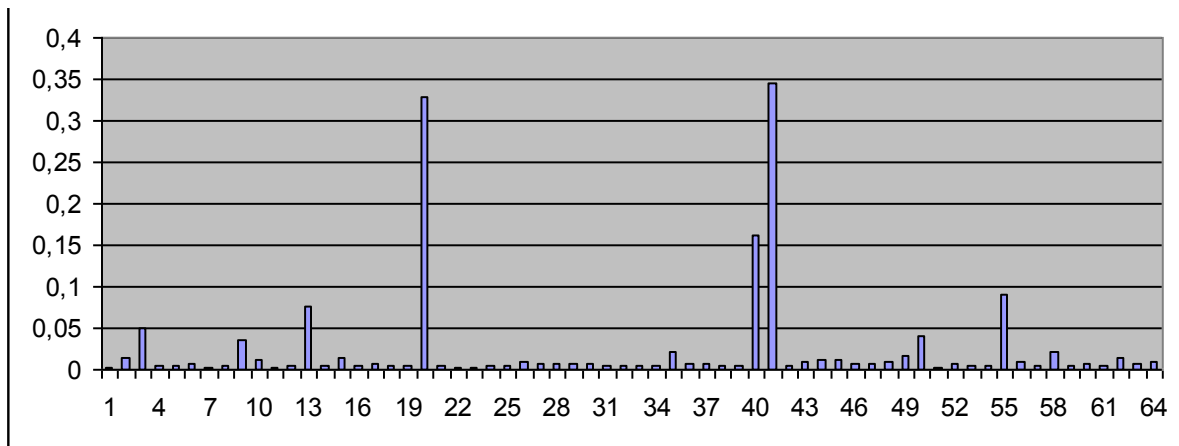
В своєму дипломі я розглядаю та порівнюю математичне очікування та дисперсію для двох еталонів. Еталони зображенні на рисунку 2.5. Спочатку будує гістораму векторів ХО(рис. 2.6).



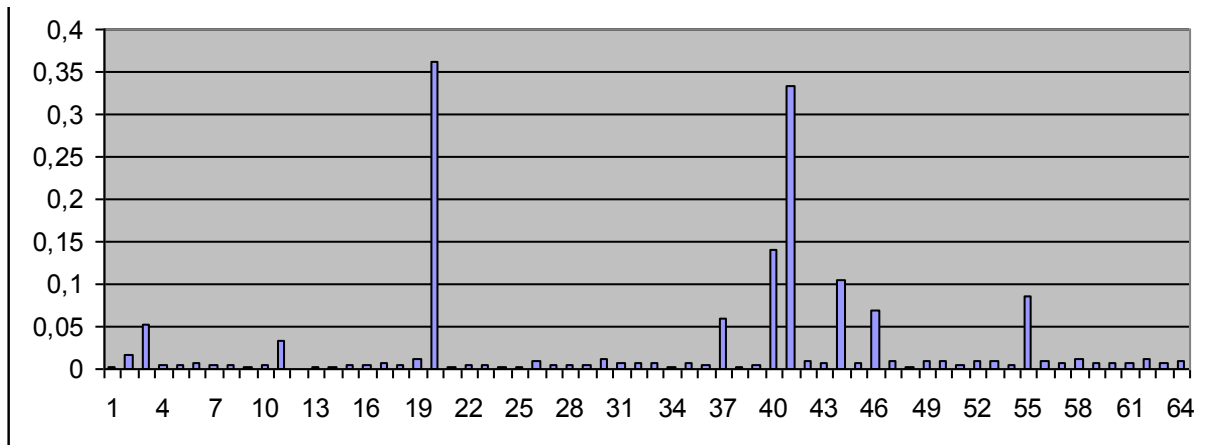
а

б

Рисунок 2.5 – Еталони, використані для порівняння



а



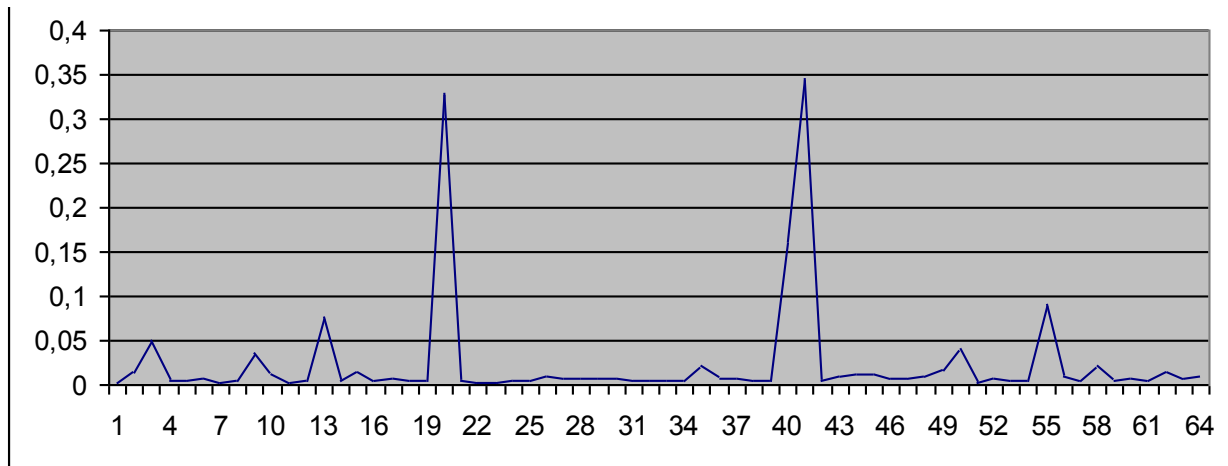
б

Рисунок 2.6 – Гістограма середнього значення векторів ХО:

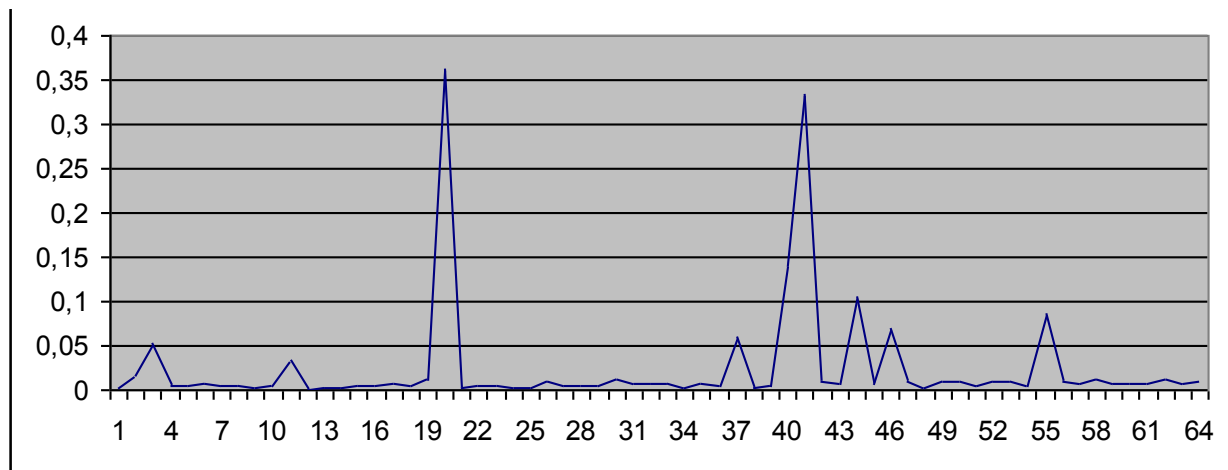
а – для рис. 2.5(а), б – для рис. 2.5(б)

Порівнюючи данні гістограми можна побачити, що вони дуже схожі. Порівнюючи всі елементи векторів, бачимо що вони дуже схожі, а в деяких

випадках – майже ідентичні. Але, дані гістограми не можуть в повному обсязі показати схожість середніх значень векторів ХО, через неможливість порівняння всіх значень лише використовуючи зорове порівняння. Тому, для більш точної оцінки, я побудував графіки по даним точкам(рис. 2.7) та порівняв значення кожного вектору.



а)



б)

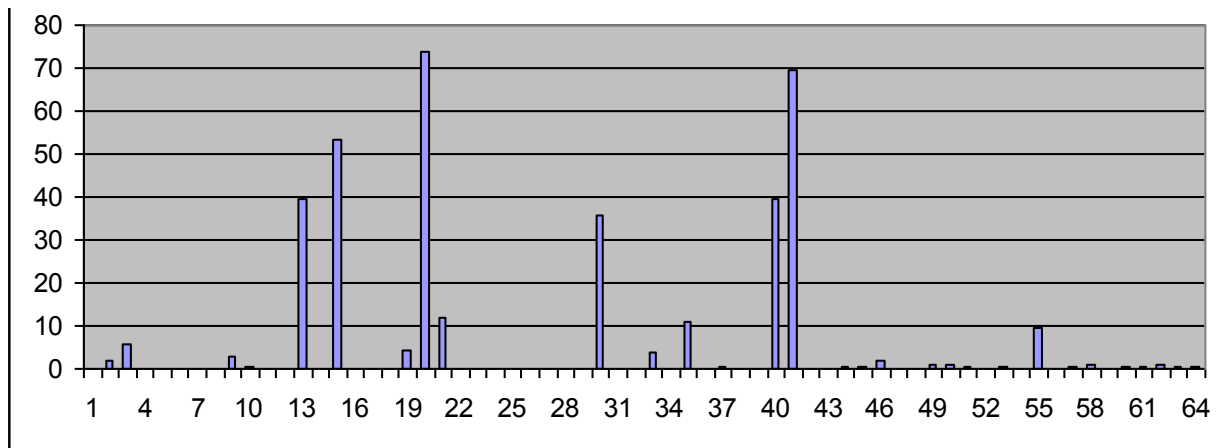
Рисунок 2.7 – Графік на основі гістограми середніх значень векторів ХО:

а) – для рис. 2.5(а), б) – для рис. 2.5(б)

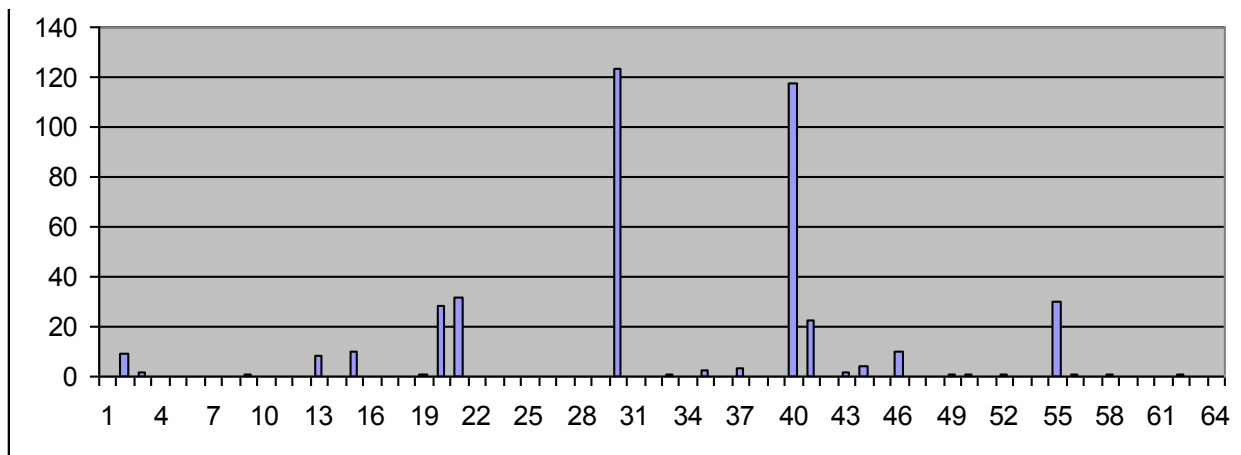
Порівнюючи графіки можна побачити, що вони дуже схожі. Хоч деякі моменти і мають різні форми, можна зробити висновок, що два еталони подібні і мають схожість, як і показано на рисунку 2.3. Так як, 100%

відповідність між 64 елементами векторів кожного зображення не можливе, буду вважати що точки співпадають з похибкою в 0,002. Таким чином отримую 36 точок які мають ~100% спів падіння. Дивлячись на рисунок 2.7 можна побачити що кількість відносно однакових точок більша.

Далі, в дипломі я порівнюю обидва рисунки за допомогою дисперсії. Для початку, як і в випадку з математичним очікуванням буду гистограми(рис. 2.8).



а)



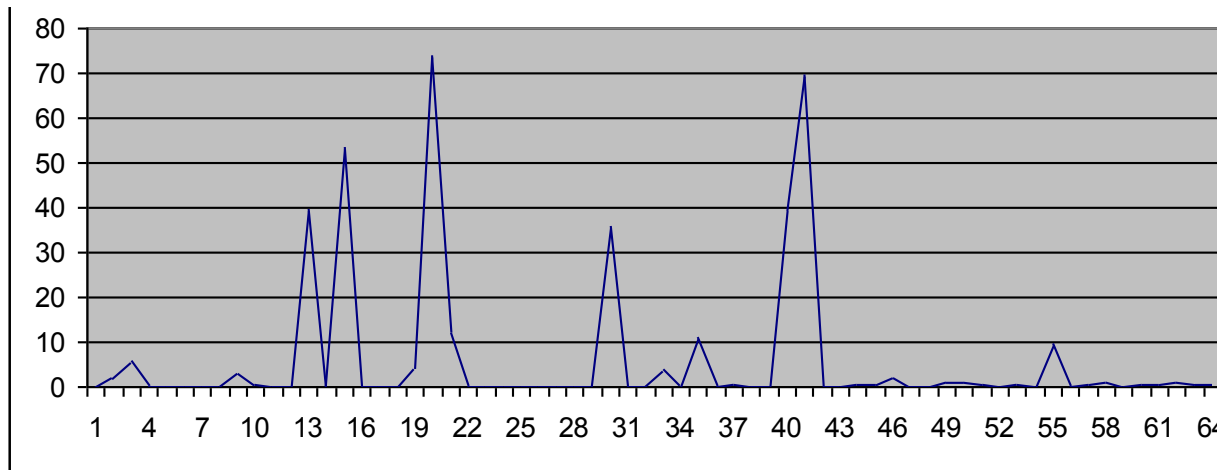
б)

Рисунок 2.8 – Гістограма дисперсій векторів ХО:

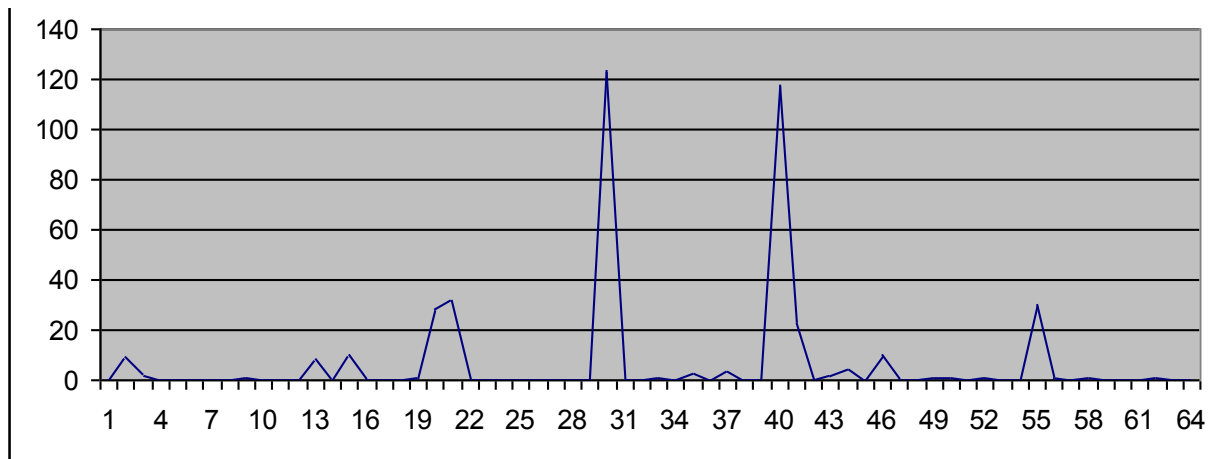
а) – для рис. 2.5(а), б) – для рис. 2.5(б)

На рисунку 2.8 можна побачити, що гістограми для рисунків 2.5 відрізняються. Дивлячись на результати середніх значень векторів ХО,

можна побачити чітку різницю, хоч математичне очікування і показує що малюнки дійсно дуже схожі, але по результатам дисперсії можна побачити чітку різницю між ними. Для більш детального та чіткого порівняння, будує графік(рисунок 2.9).



а)



б)

Рисунок 2.9 – Графік на основі гістограми дисперсій векторів  $X_0$ :

а) – для рис. 2.5(а), б) – для рис. 2.5(б)

Порівнюючи дані графіки можна чітко побачити різницю між ними. Розрахувавши дисперсію для кожного з еталонів можна впевнитися в тому, що не зважаючи на схожість самих еталонів та математичного очікування, дисперсія дуже чітко вказує на їх відмінність. Ознайомившись с цими даними можна зробити висновок, що за допомогою дисперсію векторів  $X_0$ ,

можна оцінювати на скільки схожі еталони.

Серед поширених не метричних характеристик для зіставлення фрагментів або їх ознак можна відзначити дисперсійні. Відомі підходи до обробки двовимірних сигналів, наприклад, фільтр Лі, засновані на характеристиках локального середнього і дисперсії. При відомих статистичних описах бази моделей ефективним може виявитися застосування мір, заснованих на значеннях параметрів класів структурних елементів. Якщо відомі статистичні характеристики (математичне очікування і дисперсія  $\sigma^2$ ) двох випадкових величин  $a_1, \sigma_1^2$  та  $a_2, \sigma_2^2$ , то ступінь близькості (відповідності) цих величин можна оцінити на основі статистики

$$r = \frac{[(\sigma_1^2 + \sigma_2^2)/2 + ((a_1 - a_2)/2)^2]^2}{\sigma_1^2 \sigma_2^2}. \quad (2.14)$$

В якості порівнюваних величин в (2.14) можна використовувати характеристики яскравості фрагментів, а також безпосередньо значення ХО. При збігу параметрів, тобто при  $a_1 = a_2$ ,  $\sigma_1^2 = \sigma_2^2$ , величина  $r = 1$ . З ростом відмінності між парами  $a_1, a_2$ ;  $\sigma_1^2, \sigma_2^2$  значення  $r$  збільшується до нескінченності. В умовах шуму величина набуває статистичний вигляд і залежить від законів розподілів порівнюваних величин.

Використовуючи формулу 2.14 порівнюю еталони та будує гістограму.

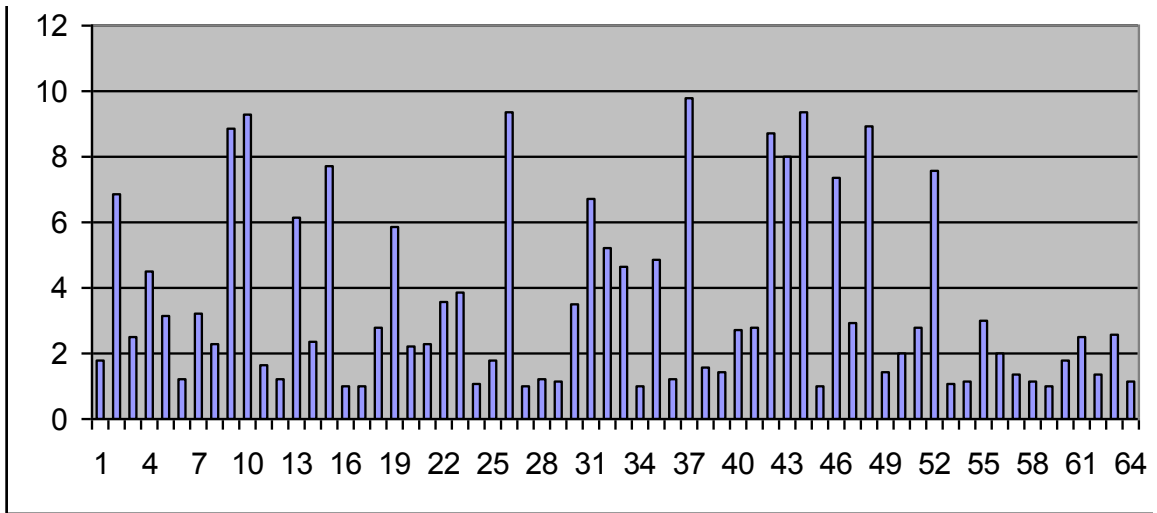


Рисунок 2.10 – Гістограма статистичних даних

Для більш детального та чіткого порівняння будує графік на основі статистичних даних(рис. 2.11).

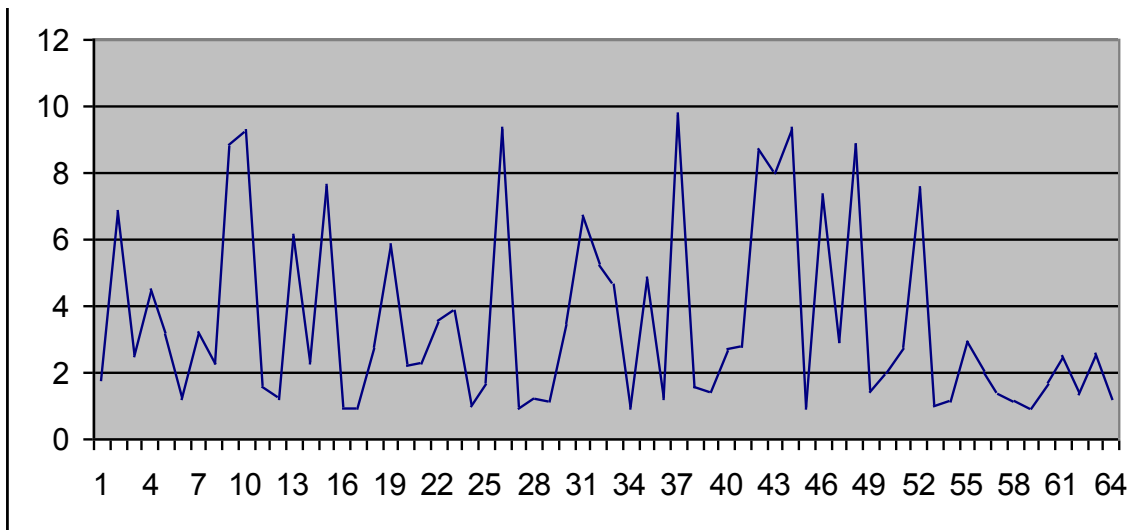


Рисунок 2.11 – Графік на основі статистичних даних

Отриманий графік підтверджує дані попередніх порівнянь. Виходячи с отриманих результатів, можна зробити висновок, що за допомогою дисперсії векторів характерних точок можна порівнювати ступінь подібності еталонів.



## 3 ОХОРОНА ПРАЦІ

### 3.1 Аналіз потенційно небезпечних і шкідливих виробничих факторів , що впливають на персонал

Персональні ЕОМ типу IBM PC AT має наступні характеристики:

- споживана потужність 350 Вт;
- робоча напруга 220 В;
- напруга джерел живлення +12 В, -12 В, 5 В;
- робоча частота 50 Гц.

Виходячи з приведених характеристик, очевидно, що для користувача існує небезпека поразки електричним струмом у разі недбалого поводження з комп'ютером і порушення правил експлуатації (невиконання огляду відкритих частин ПЕВМ, що знаходяться під напругою або знятих для ремонту вузлів і т. д.).

Джерелами підвищеної небезпеки можуть служити наступні елементи:

- розподільний щит;
- джерела живлення;

У відповідності з ДСН 3.3.6.042-99 [21] до легкої фізичної роботи відносяться всі види діяльності, вироблювані сидячи і не вимагаючи фізичної напруги. Робота користувача розробленого пакету програм відноситься до категорії Іа.

Згідно з ДСТУ Б А.3.2-13:2011 [22] приміщення для ПЕОМ по ступеню небезпеки поразки людини електричним струмом відноситься до приміщень без підвищеної небезпеки (немає струмопровідної половини, вогкості, підвищеної температури, можливості одночасного дотику до корпусів устаткування з “землею” і до струмонесучих частин).

У відповідності з ДСанПіН 3.3.2-007-98 [23] при обслуговуванні ПЕВМ мають місце фізичні і психофізичні небезпечні, а також шкідливі виробничі

чинники:

- підвищене значення напруги в електричному ланцюзі, замикання якого може відбутися через тіло людини;
- підвищений рівень статичної електрики;
- підвищений рівень електромагнітних випромінювань;
- підвищена або знижена температура повітря робочої зони;
- підвищена або знижена рухливість повітря;
- підвищена або знижена вологість повітря;
- відсутність або недолік природного світла;
- підвищена пульсація світлового потоку;
- недостатня освітленість робочого місця;
- підвищений рівень шуму на робочому місці;
- розумове перенапруження;
- емоційні навантаження;
- монотонність праці.

В промислових приміщеннях повинні бути забезпечені санітарно-гігієнічними етичні вимоги до повітря робочої зони згідно з ДСН 3.3.6.042-99 [21].

### **3.2 Заходи щодо техніки безпеки**

Основним небезпечним чинником при роботі з ЕОМ є небезпека поразки людини електричним струмом, яка усугубляється тим, що органи чуття людини не можуть на відстані знайти наявності електричної напруги на устаткуванні.

Проходячи через тіло людини, електричний струм надає на нього складну дію, що є сукупністю термічної (нагрів тканин і біологічних

середовищ), електролітичної (розкладання крові і плазми) і біологічної (роздратування і збудження нервових волокон і інших органів тканин організму) дій.

Ступінь ураження людини електричним струмом залежить від наступних факторів:

- значення сили струму;
- електричного опору тіла людини і тривалості протікання через нього струму;
- роду і частоти струму;
- індивідуальних властивостей людини і навколишнього середовища.

Даним проектом передбачаються наступні технічні способи і засоби, застережливі поразки людини електричним струмом:

- заземлення електроустановок;
- занулення;
- захисне відключення;
- електричне розділення сітей;
- використання малої напруги;
- ізоляція струмоведучих частин;
- огорожа електроустановок.

Проведемо розрахунок заземлюючого пристрою.

Початкові дані для розрахунку заземлюючого пристрою:

- напруга установки, що заземляється, - 220В;
- режим нейтралу мережі - з ізольованою нейтралюю;
- питомий опір ґрунту – 100 Ом·м(суглинок);
- гранично допустимий опір заземлюючого пристрою - 4 Ом;
- характеристика кліматичної зони (III):
  - а) середня багаторічна низька температура, °С - від –14 до -10;
  - б) тривалість замерзання вод, дні - 150;

в) коефіцієнт сезонності для вертикального електроду завдовжки 3м -1,5.

Визначимо розрахунковий опір ґрунту (Ом·м) по формулі (3.1).

$$\rho_{расч} = \psi \cdot \rho = 1,5 \cdot 100 = 150 \text{ Ом} \cdot \text{м} \quad (3.1)$$

де  $\rho$  - питомий опір ґрунту;

$\psi_i$  – кліматичний коефіцієнт, що враховує стан ґрунту під час вимірювань (таблиця 4 [22]).

Розрахуємо опір розтіканню одиночного трубчастого заземлювача за формулою (3.2).

$$R_{з.1} = \left( \frac{\rho_{расч}}{2 \cdot \pi \cdot l} \right) \cdot \ln \left( 4 \cdot \frac{l}{d} \right) \quad (3.2)$$

де  $l$  – довжина заземлювача ( $l=5\text{м}$ );

$d$  – діаметр труби і стрижня ( $d=0,05\text{м}$ );

$$R_{з.1} = \left( \frac{\rho_{расч}}{2 \cdot \pi \cdot l} \right) \cdot \ln \left( 4 \cdot \frac{l}{d} \right) = \left( \frac{150}{2 \cdot 3,14 \cdot 5} \right) \cdot \ln \left( 4 \cdot \frac{5}{0,05} \right) = 28,6 \text{ Ом}$$

Розрахуємо кількість паралельно сполучених одиночних заземлювачей по формулі (3.3).

$$n = \frac{R_{з.1}}{R_{доп} \cdot \eta} = \frac{28,6}{4 \cdot 0,47} = 15,2 \quad (3.3)$$

де  $R_{доп}=4$ . – найменш допустимий опір заземлюючого пристрою;

$\eta$  - коефіцієнт використання ґрунтового заземлення (для

шістки заземлювачей  $\eta=0,47$ ).

Округлятимемо отримане значення у більшу сторону  $n=[15,2]=16$ .

Розрахуємо довжину горизонтальної сполучної смуги по формулі (3.4).

$$L = a \cdot (n - 1) = 3 \cdot (16 - 1) = 45 \text{ м} \quad (3.4)$$

де  $a$  – відстань між вертикальними заземлювачами ( $a=3\text{м}$ );

$n$  – кількість вертикальних заземлювачей ( $n=16$ ).

Розрахуємо опір сполучної смуги по формулі (3.5).

$$R_n = \frac{\rho_{расч}}{2 \cdot \pi \cdot l} \cdot \ln\left(\frac{L^2}{d \cdot h}\right) \quad (3.5)$$

де  $d$  – еквівалентний діаметр смуги шириною  $l=5$  ( $d=0,05\text{м}$ );

$h$  – глибина заставляння смуги ( $h=0,8\text{м}$ ).

$$R_n = \frac{\rho_{расч}}{2 \cdot \pi \cdot l} \cdot \ln\left(\frac{L^2}{d \cdot h}\right) = \frac{150}{2 \cdot 3,14 \cdot 5} \cdot \ln\left(\frac{45^2}{0,05 \cdot 0,8}\right) = 51,7 \text{ Ом}$$

Розрахуємо результуючий опір заземлюючого електроду з урахуванням сполучної смуги по формулі (3.6).

$$R_{зр} = \frac{R_{з.1} \cdot R_n}{R_{з.1} \cdot \eta_n + R_n \cdot n \cdot \eta_з} \leq R_{дон} \quad (3.6)$$

де  $\eta_n$  – коефіцієнт використання сполучної смуги (для 6-і заземлювачей  $\eta_n=0,27$ ).

$$R_{zp} = \frac{R_{з.1} \cdot R_n}{R_{з.1} \cdot \eta_n + R_n \cdot n \cdot \eta_з} = \frac{26,6 \cdot 51,7}{26,6 \cdot 0,27 + 51,7 \cdot 16 \cdot 0,47} = 3,47 \text{ Ом}$$

$3,47 < 4 \Rightarrow$  умова забезпечення електробезпеки персоналу виконується.

Таким чином, остаточна кількість заземлювачей 15 шт.

### 3.3 Заходи, що забезпечують виробничу санітарію і гігієну праці

Підвищення працездатності людини і збереження його здоров'я забезпечується стабільними метеорологічними умовами.

Мікроклімат виробничих приміщень – це поєднання температури, вогкості і швидкості руху повітря, а також температури навколишніх поверхонь. Значне коливання параметрів мікроклімату приводить до порушення систем кровообігу, нервової і пітovidільної, що може викликати підвищення або пониження температури тіла, слабкість, запаморочення і навіть непритомність.

В приміщенні для виконання робіт операторського типу, пов'язаних з нервово-емоційною напругою, проектом передбачається дотримання наступних нормованих величин параметрів мікроклімату (див. табл. 3.1).

Таблиця 3.1 - Оптимальні параметри мікроклімату в робочій зоні виробничого приміщення для категорії робіт I

Період року	Температура , оС	Відносна вогкість %	Швидкіст ь руху повітря, м/с
Холодний	22.24	40.60	0,1
Теплий	23.25	40.60	0,1

Оскільки в приміщенні немає джерел виділення шкідливих речовин, можна використовувати природну вентиляцію. Площа приміщення складає 32 м<sup>2</sup>. Для забезпечення прийнятних параметрів мікроклімату в приміщенні з такою площею можна використовувати 1 кондиціонер типу БК-2000.

Спектр випромінювання монітора комп'ютера включає рентгенівську, ультрафіолетову, інфрачервону області, а також широкий діапазон хвиль інших частот. Небезпека рентгенівського проміння нехтує мала, оскільки цей вид випромінювання поглинається речовиною екрану.

Для зниження дії електромагнітного випромінювання пропонується захист часом і відстанню. Захист часом передбачає обмеження часу перебування людини в зоні дії полів. Тривалість роботи на ПЕОМ повинна складати не більше 3.5–4.5 години.

Також необхідно забезпечити раціональне освітлення в робочому приміщенні. В проекті, що розробляється, передбачається використовувати суміщене освітлення. В світлий час доби приміщення освітлюватиметься через віконні отвори, в решту часу використовуватиметься штучне освітлення.

Штучне освітлення в робочому приміщенні передбачається здійснювати з використанням люмінесцентних джерел світла в світильниках загального освітлення, оскільки люмінесцентні лампи володіють високою світловою віддачею до 75 Лам/Вт і більш, тривалим терміном служби до 10000 годин, спектральним складом випромінюваного світла, близьким до сонячного.

Зорова робота оператора ПЕВМ відповідно до ДБН В.2.5-28-2018 [25] відноситься до розряду Va з світловим потоком  $\Phi_{л}=3120$  кожна. Нормована освітленість на робочому місці ( $E_{н}$ ) при загальному освітленні складає 200 лк.

Проведемо розрахунок кількості світильників в робочому приміщенні завдовжки  $a=6$  м, шириною  $b=3$  м, заввишки  $c=4$  м. Формула розрахунку штучного освітлення при горизонтальній робочій поверхні методом

світлового потоку (3.7):

$$\Phi_{л} = \frac{E_{н} \cdot S \cdot Z \cdot K}{N \cdot U \cdot M} \quad (3.7)$$

де  $\Phi_{л}$  – світловий потік, Лм;

$E_{н}$  – нормована освітленість;

$S$  – площа підлоги, кв.м;

$Z=1.1-1.3$  - поправочний коефіцієнт світильника (для стандартних світильників);

$K$  – коефіцієнт запасу, що враховує зниження освітленості в процесі експлуатації світильників;

$N$  – число світильників;

$U=0.55-0.6$  – коефіцієнт використання, залежний від типу світильника, показника індексу приміщення і др.;

$M$  – число ламп в світильнику.

З формули (3.7) виразимо  $N$  і визначимо кількість світильників для даного приміщення:

$$N = \frac{E_{н} \cdot S \cdot Z \cdot K}{\Phi_{л} \cdot U \cdot M} \quad (3.7)$$

$$N = \frac{200 \cdot 18 \cdot 1,2 \cdot 1,5}{3120 \cdot 0,6 \cdot 2} = 1,7$$

Виходячи з цього, рекомендується використовувати 2 світильники. Світильники слід розміщувати рядами, бажано паралельно стіні з вікнами. Схема розташування світильників зображена на рис. (3.1).



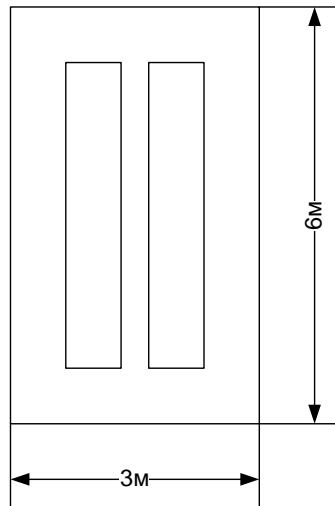


Рисунок 3.1 – Схема розташування світильників

Зниження шуму можна добитися раціонально розпланувавши приміщення, установкою устаткування на спеціальні амортизуючі прокладки. Згідно вимогам ДСН 3.3.6.037-99 [26] рівні звуку не повинні перевищувати 50 дБ.

Для зниження стомлюваності обслуговуючого персоналу в приміщеннях, де розташовані обчислювальні засоби передбачаються використовувати спокійні колірні поєднання і покриття, що не дають відблисків. Від електромагнітного випромінювання, витікаючого від ПЕОМ, використовуються захисні екрани.

Для забезпечення чистоти повітря і відповідних мікрокліматичних умов пропонується застосувати приточування-витяжну вентиляцію. Для зменшення дії шкідливих речовин і загазованості для роботи з розплавленими матеріалами робоче місце забезпечується примусовою витяжною вентиляцією. Цей метод забезпечує притоку потрібної кількості свіжого повітря ( $30 \text{ м}^3 / \text{ч}$  на одного працюючого).

Кількість повітря, яка необхідна подавати в приміщення для забезпечення необхідних параметрів повітряного середовища, визначається на підставі кількості тепла, вологи і шкідливих речовин, що поступають в приміщення, а також враховуючи видалення повітря місцевими

відсмоктуваннями від устаткування, загальнообмінною вентиляцією.

### 3.4 Рекомендації по пожежній профілактиці

Пожежі представляють небезпеку для життя людини і зв'язані як з матеріальними втратами, так і з відмовою засобів обчислювальної техніки, що спричиняє за собою порушення ходу технологічного процесу.

Горючими матеріалами в приміщенні, де розташовані ПЕОМ, є:

- поліамід - матеріал корпусу мікросхеми. Горюча речовина. Температура samozapalennya 420 °C, енергія запалення 2мДж;
- полівінілхлорид - ізоляційний матеріал. Горюча речовина. Температура samozaymannya 480 °C, енергія запалення 50мДж;
- склостоліт ДЦ - матеріал друкарської платні. Складногорючий матеріал;
- пластикат кабельний No.489 - матеріал ізоляції кабелю. Складногорючий матеріал. Температура samozaymannya 1500 °C;
- плита деревостружкова - будівельний і обробний матеріал, матеріал з якого виготовлені меблі. Складноzapalennyi матеріал. Показник горючості 1.8;
- папір – довідкова і робоча документація, література. Горючий матеріал. Показник горючості більше 2.1.

Відповідно до ДСТУ Б В.1.1-36:2016 [27] приміщення відноситься до категорії В (пожежовибухонебезпечної).

Джерелами запалення можуть бути:

- іскри при замиканні і розмиканні ланцюгів;
- іскри і дуги коротких замикань;
- перегріву від тривалого перевантаження і наявності перехідного опору;

– розряди статичної електрики.

Для того, щоб зупинити реакцію горіння, порушують умови її виникнення і підтримки. Звичайно для гасіння використовуються порушення двох основних умов сталого стану – пониження температури і режим руху газів. Пониження температури може бути досягнутий шляхом введення речовин, які поглинають багато тепла в результаті випаровування і дисоціації (наприклад, вода, порошки).

При повному тому, що згоряє органічних сполук утворюються С, SO, Н Про, N, а при тому, що згоряє неорганічних з'єднань – оксиди. Залежно від температури плавлення і тривалості реакції можуть знаходитися або у вигляді розплавів (Al O, Ti O ), або підійматися в повітря у вигляді диму (P O, Na Про, MgO).

Склад продуктів неповного згоряє горючих речовин складений і різноманітний. Це можуть бути горючі речовини:

- Н, З, СН;
- атомарний водень і кисень;
- різні радикали – ВН, СН .

Продуктами неповного згоряє можуть бути також оксиди азоту, спирти, альдегіди, кетони і високотоксичні з'єднання, наприклад, синильна кислота.

Для захисту персоналу від дій небезпечних і шкідливих чинників пожежі проектом передбачено застосування промислового фільтруючого протигаза з коробкою марки В (жовтий).

До системи запобігання пожежі відносяться: запобігання утворення горючого середовища і освіти в горючому середовищі джерел запалення, забезпечення пожежебезпеки устаткування.

Щоб запобігти пожежі в обчислювальних центрах, проектом пропонується виконання наступних вимог:

- електроживлення ЕОМ має автоматичне блокування відключення електроенергії на випадок перегріву системи, що може бути результатом

зупинки системи охолодження і кондиціонування;

– система вентиляції обчислювальних центрів обладнується блокуючими пристроями, що забезпечують її відключення на випадок пожежі. Система обладнується вогнеперегороджуючими клапанами;

– застосування устаткування, що задовольняє вимогам електростатичної іскробезпеки [22];

– після закінчення роботи, перед закриттям приміщення, всі електроустановки і персональні комп'ютери відключаються від сіті електроживлення;

– в приміщеннях обчислювальних центрів забороняється:

- 1) влаштовувати електророзетки на основах, що згорають;
- 2) використовувати синтетичні доріжки і килими;
- 3) користуватися побутовими електронагрівальними приладами;
- 4) захаращувати евакуаційні виходи і проходи;
- 5) влаштовувати на вікнах глухі ґрати;
- 6) залишати без нагляду включену в електричну мережу апаратуру, що

використовується для вимірювань і нагляду.

Для протипожежного захисту проектом пропонується обладнати приміщення площею 18 м<sup>2</sup>, яке відноситься до категорії В, автоматичною протипожежною сигналізацією із застосуванням датчиків сповіщення РІД-1 (оповіщувач димовий іонізаційний) в кількості 1 штуки і застосовується в первинних засобах пожежегасінні. Площа контролювана оповіщувачем 150 м<sup>2</sup>.

Крім того, необхідно проводити навчання робочого персоналу правилам пожежної безпеки.

Розрахуємо вірогідність виникнення пожежі у виробничому приміщенні у разі запалювання транзистора:

$$Q = l \cdot T \cdot R_{кз/отк} \cdot Q_{воспл} \cdot R_{защ} \quad (3.8)$$

де  $l$  – інтенсивність відмов пожежеопасних ЕРІ;

$T$  – час роботи пожежеопасного ЕРІ за оцінюваний інтервал часу;

$R_{кз/отк}$  - умовна вірогідність виходу ЕРІ в стан короткого замикання при його відмові;

$Q_{воспл}$  - вірогідність запалювання ЕРІ, що знаходиться в стані короткого замикання;

$R_{защ}$  – вірогідність відмови захисту пожежеопасного ЕРІ. Якщо захист відсутній,  $R_{защ}$  приймається рівній 1.

Вірогідність виникнення пожежі у разі запалювання транзистора:

$$Q = 1 \cdot 10^{-6} \cdot 1 \cdot 10^{-4} \cdot 0.1 \cdot 1 \cdot 10^{-4} = 1 \cdot 10^{-15}$$

Розрахована вірогідність виникнення пожежі значно менше допустимої, яка складає  $1 \cdot 10^{-6}$ .

В даному розділі були проаналізовані небезпечні і шкідливі виробничі чинники, що роблять вплив на персонал, розроблені заходи щодо техніки безпеки, заходу, забезпечуючи виробничу санітарію і гігієну праці, а також заходи щодо пожежної профілактики.

## ВИСНОВКИ

У кваліфікаційній випускній роботі бакалавра проведено моделювання та обчислення, а також практично реалізовано застосування структурних методів розпізнавання зображень на основі характерних ознак. За основу був взят ефективний та сучасний метод структурного аналізу SURF, який вирішує завдання пошуку та ідентифікації особливих точок зображення та створення дескрипторів, інваріантних до зсувів, масштабу і обертання.

Проведено експерименти по використанню SURF для моделювання методу зіставлень структурних описів зображень з використанням унікальності їх значень.

Програмний додаток був створений за допомогою мови програмування C++. З використанням додаткових бібліотек OpenCV, при здійсненні геометричних перетворень сформовані множини характерних ознак, що відображають властивості реальних зіставлень зображень. На основі використаної бази зображень «монстрів» проведені порівняння, які підтверджують ефективність розробленого методу.

В результаті проведеної роботи було зроблено аналіз умов праці, шкідливи та небезпечних чинників, з якими стикається робітник. Було визначено параметри і певні характеристики приміщення для роботи над запропонованим проектом написаному в кваліфікаційній роботі, описано, які заходи потрібно зробити для того, щоб дане приміщення відповідало необхідним нормам і було комфортним і безпечним для робітника. Приведені рекомендації щодо організації робочого місця, а також важливу інформацію щодо пожежної та електробезпеки. Була наведена схема, розміри приміщення та наведено значення температури, вологості й рухливості повітря, необхідна кількість і потужність ламп та інші параметри, значення яких впливає на умови праці робітника, а також – наведені інструкції з охорони праці, техніки безпеки при роботі на комп'ютері.

## ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ

- 1) Шапиро, Л. Компьютерное зрение [Текст] / Л. Шапиро, Дж. Стокман.; пер. с англ.– М.: БИНОМ. Лаборатория знаний, 2006. – 752 с.
- 2) Рассел, С. Искусственный интеллект: современный подход; 2-е изд. [Текст] / С. Рассел, П. Норвиг; пер. с англ.– М.: Изд. дом «Вильямс», 2006. – 1408 с.
- 3) Гороховатский, В.А. Структурное распознавание изображений на основе моделей голосования признаков характерных точек [Текст] / В.А. Гороховатский, Е.П. Путятин // Реєстрація, зберігання і обробка даних. – 2008. – Т. 10, № 4. – С. 75–85.
- 4) Гороховатский, В.А. Модели локально-пространственного подобия структурных описаний визуальных объектов [Текст] / В.А. Гороховатский // Системи управління, навігації та зв'язку: зб. наук. пр.– К.: ЦНДІ навігації і управління. – 2010. – Вип. 3(15). – С. 82–85.
- 5) Путятін, Є.П. Методи та алгоритми комп'ютерного зору: навч. посіб. [Текст] / Є.П. Путятін, В.О. Гороховатський, О.О. Матат. – Х.: ТОВ «Компанія СМІТ», 2006. – 236 с
- 6) Красильников, Н.Н. Статистическая теория передачи изображений [Текст] / Н.Н. Красильников. – М.: Связь, 1976.– 184с.
- 7) Шлезингер, М. Десять лекций по статистическому и структурному распознаванию [Текст] / М. Шлезингер, В. Главач. – К.: Наукова думка, 2004.–535 с.
- 8) Шлезингер, М.И. Математические средства обработки изображений [Текст] / М. И. Шлезингер. – К.: Наукова думка, 1989.–200 с.
- 9) Журавлев, Ю.И. Избранные научные труды [Текст] / Ю.И. Журавлев. –М.:Магистр, 1998. – 416 с.
- 10) Lowe, D. G. Object recognition from local scale-invariant features [Text] / D. G. Lowe // International Conference on Computer Vision. – Corfu,

Greece (September 1999). – P. 1150–1157.

11) Колмогоров, А.Н. Элементы теории функций и функционального анализа [Текст] / А.Н. Колмогоров, С.В. Фомин. – М.: Наука, 1976. – 544 с.

12) Duda, R.O. Pattern classification [Text] / Duda R.O., Hart P.E., Stork D.G. – 2ed., Wiley, 2000.–738p.

13) Kim, S. Object Recognition Using a Generalized Robust Invariant Feature and Gestalt's Law of Proximity and Similarity [Text] / S. Kim, K.-J. Yoon, I.-S. Kweon // Pattern Recognition.– 2008.– 41, No. 2.– P. 726–741.

14) Бондаренко, М.Ф. Теория интеллекта: Учебник [Текст] / М.Ф. Бондаренко, Ю.П. Шабанов-Кушнарченко. – Харьков: ООО «Компания СМІТ», 2006. –576с.

15) Колмогоров, А.Н. Элементы теории функций и функционального анализа [Текст] / А.Н. Колмогоров, С.В. Фомин. – М.: Наука, 1976. – 544 с.

16) Миркин, Б.Г. Анализ качественных признаков и структур [Текст] / Б.Г. Миркин. – М.: Статистика, 1980. –319с.

17) Браверман, Э.М. Автоматический анализ сложных изображений [Текст] / Э.М. Бравермана. – М.: Мир, 1969. –310 с.

18) Назаров, А.С. Фотограмметрия: [учеб. Пособие для студентов вузов] [Текст] / А.С. Назаров. – Мн.: Тетра-Системс, 2006. – 368 с.

19) Рубан, И.В. Метод обработки цифрового изображения для локализации информативных областей [Текст] / И.В. Рубан, О.В. Шитова, Д.П. Пашков // Збірник наукових праць Національної академії наук України. Інститут проблем моделювання в енергетиці ім. Г.Є. Пухова. Моделювання та інформаційні технології. – К., 2010. – Вип. 55. –С. 225-229.

20) Казаков, Я.В. Повышение качества киноизображения средствами вычислительной техники. Часть вторая: методы признакового описания статичного изображения [Электронный ресурс] / Я.В. Казаков. – СПб.: Санкт-Петербургский государственный университет кино и телевидения, 2003. – Режим доступа: <http://www.sciteclibrary.ru/rus/catalog/pages/5026.html>.

21) ДСН 3.3.6.042-99 Державні санітарні норми мікроклімату



виробничих. Режим доступу: [https://dnaop.com/html/34094/doc-ДСН\\_3.3.6.042-99](https://dnaop.com/html/34094/doc-ДСН_3.3.6.042-99)

22) ДСТУ Б А.3.2-13:2011 Система стандартів безпеки праці. Будівництво. Електро безпечність. Загальні вимоги. Режим доступу- : [http://online.budstandart.com/ru/catalog/doc-page?id\\_doc=27973](http://online.budstandart.com/ru/catalog/doc-page?id_doc=27973)

23) ДСанПіН 3.3.2-007-98 Державні санітарні правила і норми. Гігієнічні вимоги до організації роботи з візуальними дисплейними терміналами електронно-обчислювальних машин Режим доступу- : <https://zakon.rada.gov.ua/rada/show/v0007282-98>

24) НПАОП 0.00-7.15-18 Вимоги щодо безпеки та захисту здоров'я працівників під час роботи з екранними пристроями Міністерство доходів і зборів України Наказ від 05.09.2013 р. № 443 "Про затвердження Примірної інструкції з охорони праці під час експлуатації електронно-обчислювальних машин") Режим доступу- : [https://zakon.rada.gov.ua/laws/show/%D0%BD%D0%BF%D0%B0%D0%BE%D0%BF\\_0.00-7.15-18](https://zakon.rada.gov.ua/laws/show/%D0%BD%D0%BF%D0%B0%D0%BE%D0%BF_0.00-7.15-18)

25) ДБН В.2.5-28-2018. Природне і штучнеосвітлення. Режим доступу - : [https://okna.ua/img\\_all/oknaua/dbn-V-2-5-28-2018-ed.pdf](https://okna.ua/img_all/oknaua/dbn-V-2-5-28-2018-ed.pdf)

26) ДСН 3.3.6.037-99 Державні санітарні норми виробничого шуму, ультразвуку та інфразвуку. Режим Доступу-: <https://zakon.rada.gov.ua/rada/show/va037282-99>

27) ДСТУ Б В.1.1-36:2016 «Визначення категорій приміщень, будинків та зовнішніх установок за вибухопожежною та пожежною небезпекою». Наказ від 15.06.2016 №158. Режим доступу: [www. URL: https://dbn.co.ua/load/normativy/dstu/dstu\\_b\\_v\\_1\\_1\\_36/5-1-0-1759](http://www.url:https://dbn.co.ua/load/normativy/dstu/dstu_b_v_1_1_36/5-1-0-1759)

## Додаток А

### Лістинг програми

```

#include <cv.h>
#include <highgui.h>
#include <ctype.h>
#include <stdio.h>
#include <stdlib.h>
#include <iostream>
#include <vector>
using namespace std;
IplImage *image = 0;
// Сравнение двух особенностей
double
compareSURFDescriptors( const float* d1, const float* d2,
double best, int
length )
{
double total_cost = 0;
assert( length % 4 == 0 );
for( int i = 0; i < length; i += 4 )
{
double t0 = d1[i] - d2[i];
double t1 = d1[i+1] - d2[i+1];
double t2 = d1[i+2] - d2[i+2];
double t3 = d1[i+3] - d2[i+3];
total_cost += t0*t0 + t1*t1 + t2*t2 + t3*t3;
if( total_cost > best )
break;
}
return total_cost;
}
// Сравнивает одну особенность объекта со всеми
особенностями сцены
int
naiveNearestNeighbor( const float* vec, int laplacian,
const CvSeq* model_keypoints,
const CvSeq* model_descriptors )
{
int length = (int)(model_descriptors-
>elem_size/sizeof(float));
int i, neighbor = -1;
double d, dist1 = 1e6, dist2 = 1e6;
CvSeqReader reader, kreader;
// Начальная особенность сцены
cvStartReadSeq( model_keypoints, &kreader, 0 );
cvStartReadSeq( model_descriptors, &reader, 0 );
// Перебор всех особенностей сцены
for( i = 0; i < model_descriptors->total; i++ )

```

```

{
const CvSURFPoint* kp = (const CvSURFPoint*)kreader.ptr;
const float* mvec = (const float*)reader.ptr;
CV_NEXT_SEQ_ELEM( kreader.seq->elem_size, kreader );
CV_NEXT_SEQ_ELEM( reader.seq->elem_size, reader );
// Для ускорения сначала сравнивается лапласиан особенностей
if( laplacian != kp->laplacian )
continue;
// Сравнение особенностей
d = compareSURFDescriptors( vec, mvec, dist2, length );
if( d < dist1 )
{
// Найдена лучшее совпадение особенностей
dist2 = dist1;
dist1 = d;
neighbor = i;
}
else if ( d < dist2 )
dist2 = d;
}
if ( dist1 < 0.6*dist2 )
return neighbor;
return -1;
}

// Функция ищет совпадающие пары
void
findPairs( const CvSeq* objectKeypoints, const CvSeq*
objectDescriptors,
const CvSeq* imageKeypoints, const CvSeq* imageDescriptors,
vector<int>&
ptpairs )
{
int i;
CvSeqReader reader, kreader;
// Установка начальной особенности объекта распознавания
cvStartReadSeq( objectKeypoints, &kreader );
cvStartReadSeq( objectDescriptors, &reader );
ptpairs.clear();
// Перебор всех особенностей объекта
for( i = 0; i < objectDescriptors->total; i++ )
{
const CvSURFPoint* kp = (const CvSURFPoint*)kreader.ptr;
const float* descriptor = (const float*)reader.ptr;
CV_NEXT_SEQ_ELEM( kreader.seq->elem_size, kreader );
CV_NEXT_SEQ_ELEM( reader.seq->elem_size, reader );
// Сравнение текущей особенности со всеми особенностями из
сцены
int nearest_neighbor = naiveNearestNeighbor( descriptor, kp-
>laplacian,
imageKeypoints, imageDescriptors );
if( nearest_neighbor >= 0 )
{
// Нашлось совпадение особенностей

```

```

ptpairs.push_back(i);
ptpairs.push_back(nearest_neighbor);
}
}
}
/* Грубое нахождение местоположения объекта */
int
locatePlanarObject( const CvSeq* objectKeypoints, const
CvSeq*
objectDescriptors,
const CvSeq* imageKeypoints, const CvSeq* imageDescriptors,
const CvPoint src_corners[4], CvPoint dst_corners[4] )
{
double h[9];
CvMat _h = cvMat(3, 3, CV_64F, h);
vector<int> ptpairs;
vector<CvPoint2D32f> pt1, pt2;
CvMat _pt1, _pt2;
int i, n;
// Ищем пары особенностей на обеих картинках, которые
соответствуют
// друг другу
findPairs( objectKeypoints, objectDescriptors,
imageKeypoints,
imageDescriptors, ptpairs );
n = ptpairs.size()/2;
// Если пар мало, значит надо выходить - объект не найден
if( n < 4 )
return 0;
// Выделяем память
pt1.resize(n);
pt2.resize(n);
// Считываем координаты «особых»точек
for( i = 0; i < n; i++ )
{
pt1[i] =
((CvSURFPoint*)cvGetSeqElem(objectKeypoints,ptpairs[i*2]))->pt;
pt2[i] =
((CvSURFPoint*)cvGetSeqElem(imageKeypoints,ptpairs[i*2+1]))->pt;
}
// По полученным векторам создаём матриц
_pt1 = cvMat(1, n, CV_32FC2, &pt1[0] );
_pt2 = cvMat(1, n, CV_32FC2, &pt2[0] );
// Находим трансформацию между исходным изображением и с
тем, которое
// ищем
if( !cvFindHomography( &_pt1, &_pt2, &_h, CV_RANSAC, 5 ) )
return 0;
// По полученному значению трансформации (в матрицу _h)
находим
// координаты четырёхугольника, характеризующего объект
for( i = 0; i < 4; i++ )
{

```

```

double x = src_corners[i].x, y = src_corners[i].y;
double Z = 1./(h[6]*x + h[7]*y + h[8]);
double X = (h[0]*x + h[1]*y + h[2])*Z;
double Y = (h[3]*x + h[4]*y + h[5])*Z;
dst_corners[i] = cvPoint(cvRound(X), cvRound(Y));
}
return 1;
}
int main(int argc, char** argv)
{
// Инициализация параметров
const char* object_filename = argc == 3 ? argv[1] :
"box.png";
const char* scene_filename = argc == 3 ? argv[2] :
"box_in_scene.png";
CvMemStorage* storage = cvCreateMemStorage(0);
cvNamedWindow("Object", 1);
cvNamedWindow("Object Correspond", 1);
static CvScalar colors[] =
{
{{0,0,255}},
{{0,128,255}},
{{0,255,255}},
{{0,255,0}},
{{255,128,0}},
{{255,255,0}},
{{255,0,0}},
{{255,0,255}},
{{255,255,255}}
};
// Загрузка изображений
IplImage* object = cvLoadImage( object_filename,
CV_LOAD_IMAGE_GRAYSCALE );
IplImage* image = cvLoadImage( scene_filename,
CV_LOAD_IMAGE_GRAYSCALE );
if( !object || !image )
{
fprintf( stderr, "Can not load %s and/or %s\n"
"Usage: find_obj [<object_filename> <scene_filename>]\n",
object_filename, scene_filename );
exit(-1);
}
// Перевод в градации серого
IplImage* object_color = cvCreateImage(cvGetSize(object), 8,
3);
cvCvtColor( object, object_color, CV_GRAY2BGR );
CvSeq *objectKeypoints = 0, *objectDescriptors = 0;
CvSeq *imageKeypoints = 0, *imageDescriptors = 0;
int i;
// Инициализация структуры CvSURFParams с размером
дескрипторов в 128
// элементов
CvSURFParams params = cvSURFParams(500, 1);

```

```

// Засекаем время
double tt = (double)cvGetTickCount();
// Ищем особенности объекта распознавания
cvExtractSURF( object, 0, &objectKeypoints,
&objectDescriptors, storage,
params );
printf("Object Descriptors: %d\n", objectDescriptors-
>total);
// Ищем особенности сцены
cvExtractSURF( image, 0, &imageKeypoints, &imageDescriptors,
storage,
params );
printf("Image Descriptors: %d\n", imageDescriptors->total);
// Сколько потребовалось времени (У меня 167 милли секунд)
tt = (double)cvGetTickCount() - tt;
printf( "Extraction time = %gms\n",
tt/(cvGetTickFrequency()*1000.));
// Устанавливаем границы изображений, внутри которых будут
сравниваться
//особенности
CvPoint src_corners[4] = {{0,0}, {object->width,0}, {object-
>width, object-
>height}, {0, object->height}};
CvPoint dst_corners[4];
// Создание дополнительного изображение (в нём будет сцена и
объект)
// Запустите пример и поймёте о чём речь
IplImage* correspond = cvCreateImage( cvSize(image->width,
object-
>height+image->height), 8, 1 );
cvSetImageROI( correspond, cvRect( 0, 0, object->width,
object->height ) );
cvCopy( object, correspond );
cvSetImageROI( correspond, cvRect( 0, object->height,
correspond->width,
correspond->height ) );
cvCopy( image, correspond );
cvResetImageROI( correspond );
// Вызываем функцию, находящую объект на экране
if( locatePlanarObject( objectKeypoints, objectDescriptors,
imageKeypoints,
imageDescriptors, src_corners, dst_corners ))
{
// Обводим нужный четырёхугольник
for( i = 0; i < 4; i++ )
{
CvPoint r1 = dst_corners[i%4];
CvPoint r2 = dst_corners[(i+1)%4];
cvLine( correspond, cvPoint(r1.x, r1.y+object->height ),
cvPoint(r2.x, r2.y+object->height ), colors[8] );
}
}
}

```

```

        // Если в этом месте вывести результат на экран, то
        // получится то, что
        // показано на рисунке 23.3.
        vector<int> ptpairs;
        // Снова ищутся все совпадающие пары особенностей в обеих
        картинках
        findPairs( objectKeypoints, objectDescriptors,
        imageKeypoints,
        imageDescriptors, ptpairs );
        // Между парами особенностей на рисунке проводятся прямые
        for( i = 0; i < (int)ptpairs.size(); i += 2 )
        {
            CvSURFPoint* r1 = (CvSURFPoint*)cvGetSeqElem(
            objectKeypoints,
            ptpairs[i] );
            CvSURFPoint* r2 = (CvSURFPoint*)cvGetSeqElem(
            imageKeypoints,
            ptpairs[i+1] );
            cvLine( correspond, cvPointFrom32f(r1->pt),
            cvPoint(cvRound(r2->pt.x), cvRound(r2->pt.y+object-
            >height)),
            colors[8] );
        }
        // Результат можно посмотреть на рисунке 23.4.
        cvShowImage( "Object Correspond", correspond );
        // Выделяем особенности окружностями (Рис. 23.5)
        for( i = 0; i < objectKeypoints->total; i++ )
        {
            CvSURFPoint* r = (CvSURFPoint*)cvGetSeqElem(
            objectKeypoints, i );
            CvPoint center;
            int radius;
            center.x = cvRound(r->pt.x);
            center.y = cvRound(r->pt.y);
            radius = cvRound(r->size*1.2/9.*2);
            cvCircle( object_color, center, radius, colors[0], 1, 8, 0
            );
        }
        cvShowImage( "Object", object_color );
        cvWaitKey(0);
        cvDestroyWindow("Object");
        cvDestroyWindow("Object SURF");
        cvDestroyWindow("Object Correspond");

        //Загружаем и обучаем
        int i;
        char buf[256];
        CvSURFParams params = cvSURFParams(500, 1);
        CvSeq *objectKeypoints[1000], *objectDescriptors [1000];
        CvMemStorage* storage = cvCreateMemStorage(0);
        for(i=0;i<500;i++)
        {
            sprintf(buf,"e:\\algor\\grey2\\%d\\%d_i110.png",i+1,i+1);

```

```

IplImage* object=cvLoadImage(buf,CV_LOAD_IMAGE_GRAYSCALE);
IplImage* object_color = cvCreateImage(cvGetSize(object), 8,
3);
cvCvtColor( object, object_color, CV_GRAY2BGR );
objectKeypoints[i]=0;objectDescriptors[i]=0;
double tt = (double)cvGetTickCount();
cvExtractSURF( object, 0, &objectKeypoints[i],
&objectDescriptors[i],
storage, params );
printf("%d Image Descriptors: %d\n",
i+1,objectDescriptors[i]->total);
tt = (double)cvGetTickCount() - tt;
printf( "Extraction time = %gms\n",
tt/(cvGetTickFrequency()*1000.));
cvReleaseImage(&object);
cvReleaseImage(&object_color);
}
LOG_FILE log;
log.BLog("out1.log");
//A теперь необходимо произвести сравнение
int j,k;
int all=0;
int all_p=0;
int pai_r[1000];
int maxpair;
int il;
for(i=0;i<500;i++)
{
sprintf(buf,"Begin %d-s image:",i+1);
log.Log(buf);
for(j=0;j<11;j++)
{
k=120+j*10;
if (j>=8) k=210+(j-8)*20;
sprintf(buf,"e:\\algor\\grey2\\%d\\%d_i%d.png",i+1,i+1,k);
log.Log(buf);
CvSeq *objectKeypoints1, *objectDescriptors1;
IplImage* object=cvLoadImage(buf,CV_LOAD_IMAGE_GRAYSCALE);
IplImage* object_color = cvCreateImage(cvGetSize(object), 8,
3);
cvCvtColor( object, object_color, CV_GRAY2BGR );
objectKeypoints1=0;objectDescriptors1=0;
double tt = (double)cvGetTickCount();
cvExtractSURF( object, 0, &objectKeypoints1,
&objectDescriptors1,
storage, params );
printf("%d Image Descriptors: %d\n",
i+1,objectDescriptors[i]->total);
tt = (double)cvGetTickCount() - tt;
printf( "Extraction time = %gms\n",
tt/(cvGetTickFrequency()*1000.));
cvReleaseImage(&object);
cvReleaseImage(&object_color);
}

```



```
log.Log("Find object");
//здесь собственно предстоит найти объект
maxpair=0;
for(i1=0;i1<500;i1++)
{
vector<int> ptpairs;
findPairs( objectKeypoints[i1], objectDescriptors[i1],
objectKeypoints1, objectDescriptors1, ptpairs );
int n = ptpairs.size()/2;
pai_r[i1]=n;
if (i1>0 && n>pai_r[maxpair]) maxpair=i1;
}
sprintf(buf,"Object #%d",maxpair+1);
log.Log(buf);
all++;
//if (pai_r[maxpair]<4) maxpair=-1;
if (maxpair==i) all_p++;
}
}
sprintf(buf,"All #%d",all);
log.Log(buf);
sprintf(buf,"All D #%d",all_p);
log.Log(buf);

return 0;
}
```

Додаток Б  
Комп'ютерна презентація

Дипломна робота бакалавра

**Методи структурного  
розпізнавання візуальних  
об'єктів**

Виконав:  
ст. гр. КІ-163  
Приходько Е.В.

Керівник:  
Проф. Рязанцев О.І.

**Актуальність**

- Із завданням розпізнавання образів живі системи, у тому числі і людина, стикаються постійно з моменту своєї появи.
- Розвиток сучасних технологій штучного інтелекту викликає потребу в розробці нових, більш ефективних підходів до аналізу багатовимірної інформації, наприклад, при розпізнаванні зображень.

## Постановка задачі

- 1) провести аналіз методу структурного розпізнавання SURF;
- 2) провести аналіз можливостей бібліотеки OpenCV;
- 3) розробити програму, для моделювання та обчислення статистичних ознак;
- 4) провести аналіз і використовувати методи для порівняння статистичних ознак.

## Програмна реалізація

- Мова програмування: C ++ (з використанням бібліотеки OpenCV)
- Вхідні дані: рисунки для порівняння
- Вихідні дані: порівняння двох картинок, і підрахунок статистичних ознак
- Результат: Дані порівняння двох рисунків

## Аналіз результатів

```

C:\Users\Home\documents\visual studio 2012\Projects\Test2\Debug\Test2.exe
Mat osh 1451 - 0.00987736
Mat osh 1461 - 0.00089408
Mat osh 1471 - 0.003095833
Mat osh 1481 - 0.0102187
Mat osh 1491 - 0.0105483
Mat osh 1501 - 0.0045903
Mat osh 1511 - 0.0105424
Mat osh 1521 - 0.00093123
Mat osh 1531 - 0.00272387
Mat osh 1541 - 0.0061912
Mat osh 1551 - 0.010181
Mat osh 1561 - 0.00623909
Mat osh 1571 - 0.0129124
Mat osh 1581 - 0.00000022
Mat osh 1591 - 0.00798898
Mat osh 1601 - 0.00609132
Mat osh 1611 - 0.0115884
Mat osh 1621 - 0.00603079
Mat osh 1631 - 0.0100995
Sum = 64
OBJECT 000001
Good matches found: 26
Algorithm duration: 354.031
  
```



## Аналіз результатів

Еталони використовувані для порівняння:



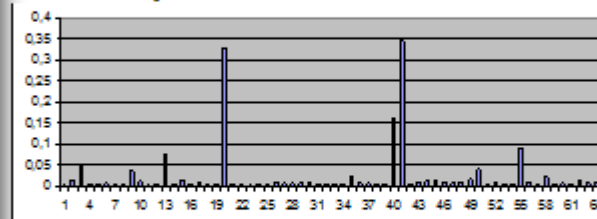
**Еталон №1**



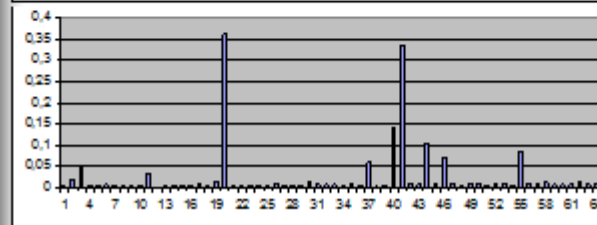
**Еталон №2**

## Аналіз результатів

### ● Гістограми середніх значення векторів ХП:



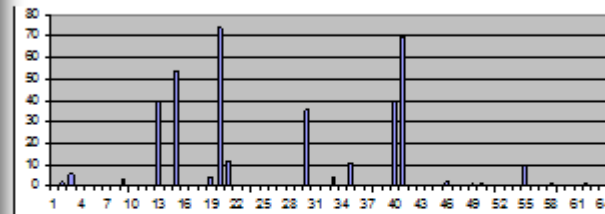
● Для еталона №1



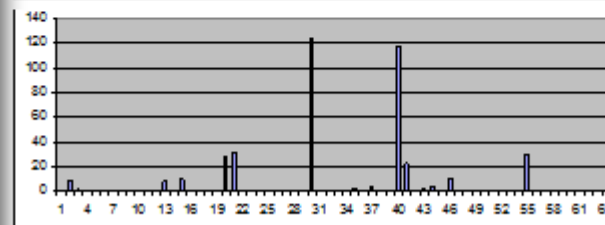
● Для еталона №2

## Аналіз результатів

### ● Гістограми дисперсій векторів ХП:



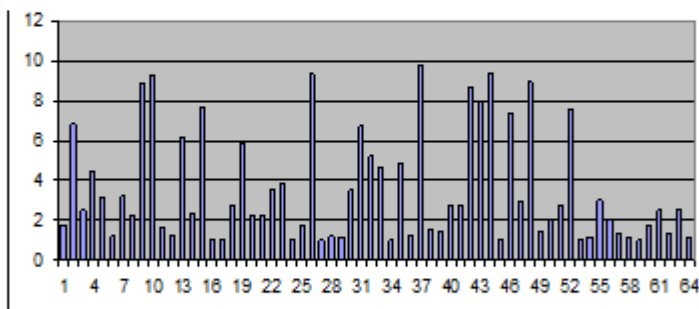
● Для еталона №1



● Для еталона №2

## Аналіз результатів

### ● Гістограма статистичних даних



## Висновки

- Реалізовано на практиці використання структурних методів розпізнавання зображень на основі ХП
- Проаналізовано використання методу SURF для моделювання структурних ознак
- Зроблені порівняння «монстрів», які підтверджують ефективність методу