

**КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ БУДІВНИЦТВА І  
АРХІТЕКТУРИ**

Факультет геоінформаційних систем і управління територіями

(повне найменування інституту, назва факультету)

Кафедра геоінформатики і фотограмметрії

(повна назва кафедри )

**Пояснювальна записка**  
до дипломного проекту (роботи)  
бакалавра  
(освітньо-кваліфікаційний рівень)

На тему «Розроблення функції Python для автоматизації процесу  
оцифрування горизонталей для побудови цифрових моделей рельєфу»

Виконав: студент 4 курсу, групи ГСТ-41  
за напрямком підготовки  
193 «Геодезія та землеустрій»  
(шифр і назва напрямку підготовки, спеціальності)

Карнарук К. А.  
(прізвище та ініціали)

Керівник: Максимова Ю. С.  
(прізвище та ініціали)

Рецензент: Лепетюк В. Б.  
(прізвище та ініціали)

Київ-2022р.

# КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ БУДІВНИЦТВА І АРХІТЕКТУРИ

Інститут, факультет Геоінформаційних систем і управління територіями  
Кафедра Геоінформатики і фотограмметрії  
Освітньо-кваліфікаційний рівень бакалавр  
Напрямок підготовки 193 “Геодезія та землеустрій”

(шифр і назва)

Спеціальність 7.08010105 “Геоінформаційні системи і технології”  
(шифр і назва)

**ЗАТВЕРДЖУЮ**

Завідувач кафедри

проф., д.т.н. Карпінський Ю. О.

“27” квітня 2022 року

## **ЗАВДАННЯ НА ДИПЛОМНИЙ ПРОЕКТ (РОБОТУ) СТУДЕНТУ**

Карнарук Кирило Андрійович

(прізвище, ім'я, по батькові)

1. Тема проекту (роботи): Розроблення функції Python для автоматизації процесу оцифрування горизонталей для побудови цифрових моделей рельєфу

керівник проекту (роботи) доц., к. г. н. Максимова Юлія Сергіївна

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

затверджені наказом вищого навчального закладу від 27 квітня 2022 року №287/2

2. Строк подання студентом проекту (роботи) 15.06.2020

3. Вихідні дані до проекту (роботи)

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

Вступ

1. Характеристика предметної сфери застосування ЦМР при актуалізації картографічних матеріалів рельєфу

1.1. Поняття цифрової моделі рельєфу та її застосування

1.2. Способи побудови ЦМР

1.3. Класифікація та порівняння алгоритмів для побудови ЦМР

Висновки до розділу 1

2. Розроблення алгоритму для автоматизації процесу оцифрування горизонталей для побудови цифрових моделей рельєфу з використанням мови Python

2.1. Загальний огляд мови Python та бібліотек для обробки зображень

2.2. Поняття колірних моделей та їх застосування при обробленні зображень та автоматичній обробці горизонталей

2.3. Технологічна схема автоматизації процесу оцифрування горизонталей для побудови цифрових моделей рельєфу з використанням мови Python

Висновки до розділу 2

3. Дослідна реалізація функції Python для автоматизації процесу оцифрування горизонталей для побудови цифрових моделей рельєфу

3.1. Підготовка вихідних даних дослідної ділянки

3.2. Функція Python для автоматизації процесу оцифрування горизонталей

3.3. Результати застосування функції Python для автоматизації процесу оцифрування горизонталей

Висновки

Список літератури

6. Консультанти розділів проекту (роботи)

Номер розділу	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв

7.Дата видачі завдання

---

### КАЛЕНДАРНИЙ ПЛАН

№з/п	Назва етапів виконання дипломного проекту (роботи)	Строк виконання етапів проекту (роботи )	Примітка
1	Розділ 1. Характеристика предметної сфери застосування ЦМР при	10.04.2022	

	актуалізації картографічних матеріалів рельєфу		
2	Розділ 2. Розроблення алгоритму для автоматизації процесу оцифрування горизонталей для побудови цифрових моделей рельєфу використанням мови Python	30.04.2022	
3	Розділ 3. Дослідна реалізація функції Python для автоматизації процесу оцифрування горизонталей для побудови цифрових моделей рельєфу	30.05.2022	
4	Розробка графічного матеріалу	15.06.2022	
5	Оформлення пояснювальної записки	15.06.2022	
6	Подача проекту на попередній захист та рецензування	15.06.2022	

**Студент**

\_\_\_\_\_ Карнарук К. А.  
 ( підпис ) ( прізвище та ініціали )

**Керівник проекту (роботи)**

\_\_\_\_\_ Максимова Ю. С.  
 ( підпис ) ( прізвище та ініціали )

## Зміст

Вступ.....	7
1. Характеристика предметної сфери застосування ЦМР при актуалізації картографічних матеріалів рельєфу.....	9
1.1. Поняття цифрової моделі рельєфу та її застосування.....	10
1.2. Способи побудови ЦМР.....	12
1.3. Класифікація та порівняння алгоритмів для побудови ЦМР.....	19
Висновки до розділу 1.....	21
2. Розроблення алгоритму для автоматизації процесу оцифрування горизонталей для побудови цифрових моделей рельєфу з використанням мови Python.....	22
2.1. Загальний огляд мови Python та бібліотек для обробки зображень.....	23
2.2. Поняття колірних моделей та їх застосування при обробленні зображень та автоматичній обробці горизонталей.....	26
2.3. Технологічна схема автоматизації процесу оцифрування горизонталей для побудови цифрових моделей рельєфу з використанням мови Python.....	28
Висновки до розділу 2.....	29
3. Дослідна реалізація функції Python для автоматизації процесу оцифрування горизонталей для побудови цифрових моделей рельєфу.....	30
3.1. Підготовка вихідних даних дослідної ділянки.....	31
3.2. Функція Python для автоматизації процесу оцифрування горизонталей..	32
3.3. Результати застосування функції Python для автоматизації процесу оцифрування горизонталей.....	34
Висновки до 3 розділу.....	37

Висновки.....	38
Список використаних джерел.....	39
Додатки.....	41

## Вступ

Цифрове моделювання це дослідження об'єктів з використанням математичних моделей на комп'ютері. Під цифровою моделлю будь-якого географічного об'єкта можна зрозуміти деяке уявлення вихідних даних і спосіб опису їх структури, що дозволяє обчислити (відновити) об'єкт шляхом інтерполяції, наближення або екстраполяції. Тобто іншими словами цифрова модель рельєфу (ЦМР) це цифрове подання рельєфу у вигляді тривимірного покриття. Слід розрізняти цифрову модель рельєфу (ЦМР) та цифрову моделі місцевості (ЦММ), різниця в тому що у ЦММ відображається не тільки форми рельєфу, а й інші об'єкти такі як споруди.

У ГІС використовуються різні методи, алгоритми для побудови цифрових моделей рельєфу. Найчастіше джерелом інформації про земну поверхню сьогодні є карти. До картографічних джерел належать топографічні карти і плани, які також можуть використовуватися для створення цифрових моделей рельєфу та місцевості. Найбільш розповсюджена технологія створення цифрових моделей рельєфу базується на оцифруванні горизонталей як основи для моделей рельєфу, а також висотних позначок і інших елементів для більш достовірного моделювання рельєфу.

Процес векторизації карт є досить довготривалим та складним, але при цьому це є одним із актуальних способів на сьогоднішній день. Тому, з метою автоматизації процесу розпізнавання горизонталей на топографічних картах та планах і побудови на їх основі ЦМР в межах цього диплому розглядатиметься можливість використання мови Python.

Функція Python розпізнає горизонталі на карті, створюється зображення лише з горизонталями. Потім для кожної горизонталі вказується значення, що відповідає висоті на карті.

Узагальнюючи, вхідними даними цієї функції є частина карти 1: 100 000 (або інших масштабів на яких чітко видно горизонталі), а вихідними продуктом функції буде цифрова модель рельєфу.

Актуальність даної роботи зумовлена необхідністю автоматизації процесів побудови ЦМР для більш швидкого та повного аналізу території.

Мета - створення функції на мові Python для автоматизації оцифрування карт та створення ЦМР.

Завданнями цієї роботи є:

1. Огляд та характеристика видів, способів побудови, інструментів для створення ЦМР.

2. Огляд синтаксису мови Python та бібліотек мови Python для обробки зображень.

3. Побудова технологічної схеми автоматизації процесу оцифрування горизонталей для побудови цифрових моделей рельєфу з використанням мови Python.

4. Дослідна реалізація функції Python для автоматизації процесу оцифрування горизонталей для побудови цифрових моделей рельєфу та її апробації

**РОЗДІЛ 1. ХАРАКТЕРИСТИКА ПРЕДМЕТНОЇ СФЕРИ  
ЗАСТОСУВАННЯ ЦМР ПРИ АКТУЛІЗАЦІЇ КАРТОГРАФІЧНИХ  
МАТЕРІАЛІВ РЕЛЬЄФУ**

# **1. Характеристика предметної сфери застосування ЦМР при актуалізації картографічних матеріалів рельєфу**

## **1.1. Поняття цифрової моделі рельєфу та її застосування**

Цифрова модель рельєфу (анг. Digital Terrain Model)- це математичне представлення поверхні землі, найчастіше у вигляді звичайної сітки, в якій кожному пікселю призначається унікальне значення висоти.

Можна сказати що головною метою створення ЦМР є при мінімальній кількості точок моделі забезпечення максимально можливої точності відображення рельєфу місцевості.

Традиційно ЦМР використовувалися для побудови моделей висот земної поверхні. Вони тривимірні за своєю природою і тому підходять для кількох важливих застосувань. Ідея полягає в створенні цінних продуктів на ЦМР, розширюючи їх корисність, але також вирішуючи реальні проблеми, які включають іншу просторову інформацію.

ЦМР може бути більш тісно пов'язаний з фізичними процесами Землі, такими як інфільтрація води, наземний стік, повені та поширення рослинності. Це пояснюється тим, що на моделі, які керують цими процесами, впливає тривимірний простір – висота. Вода швидше стікає вниз на крутішій землі тощо. Сьогодні повені є великою проблемою, тому ЦМР користуються великим попитом із міркувань страхування, наприклад. Також ЦМР грають роль у військовому застосуванні, створені на основі супутникових знімків та знімків із БПЛА, такі моделі дають більш повне розуміння місцевості. Єдиним недоліком є довге створення ЦМР, як для військових потреб.

ЦМР також може бути корисним для техногенних процесів і подій, включаючи точніше розміщення телефонних веж, пунктів спостереження за лісом та інші визначення видимості.

Наприклад побудова цифрової моделі рельєфу для створення плану дна ріки. Вихідні дані у цьому випадку складаються з фотограмметричних вимірювань, наземного (геодезичного) знімання.

Для дослідження динаміки процесів, які впливають на гідрологічний характер, створення ЦМР дна річки виконують через рівні проміжки часу, тому найкращою технологією автоматизації отримання інформації є аерофотозйомка. За допомогою ЦМР визначають довжини водних потоків, нахили схилів, деформації русла річки тощо.

Іншою областю застосування ЦМР є глобальна класифікація ґрунтового покриву. Точне картографування та класифікація земної поверхні в глобальному масштабі є найважливішою передумовою для широкомасштабного моделювання геологічних процесів. Численні дослідження показали, що радіолокаційні зображення придатні для запису та класифікації природної рослинності та сільськогосподарських районів. У дистанційному зондуванні матриці висоти використовуються для корекції зображень або для отримання тематичної інформації про геометрію датчика та місцеву топографію. Тому для спільного застосування різних сенсорних систем ГІС використання цифрових моделей місцевості є необхідною умовою для кодування супутникових зображень та корекції ефектів місцевості.

Окремо хочеться наголосити на дослідженні інших планет за допомогою цифрових моделей рельєфу. Інструментом все більшої цінності в планетарній науці стало використання орбітальної альтиметрії, яка використовується для створення цифрової карти висот планет. Основним інструментом для цього є лазерна альтиметрія, але також використовується радіолокаційна висотність. Планетарні цифрові карти висот, створені за допомогою лазерної альтиметрії, включають відображення Марса лазерного альтиметра Mars Orbiter (MOLA), місячного орбітального лазерного альтиметра (LOLA) і місячного висотоміра (LALT) Місяця. Висотомір (MLA) відображення Меркурія. У планетарному картографуванні кожне планетарне тіло має унікальну опорну поверхню.

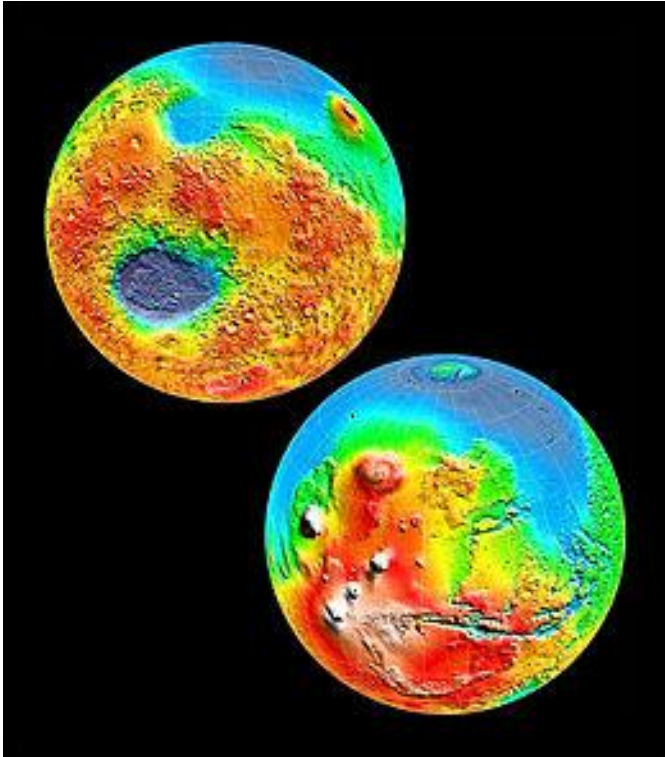


Рис. 1.1. Цифрова модель Марсу

## 1.2. Способи побудови ЦМР

Сучасні геодезія, картографія, фотограмметрія, телеметрія використовуються для збору інформації про об'єкти навколишнього світу, що дозволяє швидко і точно отримувати дискретні дані про поверхні (процеси, явища) об'єктів. Такі методи та підходи постійно вдосконалюються. Це стосується їх швидкості, ефективності, результативності, інформативності та точності.

Хоча рельєф як змодельований об'єкт може здатися простим (на перший погляд, він математично добре описується як поверхня або поле), практика надає безліч методів і прийомів для створення ЦМР. Багато типів вихідних даних, які використовуються для створення ЦМР, виникають внаслідок різних способів отримання та організації вихідних даних та їх похідних. Сюди входять геодезичні (польові) зйомки та топографічні зйомки регіону, стереофотограмметрична обробка оптоелектронних теодолітів,

аерофотознімки, альтиметрія (топографія суші), зйомки підводного рельєфу в морських водах та ехозондування, а також різноманітна вторинна інформація про топографію, наприклад топографічні карти та плани поверхів.

Просторова організація вихідних даних місцевості як набору опорних точок (точок з відомими висотами) для моделі також відрізняється з огляду на точність вимірювань висоти. Вибір вихідного матеріалу залежить від мети дослідження, вимог до точності результатів, галузі дослідження.

Цифрова модель рельєфу відображає рельєф у вигляді масиву точок з відомими їх просторовими координатами  $X, Y, Z$ . Координати одержують за даними зйомок (тахеометричної, мензульної, навігаційної і ін.) опорних точок, які вибирають в місцях перегину рельєфу (вершинах, сідловинах, хребтах, лощинах) з щільністю на 1 га: 50 – для рівнинної, 100 – горбистої, 200-300 – гірської місцевостей.

Є декілька основних методів побудови ЦМР, а саме:

- У фотограмметричному методі створюють просторові моделі об'єкта, побудовані із стереопар зображень;
- За допомогою лазерного сканування, лазерний сканер може бути розташований на повітряному носії чи на землі;
- За допомогою геодезичних (польових) вишукувань, виконуються за допомогою тахеометра;
- Картометричним методом, виконується цифрування карт або планів, вручну або автоматично.

## Переваги та недоліки методів побудови ЦМР

Методи	Переваги	Недоліки
Фотограмметричний метод	Швидкість обробки великих площ при достатньо потужному комп'ютері	Залежність точності побудови ЦМР від можливостей комп'ютера
Метод Лазерного сканування	Автоматизований процес зйомки місцевості, швидкість зйомки	Залежність можливостей сканера від його моделей
Метод геодезичних вишукувань	Достатньо велика точність вимірів	Повільний процес зйомки місцевості (у випадку тахеометричного знімання)
Картометричний метод	Відсутність польових робіт	Дрібномасштабні карти практично не придатні для створення ЦМР

Якщо цифрова модель рельєфу або геологічної будови місцевості формується на основі використання матеріалів зйомки землі, рекомендується використовувати метод топографічної зйомки, що надає інформацію про місцевість в електронному вигляді. Це дозволяє автоматизувати процес підготовки топографічних карт і ЦМР. Вимірювання велосиметрії особливо ефективні при використанні електронних тахеометрів або комп'ютеризованих вимірювальних станцій, запису вихідної інформації в режимі реального часу безпосередньо на магнітних носіях або пізніше підключення її до пам'яті

базового комп'ютера. Вимірювання оптичного теодоліту. Рекомендується обробляти результати вимірювань фотоелектричного теодоліту на стереофотограмметричному пристрої загального призначення, автоматично реєструвати вимірювані координати топографічних точок або використовувати автоматизовану систему (типу «Фотомод») для систематичної комп'ютерної обробки фотографій.

Найчастіше використовуваним джерелом про землю сьогодні є карти. До картографічних джерел належать топографічні карти і плани, що також можуть використовуватися для створення цифрових моделей рельєфу та місцевості. Найрозповсюдженіша технологія генерації ЦМР базується на цифруванні горизонталей як фундаменту для моделей рельєфу, а також висотних позначок і інших елементів для більш точного моделювання рельєфу. У загальногеографічних творах рельєф зображується:

- системою ізоліній (горизонталей, ізогіпс);
- позначками висот;
- сукупністю точкових позамасштабних, лінійних і площинних знаків

Але топографічна карта як основа для ЦМР також має і недоліки, а саме : якщо буде не достатньо інформації про рельєф, ЦМР може не правильно відобразити характер місцевості. А також для кожного із масштабів є своя точні нанесення горизонталей на карту(переріз рельєфу), тож це необхідно враховувати при створенні ЦМР.

Таблиця 1.2

## Висота перерізу рельєфу для карт різних масштабів масштабів

Характеристика району	Основна висота перерізу рельєфу (в метрах) для карт масштабів				
	1:10000	1:25000	1:50000	1:100000	1:200000
Плоскорівнинні райони із схилами місцевості до 2°	1	2,5	10	20	20
Те саме у залісених районах	2	5	10	20	20
Рівнинні, пересічені та горбисті райони з переважаючими схилами місцевості до 6° та райони піщаних пустель	(2,0); 2,0	5	10	20	20
Те саме у відкритих районах із схилами місцевості до 4°	(2,0); 2,5	2,5	10	20	20
Низькогірні та середньогірні райони	5	5	10	20	40
Високогірні (або прирівняні до них) території	—	10	20	40	80

## Середня похибка побудови горизонталей на картах різних масштабів

Висота перетину Райони зйомки	Середня похибка зйомки рельєфу			
	горизонталями, м		рельєфу (у частках висоти перетину)	
	1:10 000	1:25 000	1:10 000	1:25 000
Плоско-рівнинні з ухилом місцевості до 1°	1,0	2,5	1/4	1/3
Рівнинні з ухилом місцевості від 1° до 2°	1,0*; 2,0	2,5; 5,0**	1/3	1/3
Рівнинні пересічені і горбисті з ухилом місцевості від 2° до 6°	2,0 (2,5)	2,5***; 5,0	1/3	1/3
Гірські і передгірні	5,0	5,0	1/3	1/3
Високогірні	–	10,0	1/3	1/3

Також ще однією складністю створення ЦМР на основі топографічних творів є те, що на картах, крім так званих суцільних горизонталей, відображаються додаткові та допоміжні горизонталі. Проводяться на половині висоти перетину та мають назву напівгоризонталей і з точки зору метричності відповідають основним.

Як відомо горизонталі мають графічні межі встановлені інструкцією в 0,2 мм і такій же відстані між ними один міліметр карти може містити їх не більше трьох. Дозволяється штучне злиття горизонталей коли величина їх закладання не вкладається у графічні пороги. Наприклад коли кут ухилу місцевості дуже великий. У цьому випадку оцифрування таких горизонталей може дати велику похибку при створенні ЦМР. Для запобігання створень

таких незручностей при оцифруванні карти потрібно щоб, однойменні та різнойменні горизонталі не перетиналися, а також кожна горизонталь повинна замикатися на саму себе або на рамці карти.

Найпоширенішим типом цифрової моделі місцевості, що широко використовується в ГІС, є цифрове відображення поверхні місцевості у вигляді растру. ЦМР у даному випадку побудована з використанням існуючих обмежених наборів точкових даних про топографічні маркери поверхні в сусідніх осередках сітки з використанням просторової інтерполяції.

Побудова ЦМР саме фотограмметричним методом виконується за допомогою БпЛА (безпілотний літальний апарат) та спеціального програмного забезпечення. С початку на території, для якої потрібно створити цифрову модель рельєфу, створюються опорні точки з відомим для них координатами, вони потрібні для подальшого масштабування та геопривязки ЦМР. Далі для БпЛА задають параметри маршруту, а також параметри повздовжнього та поперечного перекриття для знімків (не менше 56% та 20 % має буде перекриття фотознімків відповідно). Наступним кроком, після завершення фотознімання, є завантаження всіх матеріалів до програми (наприклад Agisoft Metashape) та відсортування неякісних, а також вирівнювання, фотознімкі. Далі відбувається вирівнювання та оптимізація камери, після цього знімки маркуються опорними точка та задаються координати цих опорних точок. На основі вирівняних знімків будується щільна хмара точок, далі створюється модель місцевості яку потім потрібно текстуризувати. Наступним кроком є побудова ортофотоплану, який вже є основою для створення цифрової моделі місцевості.

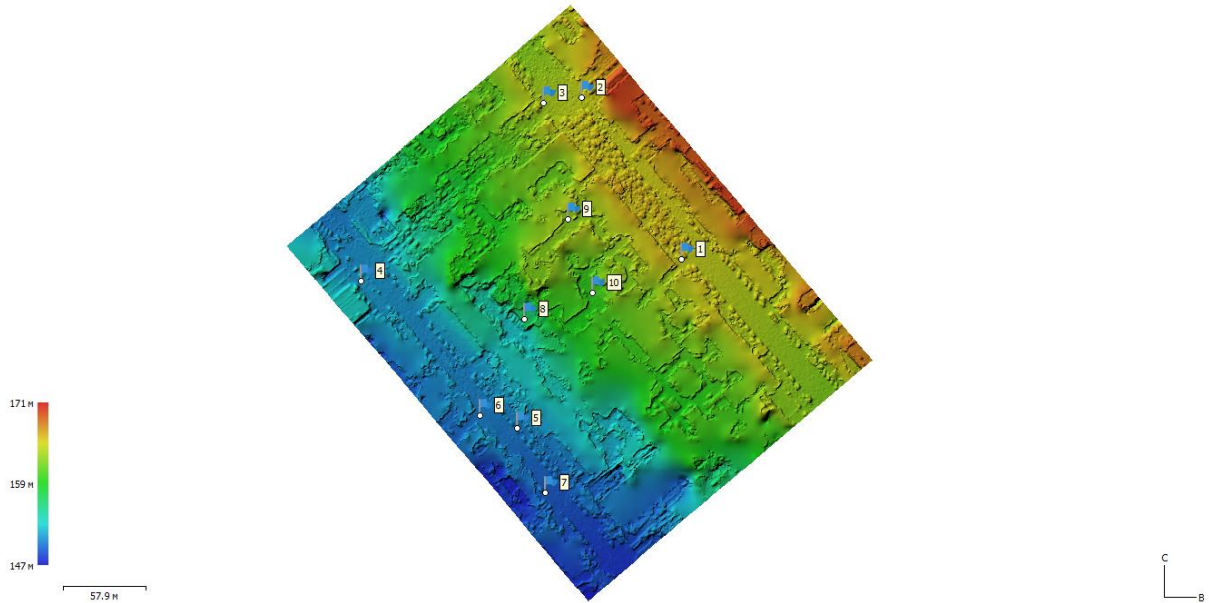


Рис. 1.2. Цифрова модель місцевості створена на основі знімків із БПЛА

### 1.3. Класифікація та порівняння алгоритмів для побудови ЦМР

ЦМР можна розділити за способами обробки та розміщення вихідної інформації:

- За регулярним розташуванням точок на прямокутних, трикутниках чи гексагональних сітках (GRID);
- За нерегулярним поданням точок по структурних лініях, профілях, центрах майданчиків, локальних точках або випадкових сітках (TIN);
- За ізолінійним поданням точок, що розташовані рівномірно на ізолініях, або з урахуванням кривизни горизонталей (TGRID);

Модель GRID є найпопулярнішим способом опису поверхонь. Подання засноване на звичайній сітці комірок з вузлами для заданих значень полів. По суті, відображення поверхні GRID є растровим методом, точність якого залежить від розміру клітинок растра. Зменшивши розмір, можна точніше описати поверхню. Однак, коли відстань сітки зменшується в 2 рази, кількість

вузлів збільшується на 4, що збільшує кількість даних, необхідних для їх зберігання.

До переваг цієї моделі можна віднести швидку комп'ютерну обробку, відносно просте зберігання та маніпулювання даними, легку інтеграцію з растровими моделями, а отриманий рельєф рівний і більш природний на вигляд.

До недоліків представлення GRID можна віднести неможливість використання сіток різного розміру, що дозволяють створювати рельєфні конструкції різної складності.



Рис. 1.3. Приклад GRID моделі

Суть моделі TIN полягає в її назві (Triangulation - означає метод побудови набору трикутників; Irregular - змінна щільність точок може використовуватися для імітації площі поверхні, де рельєф дуже сильно змінюється; Мережа відображає топологію місцевість у цій моделі). У просторовому визначенні це мережа трикутників - з висотами у вузлі її, що дозволяє представити поверхню, що моделюється, як багатогранник.

У порівнянні з GRID, TIN чіткіше висвітлюють області з високими градієнтами поля, крутизною поверхні. Цього можна досягти, використовуючи неправильну мережу - в областях з високим градієнтом кількість трикутників збільшується, а їх розмір зменшується. TIN точніше моделюють природні деталі поверхні (берегові лінії, хребти, струмки, гірські вершини). На жаль, від'ємні, нависають стіни, печери не можна відобразити за допомогою моделі TIN.



Рис. 1.4. Приклад TIN моделі

TGRID - Модель, що поєднує елементи моделей TIN і GRID.

Ізолінії це форма подання поверхонь. Ізолінії — це певна горизонтальна лінія, що представляє горизонтальний переріз поверхні.

Перетини можуть виконувати рівні кроки або довільні кроки. Висота (значення) контуру призначається як властивість кожній такій лінії.

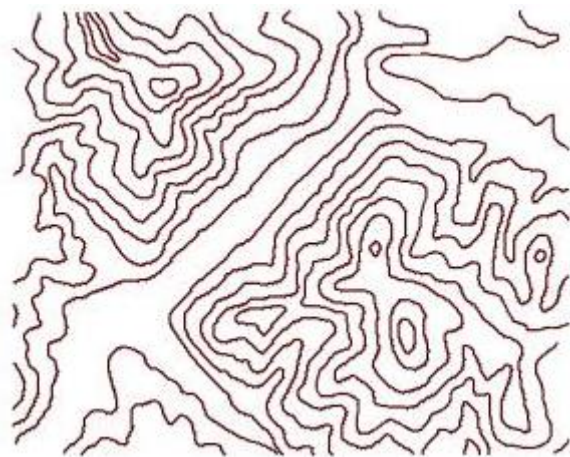


Рис. 1.5. Приклад TGRID моделі

## Висновки до розділу 1

У першому розділі проведено огляд та узагальнення поняття та використання ЦМР, проаналізовано та викладено вимоги до побудови ЦМР; виконано порівняння методів побудови ЦМР; проведено класифікація та порівняння алгоритмів для побудови ЦМР.

**РОЗДІЛ 2. РОЗРОБЛЕННЯ АЛГОРИТМУ ДЛЯ АВТОМАТИЗАЦІЇ  
ПРОЦЕСУ ОЦИФРУВАННЯ ГОРИЗОНТАЛЕЙ ДЛЯ ПОБУДОВИ  
ЦИФРОВИХ МОДЕЛЕЙ РЕЛЬЄФУ З ВИКОРИСТАННЯМ МОВИ  
Python**

## **2. Розроблення алгоритму для автоматизації процесу оцифрування горизонталей для побудови цифрових моделей рельєфу з використанням мови Python**

### **2.1. Загальний огляд мови Python та бібліотек для обробки зображень**

Python — це високорівнева, інтерпретована мова програмування загального призначення. Особливість дизайну є читабельність коду. Python динамічно типується. Він підтримує декілька парадигм програмування, включаючи структуроване (зокрема процедурне), об'єктно-орієнтоване та функціональне програмування. Його часто описують як мову повну стандартних бібліотек.

У цій мові програмування використовується динамічний тип типізації та комбінацію підрахунку посилань, що визначає цикл, для управління пам'яттю. Він використовує динамічне визначення імен, яке зв'язує імена методів та змінних під час виконання програми.

Отже до основних плюсів Python можна віднести:

- що це інтерпретована мова програмування, тобто їй не потрібен етап компіляції, а також програма на мові Python запускається прямо з вікна програми;
- це високорівнева мова програмування;
- Python не залежить від платформи на якій запускається, програми на Python можна запускати на різних операційних системах (Linux, Windows, OS X);
- Python це open source проект (відкрите джерело), будь хто може змінювати програму під свої потреби;
- це вбудовувана скриптова мова;
- це динамічна мова, що спрощує написання нескладних програм;
- для Python створена велика кількість бібліотек класів.

Але також Python має декілька недоліків, а саме:

- низька швидкість виконання задач порівняно із мовами програмування С та С++;

- динамічна типізація є мінусом при написанні складних програм.

Python підходить для вирішення широкого спектра завдань.

- Системне програмування. Вбудовані в Python інтерфейси доступу до служб операційних систем роблять його ідеальним інструментом для створення програм, що переносяться, і утиліт системного адміністрування;

- Графічні програми. Простота Python і швидкість розробки роблять його чудовим засобом створення графічного інтерфейсу. До складу Python входить стандартний об'єктно-орієнтований інтерфейс до GUI API;

- Веб-програми. За допомогою додаткових фреймворків мовою Python (Django, Flask, Pyramid) можна створювати повнофункціональні сайти;

- Веб-сценарії. Python поставляється разом із стандартними інтернет-модулями, які дозволяють програмам виконувати різноманітні мережеві операції як у режимі клієнта, так і в режимі сервера;

- Інтеграція компонентів. Можливість Python розширюватися та вбудовуватися в системи мовою С++ робить його зручним для опису поведінки інших систем та компонентів;

- Програми баз даних. У Python є інтерфейси доступу до всіх основних реляційних баз даних: Sybase, Oracle, Informix, ODBC, MySQL, PostgreSQL, SQLite та багатьох інших. З їх допомогою можна створювати програми баз даних.

Обробка зображень - це процес аналізу та роботи з цифровим зображенням, спрямований на поліпшення якості картинки або вилучення інформації для подальшого використання.

Загальні завдання зводяться до відображення зображення та виконання основних операцій (кадрування, відображення, обертання, сегментація, класифікація, вилучення ознак, відновлення та розпізнавання). Python є чудовим засобом для вирішення подібних завдань. Завдяки доступності та зростаючій популярності Python як мови науково-технічного програмування,

всередині екосистеми з'явилося безліч першокласних інструментів для обробки зображень.

NumPy — це фундаментальний пакет для наукових обчислень на Python. Це бібліотека Python, яка надає багатовимірний об'єкт масиву, різноманітні похідні об'єкти (наприклад, замасковані масиви та матриці), а також набір підпрограм для швидких операцій над масивами, включаючи математичні, логічні, маніпуляції з формою, сортування, вибір, введення/виводу. , дискретні перетворення Фур'є, базова лінійна алгебра, основні статистичні операції, випадкове моделювання та багато іншого.

В основі пакету NumPy лежить об'єкт ndarray. Це інкапсулює n-вимірні масиви однорідних типів даних, при цьому багато операцій виконуються в скомпільованому коді для підвищення продуктивності. Існує кілька важливих відмінностей між масивами NumPy і стандартними послідовностями Python:

- Масиви NumPy мають фіксований розмір під час створення, на відміну від списків Python (які можуть динамічно зростати). Зміна розміру ndarray створить новий масив і видалить оригінал.

- Усі елементи в масиві NumPy повинні мати один тип даних і, таким чином, мати однаковий розмір пам'яті. Виняток: можна мати масиви об'єктів (Python, включаючи NumPy), що дозволяє використовувати масиви елементів різного розміру.

- Масиви NumPy полегшують розширені математичні та інші типи операцій над великою кількістю даних. Як правило, такі операції виконуються ефективніше і з меншою кількістю коду, ніж це можливо з використанням вбудованих послідовностей Python.

- Все більша кількість науково-математичних пакетів на основі Python використовує масиви NumPy; хоча вони зазвичай підтримують введення послідовності Python, вони перетворюють такі вхідні дані в масиви NumPy перед обробкою, і вони часто виводять масиви NumPy. Іншими словами, для ефективного використання значної частини (можливо, навіть більшості) сучасного науково-математичного програмного забезпечення на

основі Python, недостатньо просто знати, як використовувати вбудовані в Python типи послідовностей – також потрібно знати, як використовувати масиви NumPy.

SciPy — це набір математичних алгоритмів і зручних функцій, побудованих на розширенні NumPy для Python. Він додає значної потужності інтерактивному сеансу Python, надаючи користувачеві команди та класи високого рівня для маніпуляції та візуалізації даних. З SciPy інтерактивний сеанс Python стає середовищем обробки даних і системного прототипування, що конкурує з системами, такими як MATLAB, IDL, Octave, R-Lab і SciLab.

Додаткова перевага створення SciPy на Python полягає в тому, що це також робить потужну мову програмування доступною для використання при розробці складних програм і спеціалізованих програм. Наукові програми, що використовують SciPy, отримують переваги від розробки додаткових модулів у численних нішах програмного середовища розробниками по всьому світу. Програмісту на Python стало доступним все, від паралельного програмування до веб-програм, підпрограм і класів баз даних.

## **2.2. Поняття колірних моделей та їх застосування при обробленні зображень та автоматичній обробці горизонталей**

Для автоматичного розпізнавання горизонталей необхідно використовувати ознаки, за якими горизонталі відрізняються від інших позначень на карті. Найзручнішою ознакою горизонталі є її колір, який може бути вимірний за допомогою певної колірної моделі.

Призначення колірної моделі (так званої також колірним простором або системою кольорів) полягає в тому, щоб зробити можливим опис кольорів деяким стандартним, загальноприйнятим чином. По суті, колірна модель визначає деяку систему координат і підпростір всередині цієї системи, в якому кожен колір представляється єдиною точкою. Більшість сучасних колірних

моделей орієнтовані або на пристрої відтворення кольору (наприклад кольорові монітори, принтери тощо), або на певні прикладні завдання (такі як створення кольорової графіки в анімації), коли робота з кольором є безпосередньою метою.

RGB (червоний, зелений, синій) — колірна модель, що описує спосіб синтезу кольору, за якою червоне, зелене та синє світло накладаються разом, змішуючись у різноманітні кольори. У моделі RGB кожен колір представляється червоним, зеленим і синім первинними основними кольорами (компонентами). В основі моделі лежить Декартова система координат.

HSV — колірна модель, побудована на трьох характеристиках кольору: колірному тоні (Hue), насиченості (Saturation) і яскравості (Value). Значення цих трьох параметрів варіюється в межах 0-100 або 0-1. Перевага HSV полягає в тому, що кожен з його атрибутів безпосередньо відповідає основним поняттям кольору, що робить його концептуально простим. Отже, для того, щоб краще знайти та відобразити інформацію про зображені на топографічній карті горизонталі, ми будемо використовувати колірну модель HSV. Оскільки зображення зберігаються у колірній моделі RGB, то потрібно перейти до моделі HSV.

### 2.3. Технологічна схема автоматизації процесу оцифрування горизонталей для побудови цифрових моделей рельєфу з використанням мови Python

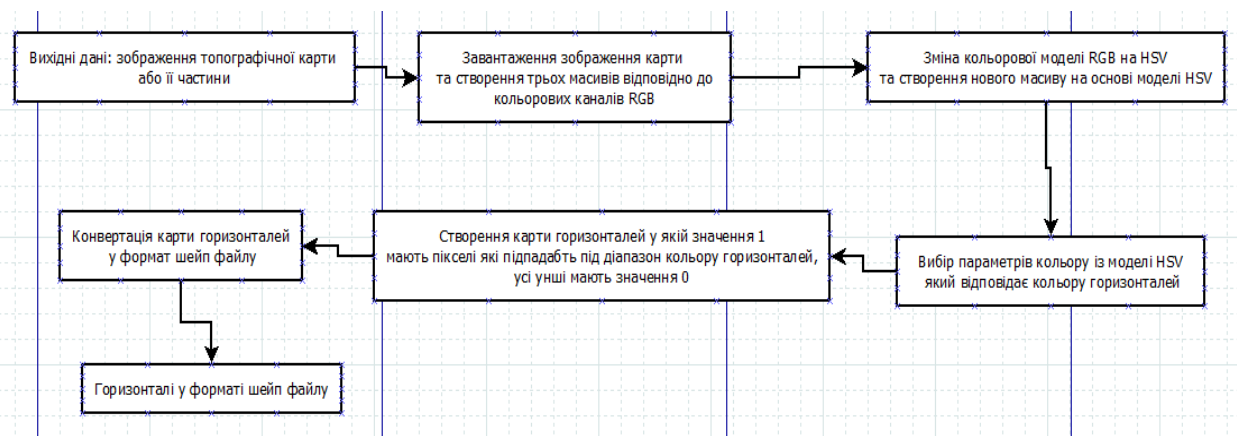


Рис. 2.3 Технологічна схема автоматизації процесу оцифрування горизонталей.

Процес автоматичного оцифрування горизонталей за допомогою функції Python подано у вигляді основних етапів як блок схему на рис. 2.3, зокрема основні етапи:

1. Першим основним етапом є прочитання карти та створення трьох масивів даних, кожен з яких відповідає своєму каналу у моделі RGB.
2. Перетворення трьох масивів даних у один кольорової моделі HSV
3. Підбір кольору горизонталей за трьома параметрами HSV та призначення цьому відтінку значення 1, усім іншим кольорам призначається значення 0
4. Створення карти горизонталей за вибіркою значення 1
5. Конвертація карти горизонталей у шейп формат для подальшого опрацювання та створення цифрової моделі рельєфу

## **Висновки до розділу 2**

У другому розділі розглянуто мову програмування Python, її переваги та недоліки. А також декілька її бібліотек які доповнюють можливості оброблення графічних даних. Розглянуто кольорові моделі RGB та HSV, а також їхні відмінності. Складено технологічну схему автоматизації оцифрування горизонталей, що є фундаментом для створення цифрової моделі рельєфу.

**РОЗДІЛ 3. ДОСЛІДНА РЕАЛІЗАЦІЯ ФУНКЦІЇ Python ДЛЯ  
АВТОМАТИЗАЦІЇ ПРОЦЕСУ ОЦИФРУВАННЯ ГОРИЗОНТАЛЕЙ  
ДЛЯ ПОБУДОВИ ЦИФРОВИХ МОДЕЛЕЙ РЕЛЬЄФУ**

### 3. Дослідна реалізація функції Python для автоматизації процесу оцифрування горизонталей для побудови цифрових моделей рельєфу

#### 3.1. Підготовка вихідних даних дослідної ділянки

Як вихідні дані для апробації розробленої функції використано частину топографічної карти Житомирської області, масштабу 1:100000. А саме М – 35 – 059, карта генерального штабу міста «Радомишль», станом місцевості на 1987 рік.

Карту отримано з мережі інтернет, формат зображення \*jpg.



Рис. 3.1 Карта генерального штабу міста «Радомишль», 1:100000 масштабу



Рис. 3.2 Досліджувана ділянка карти

### 3.2. Функція Python для автоматизації процесу оцифрування горизонталей

Використовуючи технологічну схему Рис. 2.3 реалізовано функцію яка забезпечує оцифрування горизонталей. Текст функції із роз'ясненнями до її основних блоків наведено у таблиці нижче.

Для початку потрібно прочитати фрагмент карти і створити 3 масиви даних за допомогою інструменту NumPy. Кожний масив сприймає значення яскравості у каналах R, G, B. Наступним кроком є перехід до кольорової моделі HSV, за допомогою функції `colors.rgb_to_hsv` з бібліотеки `scipy`. При цьому потрібно врахувати, що функція приймає як вхідний аргумент один тривимірний масив, що містить яскравості всіх трьох каналів. Далі створюється новий масив та заповнюється значеннями HSV.

Потім потрібно провести класифікацію по значенням HSV. На цьому етапі потрібно вручну встановити діапазон для кожного значення HSV, які відповідають кольору горизонталей.

На основі класифікації HSV були внесені основні параметри кольору горизонталей на карті. Далі створюється карта горизонталей, у якій значення 1 мають пікселі які відповідають діапазону за всіма параметрами HSV, решті

надається значення 0. У кінці створюється точкові об'єкти на основі отриманих даних з подальшим їх конвертуванням у шейп-файли за допомогою бібліотек `shapely.geometry` та `geopandas`.

Таблиця 1.4

### Розбір функції

Функція	Пояснення
<pre>import numpy as np import scipy as sc from PIL import Image</pre>	Завантаження всіх необхідних бібліотек до функції
<pre>map = np.asarray(Image.open(name).convert('RGB'))</pre>	Завантаження карти до функції
<pre>map_r = np.zeros_like(map) map_r[:, :, 0] = map[:, :, 0] #зелений канал map_g = np.zeros_like(map) map_g[:, :, 1] = map[:, :, 1] #червоний канал map_b = np.zeros_like(map) map_b[:, :, 2] = map[:, :, 2] #синій канал</pre>	Створення трьох масивів кожен з яких відповідає своєму кольоровому каналу RGB
<pre>map_hsv = mc.rgb_to_hsv(map)</pre>	Перехід до кольорової моделі HSV
<pre>if map == np.where(map == [146, 134, 82]):     horz1 = [] else:     horz0 = []</pre>	Визначення та відбір кольору горизонталей
<pre>horz_map = np.asarray(Image.new(horz1))</pre>	Створення карти горизонталей
<pre>shp_map = shp.convert(horz_map)</pre>	Конвертація у шейп файл
<pre>plt.imshow(horz_map) plt.show() map_h = map.shape</pre>	Вивід усіх даних та розмірів ділянки карти у пікселях

```
print(map_h)
```

### 3.3. Результати застосування функції Python для автоматизації процесу оцифрування горизонталей

При запуску функція дає запит на шлях зберігання файлу який потрібно завантажити.

```
6 print('Введіть шлях зберігання карти')
7 name = input()
8 map = np.asarray(Image.open(name).convert('RGB'))
9 plt.imshow(map)
10 plt.show()
```

Рис. 3.3.1 Код запиту місцезнаходження зображення та його виводу на екран

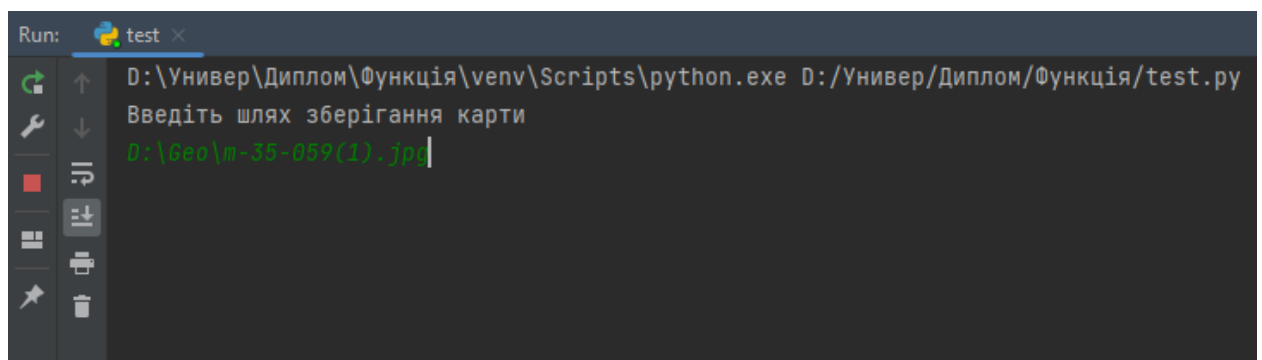


Рис. 3.3.2 Введений шлях зберігання карти

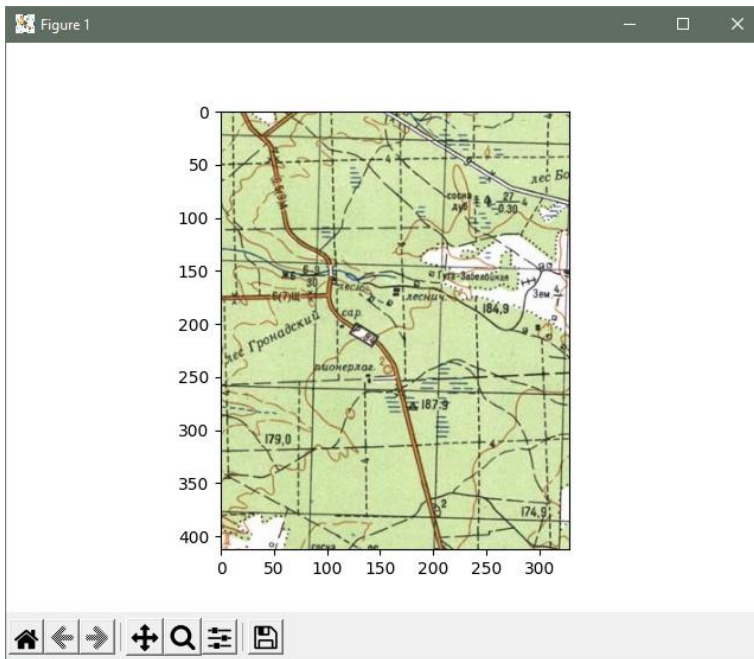


Рис. 3.3.3 Вхідні дані, виведені на екран

```

3     import numpy as np
4     import matplotlib.pyplot as plt
5     import matplotlib.colors as mc
6     from PIL import Image
7
8
9     print('Введіть шлях зберігання карти')
10    name = input()
11
12    map = np.asarray(Image.open(name).convert('RGB'))
13    map_r = np.zeros_like(map)
14    map_r[:, :, 0] = map[:, :, 0]
15    map_g = np.zeros_like(map)
16    map_g[:, :, 1] = map[:, :, 1]
17    map_b = np.zeros_like(map)
18    map_b[:, :, 2] = map[:, :, 2]
19    map_hsv = mc.rgb_to_hsv(map_r[:, :, 0], map_g[:, :, 1], map_b[:, :, 2])
20    if map == np.where(map == [146, 134, 82]):
21        horz1 = []
22    else:
23        horz0 = []
24    horz_map = np.asarray(Image.new(horz1))
25    shp_map = shp.convert(horz_map)
26
27    plt.imshow(horz_map)
28    plt.show()
29    map_h = map.shape
30    print(map_h)

```

Рис. 3.3.4 Код всієї функції



Рис. 3.3.5 Створена карта горизонталей



Рис. 3.3.6 Побудована цифрова модель рельєфу способом унікальних значень, градієнтом Spectral

### **Висновки до 3 розділу**

У третьому розділі проілюстровано вхідні дані до функції, етапи її роботи та результат виконання. Детально розібрані відмінності RGB та HSV моделі відображення графічних матеріалів та описані етапи роботи функції.

## **Висновки**

В межах диплому реалізовано та досягнуто наступні завдання:

- 1) Виконано аналіз предметної сфери, в межах якого проведено огляд та узагальнення поняття та використання ЦМР, проаналізовано та викладено вимоги до побудови ЦМР; виконано порівняння методів побудови ЦМР; проведено класифікація та порівняння алгоритмів для побудови ЦМР.
- 2) Розглянуто особливості мови програмування Python, її переваги та недоліки, бібліотеки, які доповнюють можливості оброблення графічних даних. Розглянуто кольорові моделі RGB та HSV, а також їхні відмінності.
- 3) Побудовано технологічну схему автоматизації оцифрування горизонталей, що є фундаментом для створення цифрової моделі рельєфу, та визначає основні етапи реалізації процесу автоматизації розпізнавання горизонталей на топографічних картах з використанням мови python
- 4) Реалізовано та апробовано функцію на мові Python для автоматизації процесу оцифрування горизонталей за топографічними картами на прикладі карти генерального штабу міста «Радомишль», 1:100000 масштабу.

## Список використаних джерел

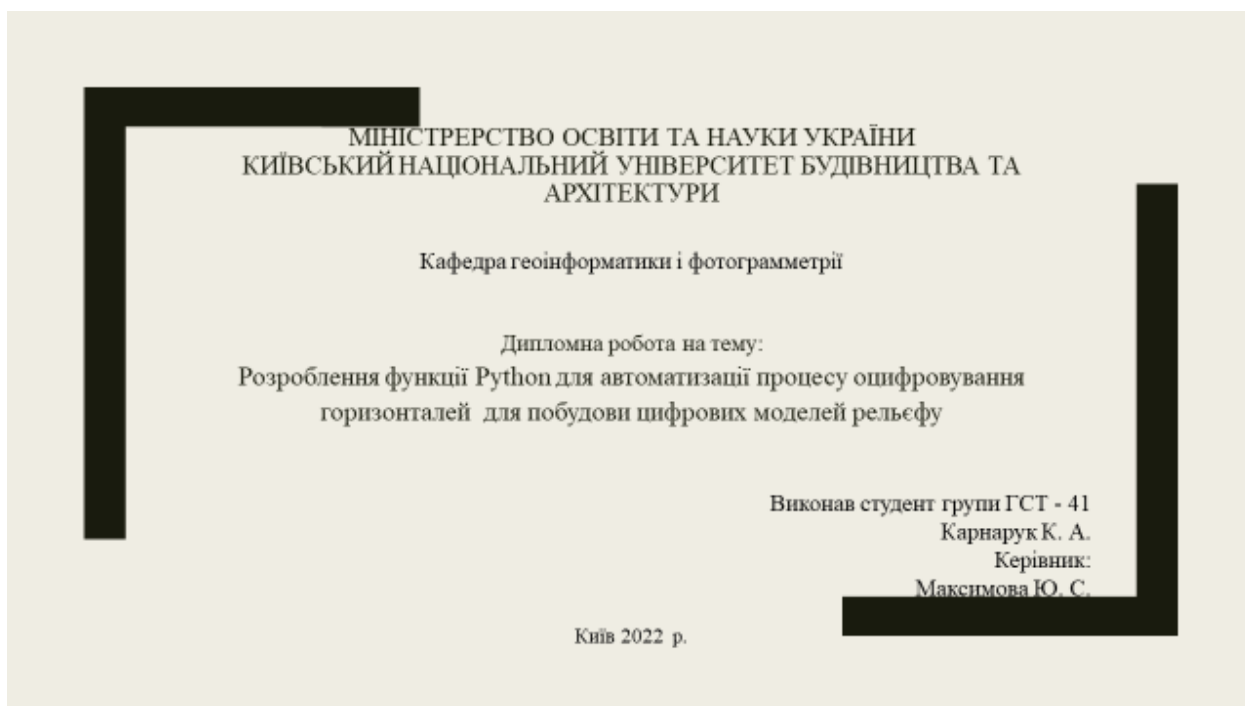
1. N.V. Rul, Yu. I. Velikodsky, V.I. Zatcerkovnyi The algorithm of automatic vectorization of contours for constructing digital elevation models // <https://eage.org>
2. [https://en.wikipedia.org/wiki/Digital\\_elevation\\_model](https://en.wikipedia.org/wiki/Digital_elevation_model)
3. В. І. Зацерковний, В. Г. Бурачек, О. О. Железняк, А. О. Терещенко Геоінформаційні системи і бази даних Книга 2
4. Х. Бурштинська, І. Василюха, П. Коваль Технологія побудови цифрової моделі рельєфу для створення плану дна ріки // Національний університет «Львівська політехніка»
5. З. Кузик Цифрові моделі рельєфу на території курорту Східниця // Національний університет «Львівська політехніка»
6. Денис Горковчук Опрацювання даних фотограмметричного знімання з БПЛА // КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ БУДІВНИЦТВА І АРХІТЕКТУРИ
7. <http://www.geoguide.com.ua/basisdoc/basisdoc.php?part=tgo&art=3203>
8. А. В. Пукас Опорний конспект лекцій з дисципліни Програмне забезпечення геоінформаційних систем // ТЕРНОПІЛЬСЬКИЙ НАЦІОНАЛЬНИЙ ЕКОНОМІЧНИЙ УНІВЕРСИТЕТ
9. Лазоренко-Гевель Надія Юріївна, Денисюк Богдан Іванович АНАЛІЗ МЕТОДІВ І МОДЕЛЕЙ ЦИФРОВОГО МОДЕЛЮВАННЯ РЕЛЬЄФУ В ОБ'ЄКТНО-РЕЛЯЦІЙНИХ БАЗАХ ТОПОГРАФІЧНИХ ДАНИХ // Київський національний університет будівництва і архітектури
10. <https://docs.python.org/3/tutorial/index.html>
11. [https://en.wikipedia.org/wiki/Python\\_\(programming\\_language\)](https://en.wikipedia.org/wiki/Python_(programming_language))
12. <https://numpy.org/doc/stable/user/whatisnumpy.html>
13. [https://coderlessons.com/tutorials/python-technologies/uchitsia-stsipi-scipy-kratkoe-rukovodstvo](https://coderlessons.com/tutorials/python-technologies/uchitsia-stsipi/scipy-kratkoe-rukovodstvo)

14. [https://docs.scipy.org/doc/scipy/install\\_upgrade.html](https://docs.scipy.org/doc/scipy/install_upgrade.html)
15. <https://pypi.org/project/scipy/>
16. [https://en.wikipedia.org/wiki/Color\\_model](https://en.wikipedia.org/wiki/Color_model)
17. [https://stud.com.ua/54440/informatika/kolirna\\_model](https://stud.com.ua/54440/informatika/kolirna_model)
18. <http://freemap.com.ua/karty-ukrainy/karty-genshtaba/karta-genshtaba-kvadrat-m-35-59>
19. [https://matplotlib.org/3.5.0/gallery/color/named\\_colors.html#sphx-glr-gallery-color-named-colors-py](https://matplotlib.org/3.5.0/gallery/color/named_colors.html#sphx-glr-gallery-color-named-colors-py)
20. <https://russianblogs.com/article/4594178757/>
21. [https://matplotlib.org/3.5.0/api/\\_as\\_gen/matplotlib.colors.rgb\\_to\\_hsv.html](https://matplotlib.org/3.5.0/api/_as_gen/matplotlib.colors.rgb_to_hsv.html)
22. <https://habr.com/ru/post/451074/>
23. <http://python-3.ru/page/sozдание-izobrazhenija-v-python>
24. <https://codebeautify.org/hsv-to-rgb-converter>
25. <https://shapely.readthedocs.io/en/stable/manual.html>
26. <https://ru.stackoverflow.com/questions/1183672/shape-файлы-Прочтение-с-помощью-geopandas>
27. <https://pythonworld.ru/moduli/modul-os.html>
28. [https://pyprog.pro/mpl/mpl\\_imshow.html](https://pyprog.pro/mpl/mpl_imshow.html)
29. [https://spec-zone.ru/matplotlib~2.2/colors\\_api](https://spec-zone.ru/matplotlib~2.2/colors_api)

## Додатки

### Графічні матеріали

Додаток А



### Мета та завдання диплому

**Мета** - створення функції на мові python для автоматизації оцифрування карт та створення ЦМР.

**Завдання:**

1. Огляд та характеристика видів, способів побудови, інструментів для створення ЦМР.
2. Огляд синтаксису мови python та бібліотек мови python для обробки зображень.
3. Побудова технологічної схеми автоматизації процесу оцифрування горизонталей для побудови цифрових моделей рельєфу з використанням мови Python.
4. Дослідна реалізація функції Python для автоматизації процесу оцифрування горизонталей для побудови цифрових моделей рельєфу та її апробації

## Поняття цифрової моделі рельєфу

**Цифрова модель рельєфу** (анг. Digital Terrain Model) - це математичне представлення поверхні землі, найчастіше у вигляді звичайної сітки, в якій кожному пікселю призначається унікальне значення висоти.

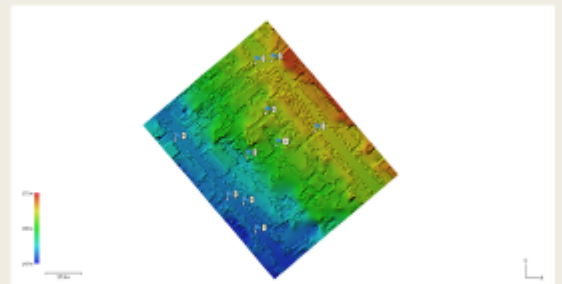
Головною метою створення ЦМР є при мінімальній кількості точок моделі забезпечення максимально можливої точності відображення рельєфу місцевості.

3

## Приклади застосування ЦМР

ЦМР можуть застосовуватися для:

- Аналізу та прогнозування повенів – ЦМР допомагає більш точно прогнозувати повені у долинах річок, та аналізувати як боротися з їх наслідками наслідки.
- Класифікації ґрунтового покриву – за допомогою ЦМР зручно досліджувати чинник які впливають на формування рельєфу.
- Космічних досліджень – завдяки ЦМР науковці аналізують поверхні веземних тіл.
- Візуалізації рельєфу місцевості – за допомогою 3D моделі краще проілюструвати особливості місцевості ніж на топографічній карті.



4

## Шляхи створення ЦМР

Методи	Переваги	Недоліки
Фотограмметричний метод	Швидкість обробки великих площ при достатньо потужному комп'ютері	Залежність точності побудови ЦМР від можливостей комп'ютера
Метод Лазерного сканування	Автоматизований процес зйомки місцевості, швидкість зйомки	Залежність можливостей сканера від його моделей
Метод геодезичних вишукувань	Достатньо велика точність вимірів	Повільний процес зйомки місцевості (у випадку тахеометричного знімання)
Картометричний методот	Відсутність польових робіт	Дрібномасштабні карти практично не придатні для створення ЦМР

5

## Способи цифрового зображення місцевості

- **GRID** моделі – регулярне розташуванням точок на прямокутних, трикутниках чи гексагональних сітках.
- **TIN** моделі – нерегулярним розміщення точок по структурних лініях, профілях, центрах майданчиків, локальних точках або випадкових сітках.
- **TGRID** моделі - ізолінійне поданням точок, що розташовані рівномірно на ізолініях, або з урахуванням кривизни горизонталей.



6

# Мова програмування Python

**Python** — це високорівнева, інтерпретована мова програмування загального призначення, вона динамічно типується та за допомогою неї можна створювати структуровані і функціональні програми.



## Переваги:

- Python не залежить від платформи на якій запускається.
- Python це open source проект.
- Наявна велика кількість бібліотек класів.

## Недоліки:

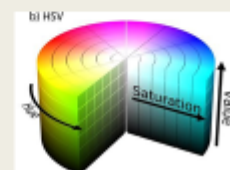
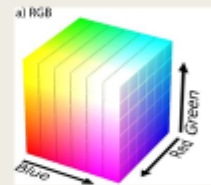
- Низька швидкість виконання задач (порівняно C та C++).
- Динамічна типізація створює тродності при написанні складних програм.

7

# Колірні моделі зображень

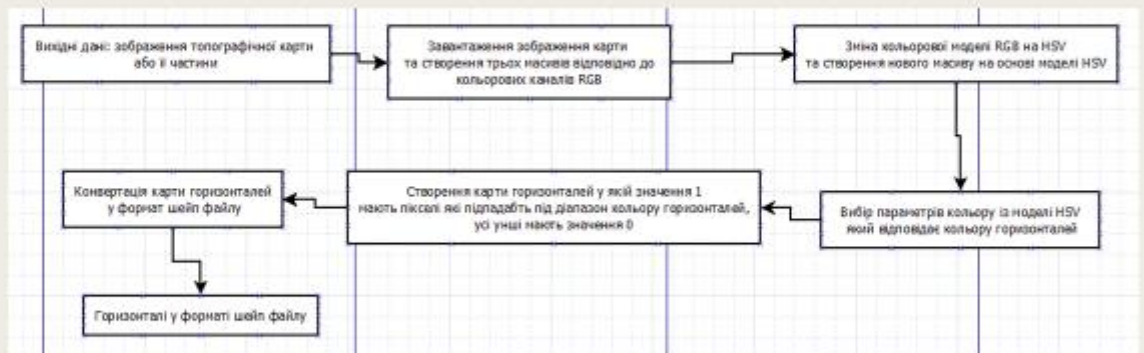
Для *автоматичного розпізнавання* горизонталей *необхідно використовувати ознаки*, за якими горизонталі відрізняються від інших позначень на карті. Найзручнішою ознакою горизонталі є її *колір*, який може бути виміряний за допомогою певної колірної моделі.

- RGB - колірна модель, що описує спосіб синтезу кольору, за якою червоне, зелене та синє світло накладаються разом, змішуючись у різноманітні кольори.
- HSV - колірна модель, побудована на трьох характеристиках кольору: колірному тоні (Hue), насиченості (Saturation) і яскравості (Value).



8

## Блок-схема процесу автоматизації оцифрування горизонталей за допомогою мови програмування Python



9

## Функція на мові програмування Python

```
import numpy as np
import matplotlib.pyplot as plt
import matplotlib.colors as mc
from PIL import Image

print('Введіть назву зображення карти')
name = input()

img = np.asarray(Image.open(name).convert('RGB'))
img_h = np.zeros_like(img)
img_r[::2,0] = img_r[::2,0]
img_g = np.zeros_like(img)
img_g[::2,1] = img_g[::2,1]
img_b = np.zeros_like(img)
img_b[::2,2] = img_b[::2,2]
img_hsv = mc.rgb_to_hsv(img_r[::2,0], img_g[::2,1], img_b[::2,2])
if img == np.where(img == [140, 136, 62]):
    horiz = 1
else:
    horiz = 0
horiz_map = np.asarray(Image.new('L', img.shape))
horiz_map = np.where(horiz == 1, 1, 0)

plt.imshow(horiz_map)
plt.show()
img_h = img.shape
print(img_h)
```

10

## Функція оцифрування горизонталей за допомогою Python, частина 1

- ❑ Для початку потрібно прочитати фрагмент карти і створити 3 масиви даних.
- ❑ Кожний масив сприймає значення яскравості у каналах R, G, B.
- ❑ Наступним кроком є перехід до кольорової моделі HSV.

Функція	Пояснення
<pre>import numpy as np import scipy as sc from PIL import Image  map = np.asarray(Image.open(name).convert('RGB'))</pre>	Завантаження всіх необхідних бібліотек до функції
<pre>map_r = np.zeros_like(map) map_g = np.zeros_like(map) map_b = np.zeros_like(map) map_r[:,0] = map[:,0] #зелений канал map_g[:,1] = map[:,1] #червоний канал map_b[:,2] = map[:,2] #синій канал</pre>	Створення трьох масивів кожен з яких відповідає своєму кольоровому каналу RGB
<pre>map_hsv = mc.rgb_to_hsv(map)</pre>	Перехід до кольорової моделі HSV

11

## Функція оцифрування горизонталей за допомогою Python, частина 2

Потім потрібно провести класифікацію по значенням HSV. На цьому етапі потрібно вручну встановити діапазон для кожного значення HSV, які відповідають кольору горизонталей. Далі створюється карта горизонталей, у якій значення 1 мають пікселі які відповідають діапазону за всіма параметрами HSV, решті надається значення 0.

Функція	Пояснення
<pre>if map == np.where(map == [146, 134, 82]):     horz1 = [] else:     horz0 = []</pre>	Визначення та відбір кольору горизонталей
<pre>horz_map = np.asarray(Image.new(horz1))</pre>	Створення карти горизонталей
<pre>shp_map = shp.convert(horz_map)</pre>	Конвертація у шейп файл
<pre>plt.imshow(horz_map) plt.show() map_h = map.shape print(map_h)</pre>	Вивід усіх даних та розмірів ділянки карти у пікселях

12

## Апробація функції Python для оцифрування горизонталей. Вихідні дані



За вихідні дані використано ділянку  
карту Житомирської області,  
масштабу 1:100000, М – 35 – 059

13

## Апробація функції Python для оцифрування горизонталей. Результати



Вхідні дані



Карта  
горизонталей



ЦМР

14

## Висновки

- Розглянуто поняття цифрової моделі місцевості, шляхи створення та використання ЦМР.
- Розглянуто особливості мови програмування Python, її переваги та недоліки, бібліотеки, які доповнюють можливості оброблення графічних даних.
- Побудовано технологічну схему процесу автоматизації оцифрування горизонталей.
- Реалізовано та апробовано функцію на мові Python для автоматизації процесу оцифрування горизонталей ділянки карти.