

**Міністерство освіти і науки України  
Національний технічний університет України  
"Київський політехнічний інститут імені Ігоря Сікорського"**

**Жученко А. І.  
Сазонов А. Ю.  
Черепанська І. Ю.**

# **СПЕЦІАЛЬНІ РОЗДІЛИ ТЕОРІЇ АВТОМАТИЧНОГО КЕРУВАННЯ. ПРАКТИКУМ**

*Рекомендовано Методичною радою КПІ ім. Ігоря Сікорського  
як навчальний посібник для здобувачів третього рівня вищої освіти,  
які навчаються за освітньо-науковою програмою  
«Автоматизація та комп'ютерно-інтегровані технології»*

**Київ  
2021**

УДК 681.5: 004.8  
С 45

Рецензенти:

**Купін Андрій Іванович**, д-р техн. наук, професор,  
Криворізький національний університет

**Казак Василь Миколайович**, д-р техн. наук, професор,  
Національний авіаційний університет

**Волощук Володимир Анатолійович**, д-р техн. наук, доцент,  
Національний технічний університет України “Київський  
політехнічний інститут імені Ігоря Сікорського”

Відповідальний редактор **Цапар В.С.**, канд. техн. наук, доцент  
Національний технічний університет України “Київський  
політехнічний інститут імені Ігоря Сікорського”

*Гриф надано Методичною радою КПІ ім. Ігоря Сікорського  
(протокол №2 від 09.12.2021 р.)  
за поданням Вченої ради Інженерно-хімічного факультету  
(протокол № 10 від 29.11.2021 р.)*

#### **Електронне мережне навчальне видання**

*Жученко Анатолій Іванович*, д-р техн. наук, проф.  
*Сазонов Артем Юрійович*, канд. техн. наук, доц.  
*Черепанська Ірина Юріївна*, д-р. техн. наук, доц.

**С 45 Спеціальні розділи теорії автоматичного керування.** Практикум [Електронний ресурс]: навч. посіб. для здобув. спеціальності 151 «Автоматизація та комп'ютерно-інтегровані технології» / КПІ ім. Ігоря Сікорського: А.І. Жученко, А.Ю. Сазонов, І.Ю. Черепанська – (1 файл 2,4 Мбайт). – Київ: КПІ ім. Ігоря Сікорського, 2021. – 99 с.

ISBN

Навчальний посібник є частиною навчально-методичного забезпечення з дисципліни та розроблений у відповідності до її програми (силабуса). Посібник призначений для надання допомоги здобувачам в процесі підготовки та виконання практичних робіт, також розглянуто деякі різнопланові за галуззю, змістом та методами задачі автоматизації, розв'язок яких ґрунтуються на використанні технологій машинного зору, наведено методичні вказівки та індивідуальні завдання для практичних робіт. До кожної роботи подано теоретичні відомості про метод виконання роботи, алгоритм та/або математичну постановку задачі, порядок та приклад проведення розрахунків, оброблення результатів та оформлення звіту з практикуму. Для аспірантів, які навчаються за освітньо-науковою програмою Автоматизація та комп'ютерно-інтегровані технології.

УДК 681.5: 004.8  
© А. І. Жученко, А. Ю. Сазонов,  
І. Ю. Черепанська, 2021  
© КПІ ім. Ігоря Сікорського, 2021

ISBN

## ЗМІСТ

|                                                           |    |
|-----------------------------------------------------------|----|
| ПЕРЕЛІК ПРИЙНЯТИХ ПОЗНАЧЕНЬ.....                          | 4  |
| ВИКОРИСТАНІ ПІКТОГРАМИ .....                              | 5  |
| ВСТУП .....                                               | 6  |
| ТЕОРЕТИЧНИЙ МАТЕРІАЛ .....                                | 7  |
| 1. Машинний зір як інструмент сучасної автоматизації..... | 7  |
| 2. Освітлення в системах машинного зору.....              | 8  |
| 3. Напрями машинного зору .....                           | 11 |
| 4. Розпізнавання за особливостями / ключовими точками ..  | 17 |
| 4.1. Scale-Invariant Feature Transform .....              | 19 |
| 4.2. Перетворення Хафа .....                              | 27 |
| 4.3. Узагальнене перетворення Хафа .....                  | 33 |
| 5. Проективна геометрія та модель камери .....            | 37 |
| 5.1. Загальні поняття проективної геометрії .....         | 37 |
| 5.2. Аберації та об'єктиви .....                          | 40 |
| 5.3. Модель камери.....                                   | 46 |
| 5.3. Стереозір .....                                      | 49 |
| 5.4. Завдання для контролю знань .....                    | 57 |
| КОМП'ЮТЕРНИЙ ПРАКТИКУМ.....                               | 62 |
| Загальні вказівки до виконання робіт.....                 | 63 |
| Загальні вимоги до оформлення звітів .....                | 63 |
| Практикум №1 .....                                        | 67 |
| Завдання 1.1. Детектор кутів Харріса .....                | 74 |
| Завдання 1.2. Патч-дескриптор .....                       | 75 |
| Завдання 1.3. Зіставлення точок інтересу .....            | 75 |
| Практикум № 2 .....                                       | 78 |
| Завдання 2.1. RANSAC .....                                | 83 |
| Завдання 2.2. Кластеризація.....                          | 84 |
| Додаток Б.....                                            | 86 |
| Штampi для текстових документів.....                      | 86 |
| Додаток В.....                                            | 87 |
| Форма титульного листа звіту з практичних робіт .....     | 87 |
| СПИСОК ЛІТЕРАТУРИ .....                                   | 88 |
| ПРЕДМЕТНИЙ ПОКАЖЧИК.....                                  | 96 |

## ПЕРЕЛІК ПРИЙНЯТИХ ПОЗНАЧЕНЬ

БД – база даних

БЗ – база знань

ГВС – гнучка виробнича система

ЕС – експертна система

ЗУС – засоби упорядкування середовища

ІДС – інтелектуальна діалогова система

МЛВ – механізм логічного виведення (висновку)

ПР – промисловий робот

СППР – система підтримки прийняття рішень

СШІ – система штучного інтелекту

ШІ – штучний інтелект

ШН – штучний нейрон

ШНМ – штучна нейронна мережа

DoG – різниця гаусіан (Difference of Gaussian)

fps – кількість кадрів на секунду (frames per second)

GHT – Узагальнене перетворення Хафа (Generalized Hough Transform)

SIFT – масштабонезалежне перетворення особливостей (Scale-Invariant Feature Transform)

## ВИКОРИСТАНІ ПІКТОГРАМИ



Важливий матеріал, який необхідно запам'ятати.



Теоретичні та практичні завдання для самостійної роботи та контролю знань студентів.



Інформація яка наводиться як приклад.



Інформація, що може бути корисною і цікавою, але не є обов'язковою для запам'ятовування.

## ВСТУП

Сучасна теорія керування є

Штучний інтелект є одним із найсучасніших напрямів розвитку інформаційно-комп'ютерних технологій, що успішно застосовують в автоматизації. Методи, технології та системи штучного інтелекту знайшли своє практичне застосування в різних сферах, де необхідно розв'язувати складні багатопараметричні та складноформалізовані задачі в режимі реального часу при швидко змінюваних умовах у неструктурованому стохастичному просторі. Зокрема при управлінні транспортними засобами, літальними апаратами, космічними об'єктами, в медичній діагностиці, в приладо- та машинобудуванні; банківській справі, фінансах, тощо.

Використання штучного інтелекту покликано підвищити якість та зменшити трудомісткість вирішуваних задач. Тому очевидною є важливість поданого в підручнику матеріалу, що може використовуватись під час підготовки фахівців з автоматизації та комп'ютерно-інтегрованих технологій. Підручник містить теоретичну та практичну складову цього сучасного інструментарію автоматизації. Зокрема, в підручнику систематизовано та подано у доступній для студентів формі базовий теоретичний та практичний матеріал щодо технологій штучного інтелекту, методів формалізації інтелектуальних задач, технології та інструменти побудови інтелектуальних систем, застосування систем машинного зору. Крім того, наведено методичні вказівки до виконання лабораторних та практичних робіт, завдання до самостійної роботи студентів, що розроблені відповідно до вимог програми та навчального плану підготовки магістрів галузі знань 15 “Автоматизація та приладобудування” спеціальності 151 “Автоматизація та комп'ютерно-інтегровані технології”. Також наведено тестові завдання для контролю знань студентів, та матеріал для самостійного вивчення, що є необхідним при підготовці фахівців за вказаною спеціальністю.

Підручник підготовлено міжвузівським колективом авторів за результатами співпраці кафедри автоматизації та комп'ютерно-інтегрованих технологій імені професора Б. Б. Самотокіна Житомирського державного технологічного університету та кафедри автоматизації хімічних виробництв Національного технічного університету України “Київський політехнічний інститут імені Ігоря Сікорського”.

# ТЕОРЕТИЧНИЙ МАТЕРІАЛ

## 1. Машинний зір як інструмент сучасної автоматизації

На сьогодні нейромережі займають одне із найважливіших місць як у промисловості так і у побуті. Зокрема, в промисловості нейромережі знайшли широке застосування в задачах керування обладнанням в умовах невизначеності. Ярким прикладом таких невизначеностей є безколізійне переміщення роботизованого технологічного обладнання у виробничому приміщенні. Складність прийняття рішень в таких умовах обумовлена неможливістю однозначно визначити об'єкти на зображенні та прийняти рішення щодо їх належності до перешкод.

Таку задачу необхідно умовно розділити на складові підзадачі:

- Розпізнавання зображення;
- Визначення характерних (ключових) точок та об'єктів на зображенні;

Аналіз зображення із використанням нейромереж для визначення перешкод та прийняття рішень.

**Машинний зір** – це застосування методів комп'ютерного зору для промислової автоматизації. **Комп'ютерний зір** – це складова штучного інтелекту, що стосується обробки зображень отриманих з реального світу.

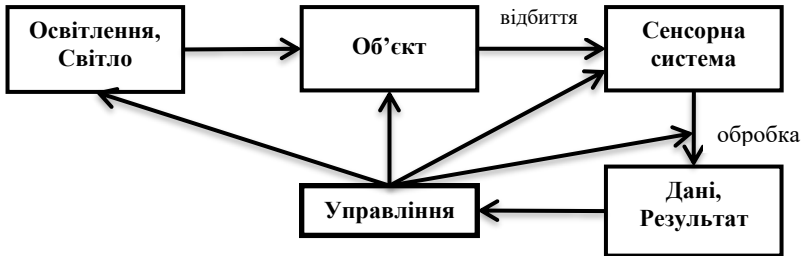
Машинний зір сьогодні став невід'ємною частиною виробничих задач: системи визначення браку використовуються у промисловості, системи навігації застосовуються для орієнтування у робочому просторі тощо. Коли ми говоримо про машинний зір, то перший акцент необхідно зробити на те, що основну роль в цьому процесів відіграють відеокамери.

Отже, які основні цілі на сьогодні ставляться перед машинним зором:

- Автоматизувати будь-який процес обробки візуальної інформації;
- Виключити людський фактор при прийнятті рішень у виробничих умовах;
- Підвищити швидкість обробки інформації.

Методи машинного зору: виявлення об'єктів, сегментація зображення, слідкування, оцінювання пози, накладання на 3D-модель, розпізнавання об'єктів (наприклад, людських облич) тощо.

В галузі машинного зору використовуються методи статистичного навчання, проективної геометрії, обробки зображень, теорії графів та інших областей. Фактично, **машинний зір** – це комп'ютеризована система, яка приймає рішення на основі аналізу цифрових зображень (рис. 1.).



*Рис. 1. Компоненти системи машинного зору*

До складу систем машинного зору (рис. 1) входять: освітлювальні прилади та системи, що забезпечують правильне освітлення аналізованого об'єкту; власне самі об'єкти; сенсори, що використовуються для отримання інформації про об'єкт. Дані з сенсорної системи обробляються з використанням алгоритмів комп'ютерного зору та використовуються для керування складовими. Вказаний підхід відповідає ідеології неперервного керування та покращення результату, що викладено у міжнародному стандарті ISO 9001.

## 2. Освітлення в системах машинного зору

Освітлення як загальне так і локальне відіграє важливу роль в промисловості при застосування систем машинного зору. Луже часто загальне освітлення виробничого приміщення, природне та штучне, не тільки ускладнює використання систем машинного зору а і унеможливилює отримання якісного зображення досліджуваного об'єкта.

**Локальне освітлення (підсвічування)** – це складова без якої практично не обходиться жодна система машинного зору. Необхідність її застосування обґрунтована багатьма факторами, такими як:



- слабкий рівень загального освітлення на підприємстві;
- низька частота мерехтіння загального освітлення (50 Гц);
- неможливість коригування кута падіння променів;
- неможливість управляти інтенсивністю освітленості;
- загальна освітленість може змінюватися протягом дня.

**Загальне освітлення** дуже часто негативно впливає на роботу камери машинного зору зокрема через:

– Імпульсний режим роботи люмінесцентних світильників. Частота мерехтіння таких ламп становить 50 Гц. При роботі камери зі швидкістю в 20-100 fps буде спостерігатись явище однорідної освітленості, тобто частина знімків світліша, а інша частина – темніша. FPS (frames per second) – параметр, що вказує на швидкість зміни кадрів протягом однієї секунди. Для людського ока це не так помітно, оскільки людське око має інертні властивості і не сприймає більше ніж 25 fps. Камера машинного зору позбавлена таких недоліків і бачить рівно те, що є насправді.

– При формуванні зображення на сенсорі камери велике значення має кут падіння світлових променів. Цьому приділяється багато уваги під час лабораторних досліджень.

– Використовуючи загальне освітлення люмінесцентних ламп, неможливо регулювати кут падіння світлових променів, що позбавляє нас гнучкості і додаткових можливостей при побудові зображення.

– При використанні загального освітлення неможливо регулювати його потік. Погіршує ситуацію ще й приміщення у якому є джерела природного освітлення – вікна. Сонячне світло також впливає на підсумкову кількість світлового потоку, яку приймає матриця. Ситуація аналогічна мерехтінню люмінесцентної лампи, лише за тим винятком, що частота мерехтіння значно нижче. Знімки, отримані в обід, будуть світлішими в 3-5 раз, ніж знімки, отримані в нічний час.

Таким чином очевидна необхідність використання додаткових джерел світла для підсвічування аналізованих об'єктів.

**Промислове освітлення** можна класифікувати за такими характеристикам:

– спектр світлового потоку, що випромінюється пристроєм підсвічування;

– кут падіння світлового потоку і конструктивні особливості пристрою підсвічування.

У сучасному виробництві застосовують два типи джерел освітлення: лазер і світлодіод.

Світлодіод, в порівнянні з лазером, дешевший і більш гнучкий у використанні оскільки дозволяє побудувати практично будь-яку конструкцію пристрою підсвічування, встановивши в нього необхідну кількість світлодіодів.

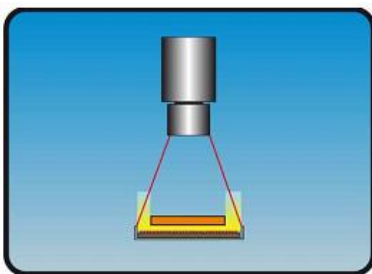
Лазер випромінює когерентний, потужний потік світла, що дозволяє, застосувавши різні фільтри-насадки, отримати “фігурне” (структуроване) освітлення.

Основними видами підсвічувань на сьогодні є [33]:

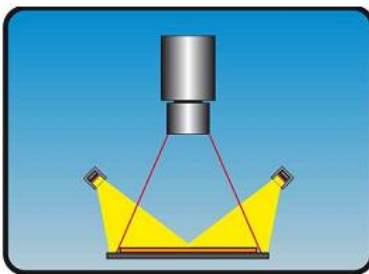
"Back Light" – прямокутне підсвічування (рис. 1.2) по всій поверхні об'єкта. Застосовується в задачах, коли об'єкт дослідження необхідно просвітити наскрізь.

"Line Light" – підсвічування для камер машинного зору з сенсором лінійного сканування (рис. 3). Застосовується для місцевого освітлення по всій ширині поверхні досліджуваного об'єкта.

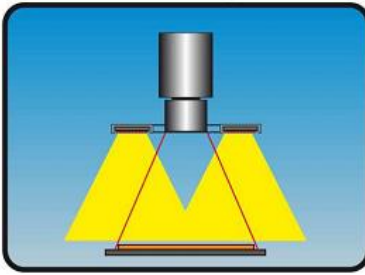
"Ring Light" – кільцеве підсвічування. Застосовується в задачах, де потрібно освітити об'єкт під кутом 90 градусів до поверхні досліджуваного об'єкта (рис. 4). Зручність застосування методу в тому, що пристрій підсвічування має отвір під об'єктив камери. Освітлення під кутом, відмінним від 90°, буде давати різну ступінь освітленості об'єкта в різних зонах.



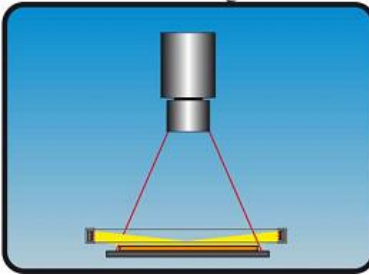
*Рис. 2. "Back Light"*



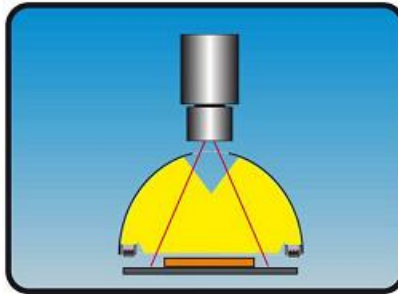
*Рис. 3. "Line Light"*



*Рис. 4. "Ring Light"*



*Рис. 5. "Dark Field Light"*



*Рис. 6. "Diffuse Light"*

"Dark Field Light" – підсвічування, що застосовується в задачах деталізації рельєфних текстур. Активну роль в ефективності даного підсвічування відіграє переломлення світлового потоку на рельєфній структурі досліджуваного об'єкта (рис. 5).

"Diffuse Light" – дифузійне підсвічування, кут розташування дозволяє виключити наявність тіней на об'ємних об'єктах, що досліджуються (рис. 6). Цей ефект досягається конструктивними особливостями пристрою підсвічування, що дозволяють світловому потоку потрапляти на об'єкт з усіх сторін.

### **3. Напрями машинного зору**

Класифікуючи задачі машинного зору, можна виділити наступні напрями:

- робота з одиночними зображеннями (кадрами);
- обробка відеопотоку:
  - стерео зір (stereo vision);
  - монокулярний зір (monocular vision);

- робота з хмарами точок (point clouds) та тримірним простором.

Кожен із напрямів передбачає володіння знаннями з базових алгоритмів обробки зображень і використовується для:

- Визначення:
  - Геометрії: форма, розмір;
  - Позиції та орієнтації (пози);
  - властивостей матеріалів: кольору, текстури, нерівностей (похибок);
- Виявлення та розпізнавання об'єктів:
  - облич, людей, рухів, дій тощо;
  - номерних знаків, площин, об'єктів на площинах;
  - промислових частин тощо.

Умовно розпізнавання, в залежності від задачі можна розділити на наступні під етапи: 1) фільтрація зображення (Filtering) → 2) виявлення ключових елементів/характеристичних точок (Key points / Features detection) → 3) розпізнавання об'єктів (Objects recognition).



**Фільтрація зображення (Filtering)** – це комплекс математичних операцій, спрямованих на зменшення впливу різного роду завад (рис. 7), наприклад, нерівномірності освітлення, аберацій, цифрового шуму на вихідне зображення (рис. 1.8).



**Цифровий шум** – це дефект зображення (рис. 7, рис.8, а), що внесений фотосенсором і електронними компонентами пристроїв, які є основною складовою цифрових фото- та відеокамер внаслідок недосконалості технологій їх виготовлення, а також фотонної природи світла.



*Рис. 7. Приклад впливу цифрового шуму на зображення*

Базовими методами фільтрації зображень є: лінійна, медіанна, Гауса, білатеральна тощо [18, 10].



*а)*

*б)*

*в)*

*Рис. 8. Результат застосування фільтрації для подавлення цифрового шуму: а) вихідне зображення; б) фільтр Гауса; в) медіанний фільтр*

Фільтра Гауса виконує згортку зображення із двовірним ядром Гауса яке описується виразом:

$$G_{\sigma}(x,y)=\frac{1}{2\cdot\pi\cdot\sigma^2}\cdot e^{-\frac{x^2+y^2}{2\cdot\sigma^2}}. \quad (1)$$

Графічна інтерпретація функції Гауса представлена на рис. 9.

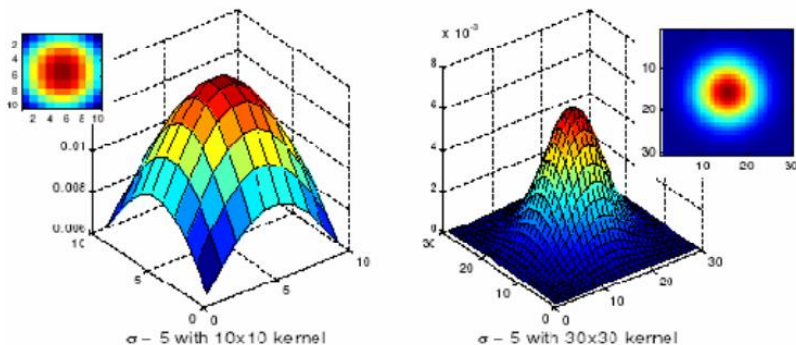


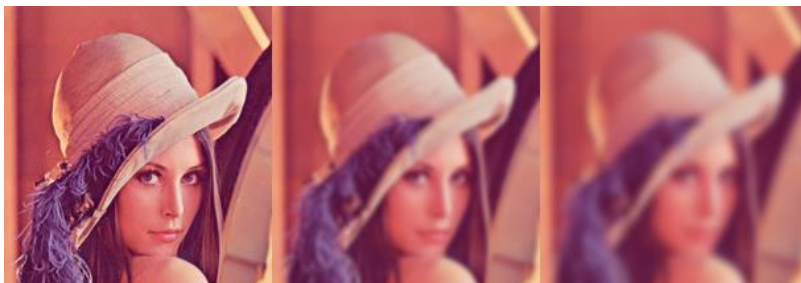
Рис. 9. Графічна інтерпретація функції Гауса  
(за К. Grauman)

Із рис. 9. видно, що для інтенсивність розмиття буде змінюватись із зміною розміру ядра.

Особливістю фільтра Гауса є його сепарабельність, тобто даний двомірний фільтр можна розділити на два одномірних і застосувати їх по черзі для обробки зображення по координаті  $x$  та  $y$ . Вираз функції в такому випадку буде мати вигляд:

$$G_{\sigma}(x) = \frac{1}{\sqrt{2 \cdot \pi \cdot \sigma}} \cdot e^{-\frac{x^2}{2 \cdot \sigma^2}} \text{ та } G_{\sigma}(y) = \frac{1}{\sqrt{2 \cdot \pi \cdot \sigma}} \cdot e^{-\frac{y^2}{2 \cdot \sigma^2}}. \quad (2)$$

Фільтр Гауса є фільтром низьких частот основною перевагою якого є можливість гнучкого налаштування параметрів, таких як розмір ядра та  $\sigma$ , що якісно впливає (рис. 10) на результуюче зображення порівняно із медіанним, білатеральним та лінійним фільтрами.



а)

б)

в)

Рис. 10. Результат зміни розміру ядра Гауса при фільтрації зображення: а) оригінал зображення; б) розмір  $5 \times 5$ ; в) розмір  $10 \times 10$

Після фільтрації зображення виконується виявлення ключових точок.



**Виявлення ключових точок / особливостей (Key points/Features detection)** – це процес аналізу зображення спрямований на обчислення абстракцій інформації зображення, та прийняття локальних рішень в кожній точці даного зображення щодо наявності ознаки зображення заданого типу в цій точці. Тобто, чи є ознака зображення в точці  $a_i \{x_i, y_i\}$ , чи ні.

Немає універсального або точного визначення того, що являє собою особливість/ключова точка, її точне визначення часто залежить від конкретної задачі або області застосування. Проте для усвідомлення термінологічного апарату машинного зору під особливостями слід розуміти:



**Особливості (Features)** – це невеликі області (patches) зображення, що використовуються для визначення подібності між зображеннями (image similarity). Особливості зображення (image features) зазвичай складаються із ключових точок (key points) та дескрипторів (descriptors) властивостей.

В цьому контексті ключові точки та дескриптори набувають наступного змісту:



**Ключова точка (Key point)** складається, як правило, із 2D області та інших складових, що притаманні цій точці, наприклад коефіцієнти масштабування, зміни орієнтації особливостей зображення тощо.

**Дескриптор (Descriptor)** містить візуальний опис області і використовується для порівняння “подібності” між різними особливостями зображення.

Отримані особливості є підмножинами області визначення зображення, часто в ролі особливостей виступають ізольовані точки, кути, неперервні криві або зв'язані області. Відомими на сьогодні методами визначення особливостей є: детектор кутів Харріса (Harris corner detector), детектор границь Кенні (Canny edge detector), масштабонезалежне перетворення ознак (SIFT – Scale-Invariant Feature Transform) тощо.

**Розпізнавання об'єктів (Object recognition)** – це комп'ютерна технологія, пов'язана з комп'ютерним зором і обробкою зображень, основним завданням якої є виявлення частин (складових) семантичних об'єктів певного класу (таких як обличчя, люди, будівлі, автомобілі тощо) в цифрових фото- та відео зображеннях. Іншими словами:



**Розпізнавання** – це віднесення досліджуваного об'єкта до певного класу з врахуванням ключових точок або його особливостей.

Різновидами задач розпізнавання є:

- виявлення (detection);
- ідентифікація (identification);
- розпізнавання (recognition).

На сьогодні відомими методами, що для цього використовуються є:

- розпізнавання об'єктів з використанням примітивів Хаара;
- розпізнавання об'єктів за його характеристичними (ключовими) точками.
- розпізнавання кінематичних характеристик об'єкта по розмиттю його зображення;



#### 4. Розпізнавання за особливостями / ключовими точками

Розпізнавання об'єктів за ключовими точками передбачає пошук особливостей та ключових точок на зображенні та їх семантичного аналізу чи співставлення їх із деяким еталонним зображенням.

Особливостями зображення можуть бути: кути, криві, границі, форми, та ключові точки. Враховуючи це, задачу розпізнавання умовно розділяють на три етапи:

Етап I – визначення особливостей/ключових точок на еталонному зображенні або об'єкті.

Етап II – визначення особливостей/ключових точок на робочих зображеннях.

Етап III – співставлення ключових точок двох зображень (еталону та робочого зображення) та прийняття рішення щодо наявності еталону на зображенні. При позитивному результаті можна вважати, що еталонний об'єкт знайдено та виділено на зображенні (рис. 5.11, б)



а)

б)

в)

Рис. 11. Візуалізація отриманого результату розпізнавання об'єкта на зображенні: а) еталонне зображення; б) робоче зображення; в) результат розпізнавання виділено зеленою рамкою

В якості ключових точок зображення можуть виступати будь-які особливості притаманні конкретному зображенню. Такими

особливостями, наприклад, можуть бути: кути, границі об'єктів на зображенні, значення градієнтів та їх напрям в околі точки тощо.

Для визначення особливостей та ключових точок на сьогодні широко використовуються наступні методи:

- детектор границь Кенні (Canny Edge Detector);
- детектори кутів: Моравеца (Moravec Corner Detector), Shi-Tomasi, SUSAN (Smallest Univalued Segment Assimilation Nucleus), Харріка (Harris Corner Detector) та інші;
- детектори блобів (Blob Detectors).

Методи виявлення **блобів** спрямовані на виявлення регіонів у цифровому зображенні, що відрізняються за властивостями, такими як яскравість або колір, у порівнянні з оточуючими його регіонами.

Під **блобом** розуміють область зображення, в якій деякі властивості є постійними або наближені до постійних; всі точки блобу можна вважати в певному сенсі схожими між собою. Найпоширенішим методом виявлення блобів є згортка.

Для більш складних випадків, коли особливість описується не просто ключовою точкою у вигляді кута чи границі зображення, вводять поняття дескриптор.



**Дескриптор ключової точки / особливості** – це ідентифікатор, який виділяє її із масиву усіх інших особливих точок.

У свою чергу, дескриптори повинні забезпечувати інваріантність до:

- масштабування;
- повороту;
- зсуву;
- зміни яскравості;
- зміни положення камери.

Існує ряд методів, які дозволяють отримати дескриптори особливостей, а саме:

- SIFT (Scale-Invariant Feature Transform) – масштабонезалежне перетворення особливостей;
- SURF (Speeded Up Robust Features) – прискорення реалізація попереднього методу;
- LIFT (Learned Invariant Feature Transform) – інтелектуалізований

- гістограма направлених градієнтів (Histogram of Oriented Gradients, HOG).

#### 4.1. Scale-Invariant Feature Transform

Основна ідея методу представлена David G. Lowe [9] і полягає у побудові піраміди гаусіан (Gaussian) та піраміди різниць гаусіан (Difference of Gaussian, DoG). Гаусіаном  $L$  є зображення  $I$ , що розмите фільтром Гауса  $G$  та представлене у вигляді:

$$L(x, y, \sigma) = G(x, y, \sigma) * I(x, y), \quad (5.3)$$

де  $L$  – значення гаусіана в точці зображення з координатами  $(x, y)$ ;

$G$  – ядро Гауса;

$\sigma$  – радіус розмиття ядра Гауса;

$I$  – значення вихідного зображення;

$*$  – операція згортки.

В загальному випадку, інваріантність до масштабування досягається за рахунок знаходження ключових точок для оригіналу зображення, отриманого при різних коефіцієнтах масштабування. Для цього будується піраміда гаусіан.

Різницею гаусіан  $D$  є зображення, що отримане шляхом попиксельного віднімання одного гаусіана вихідного зображення від гаусіана з іншим радіусом розмиття:

$$\begin{aligned} D(x, y, \sigma) &= (G(x, y, k \cdot \sigma) - G(x, y, \sigma)) * I(x, y) \\ &= L(x, y, k \cdot \sigma) - L(x, y, \sigma). \end{aligned} \quad (4)$$

**Масштабований простір зображення** – це набір усіх можливих, згладжених деяким фільтром, варіантів оригіналу зображення.

Масштабований простір зображення є лінійним, інваріантним до зсуву, обертання, масштабу, що не зсуває локальні екстремуми, і має властивість напівгруп. Важливим є те, що різна ступінь розмиття зображення фільтром гауса може бути прийнята за вихідне зображення, взяте в деякому масштабі.

Загалом, інваріантність до масштабування досягається за рахунок знаходження ключових точок для вихідного зображення, взятого в різних масштабах. Для цього будується піраміда гаусіан.

### **Методика побудови піраміди гаусіан:**

1. Весь простір, що масштабується розбивається на деякі ділянки – октави, причому частина масштабованого простору, що займається наступною октавою, в два рази більше частини, що займається попередньою.
2. При переході від однієї октави до іншої виконується зміна розміру зображення (ресемплінг), його розміри зменшуються вдвічі. Природно, що кожна октава охоплює безліч гаусіан зображення, тому будується тільки деяка їх кількість ( $N$ ), з певним кроком по радіусу розмиття.
3. З тим же кроком добудовуються два додаткових гаусіана (всього отримуємо  $N+2$ ), що виходять за межі октави. Проте це необхідно для перевірки зображення на наявність локальних екстремумів
4. Масштаб першого зображення наступної октави дорівнює масштабу зображення з попередньої октави з номером  $N$ .

Одночасно із побудовою піраміди гаусіан виконується побудова піраміди різниць гаусіан (DoG), що відповідно до (5.4) складається із різниці сусідніх зображень в піраміді гаусіан (рис. 5.12). Відповідно, кількість зображень в цій піраміді буде  $N+1$ .

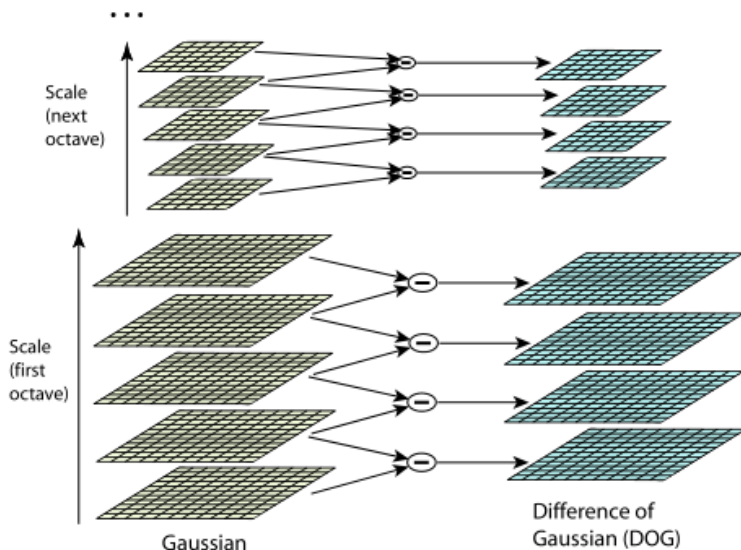


Рис. 5.12. Графічне представлення побудови піраміди гаусіан та піраміди різниць гаусіан [9]

На (рис. 5.12) зліва зображена піраміда гаусіан, а справа – DoG. Схематично показано, що кожна DoG отримується з двох сусідніх гаусіан, кількість DoG на одиницю менше ніж кількість гаусіан, при переході до наступної октави розмір зображень зменшується вдвічі

Після побудови піраміди (рис. 5.12) ключовою точкою/особливістю вважають ту точку, що є локальним екстремумом (рис. 5.13) DoG. Наприклад, ключова точка, що позначена “X” (рис. 5.13) буде вважатись локальним екстремумом якщо значення різниці гаусіан в цій точці буде меншим, чи більшим за інші в межах піраміди DoG.

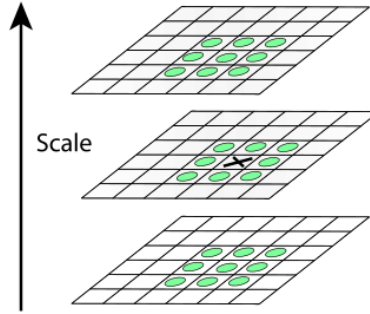


Рис. 5.13. Графічна інтерпретація процесу визначення локальних екстремумів різниці гаусіан [9]

**Пошук екстремуму** виконується порівнянням поточної точки  $i_n(x,y)$  |  $n = 1-N$ , де  $n$  – порядковий номер DoG в піраміді,  $N$  – загальна кількість DoG в піраміді, з її сусідами (рис. 5.14) в межах поточної ( $n$ -ої) DoG та з усіма сусідніми точками в межах піраміді DoG (рис. 5.13).

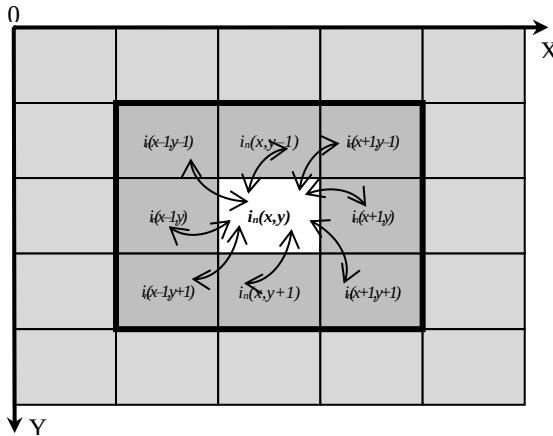


Рис. 5.14. Графічна інтерпретація пошуку локального екстремуму в межах  $n$ -ої DoG

Для того, щоб перевірити наявність точок екстремуму  $n$ -те зображення в DoG піраміді, потрібно мати  $(n+1)$ -ше зображення. А для того, щоб отримати  $(n+1)$ -ше в DoG піраміді, треба мати  $(n+1)$ -ше та  $(n+2)$ -ге зображення в піраміді гаусіан. Отже, можна сказати, що потрібно будувати 0-ве зображення (для перевірки 1-го) в піраміді DoG і ще два в піраміді гаусіан. Але 1-ше зображення в

поточній октаві має той же масштаб, що і  $n$ -те в попередній (яке вже має бути перевірено).

Після визначення точок екстремуму виконується ряд перевірок на відповідність цих точок умовам ключових.

По-перше визначаються координати особливої точки з субпіксельною точністю. Це досягається шляхом апроксимації функції DoG (5.4) багаточленом Тейлора другого порядку, що взятий в точці екстремуму.

$$D(x) = D + \frac{\partial D^{-1}}{\partial x} x + \frac{1}{2} x^T \frac{\partial^2 D}{\partial x^2} x, \quad (4)$$

де  $D$  – функція DoG;

$X$  – вектор зміщення відносно точки розкладення,  $X=(x,y,\sigma)$ .

Перша похідна DoG – градієнт, а друга похідна DoG – матриця Гессе.

Екстремум багаточлена Тейлора отримується шляхом визначення похідної та прирівнювання її до нуля. В результаті отримується (5.5) зміщення точки екстремуму відносно точного:

$$\hat{x} = -\frac{\partial^2 D^{-1}}{\partial x^2} \cdot \frac{\partial D}{\partial x}. \quad (5)$$

Всі похідні обчислюються за формулами кінцевих різниць. Як результат – отримується система лінійних алгебраїчних рівнянь розмірності  $3 \times 3$  відносно компонента вектора  $\hat{x}$ . Якщо одна із компонент вектора  $\hat{x}$  задовольняє умові (6), то це значить, що точка екстремуму визначена невірно і необхідно зсунутись до сусідньої точки у напрямку вказаних компонент вектора  $\hat{x}$ .

$$\hat{x} > 0,5 \cdot t, \quad (6)$$

де  $t$  – крок сітки у напрямку компоненти вектора  $\hat{x}$ .

Для сусідньої точки все повторюється. Якщо виконання такої перевірки змушує вийти за межі октави, то таку точку слід виключити із розгляду.

Коли положення точки екстремуму визначено, то перевіряється на мінімальність саме значення DoG в цій точці за виразом (7).

$$D(\hat{x}) = D + \frac{1}{2} \cdot \frac{\partial D^{-1}}{\partial x} \cdot \hat{x}. \quad (7)$$

Якщо ця перевірка не задовольняє заданим умовам, то точка включається із переліку оскільки у неї малий контраст.

**Фільтрація ключових точок.** На етапі фільтрації виконується видалення зі списку особливих точок тих, що мають низьку контрастність відносно сусідніх точок або сконцентровані на границях об'єктів. Ці точки мають великий вигин однієї із компонент другої похідної вздовж границі та малий вигин у перпендикулярному напрямку. Цей вигин визначається матрицею Гессе. Для перевірки та відсіювання таких точок використовується матриця  $H$  (8) розміром  $2 \times 2$ .

$$H = \begin{bmatrix} D_{xx} & D_{xy} \\ D_{xy} & D_{yy} \end{bmatrix}. \quad (8)$$

Нехай  $Tr(H)$  – слід матриці (5.9),  $Det(H)$  – її визначник (5.10), а  $r$  – відношення більшого вигину до меншого (5.11).

$$Tr(H) = D_{xx} + D_{yy} = \alpha + \beta, \quad (9)$$

$$Det(H) = D_{xx}D_{yy} - (D_{xy})^2 = \alpha \cdot \beta. \quad (10)$$

$$\alpha = r \cdot \beta. \quad (11)$$

Тоді, беручи до уваги вирази (5.7), (5.8), та (5.9) отримується співвідношення (5.12).

$$\frac{Tr(H)^2}{Det(H)} = \frac{(\alpha + \beta)^2}{\alpha \cdot \beta} = \frac{(r \cdot \beta + \beta)^2}{r \cdot \beta^2} = \frac{(r+1)^2}{r}. \quad (12)$$



Точка приймається і розглядається далі, якщо виконується умова (5.13).

$$\frac{Tr(H)^2}{Det(H)} < \frac{(r+1)^2}{r}. \quad (13)$$

**Визначення орієнтації ключової точки.** На наступному етапі виконується визначення орієнтації ключової точки.

Напрямок градієнта ключової точки обчислюється із напрямів градієнтів точок сусідніх з ключовою. Всі обчислення градієнтів виконуються на зображенні в піраміді гаусіан з масштабом, що найбільш близький до масштабу ключової точки.

У точки може бути декілька напрямів. Для розрахунку напрямів береться кілька сусідніх точок і обчислюються їх градієнти. Найчастіше в якості сусідів приймають квадрат (вікно) розміром, наприклад,  $5 \times 5$  точок, центром якого є обрана ключова точка. Для кожної точки обчислюється її градієнт і напрямок градієнта.



**Градiєнт** – це вектор, що своїм напрямком вказує напрямок найбільшого зростання деякої величини, значення якої змінюється від однієї точки простору до іншої (скалярного поля), а по величині (модулю) дорівнює швидкості зростання цієї величини в цьому напрямку.

Величина (14) і напрям градієнта (15) визначається за відповідними виразами:

$$m(x, y) = \sqrt{(L(x+1, y) - L(x-1, y))^2 + (L(x, y+1) - L(x, y-1))^2}, \quad (14)$$

де  $m$  – величина градієнта.

$$\theta(x, y) = \arctan\left(\frac{L(x, y+1) - L(x, y-1)}{L(x+1, y) - L(x-1, y)}\right), \quad (15)$$

де  $\theta(x, y)$  – величина кута, що вказує на орієнтацію градієнта. Кінцевий напрям ключової точки визначається із гістограми

напрямів ключових точок вікна. Гістограма (рис. 15) містить 36 компонент, які рівномірно охоплюють весь інтервал  $0^\circ - 360^\circ$ . Кожна точка вікна впливає на ту компоненту гістограми, яка охоплює проміжок, що містить напрям градієнта  $\theta(x,y)$  наступним чином:

$$\theta_i(x_i, y_i) = \theta(x, y) \cdot m(x, y) \cdot G(x, y, \sigma). \quad (17)$$

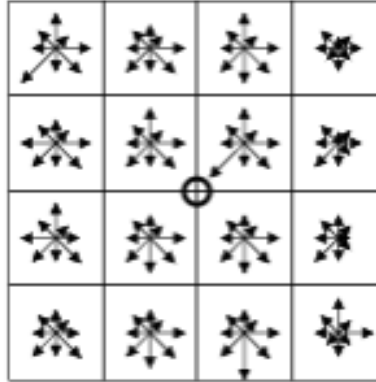


Рис. 15. Приклад ключової точки з гістограмами 16 точок

Напрямок ключової точки знаходиться у проміжку, що охоплений максимальною компонентою гістограми. Якщо в гістограмі є додатково компоненти з величинами не менше 80% від максимальної величини, то вони приймаються як додаткові напрями ключової точки.

**Побудова дескрипторів.** Перед обчисленням дескриптора обирається кількість точок в околі ключової, що необхідно врахувати. Найчастіше враховується 16 точок. На наступному кроці для кожної точки будується гістограма, що аналогічна гістограмі орієнтації ключової точки але з меншою кількістю компонент (8 замість 36). Після цього формується вектор із 128 чисел, тобто  $16 \text{ гістограм} \times 8 \text{ компонент у кожній гістограмі}$ . Цей вектор і є дескриптором SIFT.

## 4.2. Перетворення Хафа

Часто, для повноцінного аналізу зображення перед використанням складних методів розпізнавання та неймережевих підходів необхідно виявити і, при необхідності, виключити прості елементи, якими є лінії, кола, еліпси, трикутники, багатокутники тощо.

Для реалізації такої задачі використовуються наступні методи:

- Перетворення Хафа;
- Перетворення Радона;
- Перетворення Фур'є.

На сьогодні, найпоширенішим інструментарієм для автоматизованого виконання операції детектування та розпізнавання об'єктів є перетворення Хафа та його модифікації, такі як Узагальнене перетворення Хафа, прогресивно-імовірнісне перетворення хафа (Progressive Probabilistic Hough Transform), рандомізоване перетворення Хафа, ісрархічне перетворення Хафа та ін..



**Перетворення Хафа (Hough Transform)** – це алгоритм, чисельний метод, що запропонований та запатентований Полем Хафом у 1962 році [6]. Перетворення Хафа застосовується для виокремлення елементів зображення і призначене для пошуку об'єктів, що належать певному класу фігур, з використанням процедури голосування. Такими класами можуть бути: лінії, кола, еліпси тощо, тобто такі фігури, що можуть бути описані аналітичними виразами.

Класичне перетворення Хафа було запропоноване для виділення на бінарному зображенні прямих ліній. Воно ґрунтується на використанні простору параметрів, в якому і проводиться пошук прямих. Найбільш поширеним аналітичним описом прямої є рівняння виду:

$$y = mx + b, \quad (18)$$

де  $y$  та  $x$  – координати точки, що належить прямій;

$m$  – кутовий коефіцієнт (коефіцієнт нахилу), що дорівнює тангенсу кута, який утворений цією прямою і додатнім напрямом осі  $OX$ .

Головна ідея перетворення Хафа полягає у врахуванні характеристик прямої не як рівняння, що побудоване по парі точок  $(x, y)$  зображення, а в термінах її параметрів, тобто  $m$  – коефіцієнта нахилу і  $b$  – точки перетину з віссю ординат. Виходячи з цього пряма, яка задана рівнянням (18), може бути представлена у вигляді точки з координатами  $(b, m)$  в просторі параметрів.

Проте таке представлення має ряд недоліків, зокрема пряма, що паралельна осі ординат має нескінченні значення параметра  $m$ , тому зручніше представляти пряму у вигляді рівняння (19).

$$r = x \cdot \cos \theta + y \cdot \sin \theta, \quad (19)$$

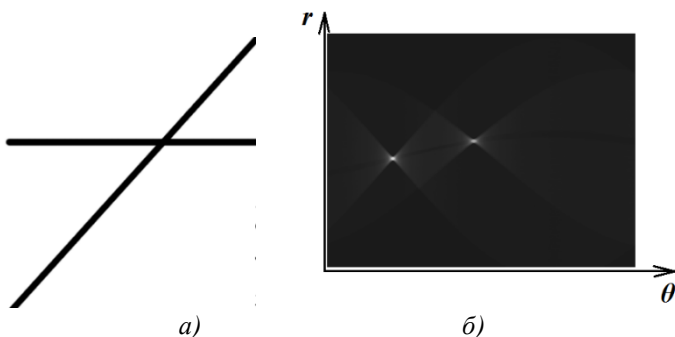
де  $r$  – довжина радіус-вектора, найближчого до початку координат точки на прямій (тобто нормалі до прямої, що проведена із початку координат);

$\theta$  – кут між радіус-вектором  $r$  та віссю абсцис

Тому, можливо з кожною прямою на зображенні (в площині  $X-Y$ ) пов'язати точку з координатами  $(r, \theta)$  в площині параметрів, яка є унікальною при умові, що  $\theta \in [0, \pi]$  та  $r \in \mathbb{R}$ , або  $\theta \in [0, 2\pi]$  та  $r \geq 0$ .



**Площина параметрів  $(r, \theta)$  прямої називається Простором Хафа, що графічно може бути зображений у вигляді множини кривих (синусоїд) (рис. 16, б).**



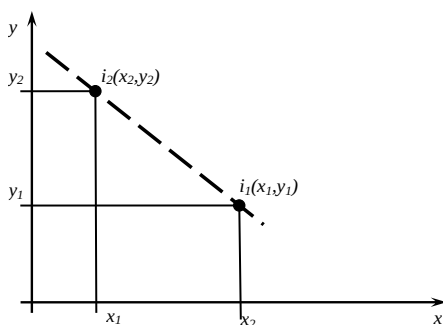
*Рис. 16. Графічна ілюстрація простору Хафа для знаходження двох прямих, що перетинаються: а) вхідне зображення, б) представлення простору Хафа для вхідного зображення*

### **Робота алгоритму перетворення Хафа.**

Алгоритм перетворення Хафа використовує масив, *акумулятор*, для виявлення прямої, яка описується виразом (19), на зображенні.

Розмірність акумулятора дорівнює кількості невідомих параметрів простору Хафа, для лінії це 2, оскільки невідомими параметрами простору Хафа є 2:  $m$  та  $b$ . Дві розмірності акумулятора відповідають дискретизованим значенням параметрів  $m$  і  $b$ .

Нехай, пряма (рис. 17) розглядається як множина точок  $i$  у вихідному просторі зображення.



*Рис. 17. Графічна інтерпретація перетворення Хафа: пряма, що представлена множиною точок*

Очевидно, що через кожну точку можна провести (рис. 18) нескінчену кількість прямих.

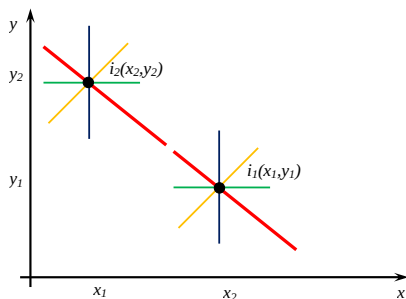


Рис. 18. Графічна інтерпретація перетворення Хафа: множина прямих проведених через кожну точку

При цьому, множину прямих, що проходять через точку  $i(x, y)$  можна представити множиною точок  $(r, \theta)$  у просторі параметрів.

Ідея перетворення Хафа полягає у тому, що для кожної точки простору параметрів підсумовується кількість голосів, що подані за неї, тобто, кількість точок вихідного простору, що породжують у просторі параметрів відклики, які проходять через дану точку  $(r_i, \theta_i)$ . Тут використовується той факт, що будь-які дві синусоїди у просторі параметрів перетнуться в точці  $(r, \theta)$  лише тоді, коли точки, у просторі координат, що породили ці синусоїди лежать на одній прямій, яка описується рівнянням (19) з параметрами  $(r, \theta)$ . Введена таким чином функція  $A(r, \theta)$  називається **аккумуляторною функцією**, причому абсолютне значення її в точці  $(r_i, \theta_i)$  дорівнює кількості точок, що лежить на одній у вихідному просторі зображення.

У тому випадку, коли на зображенні представлено  $t$  прямих, аккумуляторна функція  $A(r, \theta)$  буде мати рівно  $t$  локальних максимумів в точках, що відповідають наявним прямим. Таким чином, для виявлення прямих на зображенні необхідно знайти всі локальні максимуми аккумуляторної функції.

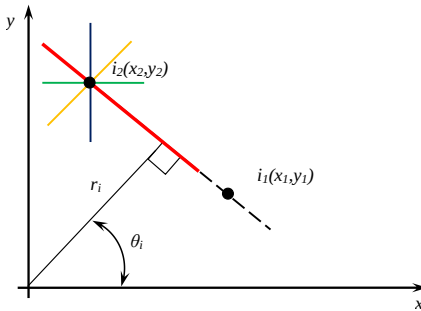
Як правило,  $A(r, \theta)$  обчислюється не для кожної точки простору параметрів, а для кожної "комірки акумулятора", тобто деякої прямокутній області (рис. 19), на які розбивається простір параметрів і розмір яких обмежує точність обчислень половинним значенням дискрети розбиття по кожному з параметрів.

|                |           |           |           |       |           |           |           |           |
|----------------|-----------|-----------|-----------|-------|-----------|-----------|-----------|-----------|
| $\theta_{i+4}$ | 0         | 0         | 0         | 0     | 0         | 0         | 0         | 0         |
| $\theta_{i+3}$ | 0         | 0         | 0         | 0     | 0         | 0         | 0         | 0         |
| $\theta_{i+2}$ | 0         | 0         | 0         | 0     | 0         | 0         | 0         | 0         |
| $\theta_{i+1}$ | 0         | 0         | 0         | 0     | 0         | 0         | 0         | 0         |
| $\theta_i$     | 0         | 0         | 0         | 0     | 0         | 0         | 0         | 0         |
| $\theta_{i-1}$ | 0         | 0         | 0         | 0     | 0         | 0         | 0         | 0         |
| $\theta_{i-2}$ | 0         | 0         | 0         | 0     | 0         | 0         | 0         | 0         |
| $\theta_{i-3}$ | 0         | 0         | 0         | 0     | 0         | 0         | 0         | 0         |
|                | $r_{i-3}$ | $r_{i-2}$ | $r_{i-1}$ | $r_i$ | $r_{i+1}$ | $r_{i+2}$ | $r_{i+3}$ | $r_{i+4}$ |

Рис. 19. Акумуляторний простір

На початку роботи алгоритму, акумуляторний простір (рис. 19) дискретизується і заповнюється нулями.

Для кожної точки  $i(x, y)$  зображення знаходяться значення  $(r, \theta)$ , що ідентифікують лінію, яка відноситься до цієї точки (рис. 20). При цьому, значення у комірці акумулятора, яка відповідає знайденим параметрам  $(r_i, \theta_i)$ , інкрементується.



|                |           |           |           |       |           |           |           |           |
|----------------|-----------|-----------|-----------|-------|-----------|-----------|-----------|-----------|
| $\theta_{i+4}$ | 0         | 0         | 0         | 0     | 0         | 0         | 0         | 0         |
| $\theta_{i+3}$ | 0         | 0         | 0         | 0     | 0         | 0         | 0         | 0         |
| $\theta_{i+2}$ | 0         | 0         | 0         | 0     | 0         | 0         | 0         | 0         |
| $\theta_{i+1}$ | 0         | 0         | 0         | 1     | 0         | 0         | 0         | 0         |
| $\theta_i$     | 0         | 0         | 0         | 1     | 0         | 0         | 0         | 0         |
| $\theta_{i-1}$ | 0         | 0         | 1         | 0     | 0         | 0         | 0         | 0         |
| $\theta_{i-2}$ | 0         | 0         | 0         | 0     | 0         | 0         | 0         | 0         |
| $\theta_{i-3}$ | 0         | 0         | 0         | 0     | 0         | 0         | 0         | 0         |
|                | $r_{i-3}$ | $r_{i-2}$ | $r_{i-1}$ | $r_i$ | $r_{i+1}$ | $r_{i+2}$ | $r_{i+3}$ | $r_{i+4}$ |

Рис. 20 Знаходження значень параметрів  $(r, \theta)$ , що ідентифікують точку  $i_1(x_1, y_1)$

Аналогічна операція (рис. 21) проводиться для всіх наступних точок зображення. При цьому, значення в комірці акумулятора, що відповідають дискретизованим параметрам  $(r_i, \theta_i)$  – інкрементуються, тим самим прямуючи до локального максимуму.

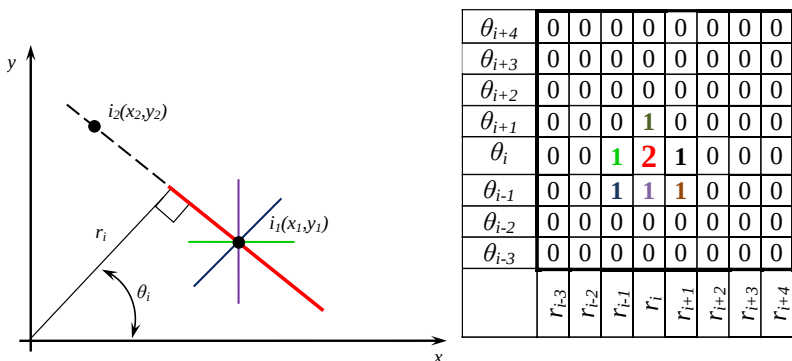


Рис. 21. Знаходження значень параметрів  $(r, \theta)$ , що ідентифікують точку  $i_2(x_2, y_2)$

Ілюстрація простору Хафа на основі акумулятора буде мати вигляд (рис. 22).

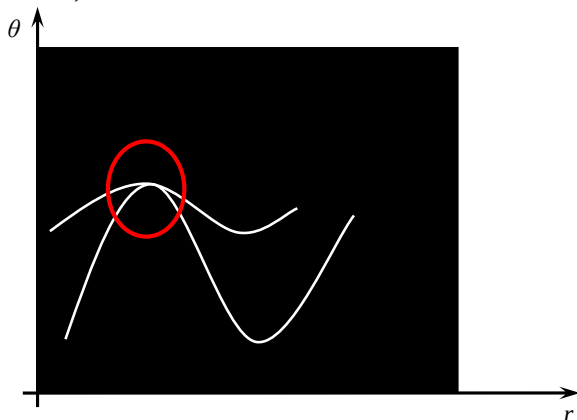


Рис. 22. Графічна ілюстрація результуючого простору Хафа

Із рисунку видно, що екстремум функції є перетином трьох синусоїд, що відповідає кількості голосів в акумуляторі у відповідній комірці.

За таким принципом окрім прямих можна виконувати пошук та детектування більш складних фігур, таких як кола, еліпси, трикутники тощо. Для їх пошуку необхідно виконання основної умови: фігури повинні мати аналітичний опис, що є досить суттєвим недоліком перетворення Хафа у класичному вигляді.



### 4.3. Узагальнене перетворення Хафа

Враховуючи неможливість абсолютно повно описати аналітичними виразами більшість об'єктів, що необхідно детектувати, Д.Х.Баллард у 1981 році запропонував [2] узагальнити перетворення Хафа *для пошуку об'єктів довільної форми*, у тому числі тих, що не мають аналітичного опису. Приймаючи до уваги даний підхід для виявлення об'єктів на зображенні необхідно провести порівняння еталону та зображення на якому він (еталон) знаходиться. Таке порівняння проводиться після виявлення ключових точок обох зображень шляхом перевірки на відповідність: чи відповідає  $i$ -та особливість еталону  $j$ -ій особливості зображення.

Таке порівняння виконується шляхом проведення голосування, що є невід'ємною складовою перетворення Хафа (Hough Transform – HT) представленого вище та його узагальнення (Generalized Hough Transform – GHT).

#### Загальний зміст алгоритму узагальненого перетворення Хафа

1. Вибрати точку відліку  $i_0(x_0, y_0)$  на еталонному зображенні (рис. 5.23) (будь-яка характерна точка, центр, верхній лівий кут зображення тощо);
2. Для кожної характеристичної точки  $i_e(x_e, y_e)$  еталону:
  - обчислити значення градієнта  $m$ ;
  - знайти вектор  $r = i_o - i_e$
3. Записати вектор  $r$  до R-таблиці:

| Напрямок градієнта               | Точки еталону:<br>Радіус-вектор / кут відносно<br>точки відліку |
|----------------------------------|-----------------------------------------------------------------|
| $0\Delta\varphi = 0^\circ$       | $(r, \beta)$                                                    |
| $1\Delta\varphi = 10^\circ$      | $(r, \beta)$                                                    |
| $2\Delta\varphi = 20^\circ$      | $(r, \beta) (r, \beta)$                                         |
| ...                              | $(r, \beta) (r, \beta)$                                         |
| $(k-1)\Delta\varphi = 170^\circ$ | $(r, \beta) (r, \beta)$                                         |
| $k\Delta\varphi = 180^\circ$     | $(r, \beta)$                                                    |

4. Завантажити вхідне зображення;

5. Створити акумуляторний масив;
6. Для кожної характеристичної точки  $j_k(x_k, y_k)$  вхідного зображення:
  - а) визначити напрям градієнта;
  - б) для кожного вектора  $r$  із відповідної комірки R-таблиці:
    - обчислити ймовірну точку відліку  $j_0(x_0, y_0)$  в залежності від  $\beta$  та  $r$ ;
    - збільшити значення акумулятора, що відповідає точці  $j_0(x_0, y_0)$ ;
7. Знайти максимальне значення акумулятора.

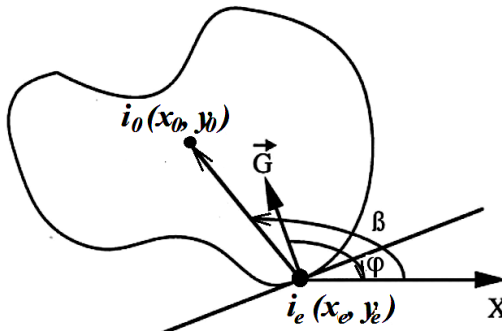


Рис. 23. Графічна ілюстрація пошуку параметрів для роботи GHT

Використання вищенаведеного алгоритму дозволяє детектувати об'єкти довільної форми, що не мають аналітичного опису. Але, за еталон може бути взято:

- зображення, яке попередньо оброблене будь-яким детектором границь, наприклад Кенні
- зображення, для якого застосований детектор особливостей, наприклад SIFT або SURF.

### **Використання алгоритмів SIFT та GHT для детектування об'єктів на зображеннях**

При використанні детектора SIFT до уваги приймаються параметри характеристичних точок: її координати та дескриптор, що її описує. В такому випадку, акумуляторний простір буде чотиримірним:  $A(d_x', d_y', \alpha, s)$  (рис. 24).

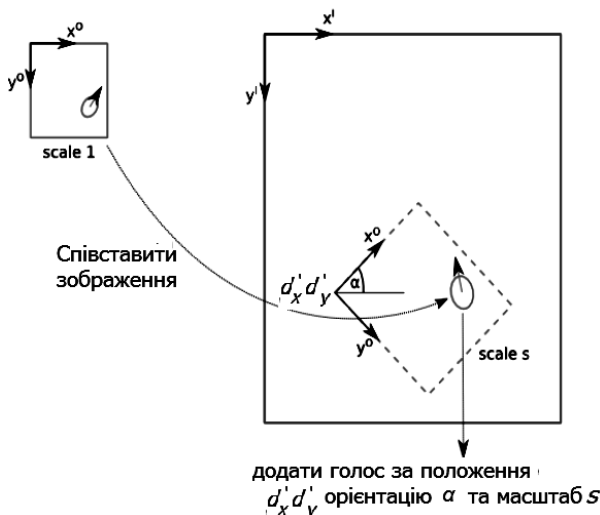


Рис. 24. Графічна ілюстрація процесу голосування з використанням SIFT та GHT

Алгоритм детектування об'єктів у такому випадку буде враховувати додаткові параметри: коефіцієнти масштабування  $s$  та кут повороту  $\alpha$ .

1. Вибрати точку відліку  $i_0(x_0, y_0)$  на еталонному зображенні (рис. 5.24) будь-яка характерна точка, центр, верхній лівий кут зображення тощо);
2. Для **кожної** характеристичної точки  $i_e(x_e, y_e)$  еталону:
  - обчислити значення градієнта  $m$ ;
  - знайти вектор  $r = i_o - i_e$
3. Записати вектор  $r$  до R-таблиці
4. Завантажити вхідне зображення;
5. Створити акумуляторний масив;
6. Для **кожної** характеристичної точки  $j_k(x_k, y_k)$  вхідного зображення:
  - а) визначити напрям градієнта;
  - б) для **кожного** вектора  $r$  із відповідної комірки R-таблиці:
    - для **кожного** допустимого кута повороту  $\alpha$ :
    - для **кожного** допустимого коефіцієнта масштабування  $s$ :

- обчислити ймовірну точку відліку  $j_0(x_0, y_0)$  в залежності від  $\beta$ ,  $r$ ,  $\alpha$ , та  $s$ ;
- збільшити значення акумулятора, що відповідає точці  $j_0(x_0, y_0)$ ;

7. Знайти максимальне значення акумулятора (рис. 25)

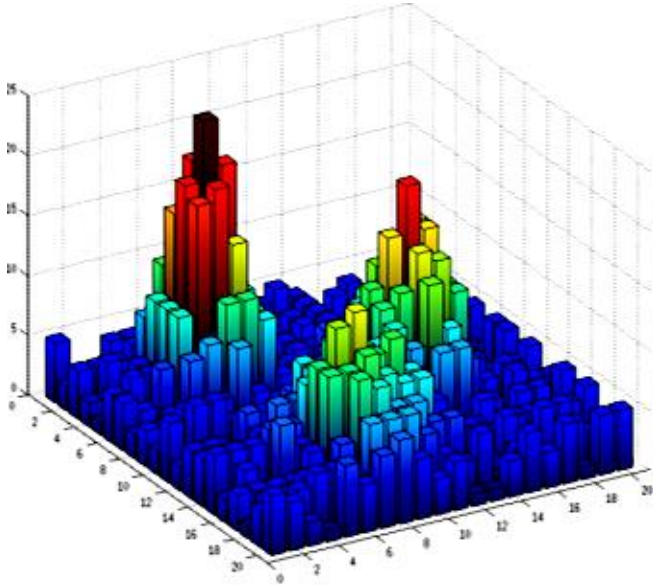


Рис. 25. Приклад простого 2-мірного акумуляторного простору голосування (наприклад, лише координати  $x$  та  $y$ )

Аналізуючи представлений на рис. 23 акумуляторний простір очевидно є наявність 3 локальних максимумів. Це свідчить, що на зображенні були виявлені 3 екземпляри еталону з різними конфігураціями.



Під конфігурацією простору голосування розуміють такі параметри  $d_x'$ ,  $d_y'$ ,  $\alpha$  та  $s$ , які визначають положення еталону в системі координат зображення.

## 5. Проективна геометрія та модель камери

### 5.1. Загальні поняття проективної геометрії

Більшість задач машинного зору неможливо розв'язати без знань порективної геометрії.

**Проективна геометрія** – розділ геометрії, що вивчає проективні площини і простори. Головною особливістю проективної геометрії є принцип подвійності, який додає витончену симетрію до більшості конструкцій.

Проективна геометрія може розглядатись як з чисто *геометричної* точки зору, так і з *аналітичної* (за допомогою однорідних координат), а також *алгебри*, розглядаючи проективну площину як структуру над полем. **Поле** в загальній алгебрі – множина, для елементів якої визначені операції додавання, взяття протилежного значення, множення і ділення (окрім ділення на нуль), причому властивості цих операцій близькі до властивостей звичайних числових операцій. Найпростішим полем є поле раціональних чисел (дробів). Хоча назви операцій поля взяті з арифметики, слід мати на увазі, що елементи поля не обов'язково є числами, і визначення операцій можуть бути далекі від арифметичних. Часто, і історично, натуральна проективна площина розглядається як Евклідова площина з додаванням «прямої у нескінченності».

Завданням проективної геометрії є вивчення властивостей фігур, інваріантних при множині проективних перетворень, а також самих цих перетворень. Проективна геометрія доповнює Евклідову, надаючи прості розв'язки для багатьох завдань, ускладнених присутністю паралельних прямих.

Проективна геометрія є основою машинного зору, адже дає пояснення щодо формування зображень навколишнього простору за допомогою камер та оптичних систем, наприклад монокулярних або стерео. Найпершою та найпростішою камерою є камера-обскура (рис. 26), основні параметри якої зображені на рис. 27.

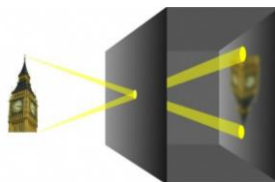


Рис. 26. Приклад камери-обскура



**Камера-обскура** (лат. *camera obscura* – «темна кімната») – це темне приміщення з одним малим отвором – *pinhole* (діаметром близько 0,3-1,0 мм), через який на протилежну стіну проектується перевернуте зменшене зображення предметів ззовні.

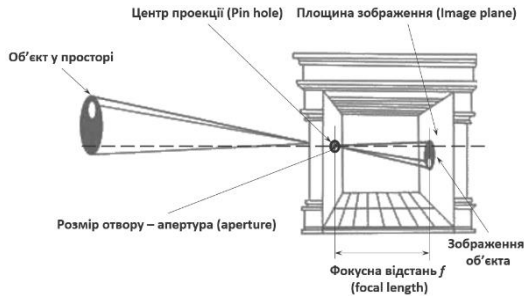
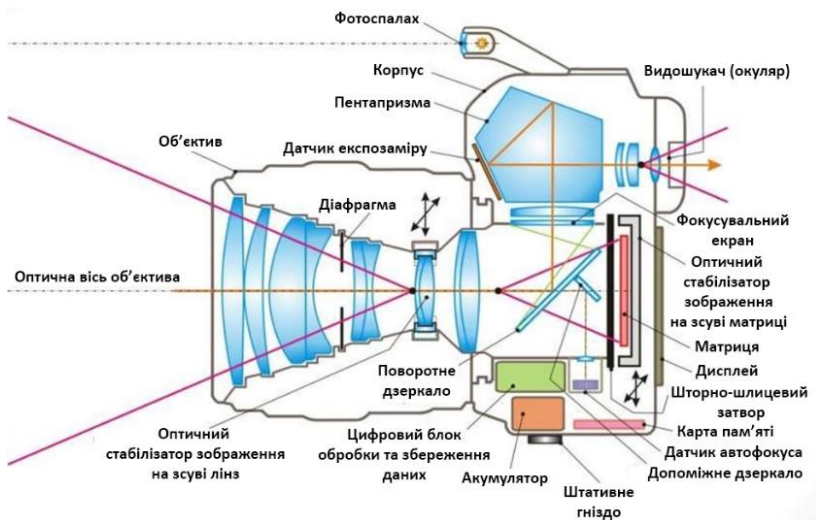


Рис. 27. Основні параметри камери-обскури

Принцип отримання зображень навколишнього простору камерою-обскура є основою сучасних оптичних систем, де апертура залежить від величини отвору затвору, та може бути змінною в одночас як і фокусна відстань. Для виробничих задач, у більшості випадків використовують камери із фіксованою фокусною відстанню, що спрощує обчислення. Сучасні цифрові камери (рис. 28) складаються із таких основних частин: об'єктив (оптична система або оптика), матриця, цифровий блок обробки та збереження даних, оптичні стабілізатори розб्राження.



*Рис. 28. Будова цифрової камери*

**Цифрова камера** – це оптико-механічний прилад, з електронним способом реєстрації, обробки і зберігання цифрових зображень, за допомогою якого проводиться фото- та відеозйомка.

Будова об'єктиву копіює людське око. Основним завданням оптики є фокусування світлового потоку, направленого від різних об'єктів і формування точки фокусу на світлочутливому матеріалі (матриці). У будові оптичної системи головну роль відіграють лінзи. Це скляні пластини круглої форми, яким за допомогою полірування і шліфування надається певна кривизна. Всі фотолінзи поділяються на наступні групи:

- опуклі;
- двоопуклі;
- увігнуті;
- двоввігнуті.

Залежно від направлення кривизни, лінзи можуть бути збиральними та розсіювальним. Перші дві групи є збиральними. Вони фокусують світловий потік в одну точку. Таким чином, роль оптичної системи може виконувати одна опукла фотолінза. Фотоб'єктив, що складається з однієї лінзи, називається меніск. Така найпростіша оптична система дозволяє отримати зображення предмета матриці, але воно буде невисокої якості, оскільки кривизна

оптичного скла змінюється від краю до центру, тому зображення об'єкта буде сильно спотворено по краях кадру.

## 5.2 Аберації та об'єктиви

Спотворення зображення називаються **абераціями**, які можуть бути хроматичними (спотворення кольору) і геометричними – дисторсія (рис. 29). При хроматичній аберації навколо предметів утворюється кольорова окантовка, що порушує нормальне сприйняття фотографії.



Рис. 29. Приклад аберації зображень:  
а) хроматична, б) геометрична (дисторсія)

**Хроматична аберація** – це різновид аберації оптичної системи, що обумовлений залежністю показника заломлення середовища від довжини хвилі, яка проходить через її випромінювання (тобто, дисперсією світла) (рис. 28).

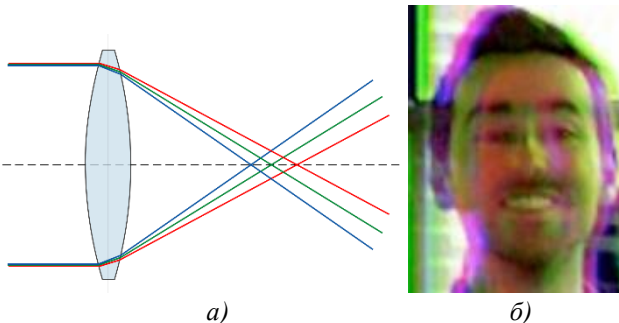


Рис. 28. Хроматична аберація: а) схематична інтерпретація сутності; б) приклад хроматичної аберації



Паразитна дисперсія призводить до того, що фокусні відстані не збігаються для променів світла з різними довжинами хвиль (променів різних кольорів). Розрізняють:

- хроматизм положення;
- хроматизм збільшення;
- хроматичні різниці геометричних аберацій.

Хроматичні аберації призводять до зниження чіткості зображення, а іноді також і до появи на ньому кольорових контурів, смуг, плям – артефактів (рис. 5.28, б).

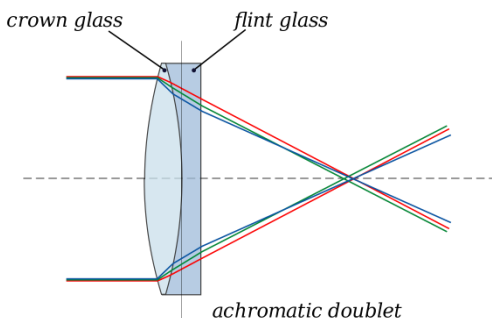
Геометричні аберації пов'язані з тим, що промені, які проходять через центральну і периферійну частину фотолінзи, переломлюються під різними кутами. Це призводить до викривлення прямих ліній або дисторсії, яка буває опуклою і увігнутою. Окрім дисторсії, бувають і інші види геометричних аберацій. Для того щоб позбутися від усіх видів аберацій, в оптичний набір фотоапарата встановлюють додаткові лінзи.

Із розвитком практичної оптики змінювалася оптична схема об'єктивів, і їх будова ставала все більш складною. Після меніска, що складається з однієї фотолінзи, були розроблені наступні варіації:

- ахромат;
- апланат;
- анастигмат.

**Ахромат** – це об'єктив, у якому виправлена хроматична аберация для променів світла двох різних довжин хвиль і частково сферична аберация. Оптичні системи з корекцією трьо і більше кольорів (довжинах хвиль) називаються *апохромат*.

У найпростішому випадку ахромат являє собою дублет (рис. 5.30, а), склеєний із двох лінз, одна з яких позитивна, а інша – негативна.



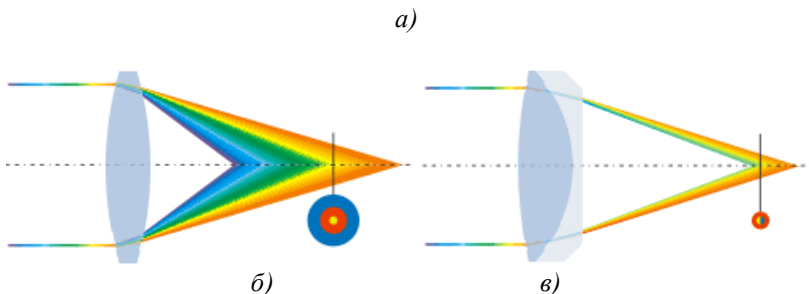


Рис. 30. Оптична схема ахромата (а) та хроматичні аберації на звичайній (б) та анахроматичній (в) лінзах

У таких випадках використовують лінзи, виготовлені з оптичних стекел з різною дисперсією. Позитивна виготовляється зі скла з великим (як правило, крона), а негативна – зі скла з меншим коефіцієнтом середньої дисперсії (як правило, флінта).



**Крон, кронглас** (від англ. crown – корона) – тип безбарвних оптичних стекел, що відрізняються великими (понад 50) значеннями коефіцієнта середньої дисперсії, що називається також числом Аббе. Такі стекла із відносно малими показниками заломлення називають легкими кронами, а з великими – важкими.



**Флінт** (від англ. flint – креміль), або **флінтглас** (від нім. flintglas) – тип безбарвних оптичних стекел, що відрізняються малими (менше 50) значеннями коефіцієнта середньої дисперсії, що називається також числом Аббе. Такі скла з відносно малими показниками заломлення називають легкими флінтами, а з великими – важкими

Для склеювання лінз застосовують оптичний клей (наприклад, канадський бальзам, смерековий бальзам або бальзамін). Склеювання лінз саме по собі ніяк не впливає на ахроматичні властивості, однак дозволяє зменшити відбиття світла від поверхонь лінз, знизити вимоги до точності виготовлення склеюваних поверхонь і полегшити подальший монтаж. Лінзи відносно великих

розмірів (з діаметром понад 10 см), як правило, не склеюють, оскільки через відмінності температурних коефіцієнтів розширення позитивної та негативної лінз, при збільшенні їх розмірів зростає ймовірність порушення цілісності склеювання, що відбувається при зміні температури навколишнього середовища.

Зведення зображення в ахроматі відбувається за двома променями, що знаходяться на краях спектрів: синього та червоного (рис. 30, б, в).

**Апланат** – об'єктив, у якому виправлені сферична і хроматична аберації, кома і дисторсія, а астигматизм виправлений для порівняно невеликого кутового поля. Апланат (рис. 5.31) складається з двох ахроматичних лінз, між якими розташована діафрагма.

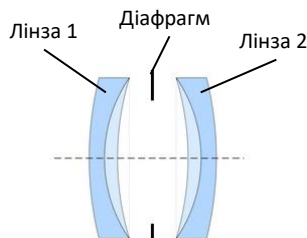


Рис. 31. Схематичне зображення апланата

**Комататична аберація** або **Кома** (від грец. Κόμη – волосся) – одна із п'яти аберацій Зейделя оптичних систем, що призводить до порушення гомоцентричності широких світлових пучків, які входять у оптичну систему під кутом до оптичної вісі (рис. 32).

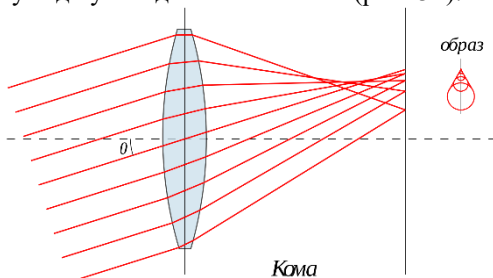


Рис. 32. Графічна інтерпретація коматичної аберації

Кожна ділянка оптичної системи, віддалена від її вісі на відстань  $d$  (кільцева зона), породжує зображення точки яка світиться у вигляді кільця, радіус якого тим більше, чим більше  $d$ ; може розглядатися як сферична аберація променів, що проходять не через оптичну вісь системи. Центри кілець не збігаються, в результаті чого їх накладення, тобто зображення точки, що дається системою в цілому, набуває вигляду несиметричної плями розсіювання, схожої

на комету. Цією схожістю обумовлено назву аберації. Розміри плями пропорційні квадрату кутової апертури системи і віддалення точки-об'єкта від осі оптичної системи.

**Сферична аберація** – це аберація оптичних систем яка викликана розбіжністю фокусів для променів світла, що проходять на різних відстанях від оптичної вісі. Приводить до порушення гомоцентричності пучків променів від точкового джерела, без порушення симетрії будови цих пучків (на відміну від коми та астигматизму). Розрізняють сферичну аберацію третього, п'ятого і вищого порядків.

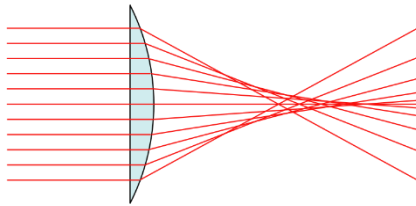


Рис. 5.33. Проходження пучків світла через лінзу зі сферичною аберацією

**Дисторсія** (від лат. *Distorsio, distortio* – викривлення) – це аберація оптичних систем, при якій коефіцієнт лінійного збільшення змінюється відповідно до віддалення предметів, що відображаються, від оптичної вісі. При цьому порушується геометрична подібність між об'єктом і його зображенням. Дисторсія неприйнятна в оптиці, призначеної для фотограмметричної аерофотозйомки і виготовлення фотошаблонів. Об'єктив з виправленою дисторсією називається ортоскопічним, оскільки задовольняє вимогам ортоскопічності.

Дисторсія виправляється на етапі розробки оптичної системи підбором лінз та інших елементів і / або шляхом обробки зображення на комп'ютері, внесенням відповідних поправок.

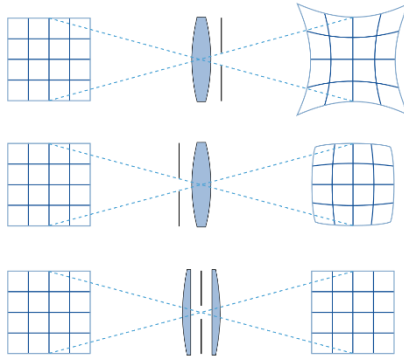


Рис. 34. Залежність дисторсії від розміщення апертурної діафрагми

Дисторсія буває радіальною і тангенціальною. **Радіальна** дисторсія симетрична відносно центра (властивість лінзи) **Тангенціальна** дисторсія несиметрична, викликана нахилом оптичної вісі відносно площини зображення. Їх враховують при побудові моделі камери у вигляді відповідних поправок (рис. 35).

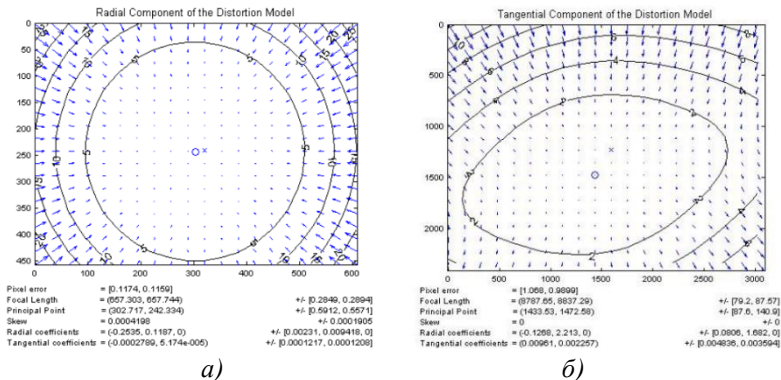


Рис. 35. Модель камери із врахуванням дисторсії:  
а) радіальної та б) тангенціальної

**Анастигмат** – об'єктив (рис. 36), у якому виправлені практично всі аберації, в тому числі астигматизм і кривизна поля зображення. Анастигматами можуть вважатися об'єктиви будь-яких конструкцій і типів, що задовольняють цим умовам. Більшість Анастигматів дають хорошу якість зображення по всьому полю при

великих значеннях відносного отвору, забезпечуючи високу світлосилу.

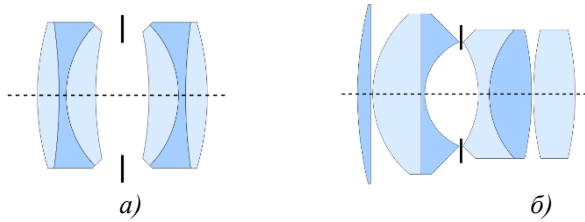


Рис. 36. Оптичні схеми анастигматів:  
а) симетрична; б) пропорційна

Оптика сучасних камер відноситься до анастигматів і має складну будову. Вона складається з декількох лінз різної кривизни і деяких технічних пристосувань, розташованих усередині оптичної системи. Кількість фотолінз може доходити до 15-20. Зазвичай вони групуються в кілька блоків. Для підвищення контрастності знімків за рахунок гасіння відблисків, на лінзи наноситься надтонке покриття з оксидів рідкісних металів. Зазвичай для цього використовується лантан і його сполуки. Цей процес називається просвітленням.

### 5.3 Модель камери

Сучасні камери описуються за допомогою моделі, що називається «проективною камерою» (projective camera, pinhole camera).

Проективна камера (рис. 37) визначається *центром* ( $C$ ), *головною віссю* ( $Cp$ ) – це промінь, який починається в центрі камери і спрямованим туди, куди камера «дивиться», *площиною зображення* (*image plane*) – це площина на яку виконується проектування точок, і системою координат на цій площині. У такій моделі, довільна точка простору  $X$  проектується на площину зображення в точку  $x$  лежить на відрізка  $CX$ , який з'єднує центр камери  $C$  з вихідною точкою  $X$ .

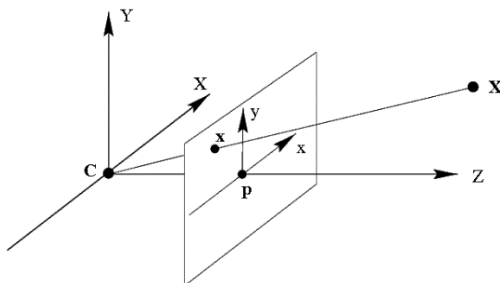


Рис.37. Схематичне зображення основних складових преоктивної камери

Перетворення координат точки простору у координати на площини зображення виконується за наступним виразом:

$$x = PX, \quad (20)$$

де  $x$  – однорідні координати точки на площині зображення;

$X$  – однорідні координати точки у просторі (глобальна система координат)

$P$  – матриця камери розміром  $3 \times 4$

Матриця  $P$  визначається наступним чином:

$$P = KR[I| -c] = K[R|t], \quad (21)$$

де  $K$  – верхня триутна матриця внутрішніх параметрів камери розміром  $3 \times 3$ ,

$R$  – ортогональна матриця розміром  $3 \times 3$ , що визначає порот камери відносно глобальної системи координат;

$I$  – одинична матриця розміром  $3 \times 3$ ;

$c$  – координати центру камери;

$$t = -Rc$$

Варто зазначити, що матриця камери визначена з точністю до постійного ненульового множника, який не змінить результатів проектування точок за виразом (20). Однак цей постійний множник зазвичай вибирається так, що б матриця камери мала вищеповисаний вид.

У самому простому випадку, коли центр камери лежить на початку координат, головна вісь камери співнаправлена вісі  $CZ$ , вісі координат на площині камери мають однаковий масштаб (що еквівалентно квадратним пікселям), а центр зображення має нульові координати, матриця камери буде мати вигляд:

$$P = K[I|0], \quad (22)$$

де 
$$K = \begin{pmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{pmatrix},$$
 тут  $f$  – фокусна відстань камери

У реальних ССД камер пікселі зазвичай несуттєво відрізняються від квадратних, а центр камери має ненульові координати, тому матриця внутрішніх параметрів  $K$  набуде вигляду:

$$K = \begin{pmatrix} \alpha_x & 0 & x_0 \\ 0 & \alpha_y & y_0 \\ 0 & 0 & 1 \end{pmatrix}, \quad (23)$$

де  $\alpha_x$  – фокусна відстань камери вздовж вісі  $x$ ;

$\alpha_y$  – фокусна відстань камери вздовж вісі  $y$ ;

$x_0$  – координати оптичного центру камери вздовж вісі  $x$ ;

$y_0$  – координати оптичного центру камери вздовж вісі  $y$ ;

Крім цього, в силу неідеальності оптики, на зображеннях, отриманих з камер, присутні аберації дисторсії (див. вище.), мають нелінійну математичну запис:

$$\begin{aligned} x'' &= x'(1 + k_1 r^2 + k_2 r^4 + k_3 r^6) + 2p_1 x' y' + p_2 (r^2 + 2x'^2), \\ y'' &= y'(1 + k_1 r^2 + k_2 r^4 + k_3 r^6) + 2p_2 x' y' + p_1 (r^2 + 2y'^2), \end{aligned} \quad (24)$$

де  $k_1, k_2, k_3, p_1, p_2$  – коефіцієнти дисторсії, що є параметрами оптичної системи;

$$r^2 = x'^2 + y'^2;$$

$x', y'$  – координати проекції точки відносно центру зображення при квадратних пікселях і відсутності спотворень;

$x'', y''$  – спотворені координати точки відносно центру зображення при квадратних пікселях;



Дисторсії не залежить від відстані до об'єкта, а залежать лише від координат точок, у які проєктуються пікселі об'єкта. Отже, для компенсації дисторсій зазвичай виконується перетворення вихідного зображення отриманого з камери. Це перетворення буде одним і тим же для всіх зображень, отриманих з цієї камери, за умови сталості фокусної відстані (математично – однієї і тієї ж матриці внутрішніх параметрів камери).

У випадку, коли відомі внутрішні параметри камери і коефіцієнти дисторсії кажуть, що камера відкалібрована.

всі ці спотворення ми наближаємо деяким поліномом, коефіцієнти якого є коефіцієнтами дисторсії, які моделюють спотворення. Якщо ми зможемо для камери порахувати ці коефіцієнти, то, тим самим, ми зможемо передбачити, які спотворення вносить ця камера (рис. 35). У цьому випадку, знаючи тривимірний світ, знаючи проєкційну матрицю камери і коефіцієнти дисторсії, ми можемо коректно передбачити, куди саме спроектується точка тримірного простору на зображення. Ідея полягає у тому, що замість ідеальної проєктивної проєкції відбудеться перехід до деякого викривленого стану, який необхідно врахувати.

### 5.3 Стереозір

Одним з напрямків машинного зору є стереозір. Він дозволяє отримати інформацію про глибину зображення і відстань до об'єктів, скласти тримірну картину навколишнього світу. В основу машинного стереозору покладено стереоскопічний зір людини та тварин. Тому, оптична система для отримання стереозображень може складатись із 2-ох і більше камер, що розміщені на відстані одна від одної і з деяким відносним кутовим зміщенням оптичних осей відносно.



**Стереоскопічний зір** (від грец. *Στερεός* - «твердий, просторовий») – це вид зору, при якому можливе сприйняття форми, розмірів і відстані до предмета, наприклад завдяки бінокулярному зору (кількість очей може бути і більше 2-ох, як, наприклад, у ос – два складних ока і три простих (вічка), скорпіонів – 3-6 пар очей) або іншим типам зору.

Головний мозок отримує два різних зображення, що надходять в від кожного ока, а сприймає їх як одне тримірне зображення. Незважаючи на те, що зображення предметів на сітківці очей двомірне, людина бачить світ тримірним, тобто вона може сприймати глибину простору стереоскопічним зором.

Людина володіє багатьма механізмами оцінки глибини. Наприклад, якщо ви знаєте величину об'єкта (людина, дерево і ін.), То можна визначити відстань до нього і дізнатися, який з об'єктів ближче, шляхом порівняння кутового розміру об'єкта. Але якщо один предмет розташований попереду іншого і частково його закриває, то людині здається, що передній об'єкт знаходиться ближче.

Опукла ділянка стіни здається світлішою у верхній своїй частині, якщо джерело світла розташований вище, а заглиблення її поверхні здається у верхній частині більш темним. Важливою ознакою віддаленості слугує паралакс руху – уявне відносне зміщення близьких і більш далеких предметів, якщо спостерігач буде рухати головою вліво і вправо або вгору і вниз.

Оцінювати віддаленість предметів можна також за величиною акомодатії очі. Найважливішим механізмом сприйняття глибини простору служить стереопсис, який залежить від одночасного використання двох очей. Розглядаючи будь-яку тримірну сцену, очі створюють на сітківці зображення, які дещо відрізняються одне від одного.

### **Матриця пари камер та калібрування.**

Нехай є дві камери, задані своїми матрицями  $P$  і  $P'$  в деякій системі координат. У такому випадку говорять, що є пара відкаліброваних камер. Якщо центри камер не збігаються, то цю пару камер можна використовувати для визначення тримірних координат спостережуваних точок.

Часто, система координат обирається так, що матриці камер мають вигляд:

$$P = K[I|0] \text{ та } P' = K'[R'|t'] \quad (25)$$

Це завжди можна зробити, якщо обрати початок системи координат, який збігається із центром першої камери, і направити вісь  $Z$  уздовж її оптичної вісі.

Калібрування камер зазвичай виконується, за рахунок багаторазової зйомки деякого калібрувального шаблону, на зображенні можна легко виділити ключові точки, для яких відомі їх

відносні положення в просторі. Далі складаються і розв'язуються (наближено) системи рівнянь, що зв'язують координати проєкцій, матриці камер і положення точок шаблону в просторі.

#### Епіполярна геометрія.

Нехай маємо дві камери, як зображено на (рис 38).  $C$  – центр першої камери,  $C'$  – центр другої камери.

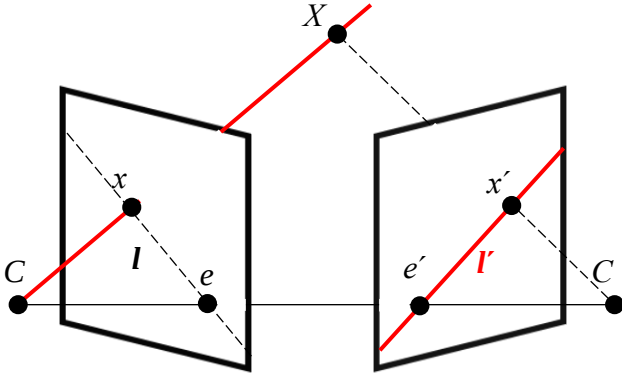


Рис. 38. Зображення елементів епіполярної геометрії

Точка простору  $X$  проєктується в  $x$  на площину зображення лівої камери і в  $x'$  на площину зображення правої камери. Прообразом точки  $x$  на зображенні лівої камери є промінь  $xX$ . Цей промінь проєктується на площину другої камери у пряму  $l'$ , що називається **епіполярною лінією** (рис. 5.38) та (рис. 5.39). Зображення точки  $X$  на площині зображення другої камери обов'язково лежить на епіполярній лінії  $l'$ .



Рис.5.39. Зображення епіполярної лінії

Таким чином, кожній точці  $x$  на зображенні лівої камери відповідає епіполярная лінія  $l'$  на зображенні правої камери. При цьому пара для  $x$  на зображенні правої камери може лежати тільки на відповідній епіполярній лінії. Аналогічно, кожній точці  $x'$  на правому зображенні відповідає епіполярная лінія  $l$  на лівому.

Епіполярну геометрію використовують для пошуку стереопар, і для перевірки того, що пара точок може бути стереопарою (тобто проекцією деякої точки простору).

Епіполярная геометрія має дуже простий запис в координатах. Нехай  $e$  пара відкаліброваних камер, і нехай  $x$  – однорідні координати точки на зображенні однієї камери, а  $x'$  – на зображенні другої. Існує така матриця  $F$  розміром  $3 \times 3$ , що пара точок  $x, x'$  є стереопарою тоді і тільки тоді, коли:

$$x'^T F x = 0 \quad (26)$$

Матриця  $F$  називається фундаментальною матрицею (*fundamental matrix*). Її ранг дорівнює 2, вона визначена з точністю до ненульового множника і залежить тільки від матриць вихідних камер  $P$  та  $P'$ .

У випадку, коли матриці камер мають вигляд (25) фундаментальна матриця може бути обчислена за виразом:

$$F = K'^{-1T} R K^T [K R^T t]_x \quad (27)$$

де для вектора  $e$  позначення  $[e]_x$  обчислюється як:

$$[e]_x = \begin{bmatrix} 0 & -e_z & e_y \\ e_z & 0 & -e_x \\ -e_y & e_x & 0 \end{bmatrix} \quad (28)$$

За допомогою фундаментальної матриці обчислюються рівняння епіполярних ліній. Для точки  $x$ , вектор, що задає епіполярну лінію, буде мати вигляд  $l' = F x$ , а рівняння самої епіполярної лінії:  $l'^T x' = 0$ . Аналогічно для точки  $x'$ , вектор, що задає епіполярну лінію, буде мати вигляд  $l = F^T x'$ .

Окрім фундаментальної матриці, існує таке поняття, як істотна матриця (*essential matrix*):  $E = K'^T F K$ . У випадку, коли матриці внутрішніх параметрів будуть одиничними істотна матриця буде збігатися з фундаментальною. Використовуючи істотну матрицю можна відновити положення і поворот другої камери відносно

першої, тому вона використовується в задачах, в яких потрібно визначити рух камери.

### Триангуляція точок (triangulation).

Це процес визначення тримірних координат за координатами їх проекцій.

Нехай є дві відкалібровані камери з матрицями  $P_1$  та  $P_2$ .  $x_1$  та  $x_2$  - однорідні координати проекцій деякої точки простору  $X$ . Тоді можна скласти наступну систему рівнянь:

$$\begin{cases} x_1 = P_1 X \\ x_2 = P_2 X \end{cases} \quad (29)$$

На практиці для розв'язання цієї системи застосовується наступний підхід. Векторно множать перше рівняння на  $x_1$ , друге на  $x_2$ , позбавляються від лінійно залежних рівнянь і призводять систему до виду  $A X = 0$ , де  $A$  має розмір  $4 \times 4$ . Далі можна виходити з того вектор  $X$  є однорідними координатами точки, покласти його останню компоненту рівною 1 і розв'язувати отриману систему з 3-ох рівнянь з 3-ма невідомими. Альтернативний спосіб – взяти будь-який ненульовий розв'язок системи  $A X = 0$ , наприклад обчислене, як сингулярний вектор, який відповідає найменшому сингулярному числу матриці  $A$ .



Невід'ємне дійсне число  $\sigma$  називається **сингулярним числом** тоді і тільки тоді, коли існує два вектори одиничної довжини  $u \in K^m$  та  $v \in K^n$  такі, що:  $Mv = \sigma u$  та  $M^*u = \sigma v$ , тут  $M^*$  - спряжено-транспонована матриця до  $M$ . Такі вектори  $u$  та  $v$  називаються відповідно, **лівим сингулярним вектором** та **правим сингулярним вектором**, що відповідають числу  $\sigma$ .

### Побудова карти глибини.

Карта глибини (depth map) – це зображення, на якому для кожного пікселя, замість кольору, зберігається його відстань до камери. Карта глибини може отримуватись за допомогою спеціальної камери глибини (наприклад, RGB-D сенсор Kinect), а також бути побудована по стереопарі зображень.

Ідея, що лежить в основі побудови карти глибини по стереопарі дуже проста. Для кожної точки на одному зображенні виконується пошук парної їй точки на іншому зображенні. А по парі відповідних точок можна виконати *триангуляцію* і визначити координати їх прообразу в тримірному просторі. Знаючи тримірні

координати прообразу, глибина обчислюється, як відстань до площини камери.

Парну точку потрібно шукати на епіполярній лінії (рис. 40).



Рис. 40. Пошук парної точки на епіполярній лінії

Відповідно, для спрощення пошуку, зображення вирівнюють так, що б всі епіполярні лінії були паралельні сторонам зображення (зазвичай горизонтальним). Більш того, зображення вирівнюють так, що б у точці з координатами  $(x_0, y_0)$  відповідна їй епіполярна лінія задавалася рівнянням  $x = x_0$ , тоді для кожної точки відповідну їй парну точку потрібно шукати у тому ж рядку на зображенні з другої камери. Такий процес вирівнювання зображень називають **ректифікацією** (*rectification*). Зазвичай ректифікацію здійснюють шляхом ремепінга (*remapping*) зображення та поєднують з позбавленням від дисторсії. Приклад ректифікованого зображень наведено на (рис. 41), зображення взяті з бази порівняння різних методів побудови карти глибини «Middlebury Stereo Vision».



Рис. 41. Приклад ректифікованих зображень [19, 20]

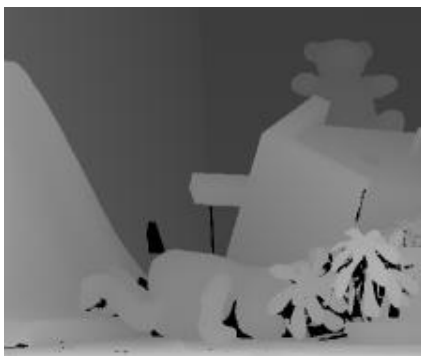


Рис. 42. Приклад карти невідповідностей (disparity map)

Після ректифікації зображень, виконують пошук відповідних пар точок. Найпростіший спосіб проілюстрований на (рис. 5.43).

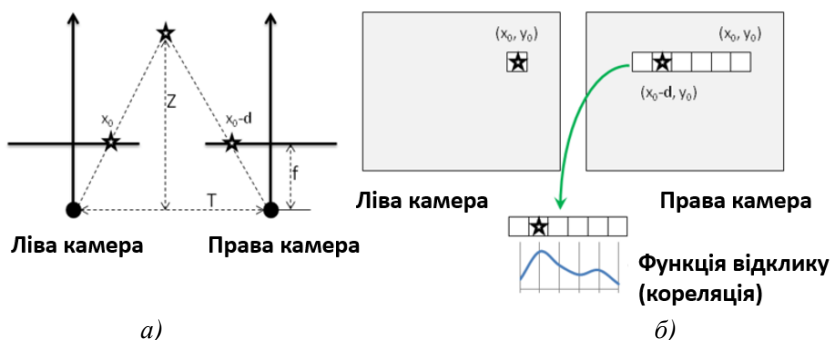


Рис. 43. Графічна інтерпритація пошуку пар точок

Цей спосіб можна описати наступним чином:

- для кожного пікселя лівої картинки з координатами  $(x_0, y_0)$  виконується пошук пікселя на правій картинці. При цьому передбачається, що піксель на правому зображенні повинен мати координати  $(x_0 - d, y_0)$ , де  $d$  – величина, що називається «невідповідність» або «зміщення» (disparity).

- пошук відповідного пікселя виконується шляхом обчислення максимуму функції відгуку, в якості якої може виступати, наприклад, кореляція околу пікселів. В результаті

отримується «карта невідповідностей» або «карта зміщень» (*disparity map*) (рис. 5.42).

Значення глибини обернено пропорційні величині невідповідності (*disparity*) пікселів. Якщо використовувати позначення наведені на (рис. 5.43, а), то залежність між *disparity* і глибиною можна виразити наступним чином:

$$\frac{T - d}{Z - f} = \frac{T}{Z} \rightarrow Z = \frac{fT}{d} \quad (30)$$

де  $T$  – відстань між оптичними центрами камер стереопари (рис. 5.43, а);

$d$  – величина невідповідності (зміщення) (*disparity*);

$Z$  – відстань від точки тримірного простору до площин зображень (рис. 5.43, а);

$f$  – фокусна відстань камери;

Через зворотню залежність глибини і зміщення, роздільна здатність систем стерео зору, які працюють на основі даного методу, краще на близьких відстанях, і погіршується по мірі віддалення від камери.



## 5.4. Завдання для контролю знань



### Теоретичні питання

**1. Комплекс математичних операцій спрямованих на зменшення впливу завад:**

- а) фільтрація; б) апроксимація;
- в) згортка;
- г) детектування;
- д) аберація.

**2. Яка перевага фільтра Гауса:**

- а) зниження шуму на зображенні;
- б) створення візуальних ефектів;
- в) гнучкість налаштування фільтра;
- г) правильними є відповіді а, б.

**3. Оберіть вірний вираз двовірного ядра Гауса:**

а)  $G_{\sigma}(x, y) = \frac{1}{2 \cdot \pi \cdot \sigma^2} \cdot e^{-\frac{y^2}{2\sigma^2}};$

б)  $G_{\sigma}(x, y) = \frac{1}{2 \cdot \pi \cdot \sigma^2} \cdot e^{-\frac{x^2}{2\sigma^2}};$

в)  $G_{\sigma}(x, y) = \frac{1}{2 + (\pi \cdot \sigma)^2} \cdot e^{-\frac{x^2 + y^2}{2\sigma^2}};$

г)  $G_{\sigma}(x, y) = \frac{1}{2 \cdot \pi \cdot \sigma^2} \cdot e^{-\frac{x^2 + y^2}{2\sigma^2}}.$

**4. Невеликі області зображення, що використовуються для визначення подібності між зображеннями – це:**

- а) дескриптор;
- б) особливості;
- в) градієнт;
- г) правильними є відповіді а, б.

**5. Містить візуальний опис області і використовується для порівняння “подібності” між різними особливостями зображення – це:**

- а) дескриптор;
- б) особливості;
- в) градієнт;
- г) правильними є відповіді а, б.

**6. Яка із наведених нижче задач не відноситься до розпізнавання:**

- а) виявлення;
- б) вимірювання;
- в) ідентифікація;
- г) вірна відповідь відсутня.

**7. Які із перерахованих нижче методів не є методами визначення особливостей:**

- а) детектор Харіса;
- б) детектор Кенні;
- в) масштабонезалежне перетворення ознак;
- г) перетворення Хафа;
- д) правильні відповіді а, б, в, г.

**8. Віднесення досліджуваного об’єкта до певного класу з врахуванням ключових точок або його особливостей:**

- а) сегментація;
- б) кластеризація;
- в) розпізнавання;
- г) класифікація.

**9. Дескриптори повинні забезпечувати інваріантність до:**

- а) масштабування; б) повороту;
- в) згортки;
- д) вірні відповіді а, б;
- г) всі відповіді вірні.

**10. Основна ідея методу SIFT полягає у:**

- а) побудові піраміди гаусіан та піраміди різниць гаусіан;
- б) визначення градієнту ключової точки;

в) фільтрації зображення; г) згортці зображення.

**11. Гаусіан – це:**

- а) ядро фільтра Гауса;
- б) зображення розмите фільтром Гауса;
- в) набір усіх можливих, згладжених деяким фільтром, варіантів оригіналу зображення;
- г) правильна відповідь відсутня.

**12. Оберіть правильний вираз Гаусіана:**

- а)  $L(x, y, \sigma) = G(x, y, \sigma) * I(x, y)$  ;
- б)  $L(x, y, \sigma) = G(x, y, \sigma) + I(x, y)$  ;
- в)  $L(x, y, \sigma) = G(x, y, \sigma) \cdot I(x, y)$  ;
- г)  $L(x, y, \sigma) = G(x, y) * I(x, y)$  .

**13. За яким виразом обчислюється різниця гаусіан:**

- а)  $D(x, y, \sigma) = (G(x, y, k \cdot \sigma) + G(x, y, \sigma)) * I(x, y)$  ;
- б)  $D(x, y, \sigma) = G(x, y, k \cdot \sigma) - I(x, y)$  ;
- в)  $D(x, y, \sigma) = (G(x, y, k \cdot \sigma) - G(x, y, \sigma)) * I(x, y)$  ;
- г)  $D(x, y, \sigma) = G(x, y, \sigma) * I(x, y)$  .

**14. Згідно методики побудов піраміди гаусіан SIFT при переході від однієї октави до іншої виконується, його розміри зменшуються вдвічі?**

- а) зображення збільшується вдвічі;
- б) попіксельне віднімання сусідніх зображень в піраміді різниць гаусіан;
- в) попіксельне віднімання сусідніх зображень в піраміді гаусіан;
- г) зображення зменшується вдвічі.

**15. Що вважають ключовою точкою зображення відповідно до методики SIFT після побудови піраміди гаусіан?**

- а) локальні максимуми; б) локальні мінімуми;
- в) локальні екстремуми; г) вірна відповідь відсутня.

**16. Вектор, що своїм напрямком вказує напрям найбільшого зростання деякої величини, значення якої змінюється від однієї точки простору до іншої – це:**

- а) градієнт; б) скаляр;
- в) дескриптор;
- г) вірна відповідь відсутня.

**17. За яким із наведених нижче виразів визначається величина градієнта:**

- а)  $m(x, y) = (L(x+1, y) - L(x-1, y))^2 + (L(x, y+1) - L(x, y-1))$ ;
- б)  $m(x, y) = \sqrt{(L(x+1, y) - L(x-1, y))^2 + (L(x, y+1) - L(x, y-1))^2}$  ;
- в)  $m(x, y) = \sqrt{(L(x+1, y) - L(x-1, y)) + (L(x, y+1) - L(x, y-1))}$  ;
- г)  $m(x, y) = \sqrt[3]{(L(x+1, y) - L(x-1, y))^2 + (L(x, y+1) - L(x, y-1))^2}$  .

**18. За яким із наведених нижче виразів визначається напрям градієнта:**

- а)  $\theta(x, y) = \arctan\left(\frac{L(x, y+1) - L(x, y-1)}{L(x+1, y) - L(x-1, y)}\right)$ ;
- б)  $\theta(x, y) = \tan\left(\frac{L(x, y+1) - L(x, y-1)}{L(x+1, y) - L(x-1, y)}\right)$ ;
- в)  $\theta(x, y) = \arcsin\left(\frac{L(x, y+1) - L(x, y-1)}{L(x+1, y) - L(x-1, y)}\right)$ ;
- г)  $\theta(x, y) = \arccos\left(\frac{L(x, y+1) - L(x, y-1)}{L(x+1, y) - L(x-1, y)}\right)$ .

**19. Що є основною ідеєю перетворення Хафа?**

- а) пошук примітивів по парі точок зображення;
- б) пошук примітивів відповідно їх параметричних форм;
- в) правильні відповіді а, б;
- г) правильна відповідь відсутня.

**20. Для пошуку яких об'єктів застосовується перетворення Хафа?**

- а) представлених у вигляді набору ключових точок;

б) представлених у вигляді набору ключових точок та дескрипторів;

в) описаних аналітичним виразом;

г) правильна відповідь відсутня.

## КОМП'ЮТЕРНИЙ ПРАКТИКУМ

### ЗАГАЛЬНІ ВІДОМОСТІ

Лабораторні заняття мають за мету:

- ознайомити студентів із елементами систем штучного інтелекту, що можуть бути застосовані для автоматизованого вирішення різноманітних проблем та розв'язання задач науково-практичного значення у різних галузях діяльності людини;
- закріпити отримані теоретичні знання про елементи систем штучного інтелекту та використовувати їх при розв'язуванні практичних задач науково-практичного значення, наприклад, при автоматизованому керуванні технологічними процесами, медичному діагностуванні, фінансовому аналізі, конструюванні вимірювальних приладів, виборі обладнання, формуванні оптимальної роботизованої механоскладальної технології тощо.

Тематика лабораторних робіт підібрана таким чином, що дозволяє студентам закріпити теоретичні знання з основних розділів дисципліни та отримати практичні навички досліджень та синтезу систем штучного інтелекту на прикладі синтезу штучних нейронних мереж.

Всі необхідні при виконанні лабораторних робіт аналітичні залежності приводяться в кінцевому вигляді, без доведень, з відповідними посиланнями на теоретичний курс.

Основні визначення та термінологія відповідають прийнятим в лекційному курсі.

Виконання лабораторних робіт припускає використання устаткування та програмних засобів, що існує та може бути встановлене в лабораторіях більшості вищих навчальних закладів та на підприємствах.

Зміст даного розділу підручника у відповідності до програми навчальної дисципліни розділений на окремі лабораторні роботи, назва та зміст яких висвітлюється нижче.

## Загальні вказівки до виконання робіт

Перед виконанням кожної лабораторної роботи студент повинен ознайомитись із теоретичним матеріалом за темою лабораторної роботи.

За результатами виконаної лабораторної роботи студент оформлює відповідний звіт. Звіт з виконаної лабораторної роботи оформлюється кожним студентом з використанням доступних текстових редакторів, наприклад, Microsoft Word, та захищається на консультації перед виконанням наступної роботи або на наступному лабораторному занятті безпосередньо перед виконанням роботи.

Результати виконаних лабораторних робіт є складовою загальної оцінки з навчальної дисципліни, що виставляється відповідно із рейтинговою системою оцінювання при пред'явленні правильно виконаних робіт, а також акуратно оформлених та зброшурованих звітів з усіх відпрацьованих та зарахованих раніше лабораторних робіт.

## Загальні вимоги до оформлення звітів

1. Звіт з кожної роботи оформлюється на аркушах білого паперу формату А4 (розміром 210×297 мм). На кожному форматі виконується обрамлююча лінія (рамка) згідно ГОСТ 2.301-68. Зокрема **на першому аркуші** виконується **основний надпис (штамп) за формою 2** для текстових документів згідно ГОСТ 2.105-79 (див. додаток Б, рис. Б.1); на наступних – **надпис (штамп) за формою 2а** для текстових документів згідно ГОСТ 2.105-79 (див. додаток Б, рис. Б.2).

2. На першому аркуші звіту розміщується розділ "Зміст". Назви розділів та підрозділів наступних листків повинні відповідати змісту звіту.

3. Звіт повинен містити наступні складові:

Назва роботи.

Мета роботи.

Короткі теоретичні відомості.

Порядок виконання роботи.

Проміжні результати досліджень та розрахунків.

Зведена таблиця результатів.

Висновки по роботі.

Відповіді на контрольні питання

4. Всі заповнені аркуші брошуруються в загальній обкладинці з титульним аркушем. Приклад оформлення титульного листа приведено в Додатку В. Справа в дужках подано відстані від нижньої лінії, яка обрамлює формат, до основи відповідного рядка та номер шифру, яким слід виконувати запис рядка.

5. Текст звіту лабораторної роботи набирається в текстовому редакторі Microsoft Word гарнітурою Times New Roman, розміром 14 пунктів, з вирівнюванням по ширині.

6. Інтервали між рядками – полуторні. Абзац – 3 символи зліва.

7. **Заголовки та назви розділів** звіту кожної лабораторної роботи виділяються жирним; крім того перед назвою розділу та після нього пропускається 1 рядок.

8. **Формули** набираються за допомогою редактора формул та розміщуються по центру рядка, після формули дається пояснення; нумерація формули вказується в крайньому правому положенні відповідно до порядкового номеру лабораторної роботи та порядкового номеру формули; перед формулою та після неї пропускається 1 рядок. Крім того, всі формули мають бути вставлені в таблиці, які не повинні бути обрамлені (таблиця без обрамлення). Нижче наведено приклад вставки формули:

|                                                                                   |                       |        |
|-----------------------------------------------------------------------------------|-----------------------|--------|
|  | № формули             |        |
|                                                                                   | № лабораторної роботи |        |
| <V, S, K, W>,                                                                     |                       | (1. 1) |

де: V – процедура вибору з бази знань і робочої пам'яті правил і фактів;

S – процедура співставлення правил і фактів, в результаті якої визначається безліч фактів до яких застосовувані правила для присвоєння значень;

K – процедура вирішення конфліктів, що визначає порядок використання правил, якщо у висновку правила вказані однакові імена фактів з різними значеннями;

W – процедура, що здійснює виконання дій, відповідно до отриманого значення факту (укладення правила).



Формула має бути вставлена в таблицю по центру без абзацу. Номер формули слід розмістити посередині висоти другої колонки з виключкою вправо.

9. **Таблиці** розміщуються по центру рядка; нумерація таблиці вказується над таблицею праворуч відповідно до порядкового номеру лабораторної роботи та порядкового номеру таблиці, назва таблиці вказується в наступному під номером рядку по центру; перед таблицею та після неї пропускається 1 рядок; назва таблиці та "шапка" таблиці можуть бути виділені жирним шрифтом, наприклад:

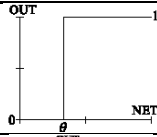
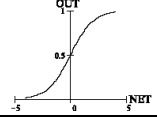


№ таблиці

№ лабораторної роботи

Таблиця 3.2

Основні види функцій активації

| Назва функції                        | Рівняння функції                                                  | Область значень функції | Графік функції                                                                    |
|--------------------------------------|-------------------------------------------------------------------|-------------------------|-----------------------------------------------------------------------------------|
| Жорсткий поріг                       | $OUT = \begin{cases} 0, & NET < 0 \\ 1, & NET \geq 0 \end{cases}$ | 0; 1                    |  |
| Логістична (сигмоїда, функція Фермі) | $OUT = \frac{1}{1 + e^{-NET}}$                                    | 0; 1                    |  |

Посилання на таблицю в тексті роботи здійснюються скорочено, наприклад, наступним чином:



...вихідні дані задачі наведені в табл. 1.2.

10. **Рисунки, схеми та графіки** у випадку оформлення звіту лабораторної роботи у рукописному вигляді необхідно креслити олівцем в масштабі з використанням креслярських інструментів. Рисунки, схеми та графіки розміщуються по центру рядка, перед ними та після них пропускається 1 рядок. Нумеруються аналогічно до вимог нумерації формул та таблиць.

Посилання на рисунок, схему, графік в тексті роботи здійснюються скорочено, наприклад, наступним чином:

---



*...результат розв'язання задачі приведено на рис. 3.1.*

---

11. Всі сторінки, рисунки, таблиці тощо повинні бути пронумеровані, підписані та / або надписані.

## Практикум №1

### Тема: Детектор Харріса та порівняння зображень

**Мета:** отримати практичні навички з пошуку ключових точок на основі детектора кутів Харіса та зіставлення зображень з врахуванням цих точок.

#### 1.1. Теоретичні відомості

Існує безліч видів особливостей (features) зображень та алгоритмів їх пошуку (detection), що призначені для різних задач, наприклад, детектори кутів.

*Кути* (corners) – особливі точки (key points) зображення, які формуються з двох або більше граней, які, зазвичай, визначають границі між різними об'єктами та/або їх частинами [**Ошибка! Источник ссылки не найден.**].

Іншими словами, кути – це точки, у яких в околиці яскравість змінюється відносно центру  $(x, y)$ . Кути визначаються їх координатами та величиною зміни яскравості сусідніх точок.

Головною властивістю таких точок є те, що в області навколо кута у градієнта зображення переважають два домінуючі напрямки, що робить їх помітними.

Градiєнт – векторна величина, що вказує напрямок найшвидшого зростання функції яскравості зображення  $I(x, y)$ . Оскільки зображення дискретне, то вектор градієнта визначається через часткові похідні по осі  $X$  і  $Y$  через зміну яскравості сусідніх точок. Більшість методів розглядають кути, що залежить від похідної 2-го порядку, тому в загальному методи чутливі до шуму.



Рис. 1.1. Види кутів

В залежності від кількості граней, що перетинаються існують (рис. 1.1.) різні види кутів: L-, Y-, (або T-) та X- зв'язні. [**Ошибка! Источник ссылки не найден.**] (деякі виділяють ще стрілоподібні зв'язкові кути [**Ошибка! Источник ссылки не найден.**]). Різні детектори по-різному реагують на кожен із прерахованих вище кутів.

Підходи до визначення ключових точок умовно поділяються на 3 категорії [Ошибка! Источник ссылки не найден.]:

*Засновані на аналізі яскравості зображення:* особливі точки обчислюються безпосередньо за значеннями яскравості пікселів.

*Використовують контури зображення:* методи, що виділяють контури і дозволяють детектувати області із максимальним значенням кривизни або виконують полігональну апроксимацію контурів і визначають перетини. Ці методи чутливі до околиць перетинів, оскільки результат часто може бути помилковим в тих місцях, де є перетинають 3-х або більше границь.

*На основі використання моделі:* використовуються моделі зі значеннями яскравості у якості параметрів, які підлаштовуються під зображення-шаблони до субпіксельної точності. Мають обмежене застосування з особливими точками спеціальних видів (наприклад, L-зв'язними кутами), залежать від шаблонів, що використовуються.

На практиці широкого застосування набули методи, що базуються на аналізі яскравості зображення.

Відомі на сьогодні наступні детектори кутів:

- Моравеца (Moravec);
- Харріса (Harris);
- Ши-Томасі (Shi-Tomasi);
- Форстнера (Förstner);
- SUSAN (Smallest Univalued Segment Assimilation Nucleus);
- Трайковіца (Trajkovic);
- FAST (Features from Accelerated Segment Test);
- CSS (Curvature scale space);
- CPDA (Chord-to-Point Distance Accumulation);

## **1.2. Виконання роботи**

**Детектор кутів Харріса** – це вдосконалений детектор Моравеца. Сутність методу полягає у наступному: для зображення  $I(x,y)$  розглядається вікно  $w(x,y)$  (за звичай розміром  $5 \times 5$ ), що зміщується на величину  $(u,v)$  вздовж границь на зображенні (рис. 1.2) і у якому виконується аналіз яскравості (рис. 1.3)

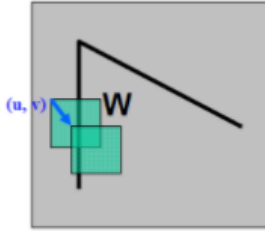
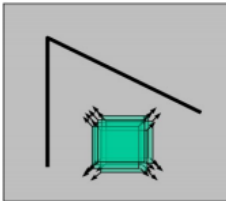
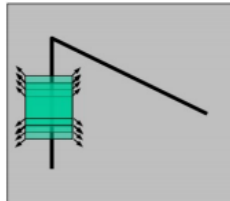


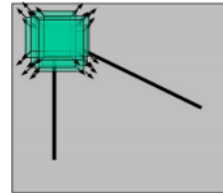
Рис. 1.2. Зміщення віка  $W(x,y)$  на величину  $(u,v)$



«плоска» область.  
Немає зміни  
яскравості при  
переміщенні у  
жодному із  
напрямів



«границя»  
Немає зміни  
яскравості при  
переміщенні вздовж  
границі



«кут»  
Суттєва зміна  
яскравості у всіх  
напрямах

Рис. 1.3. Ілюстрація ідеї детектора кутів Харріса

Зміна яскравості в межах вікна  $w(x,y)$  (автокореляція) при цьому визначається за виразом:

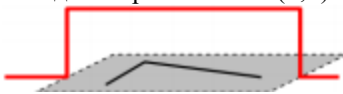
$$E(u, v) = \sum_{x,y} w(x, y) \cdot [I(x + u, y + v) - I(x, y)]^2 \quad (1.1)$$

де  $w(x, y)$  – вагова функція скануючого вікна;

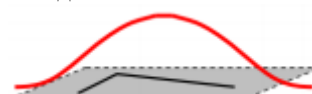
$I(x, y)$  – значення яскравості в точці;

$I(x + u, y + v)$  – зміщена яскравість;

Вагова функція скануючого вікна може бути представлена або у вигляді бінарного вікна (1,0) або у розподілену за законом Гауса:



1 – в середині вікна,  
0 – за межами вікна



функція Гауса



Рис. 1.4. Функція автокореляції в околі точок для різних регіонів зображення

Для малих зсувів проведемо білінійну апроксимацію:

$$E(u, v) \cong [u, v] \cdot M \begin{bmatrix} u \\ v \end{bmatrix} \quad (1.2)$$

де  $M$  – автокореляційна матриця, що розраховується із часткових похідних  $I_x$  та  $I_y$  по яскравості зображення

$$M = \sum_{x,y} w(x, y) \cdot \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix} \quad (1.3)$$

На практиці ця операція виконується за допомогою згортки зі зваженим ядром:

$$M = w(x, y) * \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix} \quad (1.3)$$

У випадку коли градієнти вирівняні вздовж осей (горизонтальних або вертикальних)

Кути характеризуються значною зміною функції  $E(u, v)$  у всіх напрямках вектора  $[u, v]$ , що еквівалентне великим по модулю власним значенням матриці  $M$ , яка є симетричною, тому:

$$M = R^{-1} \cdot \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix} \cdot R \quad (1.4)$$

Матрицю  $M$  можна представити у вигляді еліпса у якого довжини осей визначені власними значеннями, а орієнтація визначена матрицею  $R$ .

Рівняння еліпса:

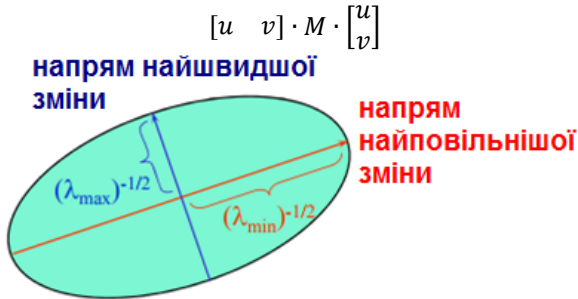
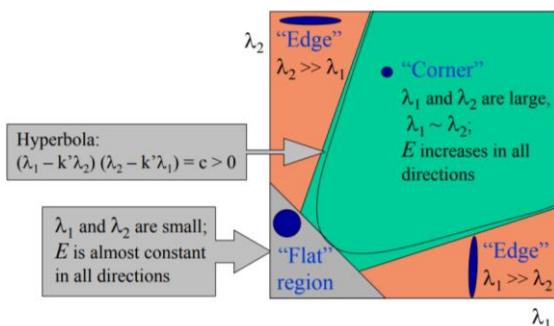


Рис. 1.5. Ілюстрація сутності параметрів  $\lambda_1$  та  $\lambda_2$  матриці  $M$

На рис. 1.5. власні значення  $\lambda_1$  та  $\lambda_2$  матриці  $M$  вказують на максимальний та мінімальний напрями зміни градієнту.

За значеннями  $\lambda_1$  та  $\lambda_2$  виконується класифікація точок зображення:



Якщо  $\lambda_1 \gg \lambda_2$  – вертикальна границя;

Якщо  $\lambda_2 \gg \lambda_1$  – горизонтальна границя;

Якщо  $\lambda_1$  та  $\lambda_2$  приймають великі значення,  $\lambda_1 \approx \lambda_2$  – кут,  $E$  зростає у всіх напрямках;

Якщо  $\lambda_1$  та  $\lambda_2$  приймають малі значення – «плоска» область,  $E$  майже не змінюється у всіх напрямках;

Оскільки безпосередньо обчислювати власні значення  $\lambda_1$  та  $\lambda_2$  є трудомісткою задачею тому Харрісом і Стефенсом [Ошибка! Источник ссылки не найден.] була запропонована міра відгуку  $R$ , що визначається за наступним виразом:

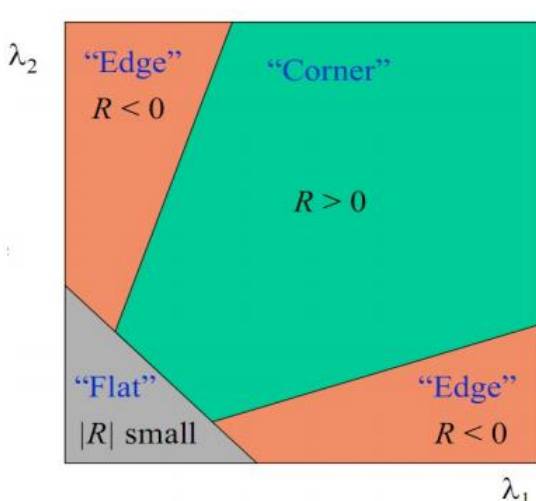
$$R = \det M - k \cdot (\text{trace } M)^2, \quad (1.5)$$

$$\det M = \lambda_1 \cdot \lambda_2, \text{ trace } M = \lambda_1 + \lambda_2$$

де  $k$  – емпіричний коефіцієнт ( $k = 0,04 \dots 0,06$ );



$R < 0$  – границя  
 $R > 0$  – кут  
 $|R| \approx 0$  (малі  
 значення) –  
 «плоска»  
 область



Отже, після обчислення значення  $R > 0$  для кутових особливих точок виконується відсікання точок за порогом  $R$  (тобто ті точки, у яких значення  $R$  менше деякого порога, виключаються із аналізу).

На наступному кроці шукають локальні максимуми функції відклику  $R$  (non-maxima suppression) які і обираються в якості кутових особливих точок.

### Особливості детектора Харріса

Детектор Харріса інваріантний до поворотів, частково інваріантний до афінних перетворень.

До недоліків варто віднести чутливість до шуму і залежність детектора від масштабу зображення.

### 1.3. Виконання роботи

Практикум може виконуватись із використанням будь-якої мови та середовища програмування: Python, C++, C#, Java, Qt, Matlab, тощо.



**При виконанні МКР із використанням Matlab,** доцільно скористатись поясненнями, що надані нижче.

**УВАГА!** При виконанні МКР **не допускається** використання вбудованих функцій та сторонніх бібліотек у яких реалізовано детектор кутів Харріса.

## Завдання 1.1. Детектор кутів Харріса

У файлі `harris_corner.m` необхідно додати власний код детектора Харріса.

*Вхідні параметри функції* – зображення у градаціях сірого із значеннями діапазону від 0 до 255 та структурою параметрів, що містить наступні 4 налаштування:

`parameters.k` – коефіцієнт формули Харріса

`parameters.threshold` – значення обмежень для порогової фільтрації кутів

`parameters.sigma1` –  $\sigma$  фільтра Гауса, що використовується на першій стадії алгоритму

`parameters.sigma2` –  $\sigma$  фільтра Гауса, що використовується після розрахунку похідних по координатам  $x$  і  $y$  (повинна бути більше, ніж `sigma1`)

Для відлагодження коду необхідно результат кожного проміжного кроку повертати у вигляді параметру. Докладні описи *вихідних параметрів* дивитись у коментарях до файлу.

*Основні кроки, які необхідно виконати:*

1. Нормалізувати зображення та перевести у тип `double` в діапазоні значень від 0 до 1.

2. Обчислити горизонтальні та вертикальні похідні зображення ( $I_x$  і  $I_y$ ) шляхом згортки зображення  $I$  з похідною гаусіан (Derivatives of Gaussians –DoG) по координаті  $x$  та  $y$ . Ви можете створити Гаусіани командою `fspecial`, використовуючи параметр `sigma1`. Розмір фільтра може бути змінним.

Щоб отримати DoG, ви можете відфільтрувати ядро Гауса, наприклад з фільтром `[-1 0 1]`.

3. Обчислити значення функції Харріса для кожного пікселя та нормалізувати значення до діапазону `[0, 1]`. Використовуйте інший фільтр Гаусса (з `sigma2`) у якості зваженого ядра.

4. Виконати порогову фільтрацію значень функції та запровадити простий метод придушення немаксимумів.

Для перевірки вашої реалізації ви можете викликати скрипт `main_harris_image.m` необхідно спробувати різні налаштування параметрів і різні вхідні зображення.

## Завдання 1.2. Патч-дескриптор

Після додавання коду детектора кутів Харріса (**harris\_corner.m**), на наступному кроці необхідно виконати зіставлення знайдених кутів подібних зображень, щоб встановити відповідності. Для того, щоб це зробити, спочатку необхідно застосувати дескриптор властивостей, який характеризує кожний виявлений кут.

У цьому завданні необхідно обчислити *простий дескриптор властивостей*. Для цього необхідно проаналізувати значення яскравості в околі кута та об'єднати їх щоб отримати великий *вектор властивостей*.

Додати свій код у файл **patch\_descriptor.m**, вхідні та вихідні параметри описані в коментарях у самому файлі.

## Завдання 1.3. Зіставлення точок інтересу

Після того як отримано вектор властивостей для кожної точки-інтересу необхідно виконати зіставлення схожих точок на різних зображеннях.

Функція зіставлення точок інтересу повинна бути реалізована у файлі **match\_descriptors.m**, який приймає дескриптори властивостей двох зображень у якості вхідних параметрів. Для кожного дескриптора  $x_A$  зображення  $A$  необхідно обчислити відстань до кожного дескриптора  $x_B$  зображення  $B$ . В якості відстані приймається евклідова відстань  $d$  між двома дескрипторами:

$$d = |x_A - x_B|$$

Дескриптор  $x_B$  за найменшою відстанню відповідає найкращому зіставленню з дескриптором  $x_A$ .

Деякі дескриптори можуть бути неоднозначними, що призведе до помилкових зіставлень. Для зменшення кількості помилкових зіставлень необхідно додати наступну евристику<sup>1</sup>:

Прийmemo два «найближчі» зіставлення дескрипторів  $x_{B1}$  та  $x_{B2}$  з дескриптором  $x_A$  з відповідними відстанями  $d_1$  та  $d_2$ .  $x_{B1}$  може бути розглянуто як унікальне найкраще зіставлення для  $x_A$  якщо буде виконуватись умова:

---

<sup>1</sup> Евристика – це сукупність прийомів і методів, що полегшують і спрощують рішення конструктивних та практичних завдань.

$$\frac{d_1}{d_2} < 0,8$$

На рис. 1.6. представлено результат гарного зіставлення.

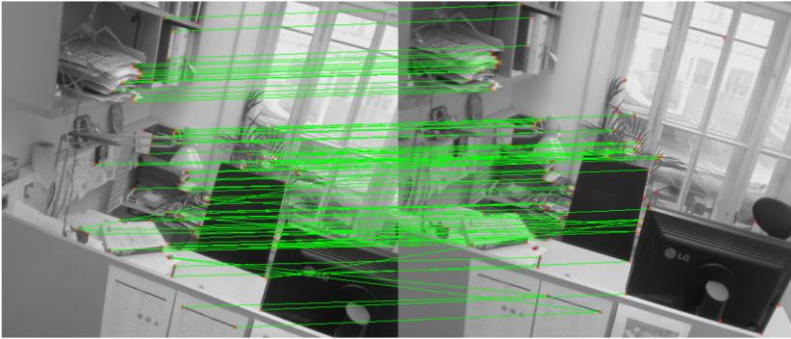


Рис. 1.6. Результат гарного зіставлення точок інтересу (лише 5 помилкових варіантів)

#### 1.3.4. Корисні команди при реалізації у Matlab

`imfilter` – фільтрація за допомогою ядра лінійного фільтра.

`fspecial` – генерування ядра фільтра

`imread` – завантаження зображення

`imshow` – відображення зображення (для RGB зображень, формантаних має бути `uint8` в діапазоні `[0, 255]`, для зображення у відтінках сірого – `double` в діапазоні `[0, 1]`)

`rgb2gray` – конвертація RGB зображення у відтінки сірого (grayscale)

`double` – перетворення у формат `double` (з плаваючою комою)

`uint8` – перетворення у цілі невід'ємні числа в діапазоні `[0, 255]`

`sort` – сортування векторів або матриць

#### 1.4. Завдання до роботи

1. Відповідно до п.1.3. реалізувати простий детектор точок інтересу (ключових точок), а саме, детектор кутів Харріса.
2. Для кожної точки інтересу обчислити дескриптор, який, може бути використаний для зіставлення аналогічних точок інтересу на різних зображеннях один з одним.
3. Обчислити детектор Харріса з різними значеннями  $\sigma$  для фільтра Гауса і перевірити їх вплив на результати зіставлення зображень.

4. Зіставити дескриптори декількох схожих зображень, використовуючи різні налаштування параметрів.
5. Зробити висновки по роботі.
6. Після висновку обов'язково подати відповіді на контрольні питання.

#### 1.5. Зміст документації

1. Тема та мета модульної контрольної роботи.
2. Блок-схема алгоритму детектора Харріса і процедур зіставлення зображень з врахуванням евристики реалізованих при виконанні лабораторної роботи та описаних у файлах **harris\_corner.m**, **patch\_descriptor.m** та **match\_descriptors.m**. У «шапці» файлів обов'язково вказати ПІБ виконавця та групу.
3. Лістинг коду детектора Харріса, наприклад з файлу **harris\_corner.m** з коментарями і поясненнями.
4. Лістинг файлів **patch\_descriptor.m** та **match\_descriptors.m** з коментарями і поясненнями.
5. Скріншоти результатів роботи детектора Харріса при запуску файлів **main\_harris\_image.m** та **main\_descriptors\_image.m**
6. Перевірити роботу детектора Харріса із зображеннями, що знаходяться у каталогах **desk/** та **pattern/a** також, щонайменше двома власними зображеннями.
7. Дати відповіді на контрольні питання.

#### 1.6. Контрольні питання

1. Як різні значення  $\sigma$  фільтра Гауса впливають на результат детектора кутів Харріса?
2. Як зміна розміру вікна дескриптора впливає на точність зіставлення?
3. Чи можете Ви запропонувати інший тип дескриптора, який можна було б використовувати для опису характеристик області навколо точок інтересу окрім значення яскравості зображення?

## Практикум № 2

### Тема: Пошук площин за допомогою RANSAC

**Мета:** отримати практичні навички роботи з тримірним простором на основі хмар точок. Пошук площин за допомогою RANSAC.

### Теоретичні відомості

Мета роботи – пошук домінуючої площини (підлоги) у хмарі точок (point clouds). Загальним методом для реалізації цього є алгоритм RANSAC (RANDOM Sample Consensus). У роботі потрібно обробити лише просторову інформацію хмари точок, інформація про колір ігнорується. Приклад хмари точок показаний на рис. 1.

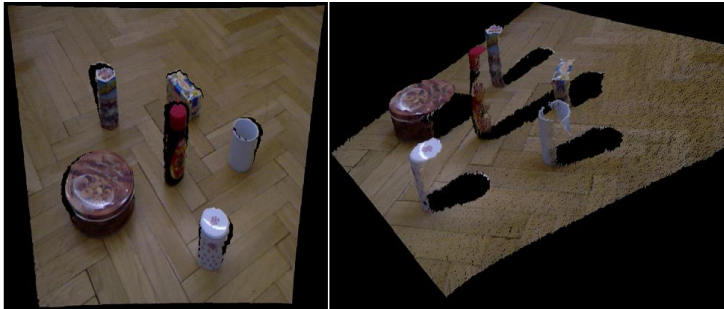


Рис. 2.1. Приклад хмари точок із набору даних, з різних точок погляду

Щоб правильно відображати 3D хмари точок, необхідно встановити бібліотеку PointCloud (PCL) [pointclouds.org](http://pointclouds.org). Бібліотека містить інструмент `pcl_viewer`, за допомогою якого можна відображати хмари з довільних точок погляду. Проте, установка бібліотеки PCL для виконання вправи не є обов'язковою.

### 2.1 Метод RANSAC

Це ітеративний метод, що використовується для оцінки параметрів математичної моделі використовуючи певний набір спостережуваних даних які містять викиди (outliers). При роботі алгоритму знаходиться прийнятне рішення тільки з певною імовірністю, імовірність зростає відповідно до зростання кількості ітерацій.

Ідея алгоритму заснована на припущенні, що початкові дані складаються із викидів (outliers) та не-викидів (inliers). Не-викиди задовольняють певну математичну модель з набором параметрів, тоді як викиди є результатом шумів, і не задовольняють модель. RANSAC припускає, що існує процедура, яка за невеликим набором не-викидів яка може оцінити параметри моделі, що оптимально відповідає вхідним даним.

Наприклад, пошук прямої за набором точок на площині. Припустимо, що набір даних складається з не-викидів (inliers) – тобто точок, які приблизно можуть належати прямій, і викидів (outliers) - точок, що не можуть належати прямій.

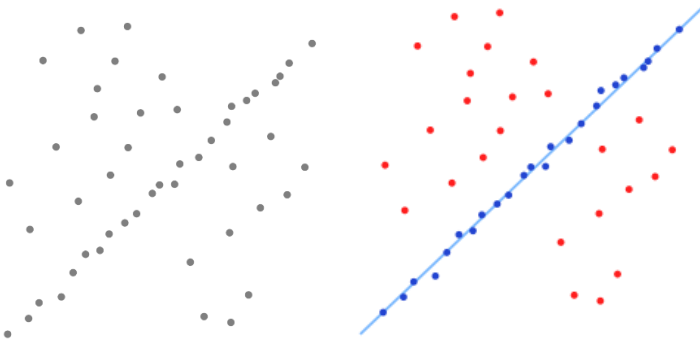


Рис. 2.2. Початковий набір даних та знайдена лінія за допомогою RANSAC

Для реалізації алгоритму необхідно виконати наступне:

1. Вибрати випадкову підмножину з початкових даних – потенційні не-викиди. Обсяг підмножини залежить від моделі. Для лінії – 2 точки, для площини – 3 точки і т.д.
2. Підібрати параметри моделі (задати набір обмежень) для вибірки потенційних не-викидів.
3. Всі інші точки перевіряються на відповідність моделі (чи належать вони прямій та задовольняють обмеження). Точки які з деякою точністю належать до прямої можна назвати набором погоджених точок.
4. Модель вважається вдалою, якщо достатньо великий відсоток точок належить до набору погоджених точок, тобто inliers.

5. Далі модель може бути уточнена використовуючи всі точки з набору погоджених точок.

### Особливості алгоритму RANSAC

Надійний ітераційний метод для оцінки параметрів математичної моделі з набору даних, що містять викиди.

Поділяє отримані дані на "inliers" та "outliers"

Результат залежить від величини обмежень та заданої достовірності

Не ідеальний, але існує багато вдосконалень.

## 2.2. Кластеризація

Для реалізації задачі розпізнавання об'єктів необхідно, після знайдення домінуючої площини (площини підлоги) та відділення від неї об'єктів виконати їх кластеризацію, щоб виділити всі точки, які відповідають потенційним об'єктам.

Кластеризація – це процес групування об'єктів (точок) за рядом ознак (рис. 2.3)

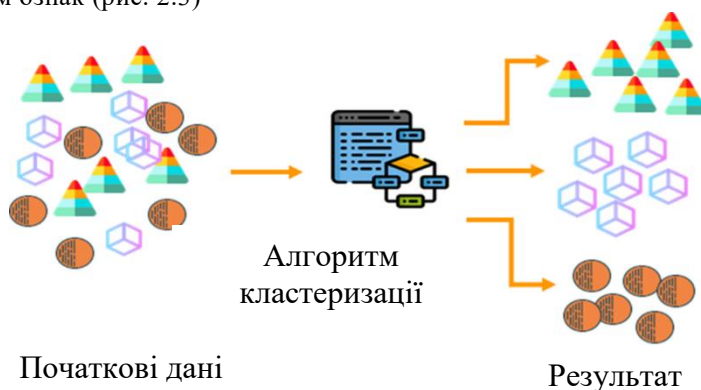


Рис. 2.3. Графічна інтерпретація сутності кластеризації

В ідеальному випадку всі точки об'єкта повинні бути пов'язані з одним кластером. Хороший результат кластеризації має вирішальне значення для надійного розпізнавання об'єктів. Приклад хорошого результату кластеризації можна побачити на рис. 2.4.



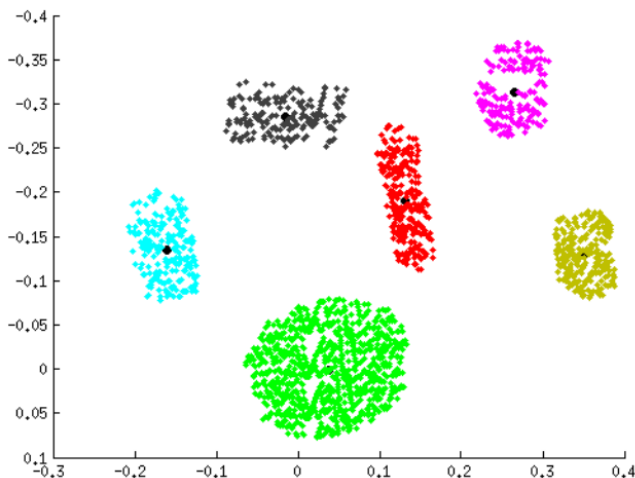


Рис. 2.4. Ідеальний результат кластеризації для 6 об'єктів.  
Різні кольори представляють різні номери кластерів

Існує багато різних підходів до кластеризації даних, всі мають свої переваги та недоліки. У роботі слід реалізувати ієрархічний алгоритм кластеризації. Його основною перевагою є те, що кількість кластерів не потрібно знати заздалегідь (як, наприклад, для  $k$ -середніх). Крім того, ієрархічна кластеризація може справлятися з різними розмірами кластерів і є досить швидкою (на відміну, наприклад, від методу Зсуву середнього значення).

### Основна ідея алгоритму

Основна ідея алгоритму полягає кластеризації точок даної хмари, враховуючи попарну відстань точок одна до одної. На початку кожна точка може розглядатися як власний кластер. Отже, кластери, які є достатньо подібними, об'єднуються. У нашому випадку показником подібності є евклідова відстань між кластерами. Таким чином, кластери, які знаходяться досить близько один до одного, повинні бути об'єднані. Для визначення відстані слід застосувати так званий метод одноканальної кластеризації (SingleLinkage), який визначає відстань між двома кластерами як найкоротшу відстань між двома точками кластерів ( $min\_distance$ ) (рис. 2.5.). Процес злиття припиняється, якщо відстань між усіма кластерами більша за допустиму відстань  $min\_distance < maxdist$ .

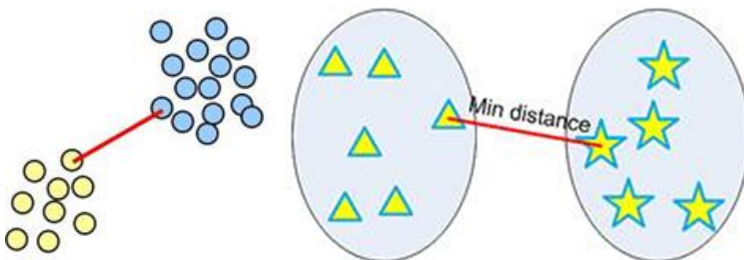


Рис. Графічна інтерпретація найкоротшої відстані між кластерами (Single-Link)

Звичайно, кількість точок у хмарі коливається від сцени до сцени. Часто розмір хмари може бути понад 10 000 точок, що суттєво ускладнює процес швидкої кластеризації. Для вирішення цієї проблеми хмару точок розбивають на підвибірку (виконують субдискретизацію) до того як викликають функцію кластеризації, наприклад аналізується лише кожна десята точка. Такий підхід різко зменшує час обробки. Якщо у подальшому всі точки необхідно пов'язати із кластером, то кожній точці, якій не було присвоєно кластер внаслідок субдискретизації, можна присвоїти номер кластера найближчого сусіда (якому присвоєний номер кластера).

## 2.4. Виконання роботи

Модульна контрольна робота може виконуватись із використанням будь-якої мови та середовища програмування: Python, C++, C#, Matlab.



**При виконанні МКР із використанням Matlab**, доцільно скористатись поясненнями, що надані нижче.

**УВАГА!** При виконанні МКР **не допускається** використання вбудованих функцій та сторонніх бібліотек у яких реалізовано алгоритм RANSAC та метод ієрархічної кластеризації, вбудованої у Matlab функції *linkage*.

## Завдання 2.1. RANSAC

У файлі **fitPlaneRANSAC.m** (MRK2\_RANSAC.rar) необхідно додати власний код алгоритму RANSAC.

*Вхідні параметри функції* – хмара точок  $p$  із набору даних, та наступні налаштування:

confidence – достовірність результату, у відсотках;

inlier\_margin – максимальна відстань від точки до площини, при якій ця точка вважається «inlier» (у метрах);

parameters.sigma1 – мінімальна відстань усіх відібраних точок одна до одної (у метрах);

*Вихідні параметри функції:*

a, b, c, d – параметри площини визначені із рівняння:

$$a \cdot x + b \cdot y + c \cdot z + d = 0$$

inliers – логічний (бінарний) масив  $1 \times N$ , що позначає точки хмари як «inliers». Містить 1 для кожного «inlier» та 0 для «outliers» для відповідної позиції у масиві  $p$ .

Наприклад, inliers = [0, 1, 0, 0, 1, 0, 1, 1, ...] означає, що точки із  $p$  з номерами колонок 2, 5, 7, 8,... є «inliers» та входять до розрахованої площини

k – необхідна кількість ітерацій для досягнення достовірності результату

Додатково, описи *вихідних параметрів* дивитись у коментарях до файлу.

Хмара точок  $p$  – це масив ( $6 \times N$ , де  $N$  – кількість точок у хмарі), що містить наступні дані:

| № рядка | Дані | Одиниці вимірювання      |
|---------|------|--------------------------|
| 1       | x    | відстань, метри          |
| 2       | y    | відстань, метри          |
| 3       | z    | відстань, метри          |
| 4       | r    | яскравість пікселя (0-1) |
| 5       | g    | яскравість пікселя (0-1) |
| 6       | b    | яскравість пікселя (0-1) |

## Завдання 2.2. Кластеризація

Після реалізації алгоритму RANSAC файл **fitPlaneRANSAC.m** (MRK2\_RANSAC.rar) скопіювати до MRK2\_Clustering.rar

```
function [clusters, point2cluster] = clustering(p, maxdist)
```

*Вхідні параметри:*

*p*: хмара точок, що містить усі  $N$  точок, які НЕ є частиною домінуючої площини (викиди після RANSAC).  $6 \times N$  матриця, рядки 1-3:  $x$   $y$   $z$ , рядки 4-6: RGB.

*maxdist*: максимальна відстань у метрах

*Вихідні параметри:*

*clusters*:  $3 \times C$  матриця ( $C$  = кількість кластерів). Містить координати центрів кластерів. Рядки 1-3:  $x$   $y$   $z$ .

*point2cluster*: вектор рядок  $1 \times N$ , який містить пов'язаний номер кластера для кожної точки. Номери кластерів повинні починатись з 1 і бути безперервним (таким чином, номер кластера відповідає індексу стовпця центру кластера)!

### 2.4.3. Корисні команди при реалізації у Matlab

*sort* – сортування масивів.

*mean* – визначення середніх значень елементів масивів

*unique* -- пошук унікальних елементів масивів

## 2.5. Завдання до роботи

7. Реалізувати алгоритм RANSAC
8. Виконати кластеризацію отриманих результатів.
9. Зробити висновки по роботі.
10. Після висновку обов'язково подати відповіді на контрольні питання.

## 2.6 . Зміст документації

8. Тема та мета модульної контрольної роботи.

9. Блок-схема алгоритму RANSAC та кластеризації **fitPlaneRANSAC.m**, **clustering.m** У «шапці» файлів обов'язково вказати ПІБ виконавця та групу.
10. Лістинг алгоритму RANSAC, наприклад з файлу **fitPlaneRANSAC.m** з коментарями і поясненнями.
11. Лістинг файлу **clustering.m** з коментарями і поясненнями.
12. Скріншоти результатів роботи при запуску файлів **main.m** та
13. Перевірити роботу детектора Харріса із зображеннями, що знаходяться у каталозі pointclouds
14. Дати відповіді на контрольні питання.

### 2.7. Контрольні питання

4. Які методи кластеризації Вам відомі?
5. Як ще, на Вашу думку, можна кластеризувати хмари точок?
6. Які вдосконалення методу RANSAC Вам відомі, наведіть їх основні переваги?

## Додаток Б

### Штampi для текстових документів

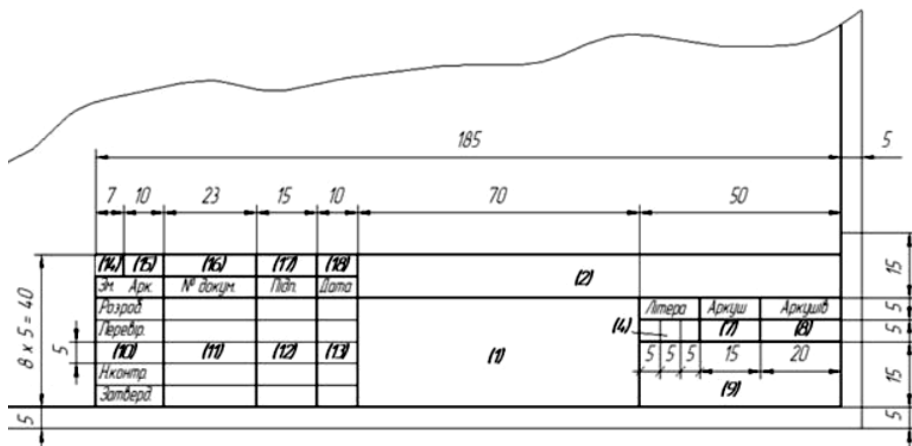


Рис. Б.1. Штамп для текстових документів за формою 2

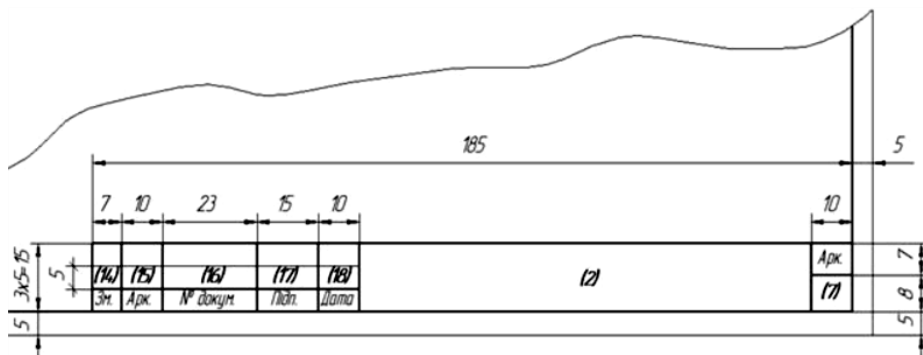


Рис. Б.2. Штамп для текстових документів за формою 2а

**Додаток В**  
**Форма титульного листа звіту з практичних робіт**

|                                                                                     |                         |                                                                                                                                               |
|-------------------------------------------------------------------------------------|-------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------|
| Міністерство освіти і науки України                                                 |                         | 272 № 3,5                                                                                                                                     |
| Національний технічний університет України                                          |                         | 260 № 5                                                                                                                                       |
| «Київський політехнічний інститут імені Ігоря Сікорського»                          |                         |                                                                                                                                               |
| Кафедра XXXXXXXXXXXX<br>група XX-XX                                                 |                         | 230 № 3,5                                                                                                                                     |
| XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX<br>(назва предмету)                           |                         | 180 № 5                                                                                                                                       |
| ЗВІТ З ЛАБОРАТОРНИХ РОБІТ                                                           |                         | 170 № 3,5                                                                                                                                     |
| ЗВІТ З ЛАБОРАТОРНИХ РОБІТ                                                           |                         | 130 № 7                                                                                                                                       |
| Код організації розробника<br>(наприклад, для студента групи АТ-2<br>код буде АТ02) | XXXX. XX 4 2 XX. X00 XX | Код номеру роботи:<br>КР – контрольна робота;<br>ЛР – звіт з лабораторних робіт;<br>ПЗ – пояснювальна записка курсової<br>роботи або проекту. |
| Дві останні цифри залікової книжки                                                  |                         | Номер роботи відповідно до<br>методичних вказівок (від 1 до 10)                                                                               |
| Галузь науки (в університеті<br>технічних наук присвоєна цифра 4)                   |                         | Характер роботи:<br>01 – звичайні роботи;<br>02 – дослідження.                                                                                |
| Вид робіт (для лабораторних робіт<br>прийнята цифра 2)                              |                         |                                                                                                                                               |
| Керівник                                                                            | (підпис)<br>(дата)      | к.т.н., доц. Сазонов А.Ю.                                                                                                                     |
| Виконавець                                                                          | (підпис)<br>(дата)      | прізвище, ім'я, по батькові                                                                                                                   |
| 20XX                                                                                |                         |                                                                                                                                               |

272 № 3,5

260 № 5

230 № 3,5

180 № 5

170 № 3,5

130 № 7

80 № 3,5

40 № 3,5

10 № 3,5

## СПИСОК ЛІТЕРАТУРИ

1. Anil K. Jain, Jianchang Mao, K. M. Mohiuddin Artificial Neural Networks: A Tutorial, Computer, Vol. 29, № 3, March/1996, pp. 31–44.
2. Ballard D. H. Generalizing the Hough transform to detect arbitrary shapes / Ballard D. H. – Pattern Recognition. Vol. 13, Issue 2, 1981, pp. 111–122.
3. C. Lindsey, T. Lindblat. Survey of neural network hardware. SPIE. Vol. 2492, pp. 1194–1205.
4. CMR releases first photographs of Versius // CAMBRIDGE, 06 September 2017 – [Електронний ресурс]. – Режим доступу: [cmrsurgical.com/cmr-releases-first-photographs-versius](http://cmrsurgical.com/cmr-releases-first-photographs-versius) – Назва з екрана.
5. Fuzzy logic air/fuel controlle [Електронний ресурс]. – Режим доступу: <http://www.google.com/patents/US5524599> – Назва з екрана.
6. Hough, P.V.C. Method and means for recognizing complex patterns, U.S. Patent 3,069,654, Dec. 18, 1962. – 6 p
7. Inceptionism: Going deeper into Neural Networks. [Електронний ресурс]. – Режим доступу: [https://photos.google.com/share/AF1QipPX0SCI7OzWilt9LnuQliattX4OUCj\\_8EP65\\_cTVnBmS1jnYgsGQAieQUc1VQWdgQ?key=aVBxWjh wSzg2RjJWLWRuVFBBZEN1d205bUdEMnhB](https://photos.google.com/share/AF1QipPX0SCI7OzWilt9LnuQliattX4OUCj_8EP65_cTVnBmS1jnYgsGQAieQUc1VQWdgQ?key=aVBxWjh wSzg2RjJWLWRuVFBBZEN1d205bUdEMnhB) – Назва з екрана.
8. K. Tahir Shah Automata, neural networks and parallel machines: some emerging principles. – World Scientific. 2002. – 500 p
9. Lowe, David G. Distinctive Image Features from Scale-Invariant Keypoints / D.G. Lowe. – International Journal of Computer Vision 60 (2). – Kluwer Academic Publishers, 2004 – pp. 91–110.
10. Paris S., Kornprobst P., Tumblin J., Durand F. Bilateral Filtering: Theory and Applications / Foundations and Trends in Computer Graphics and Vision: Vol. 4: № 1, pp. 1–73.
11. Pepper, the humanoid robot from SoftBank Robotics, a genuine companion. – [Електронний ресурс]. – Режим доступу:



<https://www.ald.softbankrobotics.com/en/robots/pepper> – Назва з екрана.

12. Posted by Alexander Mordvintsev, Software Engineer, Christopher Olah, Software Engineering Intern and Mike Tyka, Software Engineer Inceptionism: Going Deeper into Neural Networks. Wednesday, June 17, 2015. [Электронный ресурс]. – Режим доступа: <https://research.googleblog.com/2015/06/inceptionism-going-deeper-into-neural.html> – Назва з екрана.

13. Ruth Mary Ripley Neural Network Models for Breast Cancer Prognosis / Ruth Mary Ripley – Trinity Term, 1998. – 116 p.

14. Sangalli A. and Klir G. R. Fuzzy logic goes to market, New Scientist, 8 Feb., 1992 – pp. 156–170

15. Smidth F. L. Computing with a human face. New Scientist, 6 may, 1982. – pp. 203–216

16. SNAP – первый в своем роде нейропроцессор, способный самостоятельно обучаться распознаванию и подсчету определенных объектов в потоке видеоданных // DailyTechInfo – новости науки и технологий, новинки техники [Электронный ресурс]. – Режим доступа: <http://www.dailytechinfo.org/infotech/7927-snap-pervyy-v-svoem-rode-neyroprocessor-sposobnyy-samostoyatelno-obuchatsya-raspoznavaniyu-i-podschetu-opredelennyh-obektov-v-potoke-videodannyh.html> – Название с экрана.

17. SUNSPRING / Фильм, сценарий к которому написал искусственный интеллект. [Электронный ресурс]. – Режим доступа: <https://rutube.ru/video/b384979eab58c5a7f303aab834bc17ba/> – Название с экрана.

18. Szeliski R. Computer Vision: Algorithms and Applications / Szeliski R. – Springer-Verlag London, 2011 – 812 p.

19. H. Hirschmüller and D. Scharstein. Evaluation of cost functions for stereo matching. In IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2007), Minneapolis, MN, June 2007.

20. D. Scharstein, H. Hirschmüller, Y. Kitajima, G. Krathwohl, N. Nescic, X. Wang, and P. Westling. High-resolution stereo datasets with subpixel-accurate ground truth. In German Conference on Pattern

Recognition (GCPR 2014), Münster, Germany, September 2014.

21. Аналитический пакет Deductor Professional: руководство пользователя. – BaseGroupLabs, 2002. – 184 с.

22. Балдин К. В. Информационные системы в экономике: Учебник / К. В. Балдин, В. Б. Уткин. – М. : Издательско-торговая корпорация “Дашков и К<sup>О</sup>”, 2008. – 395 с.

23. Власов А. И. Нейросетевая реализация микропроцессорных систем активной акусто- и виброзащиты / А. И. Власов // Нейрокомпьютеры: разработка и применение. – № 1. – 2000. – С.40–44.

24. Вороновский Г. К. Генетические алгоритмы, искусственные нейронные сети и проблемы виртуальной реальности / Г. К. Вороновский, К. В. Махотило, С. Н. Петрашев, С. А. Сергеев; под ред. Вороновского Г. К. – Харьков: Основа, 1997. – 112 с.

25. Галушкин А. И. Некоторые исторические аспекты развития элементной базы вычислительных систем с массовым параллелизмом (80-и 90- годы) / А. И. Галушкин // Нейрокомпьютер. – № 1. – 2000. – С. 68–82.

26. Глоссарий по нейросетевым терминам. [Электронный ресурс] – Режим доступа: <http://glossary.basegroup.ru> – Название с экрана.

27. Горбань А. Н. Нейроинформатика / А. Н. Горбань, В. Л. Дунин-Барковский, А. Н. Кирдин. – Новосибирск : Наука. Сибирское предприятие РАН, 1998. – 296 с.

28. Горбань А. Н. Нейронные сети на персональном компьютере / А. Н. Горбань, Д. А. Россиев. – Новосибирск : Наука, 1996. – 275 с.

29. Гриняев С. Нечеткая логика в системах управления / С. Гриняев // Компьютерра. – 2001. – № 38. [Электронный ресурс]. – Режим доступа: <http://old.komputerra.ru/offline/2001/415/13052/> – Название с экрана.

30. Гужавин С. А. Средство быстрой разработки экспертных систем “Рапана” [Электронный ресурс]. – Режим доступа: <http://esrapana.narod.ru/> – Название с экрана.

31. Дьяконов В. П. Matlab 6.5 SP1/7/7 SP1/7 SP2 + Simulink 5/6. Инструменты искусственного интеллекта и биоинформатики / В. П. Дьяконов, В. В. Круглов. – М. : СОЛОН-ПРЕСС, 2006. – 456 с.

32. Дьяконов В. П. Matlab 6.5 SP1/7/7 SP1/7 SP2 + Simulink 5/6. Инструменты искусственного интеллекта и биоинформатики / В. П. Дьяконов, В. В. Круглов. – М. : Солон-Пресс, 2006. – 456 с.

33. Дятлов Е. И. Машинное зрение (аналитический обзор) / Е. И. Дятлов // Математичні машини і системи. – 2013. – № 2. – С. 32–40. – Режим доступа: [http://nbuv.gov.ua/UJRN/MMS\\_2013\\_2\\_5](http://nbuv.gov.ua/UJRN/MMS_2013_2_5) (станом на 09.12.2017) – Назва з екрана.

34. Жариков О. Г. Экспертные системы в медицине / О. Г. Жариков, А. А. Литвин, В. А. Ковалёв // Медицинские новости. – 2008. – № 10. – С. 15–18. [Электронный ресурс]. – Режим доступа: <http://www.mednovosti.by/journal.aspx?article=4013> – Название с экрана.

35. Заенцев И. В. Нейронные сети: основные модели / И. В. Заенцев. – Воронеж: “Воронеж”, 1999. – 76 с.

36. Искусственный интеллект: вчера, сегодня, завтра // Компьютерная газета [Электронный ресурс]. – Режим доступа: <http://www.nestor.minsk.by/kg/2005/20/kg52017.html> (По состоянию на 20.06.2016) – Название с экрана.

37. Козлов А. В. Основные направления создания электронных систем, использующих нечеткие управляющие алгоритмы / А. В. Козлов, О. С. Тамер // Вестник Волжского университета им. В.Н. Татищева. – 2011. – № 17. – С. 69–71.

38. Круг П. Г. Нейронные сети и нейрокомпьютеры: учеб. пособие по курсу “Микропроцессоры”. – М. : МЭИ, 2002. – 176 с.

39. Ларионов В. В Китае открылся первый парк роботов / В. Ларионов // Новости высоких технологий [Электронный ресурс] – Режим доступа: <http://hi-news.ru/eto-interesno/v-kitae-otkrylsya-pervyj-park-robotov.html> По состоянию на 20.06.2016) – Название с экрана.

40. Ларионов В. В США начали печатать самоходные маршрутки / В. Ларионов // Новости высоких технологий [Электронный ресурс] – Режим доступа: <http://hi->

[news.ru/technology/v-ssha-nachali-pechatat-samoxodnye-marshrutki.html](http://news.ru/technology/v-ssha-nachali-pechatat-samoxodnye-marshrutki.html) (По состоянию на 20.06.2016) – Название с экрана.

41. Малая экспертная система 2.0. Руководство пользователя [Электронный ресурс]. – Режим доступа: [http://www.aiportal.ru/downloads/expert-systems/mini\\_es\\_2\\_0.html](http://www.aiportal.ru/downloads/expert-systems/mini_es_2_0.html) <http://bourabai.kz/alg/mes2.htm> – Название с экрана.

42. Масленникова О. Е., Гаврилова И. В. Основы искусственного интеллекта / О. Е. Масленникова, И. В. Гаврилова. – М. : Флинта, 2013. – 282с. [Электронный ресурс]. – Режим доступа: [https://books.google.com.ua/books?id=Xy3UAgAAQBAJ&pg=PA236&lpg=PA236&dq=система+искусственного+интеллекта+alvinn&source=bl&ots=bhroCy3alc&sig=EoHpMGUNyv65xOo5yz\\_fIZmQoaU&hl=ru&sa=X&ved=0ahUKEwjphLXL3JPPAhWCDywKHTq\\_AVIQ6AEIIDA B#v=onepage&q&f=false](https://books.google.com.ua/books?id=Xy3UAgAAQBAJ&pg=PA236&lpg=PA236&dq=система+искусственного+интеллекта+alvinn&source=bl&ots=bhroCy3alc&sig=EoHpMGUNyv65xOo5yz_fIZmQoaU&hl=ru&sa=X&ved=0ahUKEwjphLXL3JPPAhWCDywKHTq_AVIQ6AEIIDA B#v=onepage&q&f=false) – Название с экрана.

43. Методы нейроинформатики: сб. науч. труд. Красноярского госуд. техн. ун-та. – Красноярск: КГТУ, 1998. – 205 с.

44. Муромцев Д. И. Введение в технологию экспертных систем. СПб: СПб ГУ ИТМО, 2005. – 107 с.

45. Нейронные сети, нейроинформатика. [Электронный ресурс] – Режим доступа: [nalgolist.manual.ru](http://nalgolist.manual.ru) – Название с экрана.

46. Нейронные сети. Вводный курс лекций по нейронным сетям. [Электронный ресурс] – Режим доступа: [neuronets.chat.ru](http://neuronets.chat.ru) – Название с экрана.

47. Нейронные сети. Искусственная нейронная сеть. Изложен стандартный способ классификации с помощью нейронных сетей. [Электронный ресурс] – Режим доступа: <http://www.neuroproject.ru/neuro.htm>. – Название с экрана.

48. Нейронные сети. Основные теоретические сведения: топология, активационная функция, обучение. [Электронный ресурс] – Режим доступа: <http://www.sociology.kharkov.ua/docs/stat/modules/stneunet.html> – Название с экрана.

49. Нейросети и генетические алгоритмы компании

НейроПроект. [Электронный ресурс] – Режим доступа: <http://www.neuroproject.ru> – Название с экрана.

50. Планування, моделювання та верифікація процесів в гнучких виробничих системах: практикум : навч-метод. посіб. до викон. практ., лабораторних і самостійних занять студентів спеціальності 7.05020201, 8.05020201 "Автоматизоване управління технологічними процесами" всіх форм навчання / І. Ю. Черепанська, В. А. Кирилович, А. Ю. Сазонов, Б. Б. Самотокін / [під. заг. ред. В. А. Кириловича] – Житомир, ЖДТУ 2015. – 274 с.

51. Портал искусственного интеллекта, работы с искусственным интеллектом. Статьи и файлы по основным направлениям исследований в области искусственного интеллекта, описание и виды всех роботов. [Электронный ресурс]. – Режим доступа: <http://www.aiportal.ru/> – Название с экрана.

52. Современные игровые программы / Искусственный интеллект. Системы и модели [Електронний ресурс]. – Режим доступа: <http://www.riai.org.ru/sovremennyye-igrovyie-programmyi.html> – Название с экрана.

53. Современные разработки в области искусственного интеллекта. [Электронный ресурс]. – Режим доступа: <http://www.aiportal.ru/articles/other/actual-developments.html> (По состоянию на 20.06.2016) – Название с экрана.

54. Современный искусственный интеллект. [Электронный ресурс]. – Режим доступа: <https://sites.google.com/site/simplewuu/home/sovremennyj-iskusstvennyj-intellekt> (По состоянию на 20.06.2016) – Название с экрана.

55. Структура экспертной системы. Портал искусственного интеллекта [Электронный ресурс]. – Режим доступа: <http://www.aiportal.ru/articles/expert-systems/structure.html> (По состоянию на 20.06.2016) – Название с экрана.

56. Структурные модели быстрых нейронных сетей: труды II-го Международного симпозиума „Интеллектуальные системы” / [А. Ю. Дорогов, А. А. Алексеев] ред. К. А. Пупков. – Т.2 – М. : Изво ПАИМС, 1996. – С. 138–143.

57. Терехов С. А. Лекции по теории и приложениям искусственных нейронных сетей / С. А. Терехов. – Лаборатория Искусственных Нейронных Сетей НТО-2, ВНИИТФ, Снежинск, 1994. – 120 с.

58. Уоссермен Ф. Нейрокомпьютерная техника: теория и практика / Ф. Уоссермен / [пер. с англ.] – М. : Мир, 1992. – 183 с.

59. Уоссермен Ф. Нейрокомпьютерная техника: Теория и практика – Пер. с англ., 1992. – 118 с.

60. Хабаров С.П. Семантические сети / С. П. Хабаров [Электронный ресурс]. – Режим доступа: <http://www.habarov.spb.ru/bz/bz05.htm> (По состоянию на 08.07.2016) – Название с экрана.

61. Черепанська І. Ю. Використання штучного інтелекту для керування подачею палива інжекторного двигуна / І. Ю. Черепанська, А. В. Гончаренко // Вісник Житом. держ. технол. уні-у. Серія “Технічні науки”. – Житомир : ЖДТУ. – 2014. – № 1 (68). – С. 3–11.

62. Черепанська І. Ю. Теорія та принципи побудови автоматизованої системи для лінійних та кутових переміщень об'єктів виробництва з використанням математичного апарату кватерніонів та елементів штучного інтелекту: монографія / І. Ю. Черепанська, В. А. Кирилович, О. М. Безвесільна, А. Ю. Сазонов, – Житомир: ЖДТУ, 2016. – 326 с.

63. Черепанська І. Ю. Автоматизація процесів керування вибором пристроїв орієнтування при проектуванні гнучких інтегрованих систем: дис. канд. техн. наук: 05.13.07 “Автоматизація процесів керування” / Ірина Юріївна Черепанська. – Київ, 2008. – 380 с.

64. Чиркин С. Ю. Синтез управления двигателем внутреннего сгорания на основе экспериментальных данных: автореф. дис. на соискание науч. степ. канд. тех. наук.: спец. 05.13.06 “Автоматизация и управление технологическими процессами и производствами в машиностроении” / Сергей Юрьевич Чиркин. – М. :, 2010. – 16 с.

65. Штовба С. Д. Введение в теорию нечетких множеств и

нечеткую логику. [Электронный ресурс]. – Режим доступа: <http://matlab.exponenta.ru/fuzzylogic/book1/index.php> – Название с экрана.

66. Штовба С. Д. Проектирование нечетких систем средствами MATLAB. – М. : Горячая линия – Телеком, 2007. – 288 с.

67. Экспертные системы: понятие, назначение, архитектура, отличительные особенности. Классификация ЭС. Этапы разработки ЭС. [Электронный ресурс]. – Режим доступа: <http://daxnow.narod.ru/index/0-17> – Название с экрана.

68. Ямпольский Л. С. Автоматизированные системы технологической подготовки робототехнического производства / Л. С. Ямпольский, О. М. Калинин, М. М. Ткач. – Киев : Высш. шк. Головное изд-во, 1987. – 271 с.

69. Ямпольський Л. С. Гнучкі комп'ютеризовані системи: проектування, моделювання і управління: підручник / Л. С. Ямпольський, П. П. Мельничук, Б. Б. Самотокін та ін. – Житомир: ЖДТУ, 2005. – 680 с.

70. Ямпольський Л. С. Гнучкі комп'ютерно-інтегровані системи: планування, моделювання, верифікація, управління. / Л. С. Ямпольський, П. П. Мельничук, К. Б. Остапченко, О. І. Лісовиченко та ін. – Житомир: ЖДТУ, 2010. – 784 с.

71. Стэнфордские ученые создали нейрокомпьютер, работа которого подражает работе головного мозга // Информационные технологии. – 30 апреля 2014. – [Электронный ресурс] – Режим доступа: [www.dailytechinfo.org/infotech/5856-stenfordskie-uchenye-sozdali-neyrokompyuter-rabota-kotorogo-podrazhaet-rabote-golovnogo-mozga.html](http://www.dailytechinfo.org/infotech/5856-stenfordskie-uchenye-sozdali-neyrokompyuter-rabota-kotorogo-podrazhaet-rabote-golovnogo-mozga.html) – Название с экрана.

# ПРЕДМЕТНИЙ ПОКАЖЧИК

## ***B***

Back Light ..... 15

## ***D***

Dark Field Light ..... 16

Descriptor ..... 20

Diffuse Light ..... 16

DoG ..... 23

## ***F***

Features ..... 20

Features detection ..... 19

Filtering ..... 17

## ***G***

GHT ..... 36

## ***K***

Key point ..... 20

## ***L***

Line Light ..... 15

## ***O***

Object recognition ..... 20

## ***R***

Ring Light ..... 15

## ***S***

SIFT ..... 23

## ***A***

Акумуляторна функція ..... 33

## ***Б***

**Блоб** ..... 22

## ***В***

Види підсвіток ..... 15

Виявлення особливостей  
..... См. Features detection

## ***Г***

Гаусіан ..... 23

Градiєнт ..... 29

## ***Д***

Дескриптор ..... 20, 22, 30

Детектор границь Кенні ..... 22

## ***К***

Ключова точка ..... 20

**Комп'ютерний зір** ..... 12

## ***Л***

Локальне освітлення ..... 13

## ***М***

Масштабований простір  
зображення ..... 24

**Машинний зір** ..... 12

Методи машинного зору .... 12

Методика побудови  
піраміди гаусіан ..... 24

## ***О***

Освітлення ..... 13

Особливості .... См. Features



## **П**

|                                        |    |
|----------------------------------------|----|
| Перетворення Хафа .....                | 30 |
| Підсвітка..... См. Локальне освітлення |    |
| Піраміда гаусіан.....                  | 23 |
| Піраміда різниць гаусіан ....          | 23 |
| Площина параметрів .....               | 32 |
| Простір голосування .....              | 39 |
| Простір Хафа .....                     | 32 |

## **Р**

|                    |    |
|--------------------|----|
| Розпізнавання..... | 21 |
|--------------------|----|

## **У**

|                                    |    |
|------------------------------------|----|
| Узагальнене перетворення Хафа..... | 36 |
|------------------------------------|----|

## **Ф**

|                                     |    |
|-------------------------------------|----|
| Фільтр Гауса .....                  | 18 |
| Фільтрація зображення См. Filtering |    |

## **Ц**

|                    |    |
|--------------------|----|
| Цифровий шум ..... | 17 |
|--------------------|----|

Анатолій Іванович Жученко  
Артем Юрійович Сазонов  
Ірина Юрїївна Черепанська

**СПЕЦІАЛЬНІ РОЗДІЛИ ТЕОРІЇ АВТОМАТИЧНОГО КЕРУВАННЯ.  
ПРАКТИКУМ**

**Навчальний посібник**

**для здобувачів ступеня доктора філософії  
спеціальності 151 “Автоматизація та комп’ютерно-інтегровані технології”**

Відповідальний редактор

*В.С. Цапар*

Редактори

*І. Ю. Черепанська  
А. І. Жученко*

Комп’ютерний набір і верстка

*І. Ю. Черепанська  
А. Ю. Сазонов*

Макетування

*А. Ю. Сазонов*

---

Підписано до друку \*\*.\*\*.20\*\* р. Формат 60×84 1/16 Папір офс.  
Гарнітура Times New Roman. Ум. друк. арк. \*\*. Тираж 200 екз. Замовлення № \*\*\*\*

---

\*\*\*\*\*  
\*\*\*\*\*

Свідectво про внесення до Державного реєстру суб’єктів видавничої справи  
Серія \*\*\*\*\* № \*\*\*\*\* від \*\*\*, \*\*\*, \*\*\*\*\* р.