

Форма № Н-9.02

Міністерство освіти і науки України
Національний аерокосмічний університет ім. М. Є. Жуковського
«Харківський авіаційний інститут»

Факультет систем управління літальних апаратів

Кафедра систем управління літальних апаратів

Пояснювальна записка

до дипломної роботи

магістра

(освітньо-кваліфікаційний рівень)

на тему: Розробка алгоритмів оцінювання траєкторії руху ЛА за результатами
відео спостережень

ХАІ.301.3-96ав.22О.173.007729315 ПЗ

Виконав: студент групи 396ав

Заочна форма навчання

Галузь знань 17 «Електроніка та
телекомунікації»

Спеціальність 173 «Авіоніка»

Освітня програма «Системи
автономної навігації та адаптивного
управління
літальних апаратів»

Лісовін А. О.

(прізвище та ініціали студента)

Керівник Краснов Л. О.

(прізвище та ініціали)

Рецензент Суліма Є.А.

(прізвище та ініціали)

Харків – 2022 рік

Форма № Н-9.01

Міністерство освіти і науки України
 Національний аерокосмічний університет ім. М.Є. Жуковського
 «Харківський авіаційний інститут»

Факультет систем управління літальними апаратами

Кафедра систем управління літальних апаратів

Ступінь вищої освіти магістр

Галузь знань 17 «Електроніка та телекомунікації»

Спеціальність Авіоніка
(шифр і назва)

Освітня програма Системи автономної навігації та адаптивного управління літальних апаратів

ЗАТВЕРДЖУЮ

Завідувач кафедри
 систем управління ЛА

к.т.н., доц. К. Ю. Дергачов

“ ” 20 року

З А В Д А Н Н Я
НА ДИПЛОМНУ РОБОТУ СТУДЕНТУ
Лісовіну Андрію Олександровичу
(прізвище, ім'я, по батькові)

1. Тема роботи Розробка алгоритмів оцінювання траєкторії руху ЛА за результатами відео спостережень

керівник роботи Краснов Л. О., кнт, доцент каф. 301,
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом вищого навчального закладу від 17. 11. 2022 року № 1602 -уч

2. Строк подання студентом роботи: 15 грудня 2022 року

3. Вихідні дані до роботи: програмне забезпечення для визначення траєкторії руху літального апарату.

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити): Аналітичний огляд літератури предметної області, аналіз шляхів та способів рішення завдання, опис та розробка алгоритму визначення руху, розробка програмного забезпечення, розрахунок собівартості виготовлення програмного забезпечення

5. Перелік графічного матеріалу
Слайди 1-10 (формат .ppt)

Дата видачі завдання 01.10.2022

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Стан проблеми	Краснов Л. О., доц. каф № 301		
Аналіз і синтез СУ	Краснов Л. О., доц. каф № 301		
Математичний опис СУ	Краснов Л. О., доц. каф № 301		
Дослідницька частина	Джуглаков В.Г., доц. каф № 301		
Експ.-практ. частина	Краснов Л. О., доц. каф № 301		
Економічне обґрунтування	Попов А.С., доц. каф № 605		

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломної роботи	Строк виконання етапів роботи	Примітка
1.	Початок переддипломної практики	01.10.2022	
2.	Формулювання теми роботи. Розробка технічного завдання	06.10.2022	
3.	Аналіз і синтез методів визначення координат із використанням відеопотоку. Проведення експериментальних досліджень	18.10.2022	
4.	Конструкторська частина роботи. Дослідницька частина роботи. Експериментально-практична частина. Економічне обґрунтування розробки.	22.11.2022	
5.	Оформлення розрахунково-пояснювальної записки і графічного матеріалу	10.12.2022	
6.	Попередній захист роботи	15.12.2022	
7.	Рецензування роботи. Захист роботи в ДЕК	20.12.2022	

Студент _____ Лісовін А. О.
(підпис)

(прізвище та ініціали)

Керівник роботи _____ Краснов Л. О.
(підпис) (прізвище та ініціали)

Міністерство освіти і науки України
Національний аерокосмічний університет ім. М.Є. Жуковського
«Харківський авіаційний інститут»

Кафедра 301

«ЗАТВЕРДЖУЮ»
Завідуючий кафедрою
к. т. н., с. н. с., доцент
_____ К. Ю. Дергачов
«__» _____ 2022 р.

ТЕХНІЧНЕ ЗАВДАННЯ
на дипломне проектування
Лісовіна Андрія Олександровича

1. Тема роботи: Розробка алгоритмів оцінювання траєкторії руху ЛА за результатами відео спостережень затверджена наказом по університету від «17» листопада 2022 р. № N 1602 -уч .
2. Строк здачі студентом закінченої роботи «15» _____ грудня 2022 р.
3. Область застосування розробки: робототехніка, літальні апаратів, наземного транспорт.
4. Початкові дані для системи яка розглядається.
 - 4.1 Призначення і мета створення системи: розробка системи траєкторних вимірювань для оцінки руху літального апарату або його прототипу під час випробувань.
5. Технічні вимоги до системи відеоспостереження: закріплена стаціонарно і не рухомо відеокамера й не рухається в зоні огляду. Формат відеоданих - *avi, розміри кадру 480x640 пікселів, дальність відеоз'йомки — 1км, швидкість - 30 кадрів/с.

На об'єкті дослідження встановлений маркер, положення якого вважається за положення літального апарату.

 - 5.1 Питання, що підлягають розробці: Вибрати метод та алгоритм виявлення об'єкта спостереження у відеокадрі, який забезпечить максимальну точність траєкторних вимірювань, простоту реалізації програмними засобами та гарну стійкість до перешкод. Порівняти з іншими існуючими методами виявлення. Оцінку траєкторних параметрів об'єкта проводити в системі координат відеокамери щодо центру кадру. Перерахунок траєкторних параметрів в абсолютну систему координат у рамках цього технічного завдання не проводити.
 - 5.2 Вимоги до структури й функціонування системи: в якості засобу програмування використовується система Matlab.

6. Задачі та методи роботи: дати докладний опис роботи алгоритму виявлення об'єкта у кадрі та побудувати траєкторію його руху у відеопотоці. Докладно описати програмну реалізацію цих процедур. Детально розглянути методи обчислення траєкторних параметрів (дальностей, кутових координат тощо) та дати їх програмну реалізацію.

7. Обсяг виконуваних розроблювачем робіт.

7.1 Етапи проведення роботи: 1) вибір теми; 2) розробка ТЗ на проектування ПЗ; 3) Оцінка стану проблеми, аналіз ТЗ; 4) математичний опис СУ. Дослідження СУ; 5) розробка функціональної схеми СУ; 6) статичний розрахунок СУ; 7) проектування ПЗ; 8) експериментальна частина; 9) оформлення записки; 10) захист дипломного проекту

8.2 Обсяг розробки по кожному етапу: технічне завдання (2л); реферат (1л); зміст(1л); вступ (2л); аналітичний огляд літератури і існуючих методів вирішення задач(21л); дослідження проблеми опис (25л); економічна частина (10л);

9. Вимоги до захисту інформації й надійності:

10. Дослідницька частина: опис програми, вирішення питання оцінювання траєкторії руху за результатами відео спостережень.

11. Економічна частина

11.1 Розробити (розрахувати, одержати): розрахувати собівартість і ціну розробки системи стереозору;

Керівник проектування

Краснов Л. О.

(П.І.Б.)

« » _____ 2022 р.

Прийняв до виконання

Лісовін А. О.

(П.І.Б. студента)

« » _____ 2022 р.

Погоджено з питань:

проектування

Краснов Л.О.

(П.І.Б.)

« » _____ 2022 р.

дослідницької частини

Джуглаков В.Г.

(П.І.Б.)

« » _____ 2022 р.

економіки

Попов А.С.

(П.І.Б.)

« » _____ 2022 р.

РЕФЕРАТ

Дипломна робота на тему «Розробка алгоритмів оцінювання траєкторії руху ЛА за результатами відео спостережень» 68 містить сторінок, 14 таблиць, 32 рисунка, 40 формул. Перелік посилань нараховує 12 найменувань.

Актуальність роботи. Актуальність теми дослідження зумовлено необхідністю пошуку нових методів та алгоритмів керування літальними апаратами.

Дипломну роботу виконано в Національному аерокосмічному університеті ім. М.Є. Жуковського «Харківський авіаційний інститут». Метою роботи є узагальнення теоретичних засад та обґрунтування механізмів обробки відеопотоку, завдяки якому ми можемо оприділити заздалегідь описаний об'єкт та оцінити напрям, швидкість і координати його руху.

Завдання дослідження: – отримати траєкторію руху об'єкта, на постійному фоні. Об'єкт завжди знаходиться в полі зору і з поля зору не зникає.

Ключові слова: *відеопотік, технічний зір, матриця, координати.*

Анотація

В дипломному проекті представлено огляд по сучасним електронним системам технічного зору для розпізнавання образів і траєкторії руху. Розроблено систему, що відповідає поставленому завданню.

Summary

The diploma project presents an overview of modern electronic technical vision systems for pattern recognition and movement trajectory. A system that corresponds to the task has been developed.

ЗМІСТ

ВСТУП.....	8
1. ОГЛЯД ЛІТЕРАТУРИ ТА ІСНУЮЧИХ МЕТОДІВ ВИРІШЕННЯ ЗАДАЧІ .	10
1.1. ВИБІР МЕТОДУ РОЗВ'ЯЗАННЯ ЗАДАЧІ РОЗРОБКИ АЛГОРИТМІВ ОЦІНКИ ТРАЄКТОРІЇ РУХУ ЛА ЗА ВІДЕОДАНИМИ	ERROR! BOOKMARK NOT DEFINED.
1.2. ОЦІНЮВАННЯ ПАРАМЕТРІВ ТРАЄКТОРІЇ РУХУ ОБ'ЄКТІВ НА ОСНОВІ ВІДЕОДАНИХ ЗА ДОПОМОГОЮ НЕРУХОМОЇ КАМЕРИ	13
1.3. ОСНОВНІ ФУНКЦІЇ МАТЛАВ, ЩО ВИКОРИСТОВУЮТЬСЯ ДЛЯ ВИРІШЕННЯ ЗАВДАННЯ	13
1.4. ОПИС АЛГОРИТМУ І ПРОГРАМИ ОЦІНЮВАННЯ ТРАЄКТОРІЇ ЗА ВІДЕОДАНИМИ.....	14
1.5. ОСОБЛИВОСТІ ПОБУДОВИ ПРОГРАМИ	16
1.6. РОЗРАХУНОК ІНФОРМАЦІЙНИХ ПАРАМЕТРІВ ТОЧОК ТРАЄКТОРІЇ.....	19
1.7. ОЦІНЮВАННЯ ПОХИБОК ОБЧИСЛЕННЯ ПАРАМЕТРІВ ТРАЄКТОРІЇ РУХОМОГО ОБ'ЄКТА.....	200
1.8. ПРИКЛАДИ ВИКОРИСТАННЯ РОЗРОБЛЕНОГО АЛГОРИТМУ	233
2. ОЦІНЮВАННЯ ПАРАМЕТРІВ ТРАЄКТОРІЇ РУХУ ОБ'ЄКТІВ ЗА РЕЗУЛЬТАТАМИ ВІДЕОСПОСТЕРЕЖЕННЯ,	27
2.1. ОПИС ПРОГРАМИ ОЦІНЮВАННЯ ПАРАМЕТРІВ ТРАЄКТОРІЇ РУХУ ОБ'ЄКТА (ДАЛЬНОСТІ, КУТОВОЇ КООРДИНАТИ І ВЕКТОРА ШВИДКОСТІ)	27
3. ПЕРЕТВОРЕННЯ СИСТЕМИ КООРДИНАТ	29
3.1 Короткі теоретичні відомості.....	29
3.2 Обертання фігури в тривимірному просторі.....	32
3.3 Матриця повороту в тривимірному просторі.....	32
3.4 Тривимірне обертання довкола осей координат	34
3.5 Формули напрямних косинусів	34
3.6 Система координат цифрового знімка з початком в його площині	36
3.7 Система координат знімка з початком у центрі його проєкції	37
3.8 Система координат об'єкта	39
3.9 Формули перетворення координат між системами координат похилого знімка і об'єкта	40
3.10. Перетворення координат із системи координат горизонтального знімка в систему координат об'єкта	42
4. ЕКОНОМІЧНА ЧАСТИНА.....	43
4.1 Мета економічної частини	43
4.2 Сегментування ринку	43
4.3 Аналіз конкурентоспроможності.....	45
4.4 Розрахунок собівартості і ціни виготовлення установки	46
4.5 Висновки по економічній частині.....	51
ВИСНОВОК	5252
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	53
ДОДАТКИ.....	54

ВСТУП

Розвиток технологій обробки зображень сприяє інтенсивному впровадженню відеоінформаційних комплексів для вирішення задач навігації, космічного моніторингу Землі, забезпечення безпеки найважливіших об'єктів, передачі стиснутої відеоінформації, використання в системах управління рухом роботів та ін.

Одним із напрямків створення цих комплексів є розробка систем аналізу та обробки зображень, які призначені, в першу чергу, для встановлення на рухомих носіях, таких як літаки, гелікоптери, автомобілі, безпілотні літальні апарати та інші транспортні засоби. Призначенням подібних систем є вирішення комплексу задач виявлення рухомих та нерухомих об'єктів, оцінки їх параметрів, розпізнавання, побудови траєкторій руху

Однією з центральних завдань у галузі машинного зору (Computer Vision) є завдання визначення орієнтації та розташування камери у просторі зображення, отриманому з її допомогою. Це завдання також називають завданням зовнішнього калібрування камери або визначення зовнішніх параметрів. Вирішення цього завдання потрібно у різних галузях: картографії, системах розпізнавання об'єктів, системах розширеної реальності, системах управління комп'ютером у вигляді визначення становища рук чи напрямки погляду, системах управління роботами, в системах траєкторних вимірів за результатами відеоспостереження тощо.

На сьогоднішній день велика кількість наукових досліджень проводиться якраз в рамках комп'ютерного зору, який останнім часом отримує все більше визнання та застосування. Зростаючий інтерес до аналізу руху останнім часом сильно мотивований покращення алгоритмів та моделей комп'ютерного зору, доступністю недорогого обладнання, такого як відео камери та різноманітні нові перспективні програми які можуть проводити як ідентифікацію особи так і візуальне спостереження.

Знаходження руху об'єктів може сприяти розпізнаванню об'єктів. Таким чином, основна мета дослідження полягає в тому, щоб розпізнати пікселі, що належать одному об'єкту. В даний час для виявлення рухомих об'єктів використовується в основному кадровий метод віднімання, метод віднімання фону та метод оптичного потоку. Метод оптичного потоку полягає в обчисленні поля оптичного потоку зображення і виконанні кластера обробка відповідно до характеристик розподілу оптичного потоку зображення. Метод фонового віднімання полягає у використанні методу відмінності поточного зображення та фонового зображення для виявлення переміщення об'єктів. Даний підхід є дуже чутливими до змін зовнішнього довкілля і має погану здатність до захисту від 4 перешкод. У рамковому методі віднімання наявність рухомих об'єктів визначається шляхом обчислення різниці між двома послідовні зображення. Будь-яка система виявлення руху на основі фону завжди буде чутлива до таких зовнішніх факторів як:

- зашумленість зображення через неякісне джерело зображення;
- поступові зміни умов освітлення в сцені;
- невеликі рухи нестатичних об'єктів, таких як гілки дерев та куці тощо.

Метою роботи є розробка алгоритмів для системи траєкторних вимірювань руху літального апарату чи його прототипу під час випробувань.

Для досягнення даної мети ставились наступні завдання:

- провести аналіз та класифікацію існуючих рішень щодо створення систем траєкторних вимірювань засобами технічного зору;
- проаналізувати методи аналізу зображень у відеопотоках;
- провести огляд наявних програмних засобів для виділення та класифікації об'єктів на зображеннях;
- проаналізувати існуючі методи та підходи до виділення та класифікації об'єктів на зображеннях;
- розробити алгоритми розпізнавання об'єктів на зображенні та створити алгоритми оцінки параметрів їх руху у відеопотоці;

1 ОГЛЯД ЛІТЕРАТУРИ ТА ІСНУЮЧИХ МЕТОДІВ ВИРІШЕННЯ ЗАДАЧІ

1.1 Вибір методу розв'язання задачі розробки алгоритмів оцінки траєкторії руху ЛА за відеоданими

Траєкторія — лінія, яку описує матеріальна точка, що рухається в просторі. Залежно від форми траєкторії розрізняють прямолінійний та криволінійний рухи точки.

Загалом, траєкторія – це дані, які містять лише мінімальну інформацію, наприклад положення та час, а також ідентифікатор об'єкта стеження. Ця інформація може легко доповнюватися детальною інформацією, такою як швидкість, прискорення та рух, оскільки вони можуть бути витягнуті з початкових даних траєкторії.

Траєкторія τ або T для транспортного засобу визначається як тривимірна послідовність точок, де перші дві координати представляють положення транспортного засобу в 2D просторі і третя координата представляє час:

$$\tau = (x_1, y_1, t_1), (x_2, y_2, t_2), \dots (x_n, y_n, t_n)$$

Об'єктом, що відстежуються може бути будь-що визначене як важливе для подальшого аналізу. Об'єкт визначаються за формою та виглядом. При аналізі руху нерідко проводять дослідження оптичного потоку. Є три етапи виділення динамічних об'єктів: попередня стабілізація параметрів зображення, виділення рухомих точок або областей, порівняння рухомих точок і областей між кадрами. Алгоритми знаходження руху виділяють рухомі точки та області. Розрахунок оптичного потоку (ОП) зазвичай проводять глобальними і локальними диференційними алгоритмами. Для швидкого розрахунку оптичного потоку використовують локальні диференційні алгоритми, але точність при використанні цього методу може бути недостатньою для стабільного виділення рухомих об'єктів. При використанні глобальних методів розрахунку оптичний потік є більш точним, але із-зі високої складності розрахунку не завжди цей метод може бути використаний у реальних задачах.

Автоматичне виділення і супроводження є методом цифрового оброблення, який дозволяє знаходити траєкторію руху заданого об'єкта на послідовності зображень і виконувати пошук і визначення його (їх) координат на кожному кадрі відеопослідовності.

Завдання розглядають в умовах нерухомої камери і відносно постійного фону. Воно є найбільш простим завданням, оскільки його вирішення не потребує застосування складних методів аналізу отриманої інформації про рух. Подібні завдання, як правило, виникають при розробленні охоронних систем слідкування, в яких необхідно виявити несанкціонований доступ на територію закритого об'єкта, де, наприклад, виключено рух в певний час доби.

Розглянемо алгоритми, що дозволяють знаходити траєкторію руху заданого об'єкта на послідовності зображень і передбачають пошук і визначення його координат на кожному кадрі відеопослідовності. Часто для вирішення поставленого завдання вибирають кореляційний метод, оскільки він є досить простим в реалізації, не потребує істотних обчислювальних ресурсів і забезпечує досить високу швидкодію.

Цей метод оснований на пошуку максимуму взаємно кореляційної функції $K(x, y)$ між зображенням $I(x, y)$ розмірністю $M \times N$ і зображенням $S(x, y)$, яке називають маскою або шаблоном. Суть методу пошуку максимуму взаємної кореляції наочно проілюстровано на рис. 1.1.

Усі кадри аналізованої відеопослідовності є зображенням розмірністю $M \times N$ пікселів. Наприклад, відеокадри у форматі *.avi мають розмірність 480x640 пікселів. Для формування шаблону використовують зображення, розміри якого приблизно відповідають розмірам об'єкта спостереження.

Формування взаємно кореляційної функції виконується як операція двовимірного згортання зображення шаблону $S(x, y)$ із зображенням $I(x, y)$. Для цього необхідно обчислити значення взаємно кореляційної функції $K(x, y)$ в окремій точці (x, y) і повторити процедуру шляхом послідовного переміщення шаблону $S(x, y)$ по всіх елементах зображення $I(x, y)$.

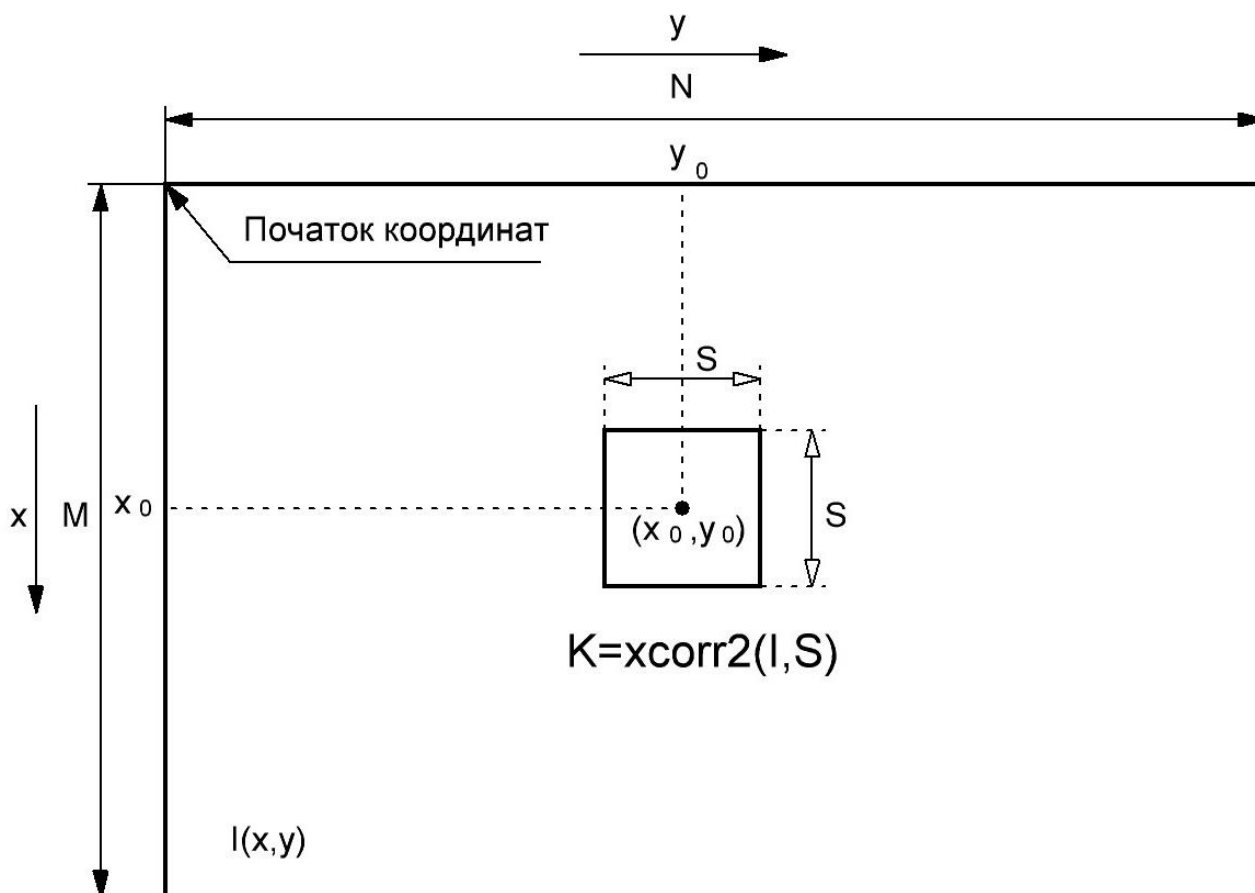


Рисунок 1.1. – Схема пошуку максимуму взаємно кореляційної функції

Як шаблон часто використовують одиничну функцію вигляду $S=ones(n,n)$, розміри якої порівняно з розмірами спостережуваного об'єкта.

Обчислення взаємно кореляційної функції з використанням одиничного шаблону практично є еквівалентним процедурі двовимірної просторової маскової низькочастотної фільтрації зображення.

Для практичної реалізації програм оцінювання траєкторії руху об'єкта відповідно до відеопослідовності необхідно використовувати апріорні дані про властивості як спостережуваного об'єкта в поточному кадрі, так і фону реєстрованої у відеокадрі сцени.

Завдання може бути успішно вирішено лише за умови істотного перевищення яскравості спостережуваного об'єкта над яскравістю фону в аналізованому кадрі. В разі рівномірного статичного фону і великої яскравості об'єкта спостереження можна безпосередньо використовувати алгоритм обчислення взаємно кореляційної функції і пошуку її максимуму. Якщо ж фон кадру має більш складну структуру через нерівномірність освітлення сцени, продуктивнішим є попереднє прибирання фону. Для цього найбільш відповідним виявляється кадр, в якому є відсутнім об'єкт спостереження. При динамічній зміні характеристик фону для процедури прибирання використовують попередній кадр.

При виконанні зазначених вище умов координати точки максимуму обчисленої взаємно кореляційної функції мають відповідати координатам положення спостережуваного об'єкта в даному кадрі відеопослідовності.

Зазвичай завдання проведення траєкторних вимірювань літальних апаратів прийнято вирішувати за допомогою радіотехнічних систем, що є комплексом дорогих антенних і апаратних засобів. При цьому складно досягти високої точності вимірювань. Істотно простіше досягти мети високоточних траєкторних вимірювань за допомогою системи відеоспостереження, коли відеокамера у заданій системі координат нерухомо закріплена. При цьому на борту літального апарату прийнято розміщувати невеликий маркер, яскравість якого перевищує яскравість відеозйомки. За становищем маркера в кадрі судять про становище ЛА на даний момент часу.

Однак, зважаючи на сучасні обставини, що склалися у нашій країні, тобто повномасштабне вторгнення, для виконання дипломної роботи неможливо було виконати дослідження у повному обсязі, тобто зробити дослідження на реальному літаючому апараті або макеті було неможливо. Замість цього були використані відеоматеріали із записом траєкторії руху маркера, розміри та колір якого відповідають розмірам та кольору маркера на борту літального апарату.

Для практичної реалізації програм оцінювання траєкторії руху об'єкта відповідно до відеопослідовності необхідно використовувати апріорні дані про властивості як спостережуваного об'єкта в поточному кадрі, так і фону реєстрованої у відеокадрі сцени.

Таким чином, завдання вирішувалося методом імітаційного моделювання. Для цього були створені та використані імітаційні відеофайли, властивості яких будуть описані далі.

1.2 Оцінювання параметрів траєкторії руху об'єктів на основі відеоданих за допомогою нерухомої камери

Завдання може бути успішно вирішено лише за умови істотного перевищення яскравості спостережуваного об'єкта над яскравістю фону в аналізованому кадрі. В разі рівномірного статичного фону і великої яскравості об'єкта спостереження можна безпосередньо використовувати алгоритм обчислення взаємно кореляційної функції і пошуку її максимуму. Якщо ж фон кадру має більш складну структуру через нерівномірність освітлення сцени, продуктивнішим є попереднє прибирання фону. Для цього найбільш відповідним виявляється кадр, в якому є відсутнім об'єкт спостереження. При динамічній зміні характеристик фону для процедури прибирання використовують попередній кадр.

При виконанні зазначених вище умов координати точки максимуму обчисленої взаємно кореляційної функції мають відповідати координатам положення спостережуваного об'єкта в даному кадрі відеопослідовності.

1.3 Основні функції Matlab, що використовуються для вирішення завдання

При обробленні кадрів відеопослідовності необхідно застосовувати різного роду перетворення типів зображень і формату використовуваних даних. Основні matlab-функції, що застосовують у цьому разі, наведено в табл. 1.1.

Таблиця 1.1

Перетворення типів зображень

Функція	Перетворення
<code>I=rgb2gray(RGB)</code>	Перетворення повнокольорового зображення в півтонове
<code>R=RGB(:,:,1)</code> <code>G=RGB(:,:,2)</code> <code>B=RGB(:,:,3)</code>	Виділення червоної компоненти Виділення зеленої компоненти Виділення синьої компоненти
<code>YIQ=rgb2ntsc(RGB)</code>	Складова Y містить інформацію про яскравість зображення, а складові I і Q – про його кольоровість
<code>I1=im2double(I)</code>	Перетворення півтонового зображення у формат double

BW=im2bw(S,threshold)	Бінаризація з відсіканням по порогу яскравості
D=imcrop(S,rect)	Вирізування фрагмента зображення

У Matlab для вирішення завдань кореляційного аналізу використовують такий набір стандартних функцій:

1. Функція оцінювання середнього значення елементів матриці (mean2). Її синтаксис описують так: $m = \text{mean2}(S)$. За допомогою функції $m = \text{mean2}(S)$ обчислюють середнє значення елементів матриці S . Дана функція є еквівалентною $\text{mean}(S(:))$.

2. Функція обчислення середньоквадратичного відхилення елементів матриці (std2). Ця функція має синтаксис $D = \text{std2}(S)$.

3. Функція обчислення коефіцієнта кореляції між двома матрицями (corr2). Ця функція має синтаксис $k = \text{corr2}(A, B)$.

На основі функції $k = \text{corr2}(A, b)$ обчислюють коефіцієнт кореляції між матрицями A і B однакового розміру, використовуючи алгоритм обчислення коефіцієнта кореляції за допомогою співвідношення

$$k = \frac{\sum_c \sum_r (A(C, r) - A_m) \cdot (B(C, r) - B_m)}{\sqrt{(\sum_c \sum_r (A(C, r) - A_m)^2) \cdot (\sum_c \sum_r (B(C, r) - B_m)^2)}}$$

де $A_m = \text{mean2}(A)$, $B_m = \text{mean2}(B)$.

За допомогою функції $D = \text{std2}(S)$ розраховують середньоквадратичне відхилення елементів матриці S . Ця функція еквівалентна $\text{std}(S(:))$.

4. Функція обчислення двовимірної взаємної кореляційної функції (xcorr2). Ця функція має синтаксис

$c = \text{xcorr2}(a, b)$

$c = \text{xcorr2}(a)$

На основі функції $c = \text{xcorr2}(a, b)$ обчислюють двовимірну взаємну кореляційну функцію з двох двовимірних сигналів за вибірками, заданими в матрицях a і b . За допомогою функції $c = \text{xcorr2}(a)$ розраховують автокореляційну функцію, що є еквівалентною функції $c = \text{xcorr2}(a, a)$.

1.4 Опис алгоритму і програми оцінювання траєкторії за відеоданими

Приведемо приклад реалізації алгоритму оцінювання траєкторії руху кольорової (в цьому разі синьої) кульки на рівномірному статичному фоні. Відеозйомку ведуть нерухомою камерою. Цей сценарій міститься у файлі `blue_ball.avi`. Структурну схему алгоритму оброблення відеоданих показано на рис. 1.2. Ця схема відображає основні перетворення вхідної відеопослідовності. Через використання алгоритму передбачають введення шумової компоненти і кількісний аналіз характеристик траєкторії руху залежно від рівня компоненти завади.

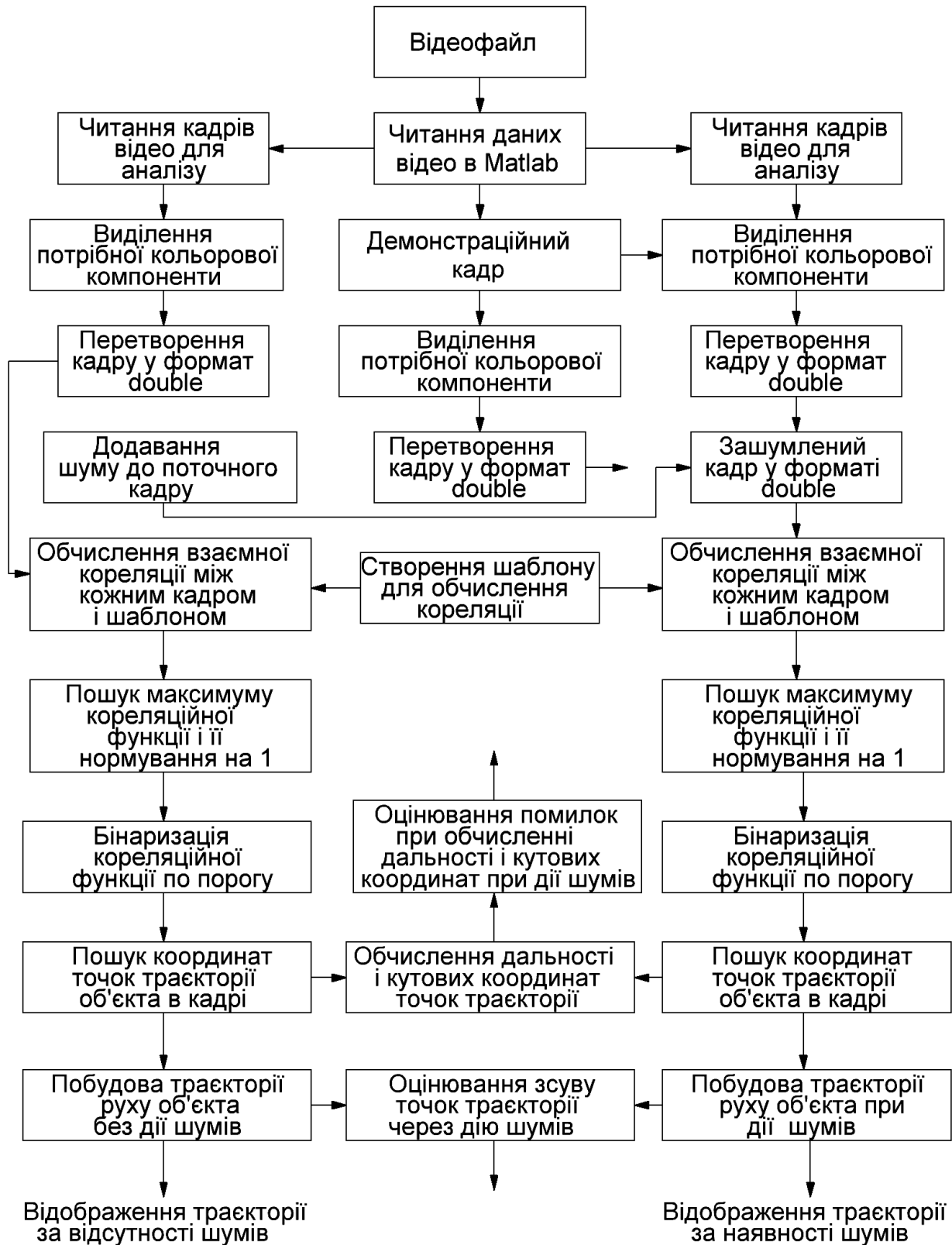


Рисунок 1.2 – Структурна схема алгоритму оцінювання руху об'єкта на основі відеоданих методом кореляційного аналізу зображень

1.5 Особливості побудови програми

Розглянемо детальніше деякі особливості запропонованої програми. На рис. 1.3 показано демонстраційний кадр вихідної відеопослідовності. Як було відзначено раніше, для стійкої роботи алгоритму необхідно забезпечити максимальне перевищення яскравості рухомого об'єкта відносно яскравості фону. Саме з цієї причини в програмі з RGB-зображення виділяється синя компонента, яка потім перетворюється у формат double (рис. 1.4).

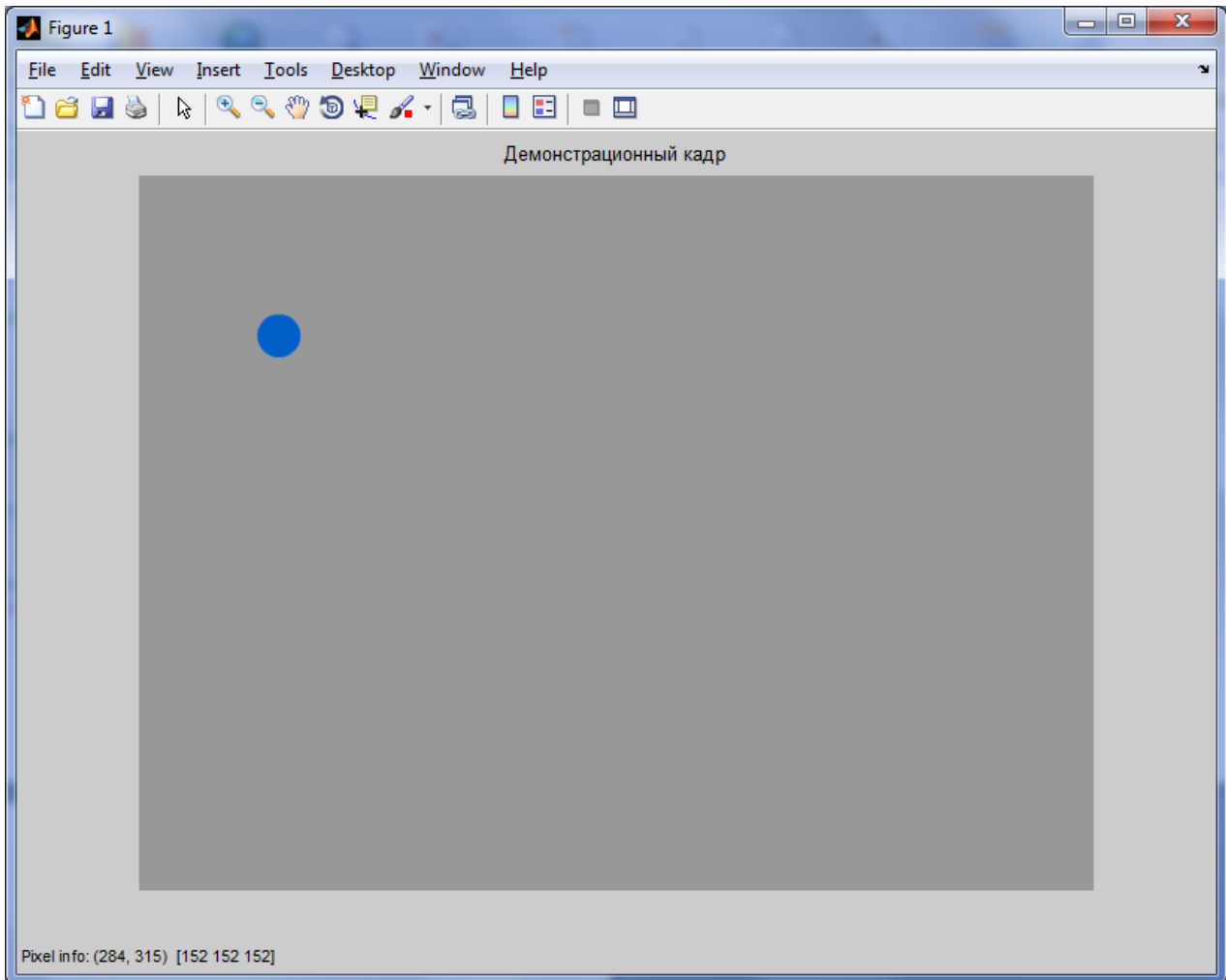


Рисунок 1.3 – Поточний кадр вихідної відеопослідовності

Відмітимо, що перетворення відеокадру у формат double є необхідним для обчислення взаємно кореляційної функції поточного кадру з шаблоном. Як шаблон було використано функцію $S=ones(25,25)$, розміри якої в пікселях приблизно дорівнюють розмірам синьої кульки.

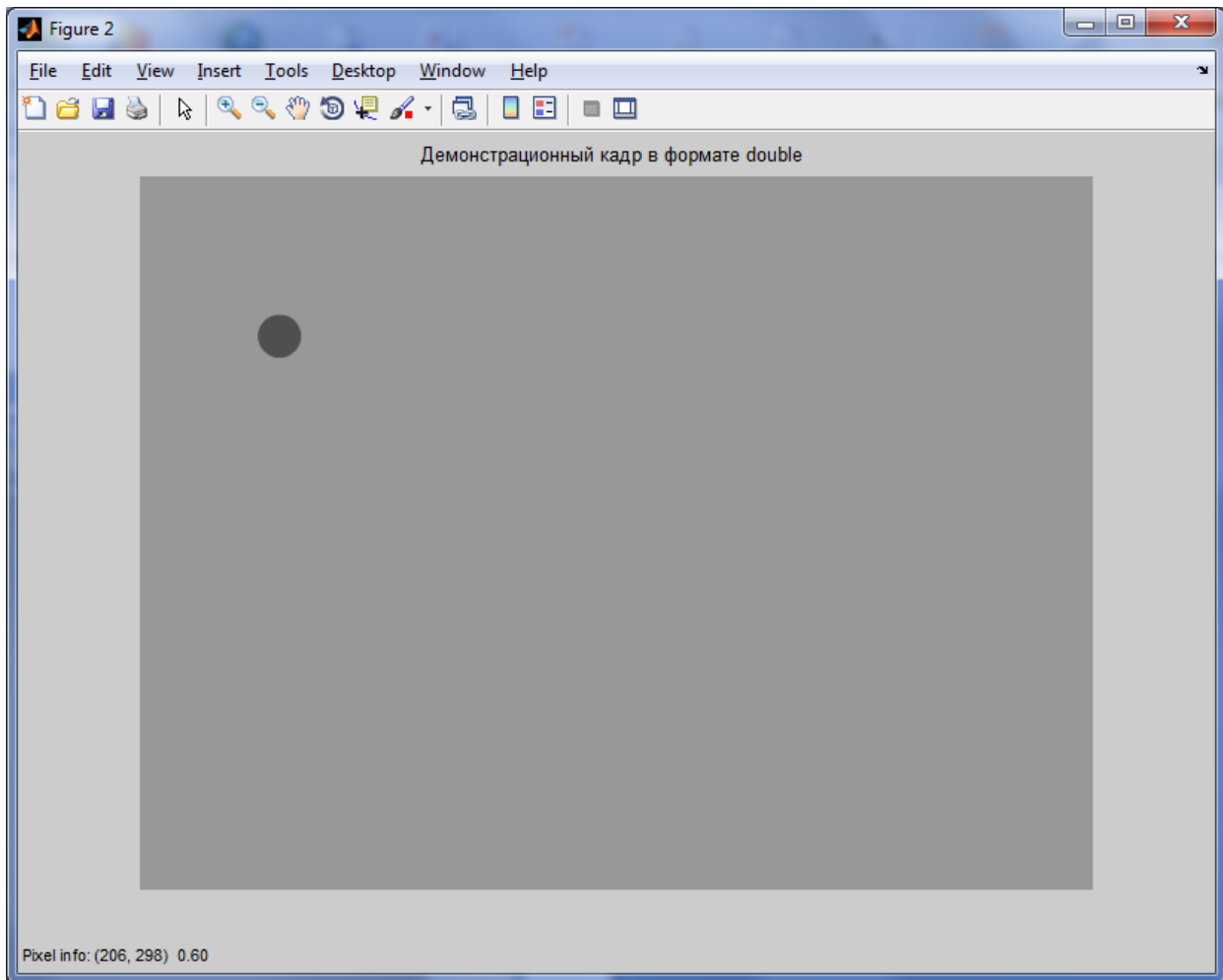


Рисунок 1.4 – Поточний кадр у форматі double

Для створення шумової компоненти використовували функцію гауссова шуму вигляду $F1=imnoise(F,'gaussian',0,q)$ з нульовим середнім і параметром дисперсії q , значення якого можна заздалегідь установлювати в діапазоні 0.05 ... 0.5 для дослідження стійкості алгоритму до дії перешкод. Приклад зашумленого відеокадру показано на рис. 1.5.

Процедура пошуку взаємно кореляційної функції поточного кадру з шаблоном відбувається за допомогою функції $K=xcorr2(I,p)$, а потім визначається максимум взаємно кореляційної функції і нормування цієї функції на 1. Після цього здійснюється бінаризація кореляційної функції з відсіканням по порозу яскравості за допомогою функції $BW=im2bw(K,p)$. Поріг відсікання слід вибирати дуже високим ($p>0.9$) для того, щоб надійно локалізувати точку максимуму кореляції.

Таким чином, після перетворень з поточного кадру отримаємо кадр з нульовими значеннями всіх пікселів, окрім однієї точки, положення якої визначається положенням центру рухомого об'єкта (кульки). Яскравість цього пікселя дорівнює 1. Процедура циклічного складання оброблених кадрів всієї відеопослідовності дозволяє отримати зображення траєкторії руху спостережуваного об'єкта (рис. 1.6).

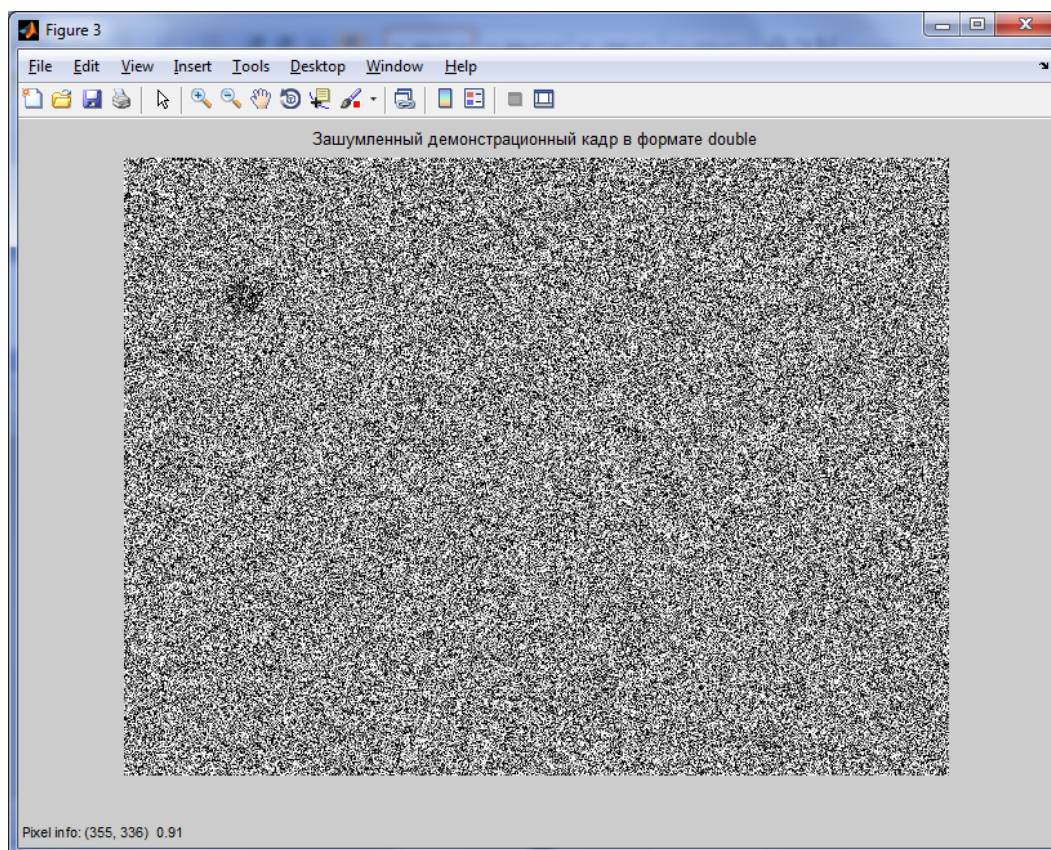


Рисунок 1.5 – Зашумлений кадр відеопослідовності у форматі double

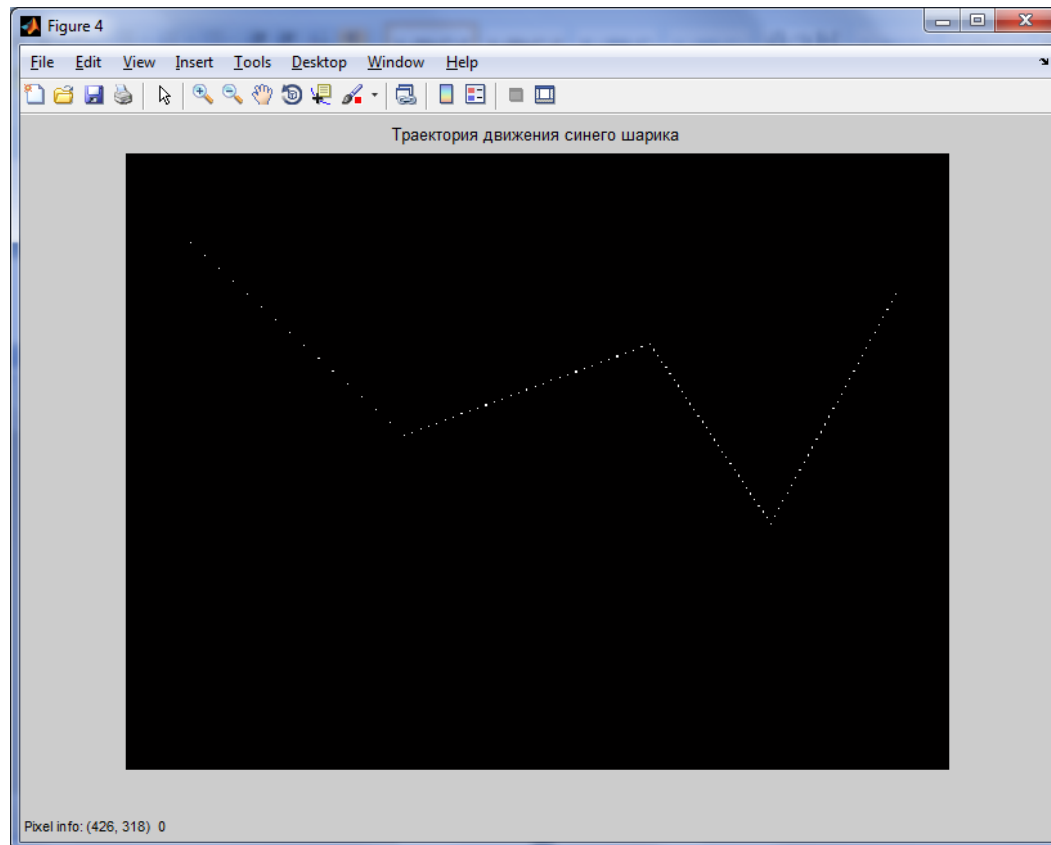


Рисунок 1.6 – Траекторія руху кульки

1.6 Розрахунок інформаційних параметрів точок траєкторії

Відображення побудованої траєкторії (див. рис. 1.6) є цілком придатним для досягнення цілей візуалізації. Проте в системах технічного зору, що орієнтовані на вирішення завдань траєкторних вимірів або управління рухом, необхідно отримувати кількісні характеристики параметрів траєкторії. Зазвичай для цього використовують параметр дальності до спостережуваного об'єкта і оцінювання його кутового положення відносно початку координат.

При розрахунку цих параметрів необхідно вибрати потрібну для вирішення поставленого завдання систему координат. Часто при вирішенні навігаційних завдань і завдань управління використовують симетричну прямокутну систему координат з відліком кутової координати в діапазоні $0 \dots 360^\circ$. На рис. 1.7 показано схему перетворення системи координат відеокадру в потрібну систему.

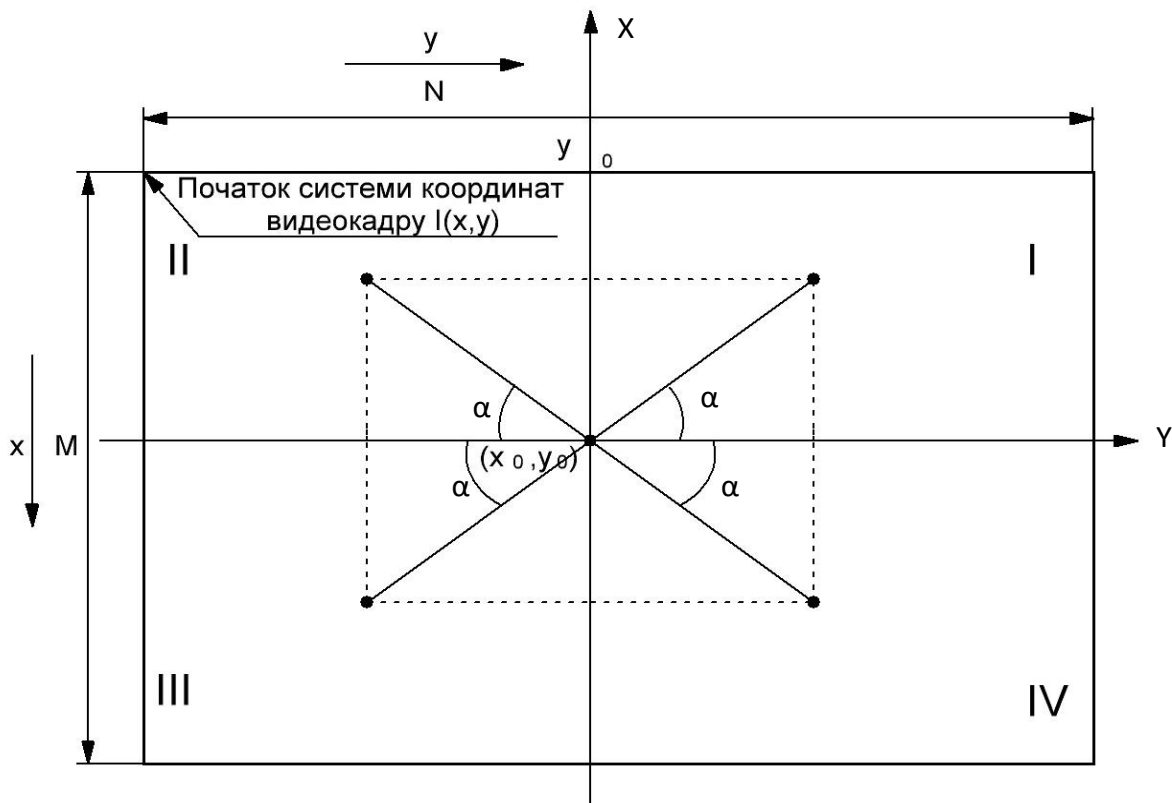


Рисунок 1.7 – Схема перетворення системи координат відеокадру в симетричну систему

Оскільки в нашому випадку використовуються відеокадри розмірністю 480×640 пікселів з початком координат у лівому верхньому куті (піксель з номером 1,1), то нова система координат утворюється зміщенням на 240 і 320 пікселів по відповідних осях. Рахуватимемо піксель з координатами (241, 321) нульовою точкою потрібної нам прямокутної системи координат і всі подальші розрахунки вестимемо відносно неї.

Обчислення дальності від початку координат до поточної точки траєкторії не є складним – її визначають як гіпотенузу прямокутного трикутника з внутрішнім кутом α , утвореного проєкціями точки на осях координат dx і dy . За допомогою обчислення кутового положення точки передбачають попереднє визначення належності координат точки до того або іншого квадранта в системі координат. Це дозволяє перерахувати внутрішній кут трикутника α в діапазоні $0 \dots 360^\circ$ за правилом:

$$\begin{array}{ll} \text{I квадрант} \rightarrow \text{Angle}=\alpha ; & \text{II квадрант} - \text{Angle}=\mathbf{180^\circ} - \alpha ; \\ \text{III квадрант} \rightarrow \text{Angle}=\mathbf{180^\circ} + \alpha ; & \text{IV квадрант} \rightarrow \text{Angle}=\mathbf{360^\circ} - \alpha . \end{array}$$

1.7 Оцінювання похибок обчислення параметрів траєкторії рухомого об'єкта

Важливим аспектом цієї роботи є кількісне оцінювання помилок побудови траєкторії через дію перешкод. Для цього в програму введено процедуру пошуку в кадрі координат точок максимуму кореляційної функції зашумленого і не зашумленого зображень з подальшим обчисленням лінійного зсуву положення цих точок відносно один одного. Відмітимо, що цей зсув обчислюють у пікселях. Проте, якщо є відомою роздільна здатність зображення (відстань між пікселями в лінійних одиницях), можна обчислити і дійсний геометричний зсув шляхом множення цих даних на відомий масштабний коефіцієнт.

За отриманими даними проводили розрахунок дисперсій (середньоквадратичних відхилень), оцінювання точок траєкторії руху (Sko_point), визначення дальності до цих точок від початку координат (Sko_range), а також їхнього кутового положення в заданій системі координат (Sko_angle). В табл. 1.2 наведено результати розрахунку SKO при різних значеннях дисперсії гауссова шуму q , а на рис. 2.8 графічно відображено залежності Sko_point і Sko_range .

Таблиця 1.2

Результати розрахунку SKO при різних значеннях q

q	0.05	0.10	0.15	0.20	0.25	0.30	0.35	0.40	0.45	0.50
SKO point	1.328	1.642	1.786	2.099	2.333	2.337	2.551	2.875	3.065	17.95 1
SKO range	0.856	1.209	1.436	1.571	1.648	1.730	2.087	2.309	2.625	2.701
SKO angle	0.532	0.657	0.739	0.859	0.969	0.996	$+\infty$	$+\infty$	$+\infty$	$+\infty$

Аналіз показав, що при істотному збільшенні рівня перешкод (дисперсії гауссова шуму $q > 0.5$) починають з'являтися аномальні помилки в положенні точок траєкторії, що істотно знижує достовірність отриманих даних і різко збільшує дисперсію оцінювання траєкторії. Відзначимо, що похибка виміру дальності точки траєкторії Sko_range виявляється менше і зростає повільніше.

На рис. 1.9 зображено графік зміни Sko_angle (обчислюють в градусах). Видно, що вже при дисперсії гауссова шуму $q > 0.3$ дисперсія оцінювання кутової координати істотно зростає. Вигляд траєкторії руху при різних рівнях дії гауссових шумів показано на рис. 1.10 і рис. 1.11.

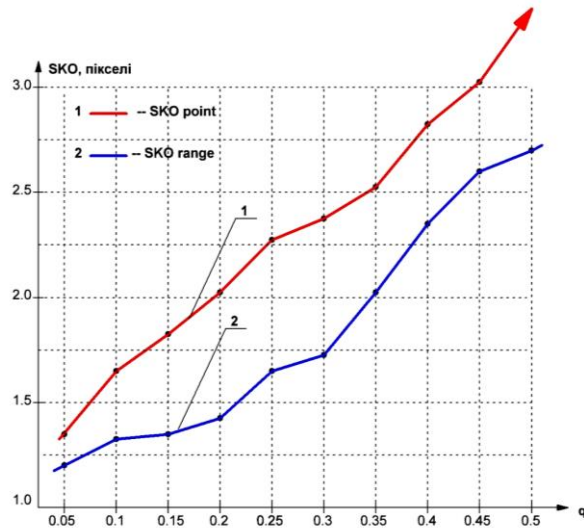


Рисунок 1.8 – Показники відхилення точок траєкторії від дійсних значень і помилок оцінювання дальності залежно від рівня перешкод

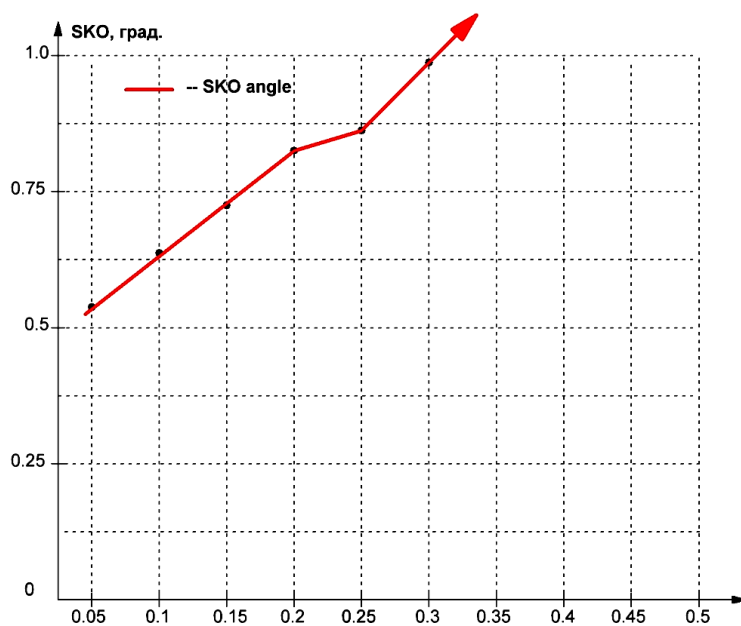


Рисунок 1.9 – Середньоквадратичне відхилення оцінки кутової координати точок траєкторії залежно від рівня перешкод

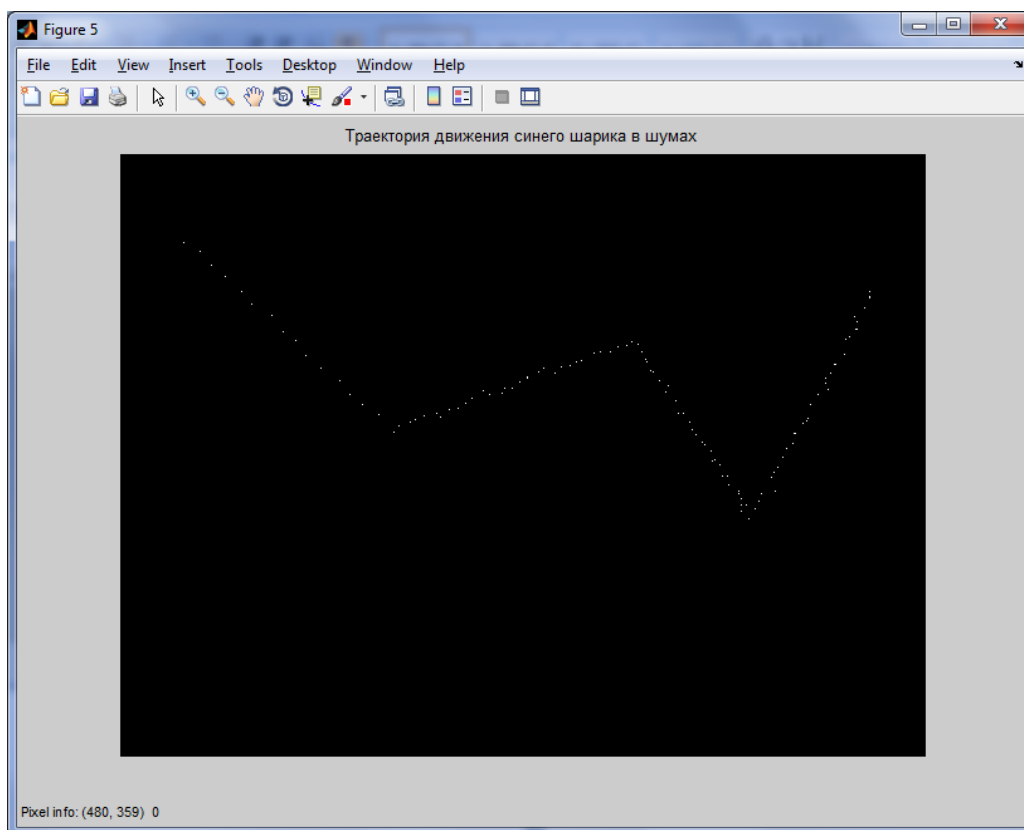


Рисунок 1.10 – Вид траекторії руху при дисперсії гауссова шуму $\sigma=0.1$

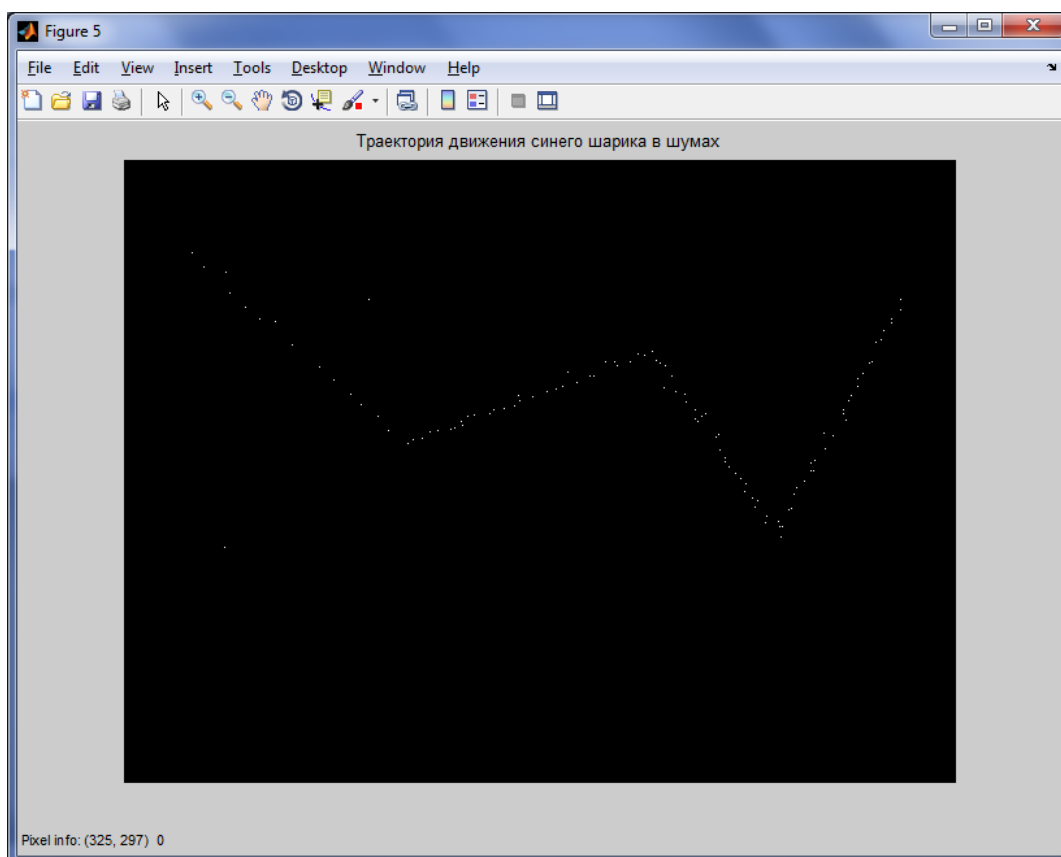


Рисунок 1.11 – Вид траекторії руху при дисперсії гауссова шуму $\sigma=0.5$

1.8 Приклади використання розробленого алгоритму

Безперечною вірогідністю даного алгоритму є виключення з програми процедури прибирання фону з поточного кадру. Це виявляється можливим за умови постійності і рівномірності фону. Проте алгоритм є стійким і при постійній, але досить складній структурі фону.

Переконаємося в цьому на такому прикладі: розглянемо сценарій руху рожевої кульки на фоні із складною структурою (файл violet-tkan.avi). Демонстраційний кадр цього сценарію показано на рис. 2.12.

При використанні нашої програми необхідно внести єдину зміну – виділити не синю, а червону компоненту з вихідного зображення, оскільки вона вносить переважний вклад у рожевий колір спостережуваного об'єкта. Це істотно поліпшить відношення яскравості цього об'єкта до яскравості фону і дозволить вирішити завдання в повному обсязі. Результати розрахунку незашумленої і зашумленої траєкторій показано на рис. 2.13 і 2.14.

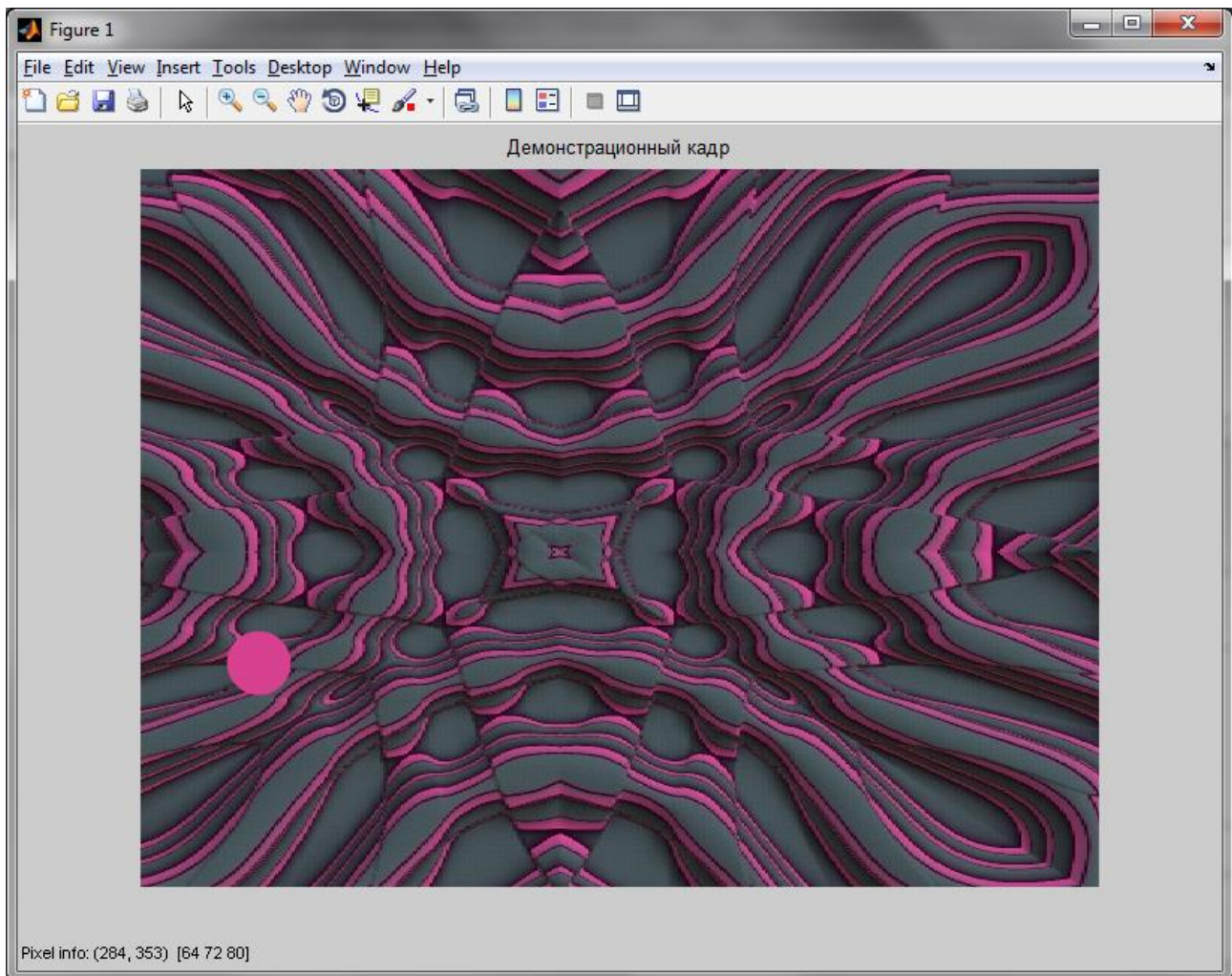


Рисунок 1.12 – Поточний кадр вихідної відеопослідовності із складною структурою фону

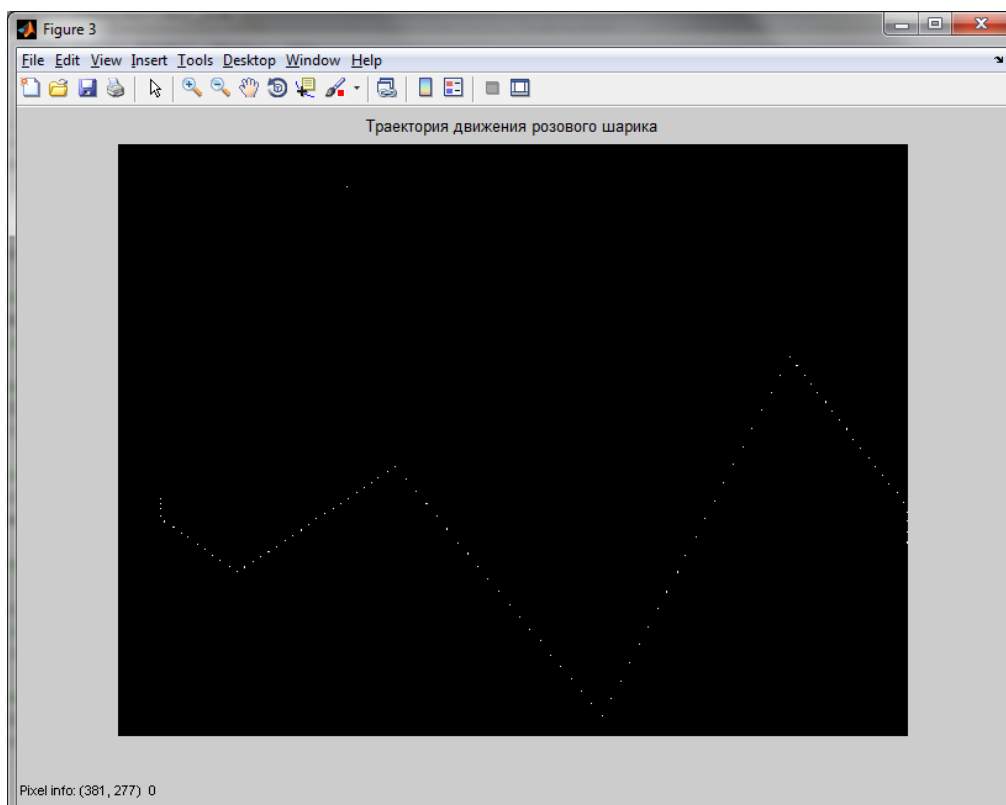


Рисунок 1.13 – Траектория руху об'єкта на фоні із складною постійною структурою за відсутності шумів

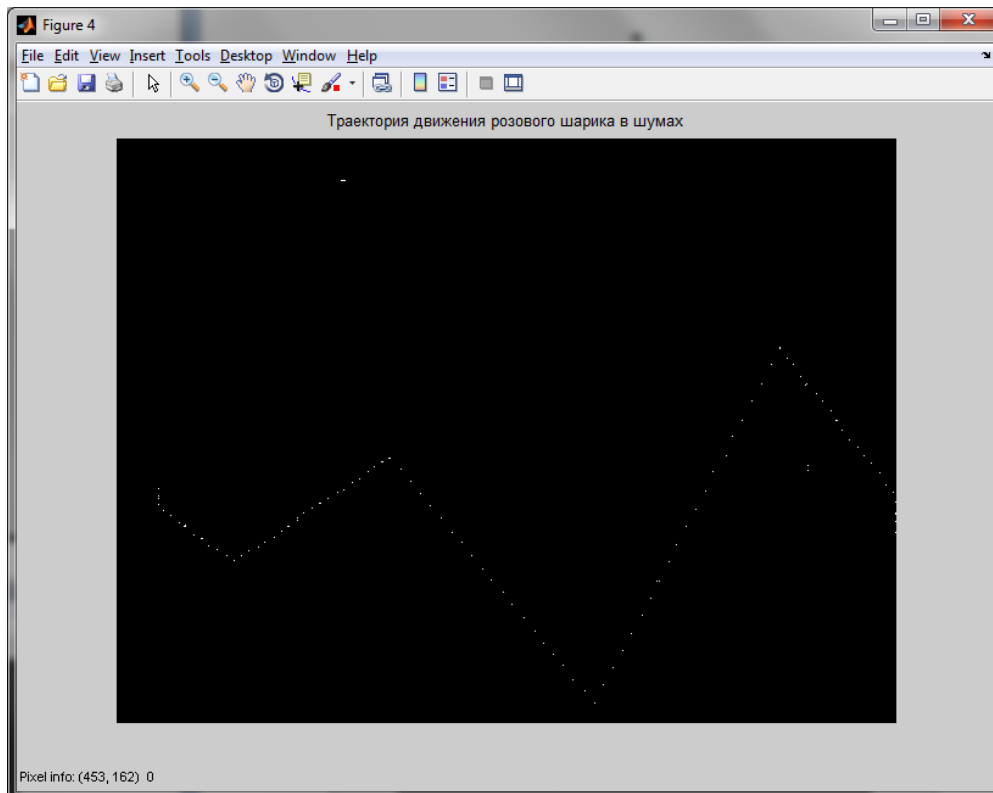


Рисунок 1.14 – Траектория руху об'єкта на фоні із складною постійною структурою за наявності шумів

Наведемо ще один приклад. Розглянемо сценарій руху червоної кульки на постійному сірому фоні (файл red_ball.avi). Демонстраційний кадр цього сценарію показано на рис. 1.15.

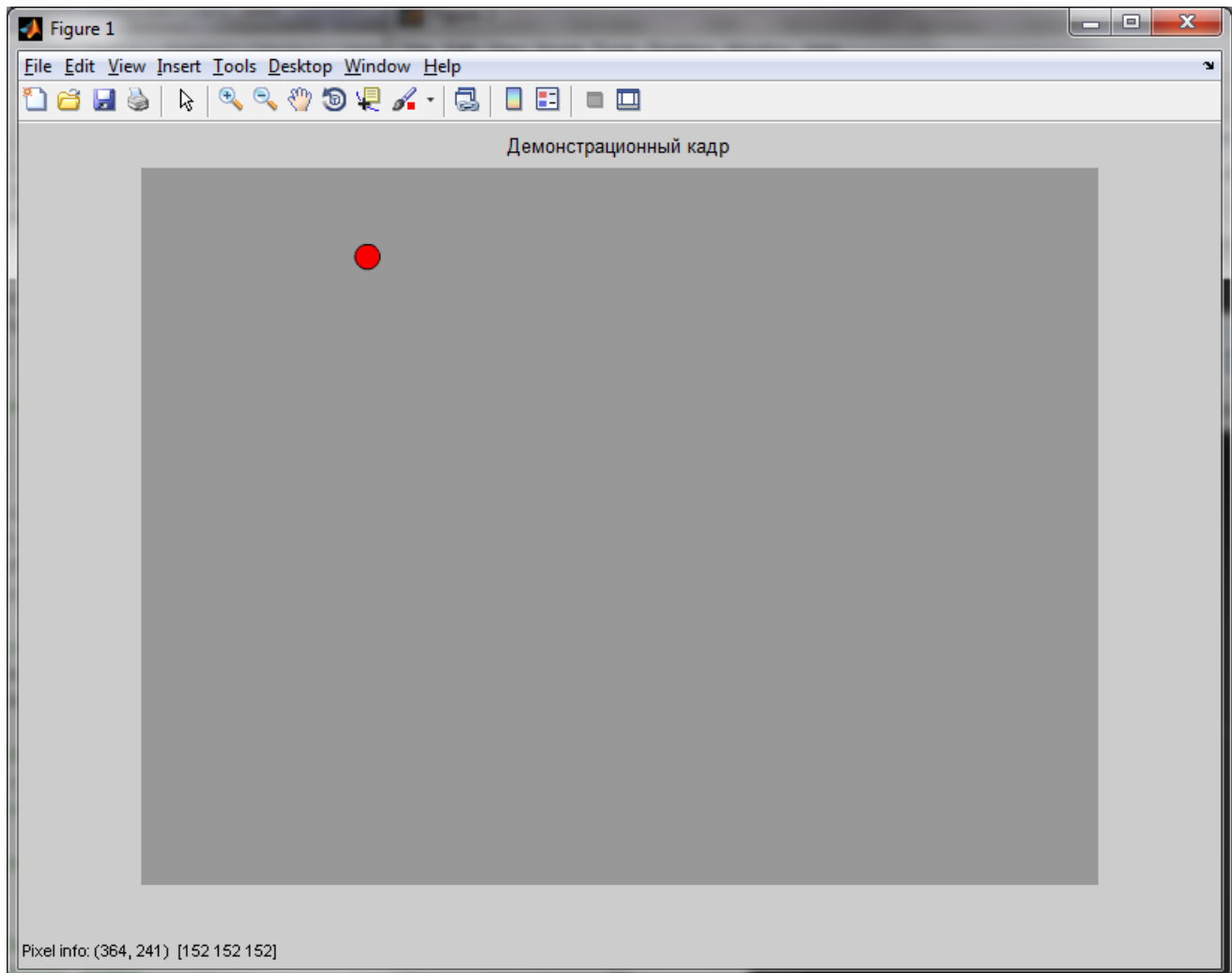


Рисунок 1.15 – Поточний кадр вихідної відеопослідовності (рух червоної кульки на сірому постійному фоні)

Для ефективного функціонування алгоритму побудови траєкторії необхідно, по-перше, виділити червону компоненту у вихідній відеопослідовності rgb-формату, що забезпечить максимальну яскравість рухомого об'єкта відносно фону, а по-друге, підібрати необхідний розмір шаблону в пікселях для обчислення взаємно кореляційної функції.

На рис. 1.16 показано результати побудови траєкторій руху червоної кульки на сірому постійному фоні. На рис. 1.17 добре помітним є вплив шумів на точність побудови траєкторії. У наступній лабораторній роботі будуть розглянуто складніші сценарії побудови траєкторій руху відразу декількох об'єктів, а також кореляційні алгоритми оцінювання руху поєднано з методом прибирання фону.

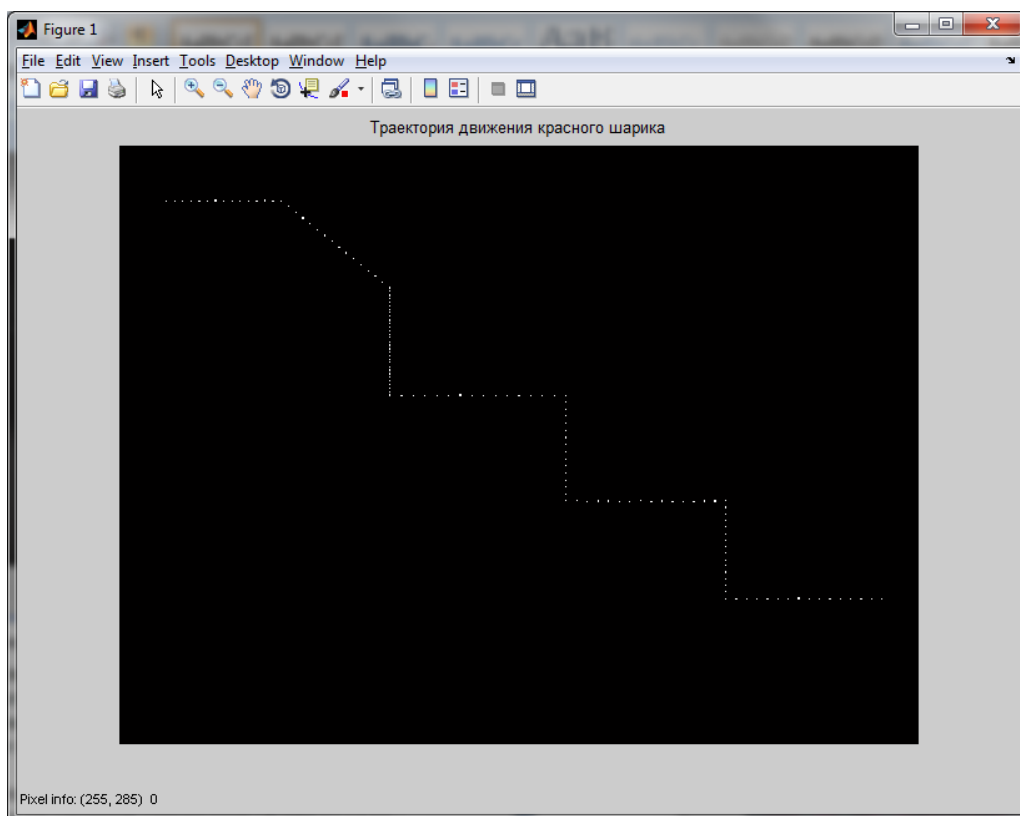


Рисунок 1.16 – Траектория руху червоної кульки на сірому фоні за відсутності шумів

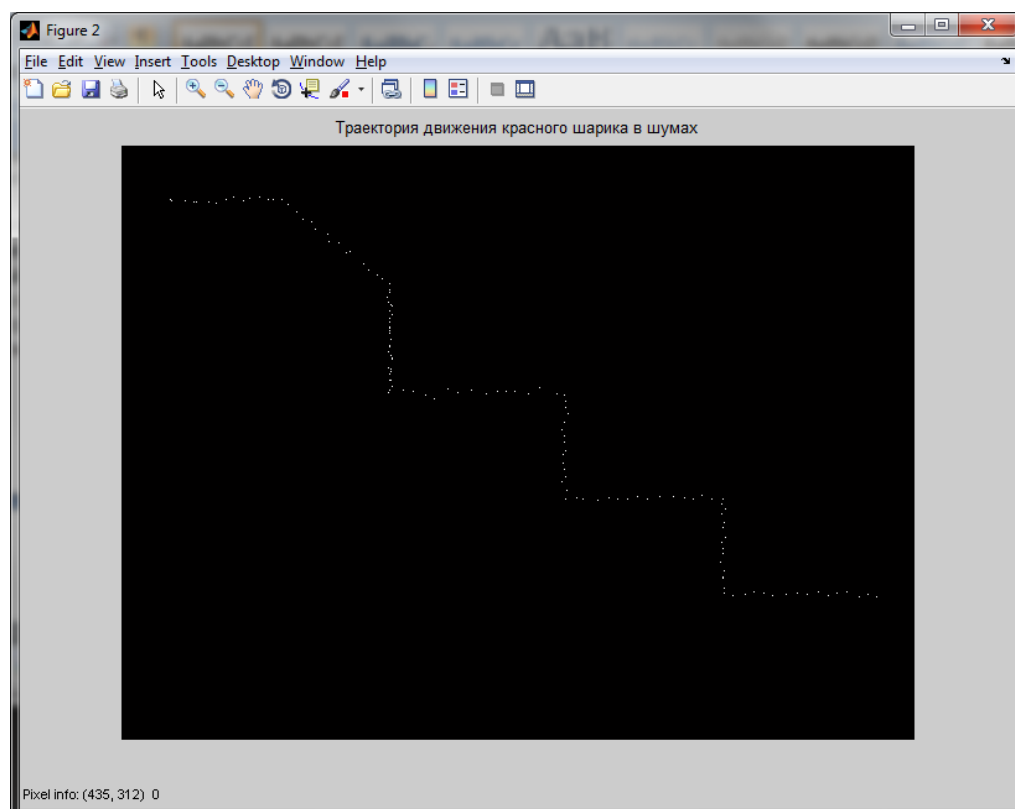


Рисунок 1.17 – Траектория руху червоної кульки на сірому фоні за наявності шумів

2 ОЦІНЮВАННЯ ПАРАМЕТРІВ ТРАЕКТОРІЇ РУХУ ОБ'ЄКТІВ ЗА РЕЗУЛЬТАТАМИ ВІДЕОСПОСТЕРЕЖЕННЯ

Для вирішення завдання оцінювання траєкторних параметрів було проведено паралельне перенесення початку прямокутної системи координат кадру з точки з координатами $(x=1, y=1)$ у точку з координатами $(x=320, y=240)$, оскільки розмір кадру у форматі аві дорівнює 640×480 пікселів.

Таке положення початку координат відповідає центру кадру. Це є зручним при визначенні траєкторних параметрів: при оцінюванні кутового положення спостережуваного об'єкта з'являється можливість використовувати праву прямокутну систему координат з виміром кута в діапазоні $0 \dots 360^\circ$.

У цьому прикладі для визначення траєкторії об'єкта (рис. 2.1) обчислюють дальність до кожної точки відносно початку системи координат (у пікселях), оцінюють модуль вектора швидкості руху об'єкта (рис. 2.2), а також кутове положення об'єкта в заданій системі координат для кожної точки траєкторії і кут вектора швидкості руху в градусах (рис. 2.3).

2.1 Опис програми оцінювання параметрів траєкторії руху об'єкта (дальності, кутової координати і вектора швидкості) (Додаток А)

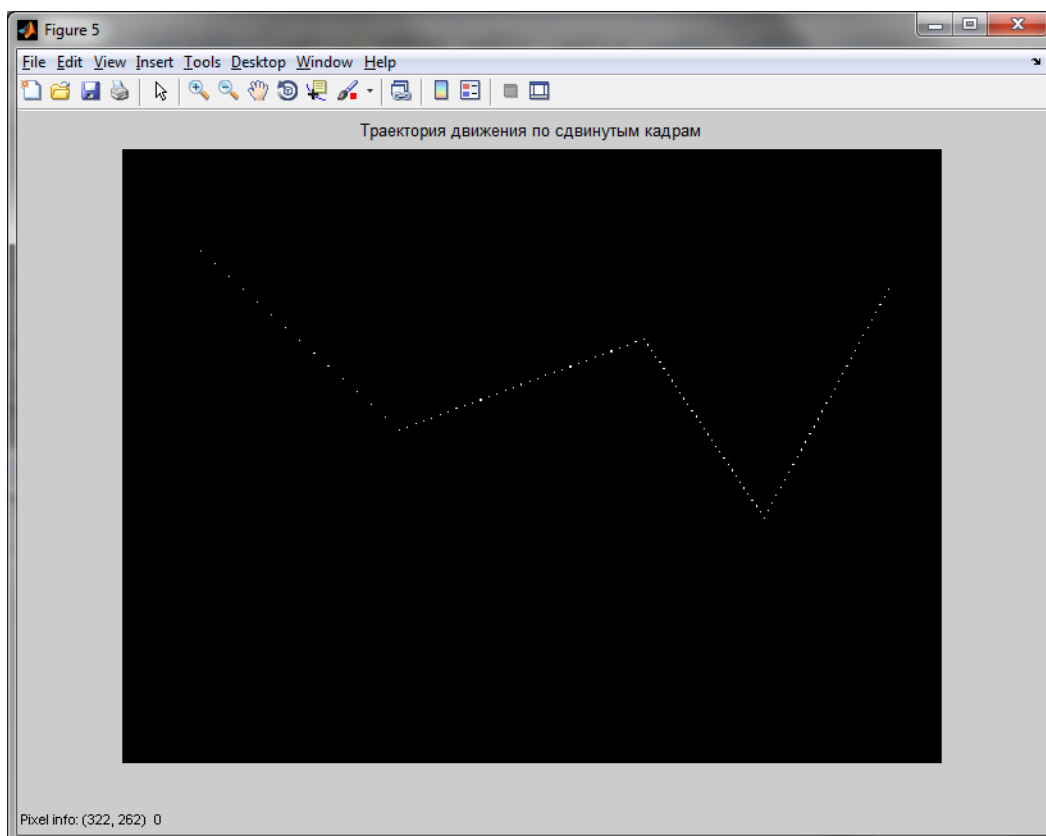


Рисунок 2.1 – Траєкторія руху об'єкта

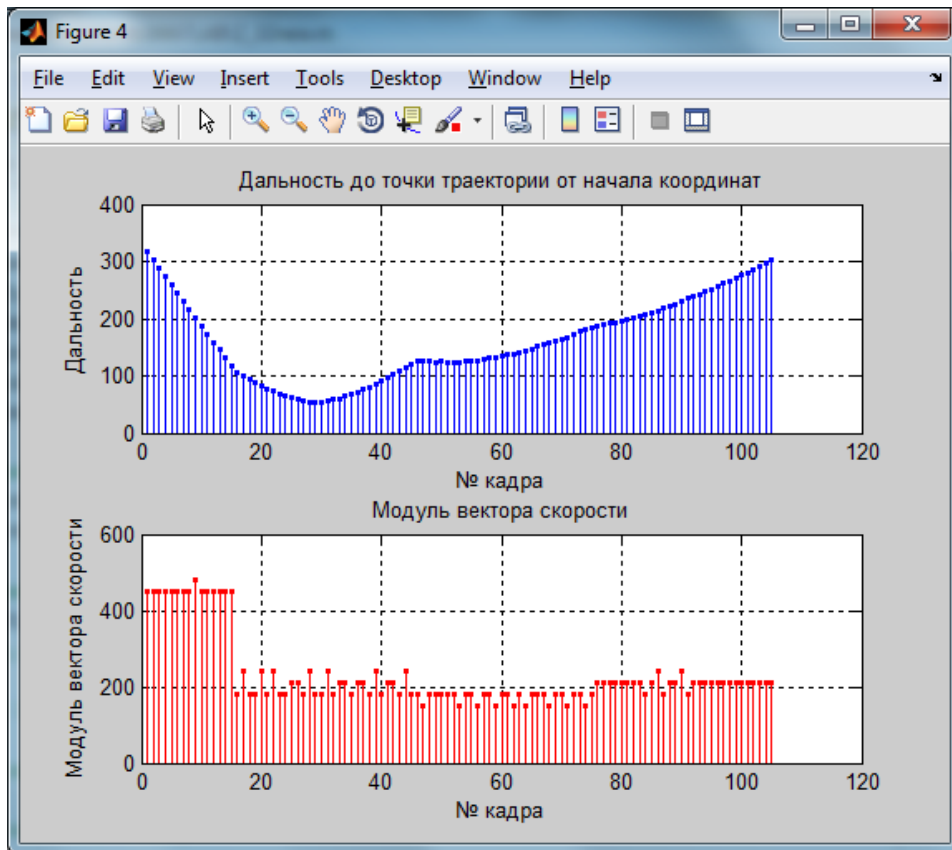


Рисунок 2.2 – Оцінювання дальності і модуля вектора швидкості об'єкта

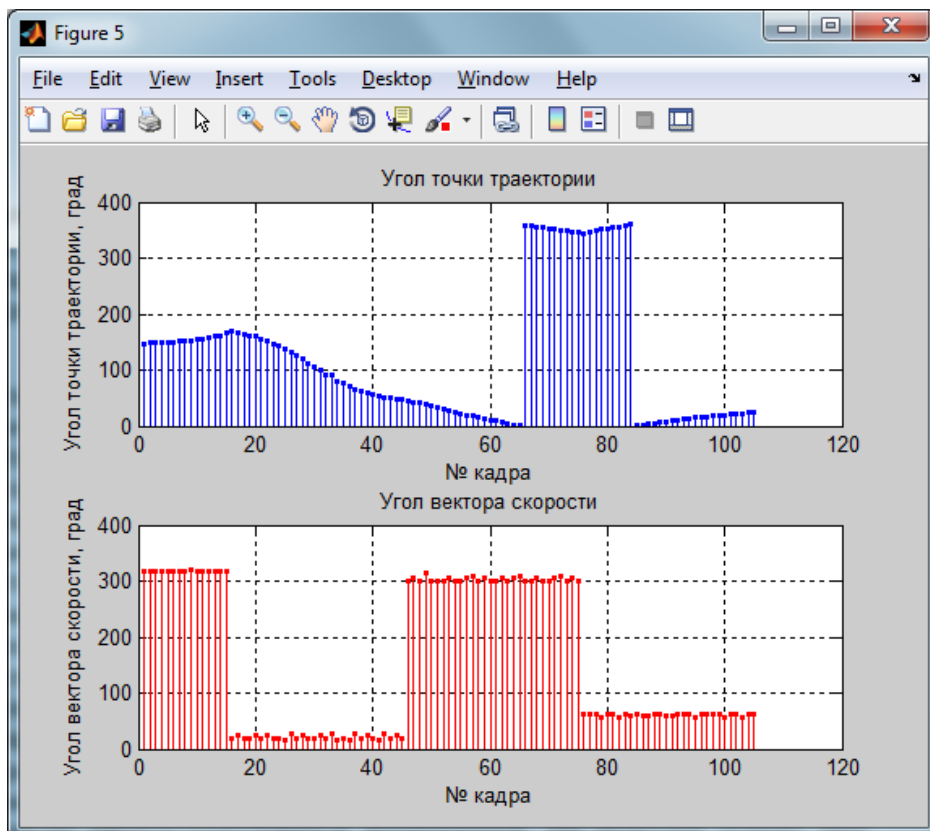


Рисунок 2.3 – Змінення кутового положення об'єкта при русі і оцінювання кутів вектора швидкості руху

3 ПЕРЕТВОРЕННЯ СИСТЕМИ КООРДИНАТ

У цьому розділі зробимо перерахунок координат із координат камери в глобальну систему координат, зважаючи що початок координат камери знаходиться в центрі кадру.

3.1 Короткі теоретичні відомості

Загальні теоретичні відомості про перетворення систем координат наведено в описі лабораторної роботи №1. Проте для практичної реалізації доцільно розглянути ці дані детальніше.

Основні види перетворень:

- паралельне перенесення;
- поворот довкола початку координат;
- центральна симетрія;
- гомотетія;
- афінне перетворення.

Розглянемо ряд перетворень, пов'язаних з переходом з однієї системи координат в іншу. Тут (x, y) і (x', y') – координати довільної точки P відповідно в старій і новій системах координат.

Паралельне перенесення. Пересунемо систему координат XOY в площині так, щоб осі OX і OY залишалися паралельні самим собі, а початок координат O змістився в точку $O'(a, b)$. Отримаємо нову систему координат $X'O'Y'$ (рис. 3.1).

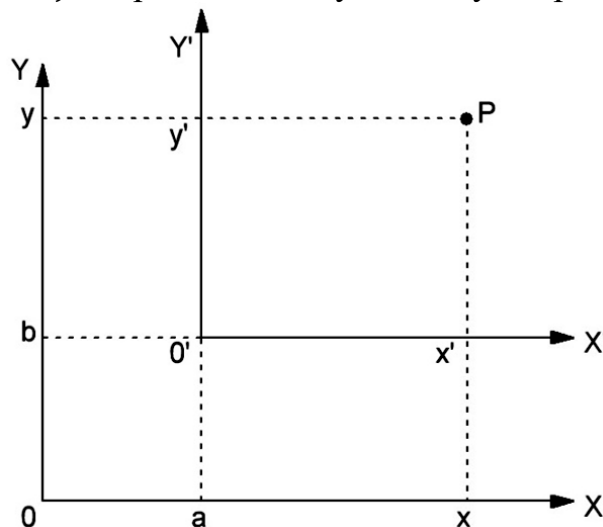


Рисунок 3.1 – Нова система координат $X'O'Y'$ після паралельного перенесення

Координати точки P в новій і старій системі координат зв'язані співвідношеннями:

$$\begin{cases} x' = x - a \\ y' = y - b \end{cases}$$

Поворот докола початку координат. Обернемо систему координат XOY в площині на кут φ (рис. 3.2).

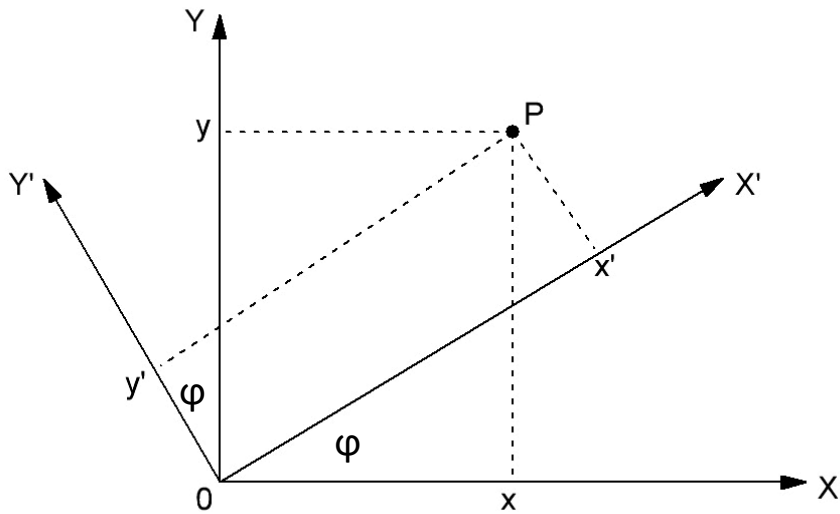


Рисунок 3.2 – Поворот системи координат на кут φ

Тепер координати точки P в новій і старій системах координат пов'язані співвідношеннями

$$\begin{cases} x' = x \cdot \cos\varphi + y \cdot \sin\varphi, \\ y' = -x \cdot \sin\varphi + y \cdot \cos\varphi. \end{cases}$$

В окремому випадку $\varphi = \pi$ отримаємо *центральну симетрію* відносно початку координат O :

$$\begin{cases} x' = -x, \\ y' = -y. \end{cases}$$

Гомотетія з центром $O(a, b)$ і коефіцієнтом $k \neq 0$:

$$\begin{cases} x' - a = k(x - a), \\ y' - b = k(y - b). \end{cases}$$

Афінне перетворення

$$\begin{cases} x' = ax + by + c, \\ y' = dx + ey + f, \end{cases}$$

де

$$\begin{vmatrix} a & b \\ d & e \end{vmatrix} \neq 0.$$

Афінне перетворення переводить прямі в прямі, пересічні прямі – в пересічні прямі, паралельні прямі – в паралельні прямі. Всі наведені вище перетворення координат є афінними.

Перетворення декартових прямокутних координат на площині. Паралельний зсув координатних осей (рис. 3.3):

$$\begin{cases} x = x' + a \\ y = y' + b \end{cases} \quad \begin{cases} x' = x - a \\ y' = y - b \end{cases}$$

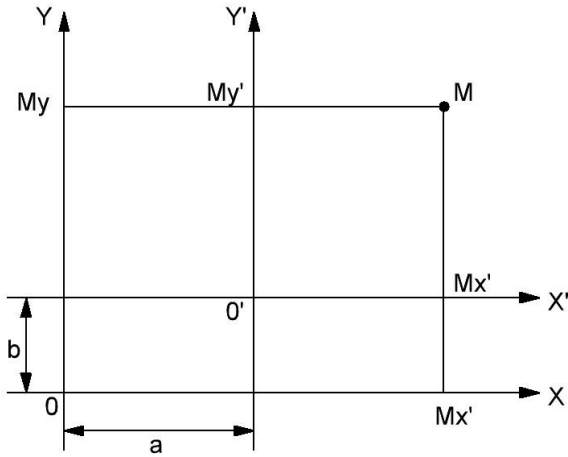


Рисунок 3.3 – Паралельний зсув координатних осей

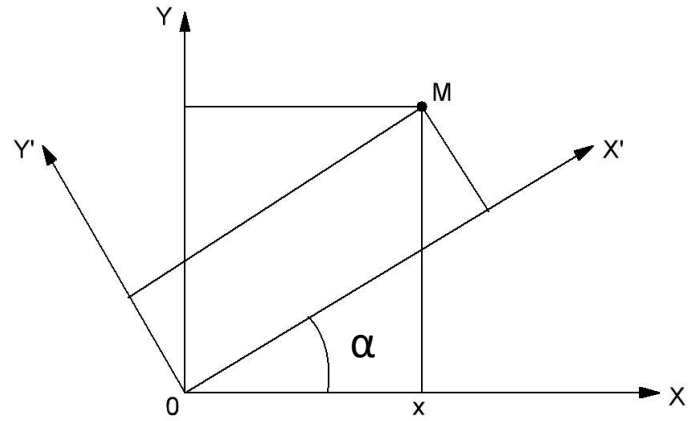


Рисунок 3.4 – Поворот координатних осей

$$\begin{cases} x = x' \cos \alpha - y' \sin \alpha \\ y = y' \sin \alpha + x' \cos \alpha \end{cases}, \quad \begin{cases} x' = x \cos \alpha + y \sin \alpha \\ y' = -x \sin \alpha + y \cos \alpha \end{cases}$$

Паралельний зсув і поворот координатних осей (рис. 3.5):

$$\begin{cases} x = x' \cos \alpha - y' \sin \alpha + a \\ y = y' \sin \alpha + x' \cos \alpha + b \end{cases}, \quad \begin{cases} x' = (x - a) \cos \alpha + (y - b) \sin \alpha \\ y' = -(x - a) \sin \alpha + (y - b) \cos \alpha \end{cases}$$

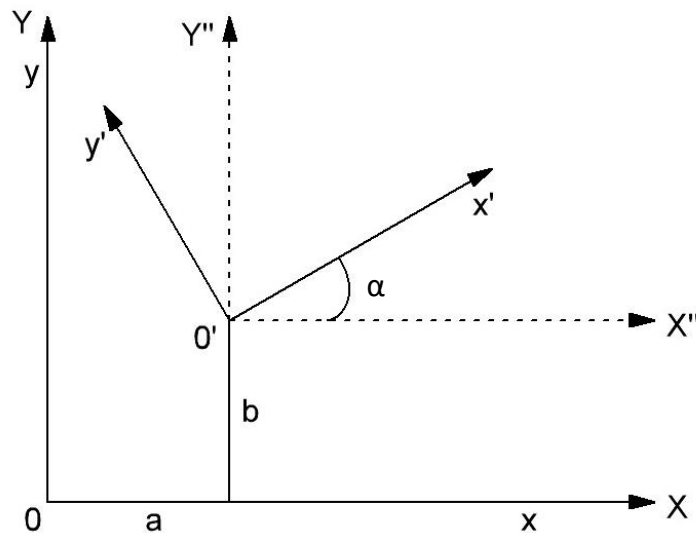


Рисунок 3.5 – Паралельний зсув і поворот координатних осей

3.2 Обертання фігури в тривимірному просторі

Матрицею повороту (або матрицею напрямних косинусів) називають ортогональну матрицю, яку використовують для виконання власного ортогонального перетворення в евклідовому просторі. При множенні будь-якого вектора на матрицю повороту довжина вектора зберігається. Визначник матриці повороту дорівнює одиниці.

Зазвичай вважають, що, на відміну від матриці переходу при повороті системи координат (базису), при множенні на матрицю повороту вектор-стовпця координати вектора перетворюються відповідно до повороту самого вектора (а не поворотом координатних осей), тобто при цьому координати поверненого вектора визначаються в тій же, нерухомій системі координат. Проте відмінність тієї й іншої матриці є лише в знаку кута повороту, і одна може бути отримана заміною кута повороту на протилежний; а інша є взаємно зворотною, і вони можуть бути отримані одна з одної транспонуванням.

3.3 Матриця повороту в тривимірному просторі

Будь-яке обертання в тривимірному просторі може бути подано як композицію поворотів довкола трьох ортогональних осей (наприклад, довкола осей декартових координат). Цій композиції відповідає матриця, що дорівнює множинню відповідних трьох матриць повороту.

На рис. 3.6 показано положення куба в прямокутній системі просторових координат. Його грані розташовані паралельно осям координат X, Y, Z , а центр є рівновіддаленим від зовнішніх граней.

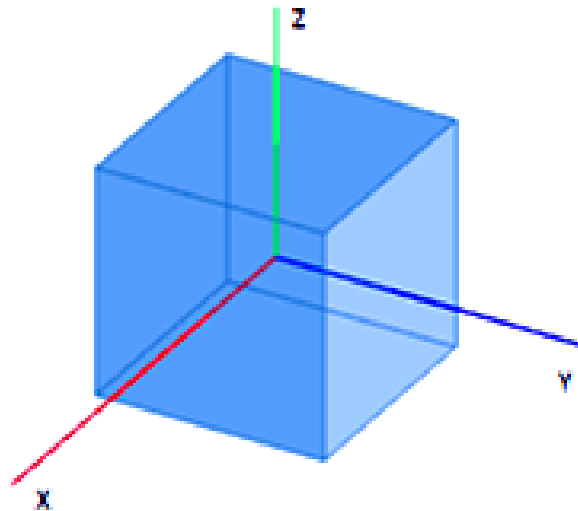


Рисунок 3.6 – Положення куба в прямокутній системі просторових координат

Матрицями обертання довкола осі декартової системи координат у тривимірному просторі є:

Оберт на кут α довкола осі x :

$$M_x(\alpha) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha & -\sin\alpha \\ 0 & \sin\alpha & \cos\alpha \end{pmatrix}.$$

Оберт на кут β довкола осі y :

$$M_y(\beta) = \begin{pmatrix} \cos\beta & 0 & \sin\beta \\ 0 & 1 & 0 \\ -\sin\beta & 0 & \cos\beta \end{pmatrix}.$$

Оберт на кут γ довкола осі z :

$$M_z(\gamma) = \begin{pmatrix} \cos\gamma & -\sin\gamma & 0 \\ \sin\gamma & \cos\gamma & 0 \\ 0 & 0 & 1 \end{pmatrix}.$$

Після перетворень отримаємо формули (рис. 3.7):

по осі X

$$x' = x; \quad y' = y \cdot \cos(\alpha) + z \cdot \sin(\alpha); \quad z' = -y \cdot \sin(\alpha) + z \cdot \cos(\alpha);$$

по осі Y

$$x' = x \cdot \cos(\beta) + z \cdot \sin(\beta); \quad y' = y; \quad z' = -x \cdot \sin(\beta) + z \cdot \cos\beta;$$

по осі Z

$$x' = x \cdot \cos(\gamma) - y \cdot \sin(\gamma); \quad y' = x \cdot \sin\gamma + y \cdot \cos(\gamma); \quad z' = z.$$

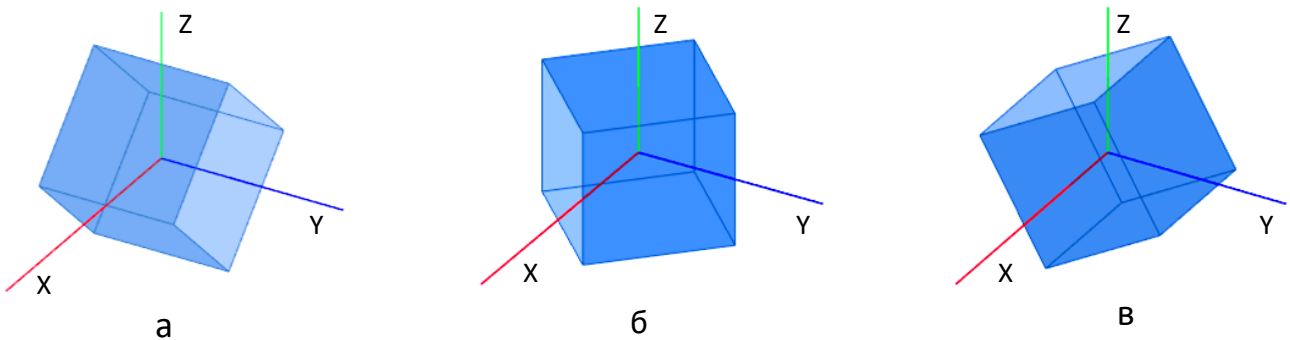


Рисунок 3.7 – Повороти по осях X, Y, Z

Всі три оберти виконуються незалежно один від одного, тобто якщо треба здійснити оберт довкола осей Ox і Oy , Oz , спочатку робиться оберт довкола осі Ox , потім стосовно отриманої точки – оберт довкола осі Oy , а потім – довкола осі Oz .

Позитивним кутам при цьому відповідає обертання вектора проти годинникової стрілки в правій системі координат і за годинниковою стрілкою в лівій системі координат, якщо дивитися проти напрямку відповідної осі. Права система координат пов'язана з вибором правого базису (див. правило гвинта).

3.4 Тривимірне обертання довкола осей координат

При обертанні довкола осі x розміри вздовж осі x не змінюються, тоді матриця перетворення, що є відповідною повороту на кут α , має вигляд (обертання позитивне):

$$T_x = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\alpha & \sin\alpha & 0 \\ 0 & -\sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

При оберті на кут β довкола осі y матриця перетворення визначається виразом:

$$T_y = \begin{bmatrix} \cos\beta & 0 & -\sin\beta & 0 \\ 0 & 1 & 0 & 0 \\ \sin\beta & 0 & \cos\beta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

Аналогічно матриця перетворення при оберті на кут γ довкола осі z має вигляд

$$T_z = \begin{bmatrix} \cos\gamma & \sin\gamma & 0 & 0 \\ -\sin\gamma & \cos\gamma & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

Слід зазначити, що тривимірні обертання є некомутативними, тобто порядок множення матриць впливає на кінцевий результат, а це означає, що обертання довкола осі x , за яким відбувається обертання довкола осі y , і обертання довкола осі y і наступне за ним обертання довкола осі x приводять до різних результатів.

3.5 Формули напрямних косинусів

Розглянемо напрямні косинуси c_{ij} , значення яких залежать від величин і знаків кутових елементів зовнішнього орієнтування α , β , γ знімків. Відомо, що кути α , β , γ мають знак плюс, якщо у момент зйомки система координат знімка повернулася проти ходу годинникової стрілки довкола трьох координатних осей від положення, при якому вона була паралельна системі координат об'єкта. З врахуванням цього на рис. 3.8 показано три послідовних повороти системи координат знімка від положення горизонтальної системи координат $Sx^0y^0z^0$ через положення $Sx_\alpha y_\alpha z_\alpha$ (поворот на кут α) і $Sx_{\alpha\beta} y_{\alpha\beta} z_{\alpha\beta}$ (поворот на кут β) до положення нахиленої системи координат $Sxyz$ знімка (після повороту на кут γ).

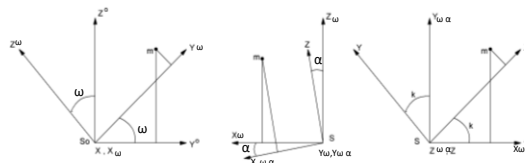


Рисунок 3.8 – Послідовні повороти системи координат

У табл. 3.1 з врахуванням рис. 3.8 наведено величини кутів між осями координат при кожному повороті.

Таблиця 3.1

Величини кутів при поворотах

1-й поворот на кут ω			2-й поворот на кут α				3-й поворот на кут k				
Вісь	x^o	y^o	z^o	Вісь	x_ω	y_ω	z_ω	Вісь	$x_{\omega\alpha}$	$y_{\omega\alpha}$	$z_{\omega\alpha}$
x_ω	0	90°	90°	$x_{\omega\alpha}$	α	90°	$90^\circ + \alpha$	x	k	$90^\circ - k$	90°
y_ω	90°	ω	$90^\circ - \omega$	$y_{\omega\alpha}$	90°	0	90°	y	$90^\circ + k$	k	90°
z_ω	90°	$90^\circ + \omega$	ω	$z_{\omega\alpha}$	$90^\circ - \alpha$	90°	α	z	90°	90°	0

Використавши таблицю, складемо матриці трьох поворотів:

$$A_\alpha = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha & \sin\alpha \\ 0 & -\sin\alpha & \cos\alpha \end{bmatrix}; A_\beta = \begin{bmatrix} \cos\beta & 0 & -\sin\beta \\ 0 & 1 & 0 \\ \sin\beta & 0 & \cos\beta \end{bmatrix}; A_\gamma = \begin{bmatrix} \cos\gamma & \sin\gamma & 0 \\ -\sin\gamma & \cos\gamma & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

При обробленні знімків потрібно виконати повороти нахиленої системи координат $Sxyz$ знімка на кути α, β, γ до положення горизонтальної системи координат $Sx^o y^o z^o$ за формулами

$$\begin{bmatrix} x^o \\ y^o \\ z^o \end{bmatrix} = A_\alpha^T A_\beta^T A_\gamma^T \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = A_{\alpha\beta\gamma} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} c_{11} & c_{12} & c_{13} \\ c_{21} & c_{22} & c_{23} \\ c_{31} & c_{32} & c_{33} \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix},$$

де $A_\alpha^T, A_\beta^T, A_\gamma^T$ – матриці, що транспоновані до матриць поворотів.

Отже, матриця $A_{\alpha\beta\gamma}$ дорівнюватиме такому множинню матриць:

$$\begin{bmatrix} c_{11} & c_{12} & c_{13} \\ c_{21} & c_{22} & c_{23} \\ c_{31} & c_{32} & c_{33} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha & \sin\alpha \\ 0 & -\sin\alpha & \cos\alpha \end{bmatrix} \begin{bmatrix} \cos\beta & 0 & -\sin\beta \\ 0 & 1 & 0 \\ \sin\beta & 0 & \cos\beta \end{bmatrix} \begin{bmatrix} \cos\gamma & \sin\gamma & 0 \\ -\sin\gamma & \cos\gamma & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

Перемноживши матриці, отримаємо формули обчислення величин напрямних косинусів:

$$c_{11} = \cos\beta \cos\gamma; c_{12} = -\cos\beta \sin\gamma; c_{13} = \sin\beta;$$

$$c_{21} = \cos\alpha \sin\gamma + \sin\alpha \sin\beta \cos\gamma; c_{22} = \cos\alpha \cos\gamma - \sin\alpha \sin\beta \sin\gamma; c_{23} = -\sin\alpha \cos\beta;$$

$c_{31} = \sin\alpha\sin\gamma - \cos\alpha\sin\beta\cos\gamma$; $c_{32} = \sin\alpha\cos\gamma + \cos\alpha\sin\beta\sin\gamma$; $c_{33} = \cos\alpha\cos\beta$.

Напрямні косинуси дозволяють обчислити кутові елементи зовнішнього орієнтування знімка за формулами:

$$\alpha = -\arctg \frac{c_{23}}{c_{33}}, \beta = \arcsin c_{31}, \gamma = \arctg \frac{c_{12}}{c_{11}}.$$

Напрямні косинуси пов'язані незалежними рівняннями, які використовують для контролю визначення їх величин:

$$\begin{aligned} c_{11}^2 + c_{12}^2 + c_{13}^2 &= 1; & c_{11}c_{21} + c_{12}c_{22} + c_{13}c_{23} &= 0; \\ c_{21}^2 + c_{22}^2 + c_{23}^2 &= 1; & c_{11}c_{31} + c_{12}c_{32} + c_{13}c_{33} &= 0; \\ c_{31}^2 + c_{32}^2 + c_{33}^2 &= 1; & c_{21}c_{31} + c_{22}c_{32} + c_{23}c_{33} &= 0. \end{aligned}$$

3.6 Система координат цифрового знімка з початком в його площині

Для визначення положення точок на цифровому знімку використовують систему піксельних координат $c'x_j y_i z'$ і систему фізичних координат $c'x' y' z'$ (рис. 3.9, а, б). У цих системах координат початок і осі збігаються, але система піксельних координат є лівою, а система фізичних координат – правою. Оскільки вимірюють плоске зображення, у всіх точок знімка є координата $z' = 0$.

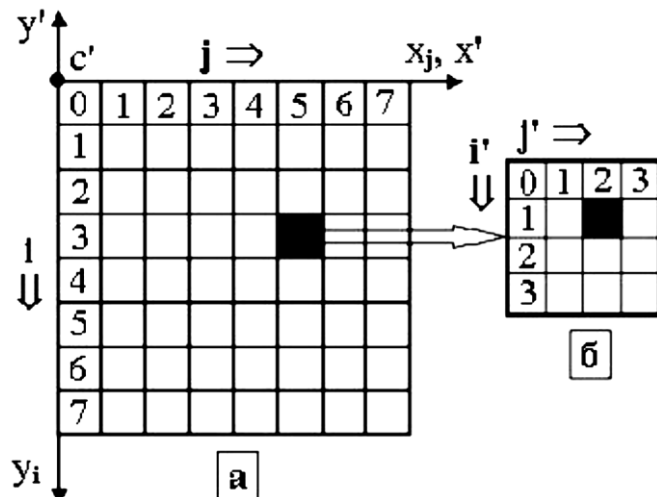


Рисунок 3.9 – Система координат: а – піксельних; б – фізичних

Початок c' обох систем координат розташовано в лівому верхньому куті. Положення пікселів задають номером стовпця j і номером рядка i . Збільшення номерів стовпців спрямовано вправо, а рядків – униз. Перший піксель є нульовим, тобто $j = 0$ і $i = 0$. Для зафарбованого пікселя $j = 5$ і $i = 3$.

Координати центрів пікселів розраховують за формулами $x_j = j + 0,5$ і $y_i = i + 0,5$. Отже, для нульового пікселя піксельні координати $x_j = 0,5$ і $y_i = 0,5$, а для зафарбованого пікселя – $x_j = 5,5$ і $y_i = 3,5$.

Для переходу від піксельних координат до фізичних, які беруть участь у подальших перетвореннях, використовують формули:

$$x' = \Delta \times x_j ; y' = -(\Delta \times y_i),$$

де Δ – фізичний розмір сторони пікселя, заданий при скануванні знімка або при виготовленні матриці цифрової фотокамери. Координата y' має знак мінус при переході до правої системи координат. Наприклад, при $\Delta = 20$ мкм фізичні координати зафарбованого пікселя будуть $x' = 5,5 \times 20$ мкм = 110 мкм і $y' = -(3,5 \times 20)$ мкм = -70 мкм.

На комп'ютері кожен піксель вихідної матриці можна замінити підматрицею $n \times n$ з пікселів меншого розміру (рис. 3.9,б). Це дозволяє збільшити зображення на екрані монітора в n разів і виконати виміри з точністю до $1/n$ вихідного розміру пікселя. На рис. 5.9 показано підматрицю розміром 4×4 зафарбованого пікселя вихідної матриці, тобто зображення збільшено в чотири рази. Підматриця має свою систему рахунку координат: j' – номер стовпця і i' – номер рядка. Координати центрів пікселів розраховують за формулами:

$$x_j = j + \frac{j' + 0,5}{n} ; y_i = i + \frac{i' + 0,5}{n}.$$

На вихідній матриці зафарбований піксель має нумерацію $j = 5, i = 3$, а на підматриці – $j' = 2, i' = 1$. Отже, піксельні координати центру зафарбованого пікселя підматриці будуть такими:

$$x_j = 5 + \frac{2 + 0,5}{4} = 5,625 ; y_i = 3 + \frac{1 + 0,5}{4} = 3,375 ,$$

а фізичні координати при $\Delta = 20$ мкм будуть $x' = 112,5$ мкм, $y' = -67,5$ мкм.

Таким чином, в результаті вимірювання цифрового знімка кожна точка в системі координат $c'x'y'z'$ матиме координати $x', y', z' = 0$. При калібруванні цифрової камери, щоб вилучити негативні координати, початок c' системи фізичних координат переносять в лівий нижній піксель.

3.7 Система координат знімка з початком у центрі його проєкції

Елементи внутрішнього орієнтування знімка. Вимірявши координати точок знімка в цифровій системі координат $c'x'y'z'$ або в заданій координатними

мітками системи координат $o'xyz$ (рис. 3.11), їх перетворять в другу систему координат $Sxyz$ знімка, початок якої знаходиться в центрі його проекції S . Вісь z цієї системи координат поєднана з головним оптичним променем S_0 . Осі x і y системи координат $Sxyz$ є паралельними відповідним осям системи координат $o'xyz$ або $c'x'y'z'$ і мають такі ж позитивні напрями. Щоб не завантажувати рисунок, на ньому показано лише систему координат $o'xyz$, оскільки перехід до системи координат $Sxyz$ для обох систем є однаковим і полягає лише в перенесенні початку відліку координат з точки o' або c' в точку S . Для цього потрібно знати координати x_s, y_s, z_s центра проекції S в системі координат $o'xyz$ або $c'x'y'z'$. Ними є координати x_0, y_0 головної точки знімка і фокусна відстань f фотокамери, тобто $x_s = x_0, y_s = y_0, z_s = f$.

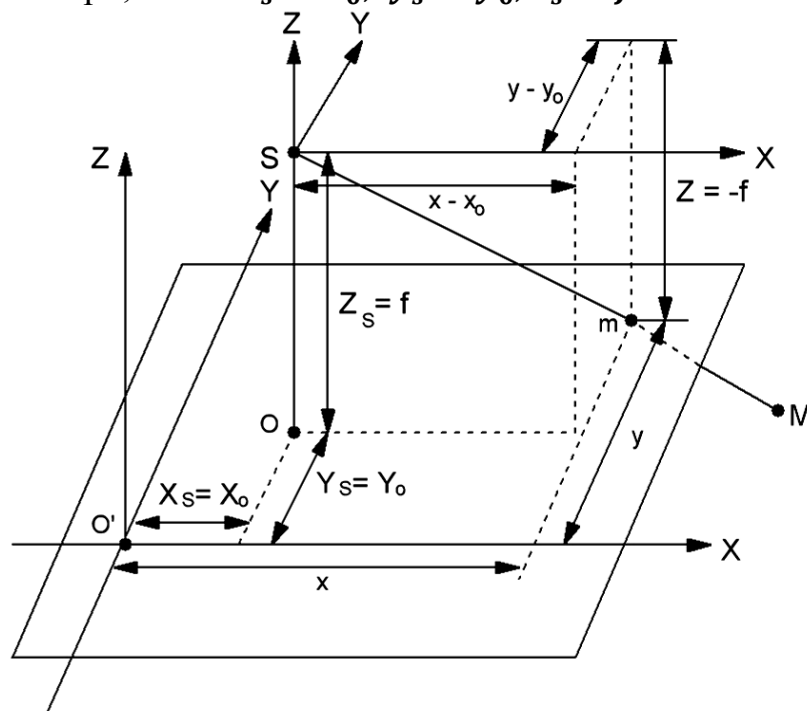


Рисунок 3.10 – Система координат з початком у центрі проекції знімка

Віднімаючи координати центру проекції з координат точок знімка, вимірених в системі координат $o'xyz$ ($c'x'y'z'$), отримують координати точок у системі координат $Sxyz$: $x - x_s = x - x_0, y - y_s = y - y_0, z - z_s = -f$, тобто точка m знімка має координати: $x - x_0, y - y_0, z = -f$.

Величини x_0, y_0, f називають елементами внутрішнього орієнтування знімка. Визначивши з їх допомогою положення центра проекції відносно знімка, можна провести проектуючі промені через центр проекції і точки знімка, наприклад, промінь SmM . У результаті буде відновлено в'язку проектуючих променів, що існувала у момент експонування зображення.

При виготовленні вимірювальних фотокамер головну точку o і точку o' прагнули поєднати. Проте досягти такого положення практично не вдається, і зазвичай величини координат x_0 і y_0 не перевищують 20 мкм. Таку умову потрібно було дотримувати при обробленні знімків на оптико-механічних

приладах. При обробленні на комп'ютері це не має значення, оскільки початок системи фізичних координат $c'x'y'z'$ знаходиться в лівому верхньому або нижньому пікселі. Отже, координати x_0 і y_0 головної точки при калібруванні цифрової фотокамери будуть дорівнювати половині відповідної сторони кадру.

3.8 Система координат об'єкта

Елементи зовнішнього орієнтування знімка і пари знімків. В результаті оброблення знімків координати точок об'єкта мають бути отримані в його системі координат $OXYZ$ (рис. 3.11), положення якої задається конструкцією об'єкта. Якщо положення цієї системи координат є невідомим, то координати точок об'єкта визначають в системі координат $SXYZ$, орієнтацію якої задають:

- 1) з урахуванням конструкції об'єкта.
- 2) з урахуванням орієнтації системи координат моделі об'єкта, або встановлюють.
- 3) паралельно системі координат $Sxyz$ знімка.

За наявності в об'єкті системи координат $OXYZ$ систему координат $SXYZ$ використовують на етапі оброблення як проміжну.

Щоб виконати перетворення координат з системи координат знімка в систему координат об'єкта, потрібно знати взаємну орієнтацію цих систем. Вона визначається елементами зовнішнього орієнтування знімка, які фіксують систему координат $Sxyz$ знімка в такому положенні відносно системи координат $OXYZ$ об'єкта, яке вона мала у момент зйомки. Елементів зовнішнього орієнтування існує шість: три лінійних і три кутових.

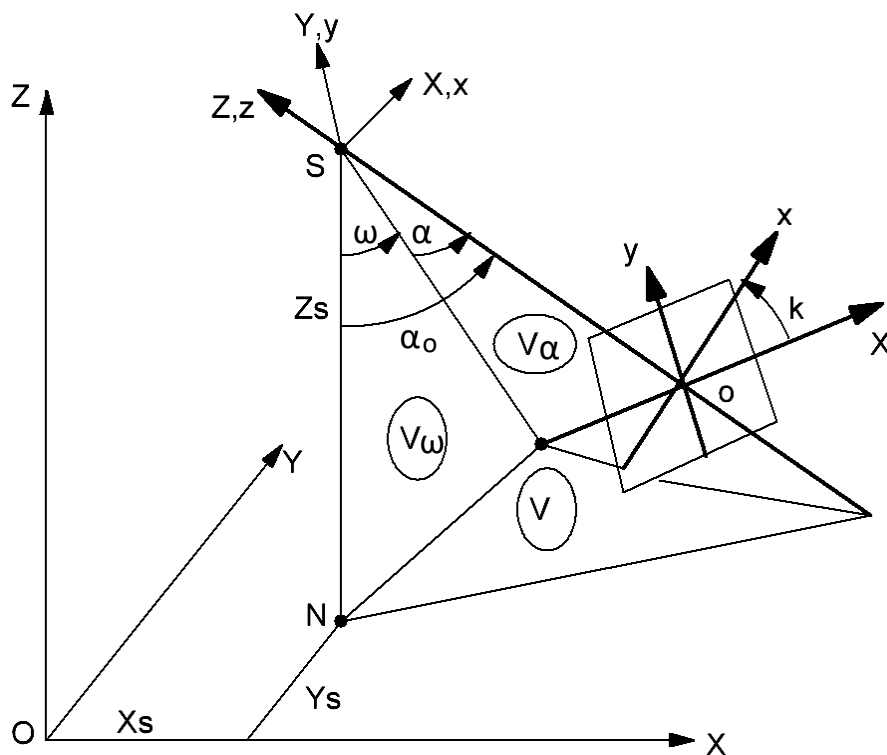


Рисунок 3.11 – Варіанти систем координат об'єкта

Лінійними елементами зовнішнього орієнтування знімка є координати X_S, Y_S, Z_S центра проекції (рис. 3.11), що визначають його положення в системі координат об'єкта і характеризують незбіг початків систем координат $Sxyz$ і $OXYZ$.

Кутовими елементами зовнішнього орієнтування знімка є кути ω, α, k , що визначають величину відхилення системи координат $Sxyz$ від положення, в якому вона була паралельною системі координат $OXYZ$.

У правій системі координат первинною вважається вісь X . Довкола цієї осі виконується перший поворот на кут ω , другий поворот – довкола осі Y на кут α , а третій поворот – довкола осі Z на кут k . Позитивним напрямом повороту правої системи координат прийнято поворот проти годинникової стрілки, якщо дивитися з боку позитивного напрямку координатної осі, довкола якої виконується поворот. Отже, якщо до моменту відображення знімка його система координат виконала три повороти проти ходу годинникової стрілки від положення, при якому вона була паралельною системі координат об'єкта, то кути ω, α, k вважаються позитивними.

Кути ω і α (див. рис. 3.11) визначають величину відхилення головного оптичного променя S_0 , з яким поєднано вісь Z системи координат знімка, від положення, при якому він був паралельним осі Z системи координат об'єкта.

Кут ω називають поперечним кутом нахилу знімка. Він лежить у вертикальній площині V_ω , яку проведено через вертикальну лінію SN паралельно координатній площині YZ .

Кут α називають подовжнім кутом нахилу знімка. Він лежить у площині V_α , яку проведено через головний оптичний промінь S_0 перпендикулярно до координатної площині YZ .

Кут k називають кутом розвороту знімка в своїй площині довкола головного оптичного променя S_0 . Кут k лежить у площині знімка і визначає відхилення осі x системи координат $Sxyz$ від лінії Ox перетину знімка площиною V_α . Ця лінія є паралельною координатній площині XZ .

Кут α_0 визначає нахил головного оптичного променя S_0 в головній вертикальній площині V і його називають кутом нахилу знімка. Кути ω і α є проєкціями кута α_0 на площинах V_ω і V_α .

3.9 Формули перетворення координат між системами координат похилого знімка і об'єкта

Використовують два варіанти формул перетворення координат між системами координат похилого знімка і об'єкта.

Перший варіант – це перетворення координат з системи координат знімка в систему координат об'єкта. Ці формули забезпечують обчислення координат точок на об'єкті за координатами, що виміряні на знімку.

Другий варіант порівняно з першим варіантом є оберненим перетворенням координат з системи координат об'єкта в систему координат знімка. Ці формули дозволяють за координатами точок об'єкта і заданими значеннями елементів

орієнтування знімка розрахувати безпомилкові координати точок знімка. Це роблять, наприклад, при визначенні помилок при нанесенні на знімок контурів майбутніх будівель, щоб визначити їх положення відносно існуючих будов, і таке інше.

Формули першого варіанта можна отримати з вихідних формул, випустивши індекс "л":

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} X_S \\ Y_S \\ Z_S \end{bmatrix} + NA_{\omega ak} \begin{bmatrix} x - x_0 \\ y - y_0 \\ -f \end{bmatrix} = \begin{bmatrix} X_S \\ Y_S \\ Z_S \end{bmatrix} + N \begin{bmatrix} c_{11} & c_{12} & c_{13} \\ c_{21} & c_{22} & c_{23} \\ c_{31} & c_{32} & c_{33} \end{bmatrix} \begin{bmatrix} x - x_0 \\ y - y_0 \\ -f \end{bmatrix}.$$

Оскільки базис фотографування і другий знімок є відсутніми, значення N можна визначити лише тоді, якщо є відомими висоти Z точок. Таким чином, зробимо висновок, що за одним знімком визначити три координати для точок об'єкта є неможливим. Визначимо масштабний коефіцієнт N , використовуючи формулу для знаходження координати Z :

$$N = \frac{Z - Z_S}{c_{31}(x - x_0) + c_{32}(y - y_0) - c_{33}f}.$$

Підставивши N у формулу для визначення координат X і Y , отримаємо формули обчислення планових координат точок об'єкта в його системі координат за координатами точок похилого знімка:

$$X = X_S + (Z - Z_S) \frac{c_{11}(x - x_0) + c_{12}(y - y_0) - c_{13}f}{c_{31}(x - x_0) + c_{32}(y - y_0) - c_{33}f};$$

$$Y = Y_S + (Z - Z_S) \frac{c_{21}(x - x_0) + c_{22}(y - y_0) - c_{23}f}{c_{31}(x - x_0) + c_{32}(y - y_0) - c_{33}f}.$$

Якщо координати точок об'єкта потрібно визначити в системі координат $SXYZ$, то в останніх формулах одержимо $X_S = Y_S = Z_S = 0$. Крім того, замінимо Z на $-H$ і отримаємо

$$X = -H \frac{c_{11}(x - x_0) + c_{12}(y - y_0) - c_{13}f}{c_{31}(x - x_0) + c_{32}(y - y_0) - c_{33}f};$$

$$Y = -H \frac{c_{21}(x - x_0) + c_{22}(y - y_0) - c_{23}f}{c_{31}(x - x_0) + c_{32}(y - y_0) - c_{33}f}.$$

Для виведення формул другого варіанта перетворимо вихідні формули на обернені, тобто отримаємо формули, за якими можна розрахувати координати точок знімка в його системі координат $Sxyz$ за координатами точок об'єкта:

$$\begin{bmatrix} x - x_o \\ y - y_o \\ -f \end{bmatrix} = N^{-1} A_{\omega\alpha k}^T \begin{bmatrix} X - X_S \\ Y - Y_S \\ Z - Z_S \end{bmatrix} = N^{-1} \begin{bmatrix} c_{11} & c_{12} & c_{13} \\ c_{21} & c_{22} & c_{23} \\ c_{31} & c_{32} & c_{33} \end{bmatrix} \begin{bmatrix} X - X_S \\ Y - Y_S \\ Z - Z_S \end{bmatrix}.$$

Запишемо формулу

$$-f = N^{-1} [c_{13}(X - X_S) + c_{23}(Y - Y_S) - c_{33}(Z - Z_S)].$$

З неї отримаємо формулу розрахунку масштабного коефіцієнта

$$N^{-1} = \frac{-f}{c_{13}(X - X_S) + c_{23}(Y - Y_S) - c_{33}(Z - Z_S)}$$

і підставимо її в два перші рівняння. В результаті отримаємо формули розрахунку за координатами точок об'єкта похилого знімка в системі координат $o'xyz$:

$$x = x_o - f \frac{c_{11}(X - X_S) + c_{21}(Y - Y_S) - c_{31}(Z - Z_S)}{c_{13}(X - X_S) + c_{23}(Y - Y_S) - c_{33}(Z - Z_S)};$$

$$y = y_o - f \frac{c_{12}(X - X_S) + c_{22}(Y - Y_S) - c_{32}(Z - Z_S)}{c_{13}(X - X_S) + c_{23}(Y - Y_S) - c_{33}(Z - Z_S)}.$$

3.10. Перетворення координат із системи координат горизонтального знімка в систему координат об'єкта

Якщо похилий і горизонтальний знімки отримано однією і тією ж фотокамерою з одного центра проєкції, запишемо формули перетворення координат точок із системи координат $Sxyz$ похилого знімка в систему координат $Sx^o y^o z^o$ горизонтального знімка. На обох знімках є зображення однієї й тієї ж ділянки місцевості, координати XYZ мають бути однаковими. З урахуванням цього прирівняємо праві частини цих формул:

$$\frac{Nx^o}{f} = -N \frac{c_{11}(x - x_o) + c_{12}(y - y_o) - c_{13}f}{c_{31}(x - x_o) + c_{32}(y - y_o) - c_{33}f};$$

$$\frac{Ny^o}{f} = -N \frac{c_{21}(x - x_o) + c_{22}(y - y_o) - c_{23}f}{c_{31}(x - x_o) + c_{32}(y - y_o) - c_{33}f}.$$

Після перетворень отримаємо

$$x^o = -f \frac{c_{11}(x - x_o) + c_{12}(y - y_o) - c_{13}f}{c_{31}(x - x_o) + c_{32}(y - y_o) - c_{33}f};$$

$$y^o = -f \frac{c_{21}(x - x_o) + c_{22}(y - y_o) - c_{23}f}{c_{31}(x - x_o) + c_{32}(y - y_o) - c_{33}f}.$$

Ці формули дозволяють перерахувати координати точок похилого знімка на площину горизонтального знімка, на якому зображення точок буде отримано із збереженням масштабу зйомки і без впливу кута нахилу знімка. Такі формули називають формулами трансформації.

4 ЕКОНОМІЧНА ЧАСТИНА

Розрахунок собівартості і ціни розробки алгоритмів оцінювання траєкторії руху ЛА за результатами відео спостереження.

4.1 Мета економічної частини

Метою даного розділу в дипломному проекті магістра є економічне обґрунтування розробки макету для візуалізації роботи алгоритму. Для досягнення поставленої мети необхідно вирішити такі завдання:

- 1) провести сегментацію ринку приладу;
- 2) визначити конкурентоспроможність даної розробки;
- 3) розрахувати трудомісткість виконання робіт;
- 4) скласти кошториси витрат на розробку;
- 5) провести розрахунок заробітної плати, визначити вартість об'єкта і розрахувати очікуваний прибуток від реалізації комплексу;
- 6) розрахувати точку беззбитковості і побудувати графік беззбитковості.

4.2 Сегментування ринку

Сегментування ринку - це виокремлення певної кількості покупців, що мають схожу або однакову реакцію на комплекс маркетингових заходів підприємства, тобто подібно сприймають параметри товару, його ціну, місце збуту та маркетингові комунікації.

Метою сегментування - є максимальне проникнення підприємства на сегменти ринку замість того, щоб розпилювати зусилля по всьому ринку. Політика сегментування ринку передбачає виділення окремих частин (сегментів) ринку, що відрізняється один від одного характеристиками попиту на товари та послуги і реакцією на маркетингові дії. Метод сегментації полягає у визначенні

на ринку груп покупців. Сегмент повинен бути досить великим в плані кількості споживачів, також в плані купівельної спроможності для того, щоб забезпечити прибутковий продаж. Сегмент, що складається з невеликої кількості покупців з низькою купівельною спроможністю не буде прибутковим для підприємства, який займається торгівлею великого обсягу товарів.

Однчасне обслуговування кількох ніш - стратегія, за якої компанія пропонує низку незалежних товарів або послуг, розрахованих на залучення клієнтів, що належать до окремих вузьких сегментів ринку.

У процесі сегментування виявляються сегменти ринку. Сегмент - це група споживачів однаково реагуючих на товар. Сегмент ринку оцінює ряд характеристик: число можливих споживачів, місткість ринку, можливі темпи зростання ємності по роках, споживчі ціни, прибутковість продажів і т.д.

Загальний порядок сегментування ринку можна представити таким чином:

- 1) виявлення основних принципів і факторів сегментації для даного виробу;
- 2) сегментування ринку: визначення складу груп споживачів, розподіл ринку на сегменти, опис профілю кожного сегмента, розрахунки річної ємності сегмента і всього ринку;
- 3) оформлення результатів сегментування у вигляді таблиці;
- 4) аналіз інформації про сегменти, вибір сегментів для подальшого аналізу;
- 5) позиціонування товару в обраних сегментах, визначення цільової ємності.

Споживачами комплексу, що розробляється, є університети та підприємства України і ближнього зарубіжжя, а також студенти, або учні, які цікавляться новими розробками.

Основною характеристикою сегмента є ємність - кількість виробів, які можуть бути продані за рік.

Розрахунок повної ємності ринку будемо проводити за формулою:

$$S_{\text{повн.}i} = N_i \cdot Q_i \cdot m_i \quad (4.1)$$

де N_i - кількість підприємств споживачів виробу в i -тому сегменті;

Q_i - середня річна програма виробів в i -тому сегменті, для яких буде поставлятися розглянутий товар;

m_i - кількість виробів, що йдуть в один виріб-споживач (1шт.).

Сегментування і розрахунок ємності ринку представлені в табл.4.1.

Таблиця 3.1 – Сегментування і розрахунок повної ємності ринку

Сегменти ринку	Код сегменту	Кількість підприємств-споживачів	Середньо – річна програма виробів підприємств - споживачів	Повна ємність шт./рік
Підприємства				

споживачі України	А	3	6	18
Підприємства споживачі ближнього зарубіжжя	Б	5	10	50
Підприємства споживачі далекого зарубіжжя	В	6	12	72
Усього		14	28	140

Маємо дані в таблиці 4.1, із них можна побачити, що повна місткість ринку становить 140 копій на рік.

4.3 Аналіз конкурентоспроможності

Конкурентоспроможність товару – це ступінь його відповідності обраному ринку з комерційних, технічних і економічних показників, забезпечуючи можливість збуту товару на цьому ринку. Це ті характеристики, які вигідно вирізняють даний товар від товарів-конкурентів.

Проведемо оцінку конкурентоспроможності розробляємої системи. Аналіз конкурентоспроможності проводимо шляхом порівняння характеристик системи з характеристиками виробів аналогів з певних параметрів, використовуючи метод комплексних показників якості з розрахунком узагальнених показників якості (з використанням гіпотетичного варіанту).

Як показники якості системи будуть використовуватись такі параметри, як: швидкість визначення об'єкту, точність визначення відстані, швидкість роботи програми. Дані характеристики впливають на роботу усієї системи управління, тобто визначають кращі техніко-економічні параметри.

Визначимо абсолютне значення i -х показників j варіантів p_{ij} у балах. Показникам якості присвоюємо коефіцієнти вагомості b_i

$$\sum_{i=1}^n b_i = 1 \text{ і } b_1 < 0, i = 1, \quad (4.2)$$

де n – номер показників якості.

Показники якості розділяють на мінімізовані та максимізовані і формують гіпотетичний (еталонний) варіант.

Розрахуємо для кожного j варіанту відносні значення i -х показників (k_{ij}) шляхом порівняння p_{ij} з p_i ГИП (з урахуванням умови $k_{ij} \leq 1$).

Таким чином, узагальнений показник якості розробленого комплексу є найбільшим – 0.65, що означає, що даний виріб є найбільш конкурентоспроможним.

Отримали узагальнені показники k_j^0 для всіх розглянутих варіантів. Розраховуємо рівні якості нового комплексу в порівнянні з комплексами конкурентами.

$$Y_{j-b} = \frac{k_j^0}{k_k^0}, \quad (4.3)$$

де k_j^0 - узагальнений показник приладу-конкуренту.

Рівні якості нової системи в порівнянні з системами-конкурентами:

$$Y_{1-5} = \frac{0.65}{0.34} = 1,91, Y_{2-5} = \frac{0.65}{0.57} = 1.14, Y_{3-5} = \frac{0.65}{0.22} = 2,95,$$

$$Y_{4-5} = \frac{0.65}{0.57} = 1.1. \quad (3.4)$$

Рівень якості нової техніки, в порівнянні із змінною досить високий. Основний показник якості систем подібного типу - швидкість конфігурування та надійність системи, чим як раз наш прототип перевершує конкурентів.

4.4 Розрахунок собівартості і ціни виготовлення установки

Собівартість продукції складається з ряду найменувань витрат. Сюди входять: витрати на основні матеріали, на великі комплектуючі вироби, пряма і додаткова заробітна плата, витрати на утримання та експлуатацію обладнання, утримання транспорту, а також цілий ряд загальнодержавних податків і відрахувань.

Для проектування установки необхідна участь наступних робочих: розробник, керівник, інженер. Тривалість робочого місяця в середньому приймемо 22 дня. Склад виконавців наведено в таблиці 4.4. Розрахуємо тривалість розробки за видами робіт.

Далі необхідно обчислити основну заробітну плату (далі - ОЗП), з урахуванням трудовитрат, кількості виконавців і середньоденна заробітна плата (далі - ЗП). Для цього кількість днів, відпрацьованих окремими виконавцями, необхідно помножити на їх денні оклади.

$$\text{ОЗП} = \sum N_i \text{ЗП}_{\text{ср}}, \quad (4.5)$$

Де N_i - кількість днів, відпрацьоване i -ми виконавцями за стадіями;

$\text{ЗП}_{\text{ср}}$ - денні оклади i -х виконавців.

Додаткова заробітня плата (ДЗП) розрахується за формулою:

$$\text{ДЗП} = \frac{20\% \cdot \text{ОЗП}}{100\%} \quad (4.6)$$

Результати розрахунків містяться в таблиці 4.2.

Таблиця 3.2 - Склад виконавців роботи 3500+3635+4277

Посада	Посадові оклади, грн.	
	За місяць	За день
Розробник	7700	350
Керівник	16000	727
Інженер	6700	305,5

Проведемо розрахунок трудомісткості робіт. Результати розрахунків наведені в таблиці 4.3

Таблиця 4.3 - Розрахунок трудомісткості робіт

Вид робіт	Тривалість, дні	Трудомісткість, чол./дні	Виконавці		
			Керівник	Розробник	Інженер
1	2	3	4	5	6
Постановка задачі	1	1	+	-	-
Розробка ТЗ	1	1	-	+	-
Погодження та затвердження ТЗ	1	3	+	+	+
Закупка товарів	2	2	-	-	+
Виробничі роботи	2	2	-	-	+
Розробка алгоритмів	5	5	-	+	-
Розробка програми	5	5	-	-	+
Налагодження продукту	2	4	-	+	+
Випробування і здача продукту в експлуатацію	2	6	+	+	+
Разом	21	29	5	10	14

Обчислимо основну заробітну плату розробників установки, з урахуванням трудовитрат, кількості виконавців і середньоденний ЗП. Для цього кількість днів, відпрацьованих окремими виконавцями, множать на їх денні оклади:

$$\text{ОЗП} = 350 \cdot 10 + 727 \cdot 5 + 305,5 \cdot 14 = 11412 \text{ грн.} \quad (4.7)$$

Додаткова заробітна плата обчислюється за формулою:

$$\text{ДЗП} = 20\% \cdot \text{ОЗП} = 11412 \cdot 20\% = 2282,4 \text{ грн.} \quad (4.8)$$

Розрахуємо вартість покупних елементів, необхідних для виготовлення установки. Перелік покупних елементів складається з урахуванням переліку блоків функціональної схеми (див. таблицю 4.4). Ціни вказані в гривнях.

Таблиця 4.4 – Перелік покупних елементів

№	Елементи	Кількість	Ціна, грн.	Вартість
1	Веб-камера	1	400,00	400,00
Загальна вартість				400

Відрахування в єдиний соціальний фонд складають 22% від основної заробітної плати і додаткової заробітної плати:

$$Z_{\text{відр}} = \frac{\text{ОЗП} + \text{ДЗП}}{100} \cdot 22 = \frac{11412 + 2282,4}{100} \cdot 22 = 3012,8 \text{ грн.} \quad (4.9)$$

Таблиця 4.5 – Перелік обладнання

Найменування	Кількість	Ціна, грн
Стіл	1 шт	3000
Стілець	1 шт	600
Лампа	1 шт	420
Комп'ютер	1 шт	20000
Інструменти	1 шт	3000
Сума		27020

Таблиця 4.6 - Розрахунок собівартості і ціни виробу за статтями

№	Статті	Сума, грн	Примітка
1	Основна заробітна плата (ОЗП)	11412	
2	Додаткова ЗП (ДЗП)	2282,4	20% від ОЗП
3	Єдиний соціальний фонд	2433,5	22%*(ОЗП+ДЗП)
4	Матеріали й куплені вироби (C_m)	400	Таблиця 6.4
5	Амортизація	537	$A_m = \frac{OC \cdot 0,25 \cdot 21}{12 \cdot 22}$
6	Витрати на утримання обладнання	2702	10% від табл.6.5
7	Додаткові витрати	3687	40% від ОЗП
8	Виробнича собівартість (С)	20820,5	п.1+п.2+п.3+...+п.7

9	Адміністративні витрати	4148	45% від ОЗП
10	Витрати на збут	520,5	2,5% від п.8
11	Собівартість власних робіт	25489	п.1+п.2+п.3+...+п.10
12	Прибуток (П)	5097	20% від п.11
13	Ціна без НДС	30586	П+п.11
14	НДС	6117,3	20% від ціни без ПДВ
15	Ціна з НДС	36730,3	п.13+п.14

Річна норма амортизаційних відрахувань (A_m) розраховується як 25% від вартості обладнання. Так, як трудомісткість становить 21 день, то амортизація обчислюється за формулою:

$$A_m = \frac{OC \cdot 0,25 \cdot 21}{12 \cdot 22} = \frac{27020 \cdot 0,25 \cdot 21}{12 \cdot 22} = 537 \text{ грн.} \quad (4.10)$$

Таким чином, собівартість власних робіт проекту на розроблення лабораторного практикуму складає 25489,5 грн, а ціна проекту (з НДС) – 36730 грн.

Кількість замовлених екземплярів має бути не менше ніж 140 штук. Виробничу собівартість одного екземпляра програмного продукту (BC) визначається за формулою:

$$BC_0 = BC / \text{кпп}, \quad (4.11)$$

де BC - виробничі витрати;
 КПП - кількість замовлених екземплярів.

$$BC_0 = \frac{25489}{140} = 182,1 \quad (4.12)$$

Повна собівартість одного екземпляру програмного продукту $СП$ складається з суми виробничої собівартості BC_0 , адміністративних витрат AB і витрат на збут B_3 , які приходяться на один екземпляр програмного продукту:

$$СП_0 = BC_0 + AB_0 + B_3_0 \quad (4.13)$$

Адміністративні витрати AB_0 , які приходяться на один екземпляр програмного продукту, визначається за формулою:

$$AB_0 = \frac{AB}{\text{кпп}} = \frac{4148}{140} = 29,6 \text{ грн.} \quad (4.14)$$

Витрати на збут B_3_0 , які приходяться на один екземпляр програмного продукту, визначаємо за формулою:

$$B_{30} = \frac{B_3}{K_{ПП}} = 3,7 \text{ грн.} \quad (4.15)$$

Таким чином, СП₀ = 182,1 + 29,6 + 3,7 = 215,4 грн.

Рентабельність продукції (норма продукту) - це відношення загальної суми прибутку до витрат виробництва і реалізації продукції (відносна величина прибутку, що припадає на 1 грн поточних витрат):

$$P_{п} = \frac{Ц - BC}{BC} * 100\% = \frac{36730 - 25489}{25489} * 100\% = 45\% \quad (4.16)$$

Отже рентабельність 45%. Розрахуємо величину оптової ціни одного виробу ЦПП (без врахування НДС).

$$ЦПП = СП * (1 + P_{п} / 100) = 215,4 * (1 + 45 / 100) = 313,49 \quad (4.17)$$

Де P_п - коефіцієнт рентабельності.

Розрахуємо точку беззбитковості. Дохід від реалізації програмних продуктів знаходимо множенням ціни одного ПП на кількість замовлених примірників ПП:

$$ДР = ЦПП * K_{ПП} = 313,49 * 140 = 43888,6 \text{ грн.} \quad (4.18)$$

Аналітичний розмір критичної програми (РКП) розраховують діленням постійних витрат РПОСТВ на різницю між ціною одного програмного продукту ЦПП і змінними витратами, які приходяться на один екземпляр програмного продукту (ЗМВ₀), тобто

$$РКП = \frac{РПОСТВ}{(ЦПП - З_{МВ_0})} = \frac{15467,5}{313,49 - 95,5} = 71 \text{ шт.} \quad (4.19)$$

Річні постійні витрати РПОСТВ складаються із суми наступних витрат:

$$РП_{ост}В = ВΟΥ + А_{м} + ДВ + АВ + ВЗ = 3460 + 546 + 7342 + 6731 + 1057 = 19136 \text{ грн.} \quad (3.20)$$

Річні зміни витрати РЗ_мВ складаються із суми наступних витрат:

$$РЗ_{м}В = ВМ + ФОП + ЄСФОП = 1621 + (13496 + 2728) + 3460,5 = 21305,5 \text{ грн.} \quad (4.21)$$

Змінні витрати, які приходяться на один екземпляр програмного продукту, визначаємо діленням річних змінних витрат на річну програму випуску продукту:

$$З_{м}В_0 = \frac{РЗ_{м}В}{K_{ПП}} = \frac{21305,5}{140} = 152,1 \text{ грн.} \quad (4.22)$$

Річний дохід в точці беззбитковості:

$$D_{БЗ} = 264 * 216 = 57024 \text{ грн.} \quad (4.23)$$

Графік критичної програми випуску продукту представлений на рис. 4.1

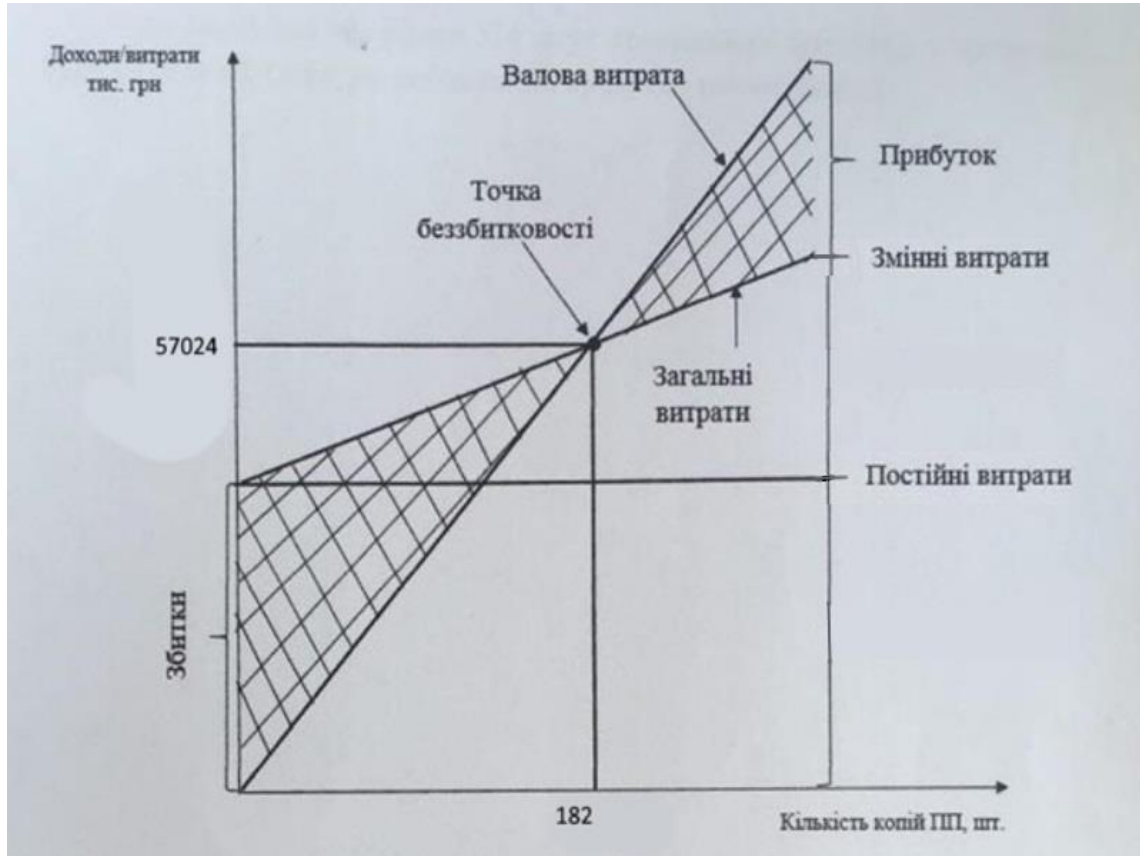


Рисунок 4.1 – Критична програма випуску продукту

4.5 Висновки по економічній частині

В даному розділі був проведений розрахунок ціни і собівартості для розробка алгоритмів оцінювання траєкторії руху ЛА за результатами відео спостережень. Ціна з НДС склала 44280. Розрахунок зроблений з урахуванням всіх необхідних трудовитрат, НДС склала 7500,3 гривень, відрахування в єдиний соціальний фонд – 2781,6 гривень.

Розроблена установка та система управління може скласти доволі високу конкуренцію на ринку збуту, так як собівартість виробництва і конструювання не дорожча аналогів на ринку.

ВИСНОВОК

У ході дипломного проектування було проведено такі роботи:

- за літературними джерелами проведено докладний аналіз існуючих методів побудови алгоритмів оцінки параметрів руху літальних апаратів та розрахунку траєкторій їх руху, виділені їх основні переваги та недоліки;
- для вирішення поставленого завдання був вибраний кореляційний метод, оскільки він є досить простим в реалізації, не потребує істотних обчислювальних ресурсів і забезпечує досить високу швидкодію;
- розроблено алгоритми стійкого виявлення та детектування ЛА та алгоритми оцінки їхнього розташування у кадрах відеопослідовності. Досліджено стійкість отриманих алгоритмів до впливу шумів різної інтенсивності;
- розроблено алгоритми оцінки параметрів траєкторії руху ЛА (кутових координат, дальності, швидкості польоту) у системі координат відеокамери, наведено проміри розрахунку параметрів траєкторії;
- докладно досліджено методи перетворення результатів розрахунку траєкторних параметрів руху ЛА у глобальну систему координат для їх комплексування з іншими засобами траєкторних вимірів;
- для практичної реалізації проекту основні алгоритми розрахунку траєкторних параметрів руху ЛА були запрограмовані та протестовані у програмі Матлаб. Повні лістинги цих програм винесено до додатків

Результати проведених досліджень є корисними. Вони можуть бути використані при вирішенні комплексного завдання проектування системи траєкторних вимірів для оцінки якості систем управління ЛА під час проведення технічних випробувань.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Основи побудови сучасних мобільних систем технічного зору [Текст] : навч. посіб. Ч. 1. Комп'ютери і засоби програмування / Л. О. Краснов, К. Ю. Дергачов, С. В. Багінський. – Харків : Нац. аерокосм. ун-т ім. М. Є. Жуковського «Харків. авіац. ін-т», 2018. – 104 с.
2. Цифрове оброблення зображень у програмі Matlab [Текст]: навч. посіб. до лаб. практикуму /Л. О. Краснов. – Харків: Нац. аерокосм. ун-т ім. М. Є. Жуковського «Харків. авіац. ін-т», 2017. – 99 с.
3. Конспект лекцій з дисципліни “Обробка сигналів та зображень” (для студентів денної форми навчання напряму 6.170101 «Безпека інформаційних і комунікаційних систем») / Укладачі: к.т.н., доцент Фриз М.Є., Стадник М. А. – Тернопіль: ТНТУ, 2015 – 97 с.
4. Обробка зображень: навч.-метод. посіб., Лавер В.О., Левчук О.М. – Ужгород : вид-во ПП «АУТДОР - ШАРК», 2021. – 51 с
5. Bertalmio M., Bertozzi A., Sapiro G. Navier-Stokes, fluid dynamics, and image and video inpainting/ Hawaii: Proc. IEEE Computer Vision and Pattern Recognition (CVPR) . — 2001. —PP. 213-226.
6. Guleryuz O.G. Nonlinear approximation based image recovery using adaptive sparse reconstructions and iterated denoising/ Part I: theory IEEE transactions on image processing. — 2006. Vol. 15(3).
7. Criminisi A., Perez P., Toyama K. Region filling and object removal by exemplar-based image inpainting/ IEEE Trans. Image Process. — 2004. Vol. 13(9). — PP. 28-34.
8. Scharstein D., Szeliski R. A Taxonomy and Evaluation of Dense Two-Frame StereoCorrespondence Algorithms // International Journal of Computer Vision. 2002. Vol. 47. no. 1-3. С. 7-42. 19, 24, 25, 28
9. Ohta Y., Kanade T. Stereo by intra- and inter- scanline search using dynamic programming// IEEE TPAMI. 7(2). 1985. С. 139-154. 20, 22
10. Jones D. G., Malik J. A computational framework for determining stereo correspondence from a set of linear spatial filters // ECCV. 12(3). 1992. С. 395-410. 22, 27, 28, 48, 49
11. Кириленко В.В. Економіка (навчальний посібник), Тернопіль: Економічна думка, 2002. – 193с.
12. Економіка підприємства: підручник / під заг. ред. д. е. н., проф. Ковальської Л.Л. та проф. Кривов'язюка І.В. Київ : Видавничий дім «Кондор», 2020. 700 с.

ДОДАТКИ

Додаток А

Лістинг програми оцінювання параметрів траєкторії руху синьої кульки в умовах дії перешкод

```

clc,close all,clear all
% Прочитування відеофайла
Video = mmreader('C:\Users\krasnov.MEDICA\Desktop\blue_ball.avi');
% Визначення характеристик лічених відеоданих
width = Video.Width;
height = Video.Height;
frameRate = Video.FrameRate;
numOfFrames = Video.NumberOfFrames;

% Читання демонстраційного кадру
frameNo = 20;
size(frameNo)
F=read(Video, frameNo);
figure,imshow(F),title('Демонстраційний кадр'),impixelinfo
F1=rgb2gray(F);
F2=im2double(F1);
figure,imshow(F2,title('Демонстраційний кадр у форматі double
'),impixelinfo

% Завдання дисперсії гауссових шумів
q=0.1

% Додавання гауссова шуму до демонстраційного кадру
F3 = imnoise(F2,'gaussian',0,q);
figure,imshow(F3),title('Зашумлений демонстраційний кадр у форматі
double'),impixelinfo

% Створення нульових кадрів
A=zeros(480,640);
B=zeros(480,640);

% Створення шаблону для обчислення кореляції
P=ones(25,25);
% figure,imshow(P),title('Шаблон у форматі double'),impixelinfo

% Читання і оброблення поточних кадрів
% Номер початкового кадру
u=15;
% Номер кінцевого кадру
z=120;
% Кількість оброблюваних кадрів
W=z-u+1
for k=u:z

```

```

% Читання поточного кадру
I = read(Video,k);

% Виділення синьої компоненти поточного кадру
I1=I(:, :, 3);
% figure,imshow(I1),title('Синя компонента поточного кадру
'),impixelinfo

% Перетворення синьої компоненти поточного кадру у формат
double
I2=im2double(I1);
% figure,imshow(I2),title(' Синя компонента поточного кадру у форматі
double'),impixelinfo

% Додавання гауссова шуму до поточного кадру
I3 = imnoise(I1, 'gaussian', 0, q);

% Перетворення зашумленого поточного кадру у формат double
I4=im2double(I3);
% figure,imshow(I4),title('Зашумлена синя компонента поточного кадру
у форматі double'),impixelinfo

% Обчислення взаємної кореляції кожного кадру з шаблоном
K=xcorr2(I2,P);
K1=xcorr2(I4,P);

% Обчислення максимумів кореляційних функцій
C=max(max(K));
C1=max(max(K1));

% Нормування кореляційних функцій на одиницю
L=K./C;
L1=K1./C1;
% figure,imshow(L),title('Нормована взаємно кореляційна функція
синьої компоненти ')
% figure,imshow(L1),title('Нормована взаємно кореляційна функція
зашумленої синьої компоненти')

% Бінаризація взаємно кореляційних функцій за порогом
BW=im2bw(L, 0.9999);
BW1=im2bw(L1, 0.9999);
% figure, imshow(BW),title('Бінаризація за порогом синьої
компоненти')
% figure, imshow(BW1),title('Бінаризація за порогом зашумленої синьої
компоненти')

% Нормування розміру зображень кореляційних функцій
Q=imcrop(BW, [0, 0, 640, 480]);
Q1=imcrop(BW1, [0, 0, 640, 480]);
% figure,imshow(Q),title('Нормування зображення кореляційної
функції'),impixelinfo

```

```

% figure,imshow(Q1),title('Нормування зображення кореляційної
функції')

% Визначення координат точок незашумленої траєкторії
% Умова пошуку -> Q(m,n)=1
for m=1:480
    for n=1:640
        if Q(m,n)==1
            x0=n;
            y0=m;
        end;
    end;
end;
% Визначення координат точок зашумленої траєкторії
% Умова пошуку -> Q1(m,n)=1
for m=1:480
    for n=1:640
        if Q1(m,n)==1
            x1=n;
            y1=m;
        end;
    end;
end;

% Оцінювання лінійного зсуву точок траєкторії через дію шумів
dx=abs(x1-x0);
dy=abs(y1-y0);
DX=(dx)^2;
DY=(dy)^2;
DT=sqrt(DX+DY);
DT1=DT;
V(k)=(DT1)^2;

% Оцінювання координат точок незашумленої траєкторії відносно початку
координат
XA=321; YA=241;
DX0=(x0-XA);
DY0=(y0-YA);
RX=(DX0)^2;
RY=(DY0)^2;

% Номер точки траєкторії
Nom=k

% Обчислення дальності від початку координат до поточної точки
незашумленої траєкторії
Range=sqrt(RX+RY)

% Обчислення кутової координати точок незашумленої траєкторії
Sangle=abs((DY0)/(Range));
Angle1=asin(Sangle)*180/pi;

```



```

if (y0<YA)
    if (x0>=XA)
        Angle=Angle1
    end;
    if (x0<XA)
        Angle=180-Angle1
    end;
end;
if (y0>YA)
    if (x0>=XA)
        Angle=360-Angle1
    end;
    if (x0<XA)
        Angle=180+Angle1
    end;
end;

if (y0==0)
    if (x0>=XA)
        Angle=0
    end;
    if (x0<XA)
        Angle=180
    end;
end;

% Оцінювання координат точок зашумленої траєкторії відносно початку
координат
XA=321; YA=241;
DX0=(x1-XA);
DY0=(y1-YA);
RX=(DX0)^2;
RY=(DY0)^2;

% Обчислення дальності від початку координат до поточної точки
зашумленої траєкторії
RangeN=sqrt(RX+RY)

% Обчислення кутової координати точок зашумленої траєкторії
Sangle=abs((DY0)/(RangeN));
Angle1=asin(Sangle)*180/pi;
if (y1<YA)
    if (x1>=XA)
        AngleN=Angle1
    end;
    if (x1<XA)
        AngleN=180-Angle1
    end;
end;
end;

```

```

    if (y1>YA)
        if (x1>=XA)
            AngleN=360-Angle1
        end;
        if (x1<XA)
            AngleN=180+Angle1
        end;
    end;

    if (y1==0)
        if (x1>=XA)
            AngleN=0
        end;
        if (x1<XA)
            AngleN=180
        end;
    end;

% Оцінювання кутового зсуву точок траєкторії через дію шумів
dangle=abs(Angle-AngleN);
V1(k)=(dangle)^2;

% Оцінювання зсуву точок траєкторії за дальністю через дію шумів
drange=abs(Range-RangeN);
V2(k)=(drange)^2;

% Перетворення точок траєкторії руху кульки у формат double
Q2=im2double(Q);
Q3=im2double(Q1);

% Побудова траєкторій руху кульки за відсутності і наявності шумів
A=A+Q2;
B=B+Q3;
end;

% Оцінювання дисперсії лінійного зсуву точок траєкторії,
обумовленої дією шумів
V(~V)=[];
MSEpoint=sum(V)/W
% Оцінювання кутового зсуву точок траєкторії через дію шумів
V1(~V1)=[];
MSEangle=sum(V1)/W
% Оцінювання зсуву точок траєкторії за дальністю через дію шумів
V2(~V2)=[];
MSErange=sum(V2)/W
% Оцінювання SKO лінійного відхилення точок траєкторії через дію
шумів
SKOpoint=sqrt(MSEpoint)
% Оцінювання SKO кутового відхилення точок траєкторії через дію
шумів
SKOangle=sqrt(MSEangle)

```

```
% Оцінювання SKO зсуву точок траєкторії за дальністю через дію шумів
SKOrange=sqrt(MSErange)

% Відображення траєкторії руху
figure,imshow(A),title('Траєкторія руху синьої кульки'),impixelinfo
figure,imshow(B),title('Траєкторія руху синьої кульки при дії шумів'),impixelinfo
```

Додаток Б

Лістинг програми оцінювання параметрів траєкторії руху об'єкта

```

clc,close all,clear all
% Прочитування відеофайла
Video = mmreader('C:\Users\Desktop\blue_ball.avi');
% Визначення характеристик лічених відеоданих
width = Video.Width;
height = Video.Height;
frameRate = Video.FrameRate;
numOfFrames = Video.NumberOfFrames;

% Читання демонстраційного кадру
frameNo = 20;
size(frameNo)
F=read(Video, frameNo);
figure,imshow(F),title('Демонстраційний кадр'),impixelinfo
F1=rgb2gray(F);
F2=im2double(F1);
figure,imshow(F2),title(' Демонстраційний кадр у форматі double
'),impixelinfo

% Створення нульових кадрів
A=zeros(480,640);
B=zeros(480,640);

% Створення шаблону для обчислення кореляції
P=ones(25,25);
% figure,imshow(P),title('Шаблон у форматі double'),impixelinfo

% ЧИТАННЯ І ОБРОБЛЕННЯ ПОТОЧНИХ КАДРІВ

% Номер початкового кадру
u=15;
% Номер кінцевого кадру
z=120;

% Число оброблюваних кадрів
W=z-u+1
% Читання кадрів через (1:2:3: ...)
v=1

nom = 0;
% Координати початку координат - середина екрана
XA=320; YA=240;
for k=u:v:z-v
    nom = nom+1
    % Читання поточного кадру
    I = read(Video,k);

```

```

    % Виділення синьої компоненти поточного кадру
    I1=I(:, :, 3);
% figure, imshow(I1), title('Синя компонента поточного
кадру'), impixelinfo

    % Перетворення синьої компоненти поточного кадру у формат
double
    I2=im2double(I1);
% figure, imshow(I2), title('Синя компонента поточного кадру у форматі
double'), impixelinfo

    % Обчислення взаємної кореляції поточного кадру з шаблоном
K=xcorr2(I2, P);
    % Обчислення максимумів кореляційних функцій
C=max(max(K));
    % Нормування кореляційних функцій на одиницю
L=K./C;
% figure, imshow(L), title('Нормована взаємно кореляційна функція
синьої компоненти')
    % Бінаризація взаємно кореляційних функцій по порогу
BW=im2bw(L, 0.9999);
% figure, imshow(BW), title('Бінаризація по порогу синьої компоненти')
    % Нормування розміру зображень кореляційних функцій
Q=imcrop(BW, [0, 0, 640, 480]);
% figure, imshow(Q), title('Кореляційні функції для поточних
кадрів'), impixelinfo

    % Читання наступного кадру
J = read(Video, k+v);
    % Виділення синьої компоненти наступного кадру
J1=J(:, :, 3);
% figure, imshow(J1), title('Синя компонента наступного
кадру'), impixelinfo

% Перетворення синьої компоненти наступного кадру у формат double
    J2=im2double(J1);
%figure, imshow(J2), title('Синя компонента наступного кадру у форматі
double '), impixelinfo

    % Обчислення взаємної кореляції кожного кадру з шаблоном
K1=xcorr2(J2, P);
    % Обчислення максимумів кореляційних функцій
C=max(max(K1));
    % Нормування кореляційних функцій на одиницю
L1=K1./C;
% figure, imshow(L1), title('Нормована взаємно кореляційна функція
синьої компоненти')
    % Бінаризація взаємно кореляційних функцій по порогу
BW1=im2bw(L1, 0.9999);
% figure, imshow(BW), title('Бінаризація по порогу синьої компоненти')

```

```

% Нормування розміру зображень кореляційних функцій
Q1=imcrop(BW1,[0,0,640,480]);
% figure,imshow(Q1),title('Кореляційні функції для наступних
кадрів'),impixelinfo

% Визначення координат точок вихідної траєкторії
% Умова пошуку -> Q(m,n)=1
for m=1:480
    for n=1:640
        if Q(m,n)==1
            x0=n-XA;
            y0=-m+YA;
        end;

        if Q1(m,n)==1
            x1=n-XA;
            y1=-m+YA;
        end;
    end;
end;

% Оцінювання дальності точки траєкторії до початку координат
Range_point(nom)=round(sqrt((x0)^2+(y0)^2));

% Оцінювання кута точки траєкторії відносно початку координат
RT2=abs((y0)/(Range_point(nom)));
RT3=asin(RT2)*180/pi;
if (y0<0)
    if (x0>=0)
        angle_point(nom)=360-RT3
    end;
    if (x0<0)
        angle_point(nom)=180+RT3
    end;
end;

if (y0>0)
    if (x0>=0)
        angle_point(nom)=RT3
    end;
    if (x0<0)
        angle_point(nom)=180-RT3
    end;
end;

if (y0==0)
    if (x0>=0)
        angle_point(nom)=0
    end;
    if (x0<0)
        angle_point(nom)=180
    end;
end;

```

```

        end;
    end;

    % Оцінювання лінійного зсуву точок траєкторії від кадру до
    кадру
    dx=x1-x0;
    dy=y1-y0;
    DX=(dx)^2;
    DY=(dy)^2;
    Vektor(nom)=(round(sqrt(DX+DY))/(v/30))

        angle=atan(abs(dy/dx))*180/pi;

    if (dy==0)
        if (dx>=0)
            angle_vector(nom) = 0
        end;
        if (dx<0)
            angle_vector(nom) = 180
        end;
    end;

    if (dx==0)
        if (dy>=0)
            angle_vector(nom) = 90
        end;
        if (dy<0)
            angle_vector(nom) = 270
        end;
    end;

    if (dy>0)
        if (dx>0)
            angle_vector(nom) = angle
        end;
        if (dx<0)
            angle_vector(nom) = 180-angle
        end;
    end;

    if (dy<0)
        if (dx>0)
            angle_vector(nom) = 360-angle
        end;
        if (dx<0)
            angle_vector(nom) = 180+angle
        end;
    end;

    % Побудова траєкторій руху початкової і зсунутої послідовностей

```

```

    A=A+Q;
    B=B+Q1;
end;

% ВІДОБРАЖЕННЯ РЕЗУЛЬТАТІВ РОЗРАХУНКУ ПАРАМЕТРІВ ТРАЄКТОРІЇ

figure,imshow(A),title('Траєкторія руху по поточних
кадрах'),impixelinfo
% figure,imshow(B),title('Траєкторія руху по зсунутих
кадрах'),impixelinfo

for i=1:1:nom
    xxx(i)=i;
end;

figure,subplot(2,1,1),stem(xxx,Range_point,'b','s','MarkerFaceColor'
,'b','MarkerSize',2),grid, title(' Дальність до точки траєкторії від
початку координат ');
xlabel('№ кадру');ylabel('Дальність');
subplot(2,1,2),stem(xxx,Vektor,'r','s','MarkerFaceColor','r','Marker
Size',2),grid,title(' Модуль вектора швидкості');
xlabel('№ кадра');ylabel('Модуль вектора швидкості');
figure,subplot(2,1,1),stem(xxx,angle_point,'b','s','MarkerFaceColor'
,'b','MarkerSize',2),grid,title(' Кут точки траєкторії');
xlabel('№ кадра');ylabel(' Кут точки траєкторії, град');
subplot(2,1,2),stem(xxx,angle_vector,'r','s','MarkerFaceColor','r','
MarkerSize',2),grid,title('Кут вектора швидкості');
xlabel('№ кадра');ylabel('Кут вектора швидкості, град');

```


Додаток В

Код програми формування матриці поворотів (напрямних косинусів) у тривимірному просторі:

```

clc,close all,clear all
% ФОРМУВАННЯ МАТРИЦІ ПОВОРОТУ (напрямних косинусів)
% Задаємо кути повороту
for alpha=-pi/2:pi/180:pi/2
    for beta=-pi/2:pi/180:pi/2
        for gamma=-pi/2:pi/180:pi/2
            % Обчислюємо синуси і косинуси кутів повороту
            A=sin(alpha);B=cos(alpha);C=sin(beta);D=cos(beta);E=sin(gamma);
            F=cos(gamma);
            % Матриця повороту на кут alpha
            Q=[1 0 0;0 B A;0 -A B];
            % Матриця повороту на кут beta
            R=[D 0 -C;0 1 0;C 0 D];
            % Матриця повороту на кут gamma
            S=[F E 0;-E F 0;0 0 1];
            % Транспонування матриць повороту
            Q1=Q';
            R1=R';
            S1=S';
            U=Q1*R1*S1
            % Обчислюємо кути повороту знімка в градусах
            Angle_alpha=-atan(-A/B)*180/pi
            Angle_beta=asin(A*E-B*C*F)*180/pi
            Angle_gamma=atan(-E/F)*180/pi
            % Формуємо елементи матриці напрямних косинусів кутів зовнішнього
            орієнтування знімка:
            C_11=D*F; C_12=-D*E; C_13=C;
            C_21=B*E+A*C*F; C_22=B*F-A*C*E; C_23=-A*D;
            C_31=A*E-B*C*F; C_32=A*F+B*C*E; C_33=B*D;
            % Формуємо матрицю напрямних косинусів
            C_ij=[C_11 C_12 C_13;C_21 C_22 C_23;C_31 C_32 C_33]
            % Перевірка правильності визначення матриці напрямних косинусів:
            P_1=(C_11)^2+(C_12)^2+(C_13)^2
            P_2=(C_21)^2+(C_22)^2+(C_23)^2
            P_3=(C_31)^2+(C_32)^2+(C_33)^2
            P_11=C_11*C_21+C_12*C_22+C_13*C_23
            P_22=C_11*C_31+C_12*C_32+C_13*C_33
            P_33=C_21*C_31+C_22*C_32+C_23*C_33
        end
    end
end

```