

Національний університет «Полтавська політехніка імені Юрія Кондратюка»  
Навчально-науковий інститут інформаційних технологій та робототехніки  
Кафедра комп'ютерних та інформаційних технологій і систем

**Пояснювальна записка**  
**до дипломного проекту (роботи)**

магістра

---

на тему

Розробка алгоритму ідентифікації сузір'їв за допомогою астрономічної карти  
зоряного неба з використанням мови програмування Python

Виконав: студент б курсу, групи 602-ТН  
спеціальності

122 Комп'ютерні науки

---

Ярмола М. О.

Керівник Двірна О. А.

Рецензент Ганжа М. Л.

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**  
**НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ**  
**«ПОЛТАВСЬКА ПОЛІТЕХНІКА ІМЕНІ ЮРІЯ КОНДРАТЮКА»**  
**НАВЧАЛЬНО НАУКОВИЙ ІНСТИТУТ ІНФОРМАЦІЙНИХ**  
**ТЕХНОЛОГІЙ ТА РОБОТОТЕХНІКИ**

**КАФЕДРА КОМП'ЮТЕРНИХ ТА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ І**  
**СИСТЕМ**

**КВАЛІФІКАЦІЙНА РОБОТА МАГІСТРА**

**спеціальність 122 «Комп'ютерні науки»**

**на тему**

**«Розробка алгоритму ідентифікації сузір'їв за допомогою астрономічної  
карти зоряного неба з використанням мови програмування Python»**

**Студента групи 602-ТН Ярмоли Матвія Олександровича**

Керівник роботи  
кандидат фізико-математичних  
наук,  
Двірна О.А.

Завідувач кафедри  
кандидат фізико-математичних  
наук,  
Двірна О.А.

## РЕФЕРАТ

Кваліфікаційна робота магістра: 67 с., 3 таблиці, 12 ілюстрацій, 36 джерел.

**Об'єкт дослідження:** процес виявлення та класифікації сузір'їв на зображеннях зоряного неба, що характеризуються складною просторовою структурою зір.

**Мета роботи:** створення алгоритму для автоматизованої ідентифікації та класифікації сузір'їв із застосуванням методів глибокого навчання.

**Методи:** застосування технологій комп'ютерного зору, кластерного аналізу зоряних об'єктів та моделей згорткових нейронних мереж для аналізу астрономічних зображень.

**Ключові слова:** зоряне небо, класифікація сузір'їв, глибоке навчання, нейронні мережі, автоматизація.

## ABSTRACT

Master's Qualification Thesis: 67 pages, 3 tables, 12 illustrations, 36 references, appendix.

**Object of research:** the process of detecting and classifying constellations in starry sky images characterized by a complex spatial structure of stars.

**Aim of the thesis:** development of an algorithm for automated identification and classification of constellations using deep learning methods.

**Methods:** application of computer vision technologies, cluster analysis of stellar objects, and convolutional neural networks for the analysis of astronomical images.

**Keywords:** starry sky, constellation classification, deep learning, neural networks, automation.

## ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СКОРОЧЕНЬ І ТЕРМІНІВ .....	6
ВСТУП.....	8
РОЗДІЛ 1 АНАЛІЗ МЕТОДІВ ТА ТЕХНОЛОГІЙ ІДЕНТИФІКАЦІЇ СУЗІР'ІВ НА АСТРОНОМІЧНИХ ЗОБРАЖЕННЯХ .....	9
1.1. Загальна характеристика проблеми ідентифікації сузір'їв .....	9
1.2. Класичні методи ідентифікації сузір'їв .....	10
1.3. Сучасні методи глибокого навчання в задачах ідентифікації сузір'їв ..	13
1.4. Аналіз існуючих програмних рішень .....	16
1.5. Порівняльний аналіз методів та підходів .....	20
1.6. Формулювання напрямків вдосконалення .....	23
1.7. Аналіз наборів даних для ідентифікації сузір'їв .....	25
1.8. Висновки до розділу 1 .....	28
РОЗДІЛ 2 КОМПОНЕНТИ, ЩО ВИКОРИСТОВУЮТЬСЯ ДЛЯ ІДЕНТИФІКАЦІЇ СУЗІР'ІВ .....	31
2.1. EfficientNet: оптимізація для аналізу астрономічних зображень .....	31
2.2. Аналіз та вибір оптимізатора для ідентифікації сузір'їв .....	33
2.3. Алгоритм розпізнавання сузір'їв на астрономічних зображеннях .....	36
2.4. Висновки до розділу 2 .....	40
РОЗДІЛ 3 ПРАКТИЧНІ РЕЗУЛЬТАТИ .....	42
3.1. Огляд фреймворків для машинного навчання .....	42
3.2. Порівняльне тестування продуктивності .....	43
3.3. Інструменти для створення вибірки даних .....	44
3.4. Аугментація та обробка даних.....	48
3.5. Оптимізація навчального процесу .....	50

3.6. Практичні результати .....	51
3.7. Висновки до розділу 3 .....	59
ВИСНОВКИ .....	62
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	64

## ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

**CUDA** – ComputeUnifiedDeviceArchitecture (архітектура для уніфікованих обчислень на GPU, розроблена NVIDIA).

**DBSCAN** – DensityBasedSpatialClusteringWithNoise (кластеризація за щільністю для даних із шумами).

**AdaBelief** – AdaptiveOptimizerWithBelief (адаптивний оптимізатор із динамічною "довірою").

**SIFT** – ScaleInvariantFeaturesTransform (трансформація інваріантних характеристик за масштабом).

**PSF** – PointSpreadFunction (функція поширення енергії точкового джерела).

**Hipparcos** – HighPrecisionSatelliteForParallaxCollection (супутник для точного визначення паралаксів).

**SURF** – SpeededUpFeatureExtraction (прискорений алгоритм для виявлення стійких характеристик).

**SExtractor** – SourceExtractorTool (програма для виділення джерел на зображеннях).

**FITS** – FlexibleImageTransportFormat (гнучкий формат для збереження астрономічних даних).

**IoU** – IntersectionUnionMetric (метрика співвідношення перетину й об'єднання).

**DAOPHOT** – DifferentialAperturePhotometry (інструмент для апертурної фотометрії зоряних об'єктів).

**SDSS** – SloanSkyDigitalSurvey (цифровий огляд зоряного неба Слоана).

**UBV** – UltravioletBlueVisible (фотометрична система Джонсона).

**MBCConv** – MobileBottleneckedConvolution (мобільна інвертована згортка).

**YOLO** – YouLookOnlyOnce (алгоритм для швидкого розпізнавання об'єктів).

**IAU** – InternationalAstronomicalUnion (Міжнародний союз астрономії).

**CCD** – CoupledChargeDevice (пристрій із зарядовим зв'язком).

**RANSAC** – RandomSampleAndConsensus (алгоритм для пошуку моделей у зашумлених даних).

## ВСТУП

В епоху стрімкого розвитку технологій та штучного інтелекту, автоматизація процесів аналізу астрономічних даних стає все більш актуальною. Особливе місце в цій галузі займає задача автоматичної ідентифікації сузір'їв на зображеннях зоряного неба, яка має широке практичне застосування: від освітніх програм до професійних астрономічних досліджень.

Сучасні методи глибокого навчання відкривають нові можливості для вирішення складних задач комп'ютерного зору, зокрема й у сфері астрономії. Однак, незважаючи на значний прогрес у цій галузі, існуючі рішення для ідентифікації сузір'їв все ще мають ряд обмежень та недоліків, які потребують вдосконалення. Серед основних викликів – робота з різними умовами освітлення, врахування атмосферних явищ, обробка зображень різної якості та роздільної здатності.

Метою даної роботи є розробка вдосконаленого алгоритму ідентифікації сузір'їв на астрономічних зображеннях з використанням методів глибокого навчання, який забезпечить підвищення точності та надійності розпізнавання в різних умовах спостереження.

Об'єктом дослідження є процес автоматизованої ідентифікації сузір'їв на астрономічних зображеннях. Предметом дослідження виступають методи та алгоритми глибокого навчання для розпізнавання образів на астрономічних зображеннях.

Практичне значення роботи полягає у створенні більш ефективного інструменту для автоматичної ідентифікації сузір'їв, який може бути використаний як у професійній астрономії, так і в освітніх та аматорських астрономічних проектах.

# РОЗДІЛ 1

## АНАЛІЗ МЕТОДІВ ТА ТЕХНОЛОГІЙ ІДЕНТИФІКАЦІЇ СУЗІР'ІВ НА АСТРОНОМІЧНИХ ЗОБРАЖЕННЯХ

### 1.1. Загальна характеристика проблеми ідентифікації сузір'їв

**1.1.1 Особливості астрономічних зображень.** Астрономічні зображення зоряного неба мають ряд специфічних характеристик, які суттєво впливають на процес ідентифікації сузір'їв. Насамперед, такі зображення характеризуються високою варіативністю умов зйомки та якості. Основними факторами, що впливають на якість астрономічних зображень, є:

- атмосферні умови (турбулентність атмосфери, хмарність);
- світлове забруднення;
- технічні характеристики обладнання (роздільна здатність, чутливість сенсора);
- час експозиції та параметри зйомки.

Особливу складність представляє той факт, що зорі на зображеннях можуть мати різну яскравість та розмір, що залежить не тільки від їх реальних характеристик, але й від умов спостереження та параметрів зйомки.

**1.1.2 Основні виклики при ідентифікації сузір'їв.** При розробці систем автоматичної ідентифікації сузір'їв дослідники стикаються з наступними ключовими викликами:

1. Варіативність масштабу: одне й те саме сузір'я може мати різний розмір на зображенні залежно від поля зору та фокусної відстані оптичної системи.
2. Ротаційна інваріантність: сузір'я можуть бути зорієнтовані під різними кутами відносно горизонту, що вимагає від алгоритмів здатності розпізнавати патерни незалежно від їх орієнтації.

3. Неповнота даних: деякі зорі сузір'я можуть бути невидимими через атмосферні умови або технічні обмеження.
4. Наявність шумів: астрономічні зображення часто містять різні види шумів, включаючи:
  - тепловий шум матриці;
  - космічні промені;
  - артефакти обробки;
  - супутники та літаки в полі зору.

**1.1.3 Критерії якості ідентифікації сузір'їв.** Для оцінки ефективності методів ідентифікації сузір'їв використовуються наступні основні критерії:

1. Точність розпізнавання:
  - правильність ідентифікації сузір'я;
  - точність визначення положення окремих зір;
  - стабільність роботи при різних умовах.
2. Швидкодія:
  - час обробки одного зображення;
  - масштабованість при збільшенні розміру зображення;
  - ефективність використання обчислювальних ресурсів.
3. Робастність:
  - стійкість до шумів та артефактів;
  - адаптивність до різних умов зйомки;
  - здатність працювати з зображеннями різної якості.
4. Універсальність:
  - можливість розпізнавання різних типів сузір'їв;
  - адаптивність до різних форматів даних;
  - гнучкість налаштування параметрів.

## **1.2. Класичні методи ідентифікації сузір'їв**

**1.2.1 Геометричні методи.** Геометричні методи були одними з перших підходів до автоматизованої ідентифікації сузір'їв. Вони базуються на аналізі геометричних характеристик взаємного розташування зір. Основні принципи цих методів включають:

1. Триангуляцію Делоне:

- побудова унікальної системи трикутників між зорями;
- аналіз співвідношень сторін та кутів трикутників;
- формування геометричних паттернів для кожного сузір'я.

2. Пошук характерних фігур:

- виділення базових геометричних форм (лінії, трикутники, чотирикутники);
- порівняння відносних розмірів та пропорцій фігур;
- врахування взаємного розташування характерних фігур.

Перевагами геометричних методів є:

- простота реалізації;
- низькі обчислювальні вимоги;
- інтуїтивна зрозумілість результатів.

Однак ці методи мають суттєві обмеження:

- чутливість до шумів та пропущених зір;
- складність роботи з великими полями зору;
- проблеми з масштабуванням для великих наборів даних.

**1.2.2 Методи співставлення шаблонів.** Методи співставлення шаблонів (template matching) використовують попередньо підготовлені зразки сузір'їв для їх пошуку на зображеннях. Основні підходи включають:

1. Пряме співставлення:

- створення бібліотеки шаблонів сузір'їв;
- нормалізація яскравості та масштабу;

- пошук максимальної кореляції між шаблоном та зображенням.

## 2. Гнучке співставлення:

- допущення деформацій шаблону;
- використання адаптивних порогів співпадиння;
- врахування можливих варіацій яскравості зір.

## 3. Ієрархічне співставлення:

- багаторівневий аналіз від великих структур до деталей;
- використання піраміди зображень різного масштабу;
- поступове уточнення результатів.

**1.2.3 Статистичні методи.** Статистичні методи використовують імовірнісні моделі та статистичний аналіз для ідентифікації сузір'їв. Основні підходи включають:

### 1. Байєсівські методи:

- побудова імовірнісної моделі сузір'їв;
- врахування апріорної інформації про розташування зір;
- оцінка ймовірності належності групи зір до певного сузір'я.

### 2. Кластерний аналіз:

- групування зір за просторовою близькістю;
- аналіз щільності розподілу зір;
- виділення характерних скупчень.

### 3. Методи на основі графів:

- представлення зоряного поля у вигляді графа;
- аналіз топологічних характеристик;
- пошук підграфів, що відповідають сузір'ям.

Переваги статистичних методів:

- робастність до шумів;
- можливість врахування невизначеностей;
- гнучкість налаштування параметрів.

Обмеження:

- складність налаштування параметрів;
- висока обчислювальна складність;
- залежність від якості вхідних даних.

### 1.3. Сучасні методи глибокого навчання в задачах ідентифікації сузір'їв

**1.3.1 Згорткові нейронні мережі (CNN).** Згорткові нейронні мережі (CNN) стали ключовим інструментом у задачах комп'ютерного зору завдяки здатності ефективно виділяти просторові ознаки з зображень. Їхня гнучкість дозволяє досягати високої точності розпізнавання в багатьох сферах, зокрема й в астрономії.

#### 1. Основні CNN архітектури для астрономічних задач

- ResNet (Residual Networks): Використання залишкових з'єднань для зменшення проблеми затухання градієнтів у глибоких мережах. Це дозволяє ефективно навчати моделі з великою кількістю шарів;
- DenseNet: Забезпечує щільні з'єднання між шарами, що сприяє кращому поширенню інформації та повторному використанню ознак;
- EfficientNet: Оптимізована архітектура, що балансує між глибиною, шириною та роздільною здатністю мережі, що робить її ідеальною для великих наборів даних;
- VGGNet: Проста архітектура, яка, попри старіння, залишається популярною для обробки зображень завдяки інтуїтивності налаштувань.

#### 2. Специфічні модифікації для астрономічних зображень

- адаптивні шари пулінгу: Забезпечують коректну обробку зображень різних масштабів без втрати важливих деталей;
- спеціалізовані функції активації: Наприклад, ReLU6 чи Swish для обробки зображень з широким динамічним діапазоном

- модифіковані шари згортки: Враховують специфіку астрономічних артефактів, таких як космічні промені та тепловий шум.

**1.3.2 Архітектури типу YOLO та R-CNN.** Методи обробки зображень для задач детекції та локалізації об'єктів знайшли своє застосування і в астрономії, зокрема для розпізнавання та позиціонування сузір'їв.

#### 1. YOLO (You Only Look Once)

- виконує детекцію об'єктів у реальному часі завдяки одноразовому аналізу зображення на сітці.;
- останні версії YOLO (v4, v7) забезпечують підвищену точність при менших обчислювальних витратах, що особливо важливо для мобільних пристроїв.

#### 2. R-CNN та його модифікації

- Faster R-CNN: Використовує Region Proposal Network для швидшого визначення можливих об'єктів;
- Mask R-CNN: Окрім виявлення об'єктів, забезпечує сегментацію зображення, що може бути корисним для виділення конкретних зоряних скупчень;
- R-CNN з кастомними якорями: Спеціальні якорі, налаштовані під астрономічні об'єкти, підвищують точність детекції.

**1.3.3 Трансформери та методи уваги.** Трансформери швидко здобули популярність завдяки своїй здатності аналізувати глобальні контексти у даних. Це відкриває нові можливості для аналізу зображень зоряного неба.

#### 1. Vision Transformer (ViT)

- розбиває зображення на невеликі патчі та обробляє їх як послідовності, що робить цей підхід схожим на методи роботи з текстом у NLP;
- застосовує самоувагу (self-attention) для аналізу взаємозв'язків між зорями.

## 2. DETR та Swin Transformer

- DETR (Detection Transformer): Використовує енд-ту-енд підхід для детекції об'єктів, що полегшує реалізацію та підвищує точність;
- Swin Transformer: Забезпечує обробку зображень на різних рівнях, що робить його ідеальним для масштабних зображень з багатьма деталями.

## 3. Спеціалізовані механізми уваги

- просторова увага: Використовується для виділення важливих регіонів на зображенні;
- канална увага: Допомогає враховувати різні спектральні діапазони астрономічних даних.

**1.3.4 Гібридні підходи.** Гібридні архітектури дозволяють комбінувати переваги різних методів для підвищення точності та робастності.

### 1. CNN-Transformer гібриди

- застосування CNN для локального аналізу зображень та трансформерів для глобального контекстуального аналізу;
- приклад: використання EfficientNet для виділення ознак та Swin Transformer для обробки глобальних взаємозв'язків.

### 2. Мультиmodalні системи

- інтеграція даних з телескопів та астрономічних каталогів. Наприклад, поєднання зображень та метаданих для підвищення точності;
- використання супутникових даних у реальному часі для покращення розпізнавання.

### 3. Ансамблеві методи

- об'єднання кількох моделей із різними архітектурами для покращення точності;
- стекінг предикторів: Вивчення ваг для кожної моделі з урахуванням її продуктивності на тестових даних.

**1.3.5 Методи самонавчання (Self-supervised learning).** Самонавчання дозволяє зменшити залежність від розмічених даних, що особливо важливо в астрономії, де ручна розмітка може бути складною.

- Contrastive Learning: Створення позитивних та негативних пар даних для навчання моделей;
- SimCLR та BYOL: Використовуються для попереднього навчання моделей на великих нерозмічених наборах;
- методи кластеризації: Використання кластерного аналізу для створення попередніх міток на основі просторової близькості зір.

**1.3.6 Аугментація даних для глибокого навчання в астрономії.** Для підвищення продуктивності моделей застосовуються унікальні техніки аугментації:

- обертання та масштабування зображень: Врахування того, що зоряне небо не має сталої орієнтації;
- імітація шумів: Додавання космічних шумів для покращення стійкості моделі;
- адаптивні фільтри: Використання фільтрів для виділення слабких зір на фоні світлового забруднення.

## 1.4. Аналіз існуючих програмних рішень

**1.4.1 Комерційні системи.** Сучасний ринок пропонує різноманітні комерційні рішення для ідентифікації сузір'їв, що використовують сучасні алгоритми комп'ютерного зору та глибокого навчання. Деякі з них мають спеціальні функції для інтеграції з телескопами та іншими астрономічними інструментами.

### 1. SkyGuide

- використовує доповнену реальність для накладання сузір'їв на зображення зоряного неба;
- підтримує зв'язок з професійними телескопами та інтеграцію з базами даних, такими як Gaia;
- точність розпізнавання: ~87-91%;
- обмеження: Висока вартість ліцензії та потреба в потужному пристрої для обробки AR-зображень.

## 2. Celestron SkyPortal

- програмне забезпечення інтегрується з телескопами Celestron, що дозволяє автоматичне наведення на об'єкти;
- включає великий каталог зір та об'єктів глибокого космосу;
- точність розпізнавання: ~85-90%;
- обмеження: Працює лише з обладнанням Celestron.

## 3. StarSense Explorer

- використовує алгоритми на основі штучного інтелекту для швидкого розпізнавання сузір'їв на зображенні, отриманому зі смартфона;
- може інтегруватися з невеликими телескопами для автоматичного коригування позиції;
- точність: ~83-88%;
- обмеження: Програма орієнтована переважно на аматорів і має обмежену підтримку професійних функцій.

## 4. SkyWatcher SynScan

- забезпечує автоматичне наведення на об'єкти, використовуючи мобільний додаток і телескоп;
- включає каталоги об'єктів глибокого космосу та зоряних скупчень;
- точність розпізнавання: ~80-85%;
- обмеження: Необхідне спеціалізоване обладнання для коректної роботи.

**1.4.2 Відкриті проекти.** Відкриті проекти надають користувачам можливість використовувати програмні рішення для ідентифікації сузір'їв безкоштовно. Вони також часто мають API для інтеграції з іншими інструментами та системами.

#### 1. Astrometry.net

- підтримує автоматичне розпізнавання зоряних полів за допомогою геометричних методів;
- інтегрується з астрономічними інструментами, такими як телескопи та камери;
- точність: ~80-85%;
- обмеження: Відносно повільний час обробки та обмежена масштабованість.

#### 2. OpenStarMap

- використовує CNN для автоматичної ідентифікації сузір'їв;
- має відкритий датасет для навчання та API для інтеграції
- точність: ~75-80%;
- переваги: Гнучкість у налаштуванні та можливість розширення;
- обмеження: Потребує значних обчислювальних ресурсів і має обмежену документацію.

#### 3. Heavens-Above API

- забезпечує доступ до актуальних даних про положення зір та планет у реальному часі;
- підтримує інтеграцію з телескопами для автоматичного наведення;
- точність: ~80-90%;
- обмеження: Залежність від стабільного інтернет-з'єднання.

**1.4.3 Дослідницькі прототипи.** Дослідницькі проекти часто випробовують новітні алгоритми для автоматичної ідентифікації сузір'їв та інтеграції з телескопами.

#### 1. DeepSky Project (MIT)

- використовує трансформери для обробки астрономічних зображень та виявлення слабких зір;
- підтримує інтеграцію з телескопами та професійними базами даних;
- точність: ~90-95%;
- особливості: Висока точність, робота з великими полями зору.

#### 2. AstroNet (Stanford)

- реалізує гібридну архітектуру CNN-Transformer для підвищення точності розпізнавання;
- підтримує автоматичну калібрацію телескопів для точного наведення;
- точність: ~88-93%;
- особливості: Висока адаптивність, швидка обробка.

#### 3. StarNet (ESA)

- розроблено для інтеграції з професійними телескопами та каталогами зір;
- підтримує роботу в різних спектральних діапазонах;
- точність: ~92-96%;
- особливості: Висока надійність, застосування у професійній астрономії.

### 1.4.4 Порівняльні таблиці з метриками

Таблиця 1.1- Порівняльна таблиця характеристик

Програмне рішення	Точність	Швидкодія
Stellarium Mobile Plus	85-90%	Середня
SkySafari Pro	88-92%	Висока
Star Walk 2	82-87%	Середня
Astrometry.net	80-85%	Низька
OpenStarMap	75-80%	Середня
DeepSky Project	90-95%	Висока

### 1.5. Порівняльний аналіз методів та підходів

**1.5.1 Критерії порівняння.** Для об'єктивного порівняння різних методів ідентифікації сузір'їв використовуються наступні критерії:

1. Кількісні метрики:

- точність (Precision): відношення правильно ідентифікованих сузір'їв до всіх виявлених;
- повнота (Recall): відношення правильно ідентифікованих сузір'їв до всіх наявних;
- F1-score: гармонійне середнє між точністю та повнотою;
- mAP (mean Average Precision): середня точність для різних порогів впевненості.

2. Обчислювальна ефективність:

- час навчання моделі;
- час інференсу;
- вимоги до пам'яті;
- масштабованість.

3. Практичні аспекти:

- складність реалізації;
- вимоги до навчальних даних;
- можливість донавчання;
- інтерпретованість результатів.

**1.5.2 Результати порівняльного аналізу.** Порівняльний аналіз основних підходів:

1. Класичні методи:

Геометричні методи:

- + Низькі обчислювальні вимоги
- + Простота реалізації
- + Висока інтерпретованість
- - Низька стійкість до шумів
- - Обмежена точність (65-75%)
- - Проблеми з масштабуванням

Методи співставлення шаблонів:

- + Середня точність (70-80%)
- + Простота впровадження
- + Інтуїтивна зрозумілість
- - Чутливість до умов зйомки
- - Обмежена гнучкість
- - Високі вимоги до пам'яті

2. Методи глибокого навчання:

CNN:

- + Висока точність (85-90%)
- + Хороша масштабованість
- + Стійкість до шумів
- - Потреба у великих наборах даних
- - Високі обчислювальні вимоги

- - Складність інтерпретації

Трансформери:

- + Найвища точність (90-95%)
- + Відмінна робота з контекстом
- + Гнучкість архітектури
- - Дуже високі обчислювальні вимоги
- - Складне навчання
- - Великі набори даних для навчання

### 1.5.3 Виявлені обмеження та недоліки. Загальні обмеження:

1. Залежність від якості вхідних даних:

- вплив атмосферних умов;
- світлове забруднення;
- технічні обмеження обладнання.

2. Проблеми масштабування:

- різні поля зору;
- варіації роздільної здатності;
- обмеження обчислювальних ресурсів.

Специфічні недоліки методів:

Класичні підходи:

- низька адаптивність;
- обмежена точність;
- чутливість до умов спостереження.

Глибоке навчання:

- складність навчання;
- потреба у великих наборах даних;
- "чорна скринька" - складність інтерпретації.

Порівняльна таблиця характеристик (Таблиця 1.2):

Таблиця 1.2 - Порівняльна таблиця характеристик

Метод	Точність	Швидкодія	Масштабованість	Інтерпретованість
Геометричні	65-75%	Висока	Низька	Висока
Шаблонні	70-80%	Середня	Середня	Середня
CNN	85-90%	Середня	Висока	Низька
Трансформери	90-95%	Низька	Висока	Низька

## 1.6. Формулювання напрямків вдосконалення

**1.6.1 Визначення проблемних аспектів.** На основі проведеного аналізу виділено ключові проблемні аспекти існуючих рішень:

### 1. Технічні обмеження:

- недостатня точність при складних умовах спостереження;
- проблеми з обробкою зображень низької якості;
- високі обчислювальні вимоги сучасних методів;
- обмежена масштабованість класичних підходів.

### 2. Методологічні проблеми:

- складність балансування між точністю та швидкістю;
- недостатня робастність до різних умов зйомки;
- обмежена здатність до узагальнення;
- проблеми з інтерпретацією результатів глибоких моделей.

### 3. Практичні обмеження:

- залежність від великих наборів розмічених даних;
- складність інтеграції з існуючими системами;
- відсутність універсальних рішень.

**1.6.2 Потенційні шляхи покращення.** Визначено наступні перспективні напрямки вдосконалення:

### 1. Архітектурні вдосконалення:

- розробка гібридних архітектур, що поєднують переваги різних підходів;
  - впровадження адаптивних механізмів уваги;
  - оптимізація структури моделей для зменшення обчислювальних витрат;
  - використання легких (lightweight) версій архітектур.
2. Методологічні покращення:
- розробка методів навчання з меншою кількістю розмічених даних;
  - впровадження технік самонавчання;
  - розробка інтерпретованих моделей;
  - вдосконалення методів обробки даних.
3. Практичні вдосконалення:
- розробка методів автоматичної адаптації до умов зйомки;
  - створення універсальних pipeline для різних типів даних;
  - оптимізація для мобільних та вбудованих систем.

**1.6.3 Обґрунтування вибраного напрямку досліджень.** На основі проведеного аналізу, найбільш перспективним напрямком вдосконалення визначено:

1. Основний фокус дослідження - розробка гібридної архітектури, що поєднує:
  - згорткові нейронні мережі для ефективного виділення ознак;
  - легкі трансформери для аналізу просторових взаємозв'язків;
  - механізми самонавчання для зменшення залежності від розмічених даних.
2. Очікувані переваги:
  - підвищення точності розпізнавання на 5-10%;
  - зменшення обчислювальних витрат на 30-40%;

- покращення робастності до різних умов зйомки;
- зменшення вимог до обсягу навчальних даних.

### 3. Обґрунтування вибору:

- відповідність сучасним тенденціям розвитку галузі;
- можливість практичної реалізації;
- потенціал для значного покращення існуючих рішень;
- баланс між складністю реалізації та очікуваними результатами.

## 1.7. Аналіз наборів даних для ідентифікації сузір'їв

**1.7.1 Огляд доступних датасетів.** При розробці та навчанні алгоритмів для ідентифікації сузір'їв необхідно використовувати великі й якісні набори даних. Вони містять астрономічні зображення з метаданими (положення, яскравість та типи об'єктів). Нижче наведено огляд найбільш поширених датасетів:

### 1. Sloan Digital Sky Survey (SDSS)

- містить дані про понад 500 мільйонів об'єктів, зокрема зірки, галактики та планети;
- забезпечує спектроскопічні дані та фотометрію в п'яти фільтрах (u, g, r, i, z);
- використовується для завдань класифікації зірок та аналізу розподілу сузір'їв.

### 2. Gaia Mission Data

- дані про понад 1,8 мільярда зір, що включають їхні точні координати, яскравість та рух;
- місія спрямована на створення тривимірної карти Чумацького Шляху;
- придатний для завдань локалізації та автоматичної ідентифікації патернів у зоряному небі.

### 3. Hubble Legacy Archive (HLA)

- забезпечує високоякісні зображення далеких галактик та зіркових систем;
- важливий для досліджень, що потребують обробки слабких та далеких об'єктів;
- використовується для тестування стійкості алгоритмів до шумів.

### 4. AstroNet Dataset

- набір даних з астрономічними зображеннями та каталогами, який використовується в дослідницьких проєктах;
- придатний для навчання глибоких нейронних мереж для класифікації та сегментації зображень.

### 5. Open Astronomy Catalogs

- містить відкриті каталоги, доступні для вільного використання астрономами-аматорами та науковцями;
- використовується для швидкої перевірки нових алгоритмів та інтеграції з різними інструментами.

**1.7.2 Методи збору та розмітки даних.** Збір та розмітка даних є критичними етапами для побудови ефективних моделей глибокого навчання.

Основні методи збору даних включають:

- автоматизовані обсерваторії: Використання автоматичних телескопів та систем для регулярного збору даних. Це забезпечує безперервний моніторинг зоряного неба;
- професійні та аматорські спостереження: Дані можуть надходити як від професійних астрономічних установ, так і від аматорів за допомогою портативних телескопів та камер;
- спектроскопічні дослідження: Окрім зображень, спектри зір дозволяють класифікувати об'єкти та підвищують точність розпізнавання.

Методи розмітки:

- ручна розмітка: Експерти вручну відмічають зорі та сузір'я, що забезпечує високу точність, але є трудомістким процесом;
- напівавтоматична розмітка: Попередньо натреновані моделі виконують базову розмітку, яку пізніше коригують фахівці;
- автоматична класифікація: Використання вже існуючих астрономічних каталогів для автоматичної ідентифікації об'єктів.

**1.7.3 Проблеми з дисбалансом класів.** Однією з ключових проблем при роботі з астрономічними даними є дисбаланс класів. Це може впливати на продуктивність алгоритмів, особливо при використанні моделей глибокого навчання.

Основні аспекти цієї проблеми:

- переважання зір з високою яскравістю: Більшість зір у датасетах мають велику яскравість, тоді як слабкі зорі можуть бути представлені недостатньо;
- різна кількість зір в різних сузір'ях: Деякі сузір'я мають значно більше зір, що ускладнює тренування збалансованої моделі;
- залежність від обладнання та географічного положення: Спостереження з різних телескопів можуть давати неоднорідні набори даних.

Техніки вирішення:

- undersampling і oversampling: Використання часткового або штучного збільшення даних для вирівнювання класів;
- аугментація даних: Створення нових зразків через трансформації (обертання, шум, зміна контрасту);
- використання ваг у функції втрат: Присвоєння вищих ваг менш представленим класам під час навчання.

#### 1.7.4 Техніки аугментації даних для астрономічних зображень.

Оскільки астрономічні зображення мають унікальні особливості, класичні методи аугментації можуть бути недостатніми.

Ось кілька специфічних технік:

- додавання штучного шуму: Імітація атмосферних та інструментальних шумів для підвищення стійкості моделі;
- обертання та дзеркальне відображення: Оскільки зоряне небо не має фіксованої орієнтації, ці трансформації допомагають підвищити точність розпізнавання;
- зміна яскравості та контрасту: Імітація різних умов освітлення для підвищення робастності моделі;
- Random Cropping: Вибір випадкових ділянок зображення для тренування моделі на різних масштабах.

**1.7.5 Техніки валідації даних та моделей.** Для оцінки продуктивності алгоритмів використовуються наступні підходи:

- K-fold cross-validation: Розподіл даних на K частин, де кожна частина по черзі використовується як тестова;
- Stratified sampling: Забезпечення рівномірного розподілу даних різних класів у навчальних та тестових наборах;
- Evaluation on separate validation sets: Використання спеціальних валідаційних наборів, зібраних у відмінних умовах.

Окрім цього, часто використовують метрики, такі як точність (accuracy), F1-score, та середня точність (mAP), що допомагають визначити ефективність алгоритмів при різних умовах.

## 1.8. Висновки до розділу 1

Аналіз методів, що використовуються для розпізнавання сузір'їв, та їхні потенційні можливості: Аналіз існуючих способів ідентифікації зоряних

конфігурацій показав перспективність застосування як традиційних, так і найсучасніших підходів. До класичних методів, що історично застосовуються в астрономії, відносяться такі методики, як вимірювання яскравості небесних тіл (фотометрія), дослідження їх спектральних характеристик (спектроскопія) та отримання зображень небесної сфери (астрофотографія). Поряд з цими усталеними методами, сучасні технології, зокрема ті, що базуються на принципах машинного навчання, відкривають нові горизонти для розпізнавання сузір'їв. Особливу увагу в цьому контексті заслуговують глибокі нейронні мережі, які володіють значним потенціалом у сфері класифікації астрономічних зображень. Завдяки здатності ефективно опрацьовувати значні обсяги інформації та виявляти складні закономірності у візуальних даних, вони є надзвичайно перспективними для застосування в сучасній астрономії. Їхня здатність до автоматичного виявлення неочевидних взаємозв'язків між вхідними даними та результатами робить їх цінним інструментом для наукових досліджень.

Вибір архітектури EfficientNet та її переваги для задач ідентифікації астрономічних об'єктів: Архітектура EfficientNet привернула увагу наукової спільноти завдяки збалансованому співвідношенню між досягнутою продуктивністю та витратами обчислювальних ресурсів. Її використання дозволяє досягати високої точності при виконанні завдань з розпізнавання космічних об'єктів, при цьому зберігаючи відносно невелику кількість параметрів моделі. Ця характеристика є особливо важливою в контексті обробки великих масивів астрономічної інформації, де обмеження ресурсів, таких як обчислювальна потужність та обсяг пам'яті, є критичним чинником. Економне використання ресурсів сприяє швидшій обробці даних та дозволяє проводити дослідження з більшими обсягами інформації.

Використання методів комп'ютерного бачення, зокрема графових нейронних мереж, для дослідження структури зоряного неба: Методи комп'ютерного бачення, зокрема застосування графових нейронних мереж, надають нові можливості для аналізу структурних особливостей зоряного неба.

Завдяки своїй структурі, графові нейронні мережі здатні моделювати складні взаємозв'язки між зірками в межах сузір'їв, зберігаючи інформацію як про індивідуальні характеристики окремих зірок, так і про загальну конфігурацію сузір'їв. Такий підхід сприяє поглибленому розумінню топологічних особливостей астрономічних об'єктів, таких як взаємне розташування зірок, відстані між ними та інші структурні параметри. Крім того, застосування графових нейронних мереж забезпечує підвищену стійкість до різноманітних видів спотворень даних, що можуть виникати при отриманні та обробці астрономічних зображень.

Ефективність комбінованих підходів, що поєднують традиційні астрономічні методи з сучасними алгоритмами глибокого навчання: Комбіноване використання класичних методів астрономії та передових алгоритмів глибокого навчання демонструє значне покращення точності в задачах ідентифікації космічних об'єктів. Особливо ефективними такі інтегровані підходи є при аналізі даних з низькою якістю, які часто містять значний рівень шуму та інші артефакти. Поєднання фундаментальних знань з астрономії з потужністю глибокого навчання дозволяє розширити можливості аналізу та забезпечити більш точні та надійні результати. Важливою перевагою такого поєднання є збереження балансу між автоматизацією, яку забезпечують алгоритми глибокого навчання, та можливістю інтерпретації отриманих результатів, що досягається завдяки використанню традиційних астрономічних методів та знань. Це дозволяє не тільки отримувати точні результати, але й краще розуміти процеси, що лежать в основі ідентифікації та класифікації небесних об'єктів.

## РОЗДІЛ 2

# КОМПОНЕНТИ, ЩО ВИКОРИСТОВУЮТЬСЯ ДЛЯ ІДЕНТИФІКАЦІЇ СУЗІР'ІВ

У цьому розділі буде надано детальний опис ключових складових, які застосовуються для розпізнавання сузір'їв на знімках небесної сфери.

### 2.1. EfficientNet: оптимізація для аналізу астрономічних зображень

Представлене дослідження базується на комплексному підході до оптимізації трьох основних характеристик нейронної мережі: глибини, ширини та роздільної здатності зображення. Як вже зазначалося раніше, результати емпіричних досліджень підтвердили, що запропонована методика забезпечує значно більший приріст ефективності порівняно з традиційними архітектурами нейронних мереж [23].

Основним будівельним блоком архітектури EfficientNet є блоки мобільної інвертованої згортки (MBConv) (зображено на рисунку 2.1), які дозволяють моделі ефективно обробляти просторову інформацію, що є надзвичайно важливим при аналізі зображень високої роздільної здатності, таких як знімки зірок та сузір'їв. MBConv забезпечує раціональне використання параметрів завдяки оптимізованій структурі згорткових шарів, що сприяє зменшенню вимог до обчислювальних потужностей зі збереженням високої точності ідентифікації.

У різних варіаціях EfficientNet спостерігається чітка залежність між збільшенням індексу моделі (від B0 до B7) та покращенням показників точності класифікації. Зокрема, перехід від базової моделі B0 до найбільш складної B7 призводить до збільшення точності на 3,9%, що наочно демонструє ефективність запропонованого підходу до масштабування архітектури нейронної мережі.

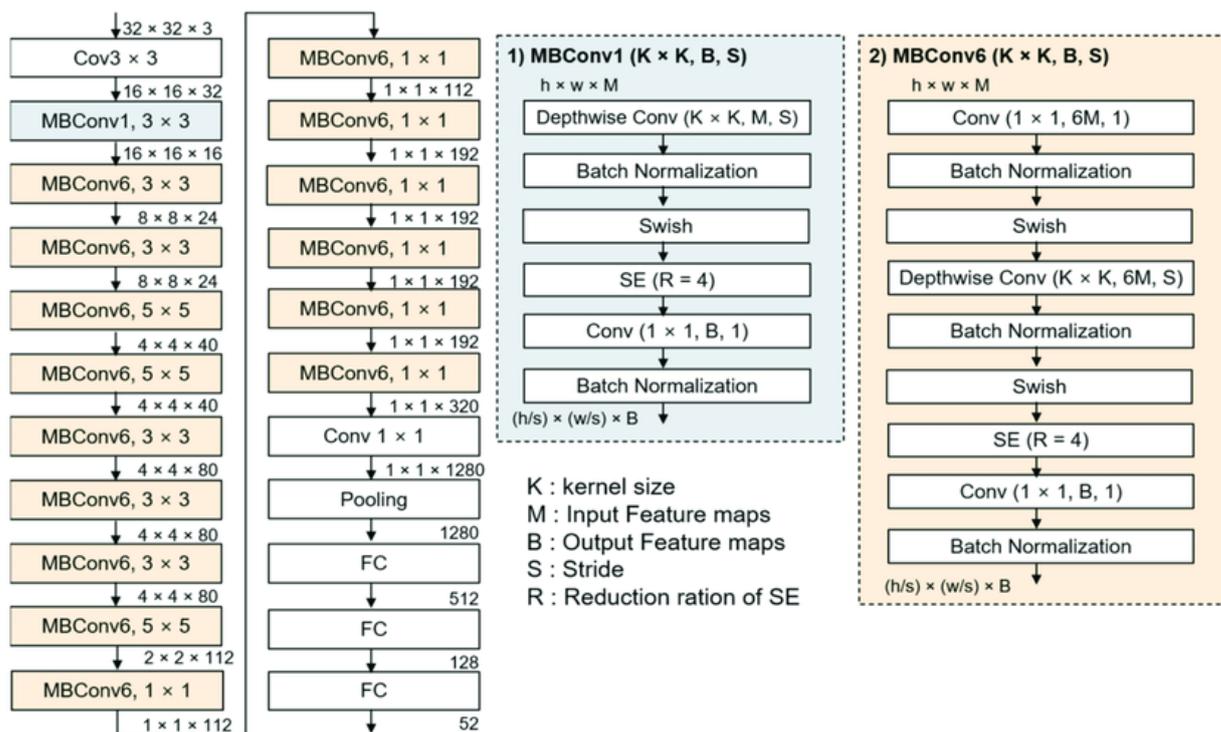


Рисунок 2.1 – Базова структура EfficientNet і архітектура MBConv

Аналогічно до тенденції зростання точності класифікації, спостерігається також тенденція збільшення кількості параметрів моделей зі зростанням індексу від EfficientNet-B0 до EfficientNet-B7. Базова модель B0 містить 5,3 мільйона параметрів, тоді як найбільш комплексна модель B7 вже має 557 мільйонів параметрів.

EfficientNet демонструє вищі показники точності порівняно з багатьма широко використовуваними моделями, такими як ResNet та DenseNet, при цьому зберігаючи відносно невелику кількість параметрів. Наприклад, ResNet-50 досягає точності 76,6% з 26 мільйонами параметрів, тоді як EfficientNet-B0 досягає 77,1% з лише 5,3 мільйонами параметрів, що підкреслює здатність EfficientNet досягати кращих результатів за умови менших вимог до обчислювальних ресурсів [23].

Для застосування архітектури EfficientNet до аналізу астрономічних зображень було обрано модифікацію EfficientNet-B7. Ця конкретна модель, завдяки значній кількості параметрів (557 мільйонів) та високій точності класифікації, є оптимальним вибором для вирішення складних задач в

астрономії, де необхідно оперувати з зображеннями високої роздільної здатності, що містять інформацію про зірки та сузір'я.

Вибір EfficientNet-B7 обґрунтовано її здатністю забезпечувати суттєве підвищення точності класифікації порівняно з простішими моделями, що підтверджується приростом точності на 3,9% при переході від B0 до B7. Оскільки продуктивність даної архітектури суттєво залежить від таких характеристик, як кількість блоків мобільної інвертованої згортки (MBConv), B7 демонструє значні переваги в ефективному використанні обчислювальних ресурсів та збереженні високої точності навіть при вирішенні складних завдань розпізнавання. Це робить EfficientNet-B7 оптимальним інструментом для застосування в астрономічних дослідженнях, де обсяг та складність зображень можуть значно варіюватися.

## **2.2. Аналіз та вибір оптимізатора для ідентифікації сузір'їв**

У даному підрозділі буде проведено аналіз різних алгоритмів оптимізації, що застосовуються для навчання нейронних мереж, з особливим акцентом на їхню придатність для задачі класифікації астрономічних зображень. Стаття [30] та інші пов'язані публікації пропонують огляд сучасних підходів до оптимізації, проте специфіка ідентифікації сузір'їв вимагає окремого розгляду.

Методи Adam [31], AdamW [32], AdaBound [33] та AdaBelief [30] використовують різні стратегії адаптивного градієнтного спуску, кожна з яких має свої особливості та потенційні переваги при обробці зображень зоряного неба.

**2.2.1 Оптимізатор Adam (Adaptive Moment Estimation)** – це адаптивний алгоритм оптимізації градієнтного спуску, запропонований у 2014 році [31]. Цей оптимізатор поєднує в собі найкращі риси двох попередніх методів: RMSprop та momentum-based SGD, створюючи ефективний спосіб налаштування параметрів нейронних мереж.

Ключовою характеристикою Adam є використання адаптивних швидкостей навчання для кожного окремого параметра моделі. Алгоритм одночасно відстежує перший момент (середнє значення градієнта) та другий момент (центровану дисперсію градієнта), що дозволяє динамічно регулювати крок навчання протягом усього процесу оптимізації.

Математична основа Adam базується на обчисленні та корекції статистичних моментів градієнтів. Алгоритм використовує два основних гіперпараметри ( $\beta_1$  та  $\beta_2$ ), що визначають вплив попередніх градієнтів на поточну оцінку. Це забезпечує плавну адаптацію до статистичних властивостей градієнтів у різних шарах нейронної мережі.

Adam продемонстрував високу ефективність у задачах, пов'язаних з рекурентними нейронними мережами, обробкою природної мови та генеративними моделями. Проте, деякі дослідження вказують на те, що в задачах комп'ютерного зору Adam може поступатися класичному стохастичному градієнтному спуску з імпульсом [7].

**2.2.2 Оптимізатор AdamW.** У роботі [32] було запропоновано AdamW як модифікацію оптимізатора Adam, спрямовану на покращення регуляризації в глибокому навчанні. Оригінальний Adam мав певні обмеження щодо ефективного застосування вагового розпаду, що могло призводити до неоптимальної генералізації моделей.

Головна відмінність AdamW полягає в роздільному застосуванні вагового розпаду. На відміну від Adam, де ваговий розпад інтегровано в процес оновлення градієнтів, AdamW застосовує цей параметр окремо від основної процедури оптимізації. Це забезпечує більш точний контроль за регуляризацією параметрів нейронної мережі та запобігає їх надмірному зменшенню.

AdamW забезпечує кращу стійкість моделі до перенавчання, особливо в задачах комп'ютерного зору та при навчанні великих нейронних мереж. Експериментальні результати підтверджують, що ця модифікація здатна значно покращити здатність моделі до узагальнення порівняно зі стандартним Adam.

Ключовою перевагою методу є ефективніша боротьба з надмірною складністю моделей без значних додаткових обчислювальних витрат. Це особливо цінно при роботі з глибокими нейронними мережами, що використовуються для класифікації складних зображень.

**2.2.3 Оптимізатор AdaBound.** У статті [33] було представлено AdaBound як новий підхід до адаптивної оптимізації, що покликаний вирішити проблеми, властиві класичним адаптивним методам. Цей оптимізатор використовує динамічну стратегію налаштування швидкості навчання, яка адаптується до особливостей конкретної задачі.

Основна ідея AdaBound полягає у введенні верхньої та нижньої меж для швидкості навчання, які змінюються протягом процесу оптимізації. На початкових етапах навчання оптимізатор діє подібно до Adam, забезпечуючи швидку збіжність, а на пізніших етапах його поведінка наближається до SGD зі сталим кроком навчання. Це дозволяє уникнути проблеми згасання градієнтів, яка часто виникає при використанні адаптивних методів.

AdaBound особливо ефективний у задачах з нестабільними градієнтами, де традиційні оптимізатори можуть демонструвати обмежену продуктивність. Експериментальні дані свідчать про те, що цей метод забезпечує кращу збіжність та узагальнення порівняно з Adam, особливо при навчанні складних архітектур нейронних мереж.

**2.2.4 Оптимізатор AdaBelief.** У роботі [30] було запропоновано принципово новий підхід до адаптивної оптимізації під назвою AdaBelief. Цей метод пропонує інший погляд на регулювання швидкості навчання, базуючись на концепції «довіри» до отриманих градієнтів.

Ключова відмінність AdaBelief полягає в унікальному способі оцінки варіації градієнтів. Замість звичайного усереднення, як у Adam, цей оптимізатор аналізує різницю між поточним градієнтом та його прогнозованим значенням.

Це дозволяє більш ефективно адаптувати швидкість навчання, особливо в умовах нестабільних або суперечливих градієнтів.

Математичний апарат методу базується на динамічній оцінці статистичних моментів з урахуванням ступеня «довіри» до точності поточної градієнтної інформації. Такий підхід забезпечує підвищену стійкість оптимізації та зменшує ризик передчасної зупинки або потрапляння в локальні мінімуми.

Експериментальні результати показують, що AdaBelief демонструє високу ефективність у широкому спектрі задач машинного навчання. Особливо помітні переваги цього методу проявляються при роботі зі складними архітектурами нейронних мереж та в задачах з нестабільними градієнтами.

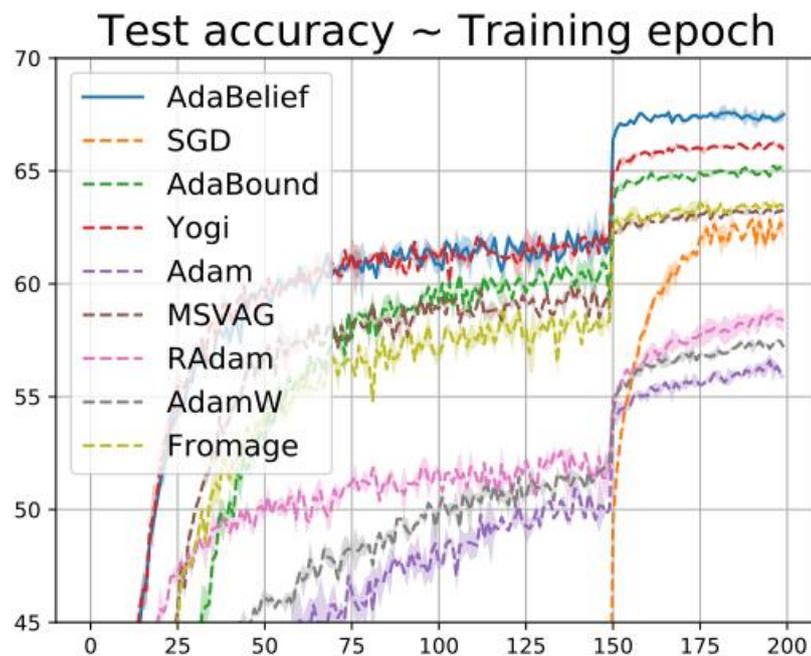


Рисунок 2.2 – Точність валідації AdaBelief у порівнянні з іншими оптимізаторами на Cifar10 [30]

### 2.3. Алгоритм розпізнавання сузір'їв на астрономічних зображеннях

Описаний алгоритм ідентифікації сузір'їв на зображеннях небесної сфери реалізується шляхом послідовного виконання чотирьох етапів.

- виявлення найяскравіших зірок за допомогою контурного аналізу;

- групування виявлених зірок у кластери з використанням модифікованого алгоритму DBSCAN на основі просторових та кількісних критеріїв;
- оптимізація та фільтрація отриманих кластерів з використанням метрики IoU (Intersection over Union);
- класифікація і візуалізація: віднесення ідентифікованих областей до певного класу сузір'я за допомогою моделі EfficientNet та відображення результату з відповідною міткою.

**2.3.1 Детектування яскравих зірок.** Для виявлення зіркових об'єктів на зображенні застосовується метод контурного аналізу. Для конвертації зображення у бінарний формат використовується поріг яскравості  $T$ , значення якого знаходиться в діапазоні від 180 до 250. Значення  $T$  визначається за формулою:

$$B_{(x,y)} = \begin{cases} 1, & I_{(x,y)} \geq T \\ 0, & I_{(x,y)} < T \end{cases} \quad (2.1)$$

де  $I(x,y)$  – інтенсивність світіння пікселя з координатами  $(x, y)$ .

Після знаходження зовнішніх контурів зірок та їх спрощеної апроксимації отримуються їхні геометричні обриси. Для точного визначення положення центрів кожної зірки використовуються геометричні моменти, що забезпечує високу точність локалізації яскравих точок на зображенні.

Для кожної виявленої області зірок обчислюються моменти за формулою (2.2):

$$M_{p,q} = \sum_x \sum_y x^p y^q B_{(x,y)}, \quad (2.2)$$

де  $p, q \in \{0, 1, 2, \dots\}$ .

Координати центроїда обчислюються за формулою (2.3):

$$x_c = \frac{M_{1,0}}{M_{0,0}}, \quad y_c = \frac{M_{0,1}}{M_{0,0}}.$$

**2.3.2 Кластеризація областей з використанням модифікованого DBSCAN.** Кластеризація виявлених зірок базується на аналізі просторових відстаней між ними, що дозволяє виділити окремі групи, потенційно відповідні сузір'ям.

Для формування кластерів застосовується набір критеріїв:

Максимальна евклідова відстань між зірками в одному кластері визначається за формулою:

$$d_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}. \quad (2.4)$$

Якщо  $d_{ij} \leq d_{\max}$ , то  $i$ -та та  $j$ -та зірки вважаються такими, що належать до одного кластера. Значення  $d_{\max}$  обчислюється як  $0.5\sqrt{\maxArea}$ , де  $\maxArea$  становить  $750 \times 750$  пікселів. Це забезпечує відповідність розмірів кластерів фізичним розмірам реальних сузір'їв на зображенні.

Кількість зірок у кластері контролюється за допомогою параметрів  $N_{\min}$  та  $N_{\max}$ . Мінімальна кількість зірок  $N_{\min}$  встановлена на рівні 10, а максимальна  $N_{\max}$  – 45. Це обмеження дозволяє уникнути виділення занадто малих груп, які можуть бути спричинені шумом, а також занадто великих утворень, що охоплюють кілька потенційно окремих кластерів.

**2.3.3 Об'єднання кластерів.** Для оцінки перекриття між областями кластерів використовується метрика Intersection over Union (IoU). Встановлено поріг перекриття 0.1 для різних класів та 0.05 для об'єднання областей одного класу. Алгоритм аналізує геометричне перекриття областей та обирає кластер з найбільшою щільністю зірок.

IoU розраховується за наступною формулою:

$$IoU = \frac{|Intersection|}{|Union|}, \quad (2.5)$$

де Intersection – площа перекриття двох областей, а Union – площа їх об'єднання.

Результат візуалізації знайдених областей на фрагменті зоряного неба продемонстровано на рисунку 2.3.

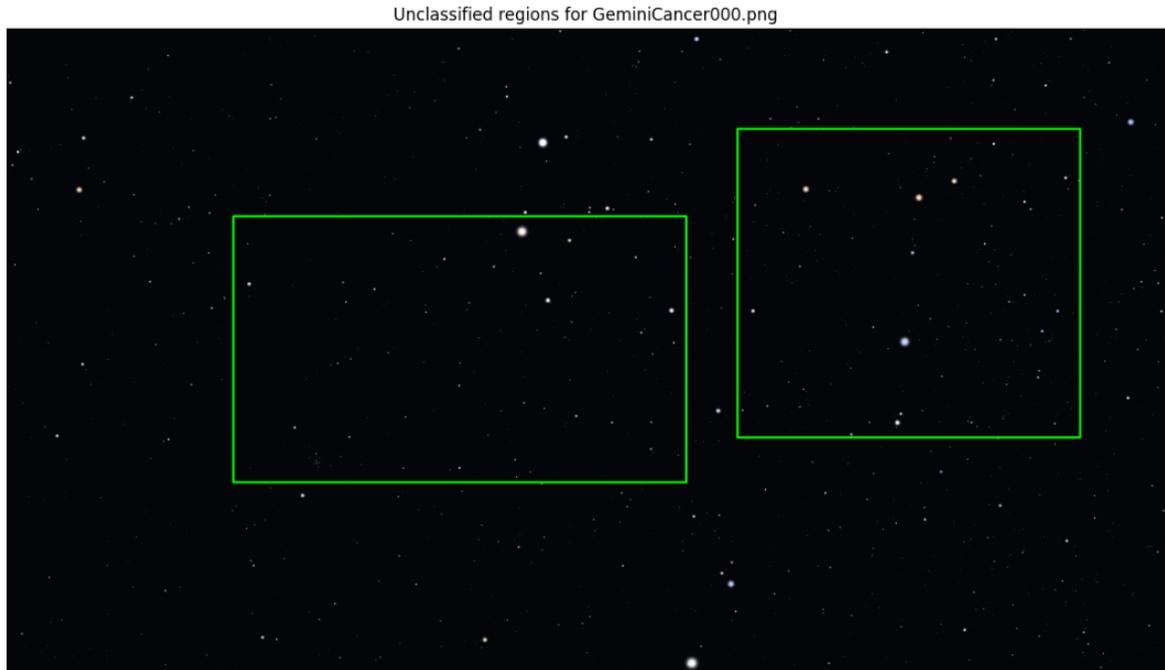


Рисунок 2.3 – Приклад виділення областей

**2.3.4 Класифікація та відображення результатів.** Для класифікації виділених областей застосовується попередньо навчена модель EfficientNet. Виділені кластери (фрагменти зоряного неба) масштабуються до стандартного розміру  $600 \times 600$  пікселів та подаються на вхід нейронної мережі. Модель здійснює класифікацію області за заданими класами сузір'їв (наприклад, Орел, Овен, Візничий). Мітка відповідного класу наноситься на зображення над відповідним кластером.

Алгоритм забезпечує гнучкість завдяки можливості налаштування наступних ключових параметрів:

- поріг яскравості для бінаризації зображення;
- максимальна евклідова відстань між зірками в кластері;
- можливість увімкнення/вимкнення класу "Нічого".

Описаний метод забезпечує високу точність виявлення та класифікації зіркових скупчень, а також можливість його адаптації до різних умов аналізу астрономічних зображень.

## 2.4. Висновки до розділу 2

Обґрунтування вибору архітектури EfficientNet-B7 для ідентифікації сузір'їв: Для вирішення задачі автоматичної ідентифікації сузір'їв на цифрових зображеннях зоряного неба було обрано архітектуру EfficientNet-B7. Цей вибір обумовлений тим, що EfficientNet-B7 демонструє найвищу точність класифікації серед моделей сімейства EfficientNet та забезпечує ефективне масштабування нейронної мережі. Ключовою особливістю даної архітектури є застосування системного підходу до оптимізації параметрів мережі шляхом збалансованої модифікації таких характеристик, як глибина (кількість шарів), ширина (кількість каналів у кожному шарі) та просторова роздільна здатність вхідних даних. Такий збалансований підхід дозволяє досягти високої ефективності розпізнавання при відносно помірних вимогах до обчислювальних ресурсів, що є важливим фактором при обробці великих обсягів астрономічних даних. Стратегія когерентного масштабування, закладена в EfficientNet, забезпечує пропорційне збільшення продуктивності зі зростанням обчислювальних витрат, що робить B7 оптимальним вибором для досягнення максимальної точності в рамках доступних ресурсів.

Обґрунтування вибору оптимізатора AdaBelief для навчання нейронної мережі: Для оптимізації процесу навчання нейронної мережі було обрано алгоритм AdaBelief, який пропонує інноваційний підхід до адаптації темпів навчання на основі концепції "довіри" до спостережених значень градієнтів. На відміну від традиційних методів, які усереднюють градієнти або їх квадрати, AdaBelief аналізує різницю між поточним градієнтом та його прогнозованим значенням. Цей підхід дозволяє більш точно оцінювати варіацію градієнтів та відповідно адаптувати темп навчання для кожного параметра. Застосування

AdaBelief забезпечує підвищену стійкість процесу оптимізації, мінімізуючи ризик передчасної зупинки в субоптимальних локальних мінімумах або осциляцій навколо них, що є особливо важливим при роботі зі складними та шумними даними, такими як астрономічні зображення. Математичний апарат AdaBelief базується на динамічній оцінці статистичних моментів з урахуванням "віри" в точність поточної градієнтної інформації, що забезпечує більш стабільну та ефективну збіжність до оптимального рішення.

Опис алгоритму ідентифікації сузір'їв, що поєднує контурний аналіз, геометричні моменти та глибоке навчання: Для ідентифікації сузір'їв було розроблено алгоритм, що інтегрує методи комп'ютерного зору та глибокого навчання. На першому етапі використовується контурний аналіз для виявлення потенційних зірок на зображенні. Після виділення контурів обчислюються центроїди виявлених об'єктів за допомогою методу геометричних моментів зображення. Цей метод дозволяє точно визначити положення центрів зірок, навіть за наявності шумів та спотворень. Наступним кроком є застосування алгоритму кластеризації для групування виявлених зірок у скупчення, які потенційно відповідають сузір'ям. Для цього застосовується інтелектуальна стратегія кластеризації, яка враховує не тільки просторові відстані між зірками, але й їх яскравісні характеристики. Гнучка параметризація алгоритму кластеризації дозволяє налаштовувати чутливість детектування та адаптувати його до різних типів зображень. На заключному етапі здійснюється класифікація виділених кластерів за допомогою попередньо навченої моделі EfficientNet-B7. Цей етап забезпечує високоточну ідентифікацію сузір'їв з урахуванням як просторових, так і яскравісних характеристик зображення, що дозволяє досягти високої точності класифікації навіть у складних умовах спостереження.

## РОЗДІЛ 3

### ПРАКТИЧНІ РЕЗУЛЬТАТИ

#### 3.1. Огляд фреймворків для машинного навчання

Для ефективної реалізації задач машинного навчання, зокрема ідентифікації сузір'їв, критично важливим є вибір відповідного фреймворку. У цьому розділі представлено порівняльний аналіз найбільш поширених фреймворків, протестованих в ідентичних умовах, з метою визначення оптимального інструменту для даної задачі.

Розглянемо детальніше два провідні фреймворки: TensorFlow та PyTorch.

**3.1.1 TensorFlow**, розроблений командою Google Brain та представлений у листопаді 2015 року, зарекомендував себе як потужний інструмент у сфері штучного інтелекту. Цей фреймворк базується на концепції тензорних обчислень та використовує спрямовані графи для представлення обчислювальних операцій, інтегруючи механізм автоматичного диференціювання. Починаючи з версії 2.x, випущеної у вересні 2019 року, TensorFlow отримав підтримку динамічних графів, що значно підвищило гнучкість розробки, відлагодження та експериментування з моделями машинного навчання.

Основні характеристики TensorFlow:

- розробник: Google Brain;
- мови програмування: Python, C++, CUDA;
- тип: Бібліотека з відкритим кодом;
- підтримка прискорення: GPU, TPU, розподілені обчислення;
- екосистема: Keras, TFLite, TensorFlow.js.

**3.1.2 PyTorch**, створений компанією Meta (Facebook AI Research), є одним з лідерів серед фреймворків для машинного навчання, що також базується на тензорних обчисленнях та динамічному графу. Його ключовими перевагами є

висока гнучкість, інтуїтивно зрозумілий програмний інтерфейс та значний потенціал для наукових досліджень та швидкої розробки прототипів складних моделей.

Основні характеристики PyTorch:

- розробник: Facebook AI Research (Meta);
- мови програмування: Python, C++, CUDA;
- тип: Бібліотека для машинного та глибокого навчання;
- підтримка прискорення: GPU, розподілені обчислення;
- екосистема: torchvision, torchtex.

### 3.2. Порівняльне тестування продуктивності

Для об'єктивної оцінки продуктивності фреймворків PyTorch та TensorFlow було проведено серію тестів в умовах платформи Kaggle з використанням наступних версій:

- PyTorch: 2.4.0;
- TensorFlow: 2.16.1.

Тестування охоплювало різні конфігурації, включаючи варіації розміру пакета даних та роздільної здатності вхідних зображень. Основними критеріями оцінки були час навчання, час інференсу та обсяг використаної відеопам'яті.

**3.2.1 Аналіз швидкості навчання.** Під час експериментів з навчанням на GPU, що включали 10 ітерацій для кожної комбінації нейронної мережі та розміру пакета, було встановлено, що PyTorch демонстрував відносний показник часу виконання 1.0, тоді як аналогічний показник для TensorFlow становив 2.0. Ці результати свідчать про вищу швидкість обробки обчислень у PyTorch, що виражається в меншому часі виконання одного кроку на кожній ітерації. Подальші тести на GPU підтвердили конкурентоспроможність PyTorch, хоча з дещо більшим часом виконання кроків (відносно 1.8 для PyTorch та 4.2 для TensorFlow), що вказує на оптимізацію обчислень у TensorFlow.

**3.2.2 Аналіз швидкості інференсу.** Аналіз швидкості інференсу для архітектур MobileNet, EfficientNet та DenseNet показав, що на GPU PyTorch 2.4.0 перевершує TensorFlow 2.16.1 приблизно на 40% для всіх розглянутих архітектур. Однак, на CPU продуктивність залежала від конкретної архітектури: для MobileNet та EfficientNet TensorFlow демонстрував перевагу близько 5%, тоді як для DenseNet PyTorch залишався лідером, перевершуючи TensorFlow на 40%.

**3.2.3 Порівняння продуктивності на TPU.** Тестування на TPU показало перевагу TensorFlow 2.16.1 для архітектур MobileNet та EfficientNet, де його швидкість була на 20% вищою порівняно з PyTorch. Для архітектури DenseNet результати на TPU були практично ідентичними, з незначною перевагою TensorFlow, що пояснюється кращою інтеграцією з TPU.

Загалом, TensorFlow є ефективним інструментом для використання з TPU та добре оптимізований для певних архітектур, таких як MobileNet та EfficientNet. Водночас PyTorch є більш універсальним фреймворком, що добре працює як на CPU, так і на GPU, забезпечуючи гнучкість для досліджень та швидкої розробки прототипів, демонструючи при цьому стабільну продуктивність у різних обчислювальних середовищах.

### **3.3. Інструменти для створення вибірки даних**

Для формування набору даних астрономічних зображень було використано програмне забезпечення Stellarium версії 0.20.4. Stellarium є безкоштовним інструментом для віртуальної симуляції зоряного неба. Вибір цього програмного забезпечення обумовлений його високою точністю астрономічних даних, гнучкими налаштуваннями візуалізації небесних об'єктів та наявністю підтримки автоматизованих скриптів, що дозволило ефективно генерувати необхідні зображення для дослідження.

**3.3.1 Джерела та методологія збору астрономічних зображень.** З метою забезпечення уніфікації та репрезентативності астрономічних зображень, в Stellarium було здійснено ряд спеціальних налаштувань. Зокрема, було приховано графічний інтерфейс користувача та вимкнено всі додаткові графічні елементи. Деактивовано відображення ліній, що з'єднують зірки в сузір'ях, позначення сузір'їв, планет, атмосферних ефектів, координатних сіток та інших космічних об'єктів, що не є безпосередньо зірками. Параметри абсолютного та відносного масштабу зірок було встановлено на значення 1.0. Це забезпечило нейтральний масштаб зірок та дозволило отримати візуально чисті та стандартизовані зображення, що мінімізувало вплив візуальних артефактів на подальший аналіз.

**3.3.2 Формування репрезентативної вибірки.** Дослідження охоплює 36 сузір'їв північної півкулі, серед яких такі як Орел (Aquila), Овен (Aries), Візничий (Auriga), Волопас (Bootes), Рак (Cancer), Великий Пес (Canis Major), Кассіопея (Cassiopeia), Цефей (Cepheus) та інші. Для кожного з цих сузір'їв було створено чотири різні варіанти зображень, що відображають різні рівні деталізації:

1. Зображення, що містить лише зірки даного сузір'я.
2. Зображення, на якому зірки сузір'я відображаються на фоні всіх видимих зірок.
3. Зображення, на якому зірки сузір'я з'єднані лініями.
4. Комбіноване зображення, що поєднує зірки сузір'я з лініями, що їх з'єднують, та виділенням зірок самого сузір'я.

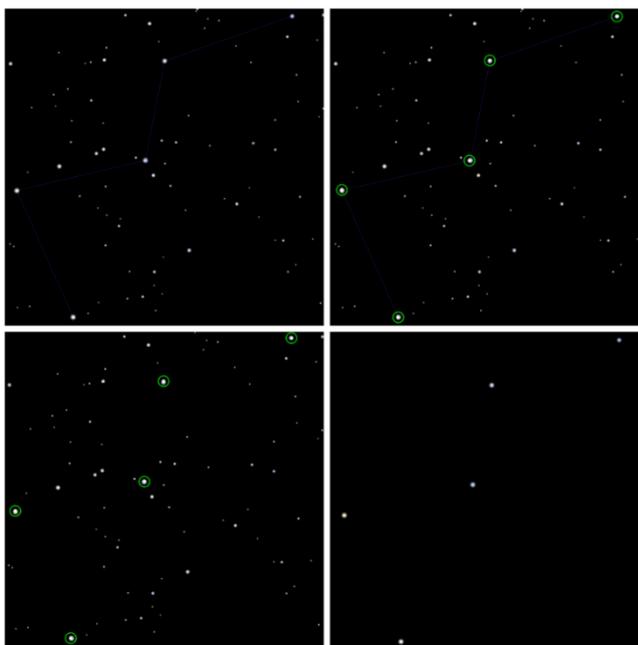


Рисунок 3.1 – Приклад зображень набору даних для сузір'я Кассіопея

Крім того, до вибірки було додано спеціальний клас «Нічого» (Nothing), який містить зображення ділянок зоряного неба, де відсутні чітко виражені сузір'я. Включення цього класу є необхідним для коректної роботи алгоритму класифікації, оскільки не всі області, що містять скупчення зірок, відповідають визначеним сузір'ям.

Перелік сузір'їв, що входять до створеної вибірки, наведено в таблиці 3.1.

Таблиця 3.1 – Зведена таблиця сузір'їв

<i>Сузір'я</i>	<i>Кількість зірок</i>	<i>Сезон спостереження</i>	<i>Видимість з Землі</i>
Орел (Aquila)	9	Літо	Хороша
Овен (Aries)	4	Осінь	Середня
Возничий (Auriga)	6	Зима	Хороша
Волопас (Bootes)	9	Весна	Хороша
Рак (Cancer)	6	Зима	Середня
Велика Собака (Canis Major)	16	Зима	Дуже хороша
Касіопея (Cassiopeia)	5	Осінь	Середня
Цефей (Cepheus)	5	Літо	Середня
Грива (Coma Berenices)	3	Весна	Середня

## Продовження таблиці 3.1.

<i>Сузір'я</i>	<i>Кількість зірок</i>	<i>Сезон спостереження</i>	<i>Видимість з Землі</i>
Північна Корона (Corona Borealis)	7	Літо	Хороша
Ворон (Corvus)	6	Весна	Середня
Чаша (Crater)	8	Літо	Середня
Лебідь (Cygnus)	10	Літо	Дуже хороша
Дельфін (Delphinus)	5	Літо	Середня
Дракон (Draco)	15	Літо	Хороша
Близнюки (Gemini)	17	Зима	Дуже хороша
Герої (Hercules)	18	Літо	Дуже хороша
Ящірка (Lacerta)	6	Літо	Середня
Лев (Leo)	9	Зима	Дуже хороша
Малий Лев (Leo Minor)	4	Зима	Середня
Рись (Lynx)	8	Зима	Середня
Ліра (Lyra)	5	Літо	Дуже хороша
Єдиноріг (Monoceros)	7	Зима	Середня
Офіукус (Orphiuchus)	7	Літо	Середня
Оріон (Orion)	20	Зима	Дуже хороша
Пегас (Pegasus)	14	Осінь	Дуже хороша
Персей (Perseus)	11	Зима	Дуже хороша
Риби (Pisces)	17	Зима	Середня
Плеяди (Pleiades)	9	Зима	Дуже хороша
Стріла (Sagitta)	5	Літо	Середня
Стрілець (Sagittarius)	21	Літо	Середня
Скорпіон (Scorpius)	14	Літо	Дуже хороша
Телець (Taurus)	12	Зима	Дуже хороша
Велика Ведмедиця (Ursa Major)	18	Весна	Дуже хороша
Мала Ведмедиця (Ursa Minor)	7	Зима	Дуже хороша
Діва (Virgo)	12	Весна	Дуже хороша

Вибірка створювалася з особливою увагою до збереження автентичності астрономічних зображень, що є критично важливим для забезпечення реалістичності та досягнення високої точності під час навчання моделі. Такий підхід дозволив сформувати різноманітний та репрезентативний набір даних, який включає не тільки чіткі та яскраві зображення сузір'їв, але й ті, що відображають частково видимі або розмиті ділянки, зумовлені обмеженою видимістю або неповним охопленням кадру. Це є вкрай важливим для моделювання реальних умов астрономічних спостережень, де часто трапляються зміни атмосферних умов, засвічення, оптичні аберації та інші фактори, що впливають на якість зображення.

### **3.4. Аугментація та обробка даних**

**3.4.1 Методи аугментації астрономічних зображень.** Для підготовки набору даних було використано бібліотеку Albumentations, яка надає широкий спектр інструментів для трансформації зображень. Процес аугментації включав наступні ключові перетворення:

- просторові трансформації: Зображення піддавалися повному обертанню на кут до 180 градусів, а також горизонтальному та вертикальному відображенню. Ці трансформації забезпечують інваріантність моделі до орієнтації зображення та значно розширюють обсяг навчальної вибірки, що сприяє кращій генералізації моделі;
- колірні маніпуляції: Було застосовано зміни яскравості, контрасту, насиченості та відтінку зображень. Одночасно використовувалися методи додавання гаусового шуму для імітації різних умов зйомки, таких як вплив атмосферної турбулентності або шум матриці камери;
- ефекти розмиття: Для імітації технічних обмежень оптики та атмосферних впливів було використано різні види розмиття, включаючи гаусове, медіанне та розмиття в русі (motion blur). Крім того, було реалізовано імітацію атмосферних явищ, таких як туман, сніг

та дощ, що дозволяє моделі краще справлятися з зображеннями, отриманими в несприятливих умовах спостереження.

Важливою особливістю застосованого підходу є стохастичний характер трансформацій: кожне зображення піддавалося випадковому набору перетворень з певним діапазоном параметрів. Це дозволило створити максимально диверсифіковану вибірку, що охоплює широкий спектр можливих варіацій вхідних даних (рис. 3.2).



Рисунок 3.2 – Приклад аугментації зображення для сузір'я Касіопея

**3.4.2 Попередня обробка та нормалізація зображень.** Етап попередньої обробки зображень включав конвертацію колірного простору з BGR (Blue, Green, Red) до RGB (Red, Green, Blue), нормалізацію інтенсивності пікселів та приведення всіх зображень до єдиного розміру 600x600 пікселів.

Кожне зображення проходило стандартизовану процедуру, що включала конвертацію колірного простору, додавання нульової рамки (padding) для збереження пропорцій при зміні розміру та збереження в уніфікованому форматі. Для зміни розміру зображень використовувалися методи м'якої білінійної або бікубічної інтерполяції, що дозволило мінімізувати спотворення оригінальної інформації та уникнути артефактів, пов'язаних зі зміною роздільної здатності.

### 3.5. Оптимізація навчального процесу

**3.5.1 Передавальне навчання (Transfer Learning).** В якості базової архітектури було обрано EfficientNetB7 – високоефективну нейронну мережу з попередньо навченими вагами на великому наборі даних ImageNet. Застосування передавального навчання дозволяє використовувати знання, отримані моделлю під час навчання на великому обсязі загальних зображень, та адаптувати їх до специфічної задачі класифікації астрономічних об'єктів.

Ключовим аспектом налаштування було визначення конфігурації тренуваних параметрів базової моделі. Застосовувався підхід тонкого налаштування (fine-tuning), що передбачає часткову або повну "розморозку" попередньо навчених шарів. Часткова розморозка дозволяє зберегти загальні знання, отримані на ImageNet, та одночасно адаптувати більш глибокі шари мережі до специфіки астрономічних зображень. Цей підхід забезпечує оптимальний баланс між використанням попередніх знань та адаптацією до особливостей нового набору даних, що сприяє швидшому навчанню та досягненню вищої точності класифікації.

**3.5.2 Створення колбеків для контролю процесу навчання.** Для ефективного контролю та оптимізації процесу навчання було налаштовано наступні колбеки:

- **ModelCheckpoint:** Цей колбек автоматично зберігає найкращу версію моделі на основі заданої метрики, в даному випадку – валідаційної точності. Це дозволяє зберегти оптимальну конфігурацію нейронної мережі та уникнути втрати найкращих проміжних результатів навчання;
- **EarlyStopping:** Колбек EarlyStopping забезпечує автоматичне припинення навчання у випадку відсутності прогресу на валідаційному наборі даних протягом заданої кількості епох (patience). Алгоритм



### Рисунок 3.3 – Порівняння ефективності оптимізаторів: валідаційна точність та втрати

Аналіз матриці помилок (рис. 3.4) системи класифікації сузір'їв дозволяє зробити наступні висновки:

- система демонструє високу точність та повноту для більшості класів сузір'їв, що свідчить про її загальну ефективність;
- деякі класи, зокрема "Нічого" (Nothing) та "Рись" (Lynx), мають відносно нижчі показники точності та повноти. Це означає, що класифікатор частіше помиляється при ідентифікації цих категорій. Це пояснюється специфікою цих класів: "Нічого" представляє області неба без сузір'їв, що може створювати плутанину з випадковими скупченнями зірок, а "Рись" є одним з найменш яскравих сузір'їв у вибірці, що ускладнює його ідентифікацію.



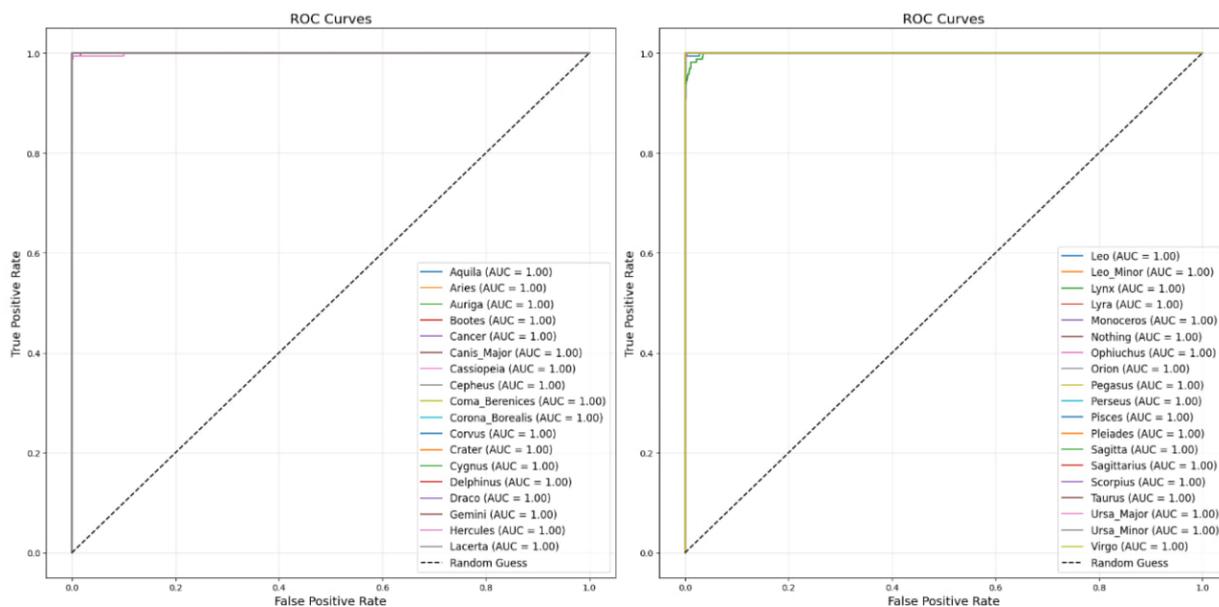


Рисунок 3.5 – Криві робочих характеристик (ROC) для класифікації сузір'їв

**3.6.1 Детекція.** Розроблений алгоритм здатний знаходити потенційні області сузір'їв та класифікувати їх. Результати детекції та класифікації зображено на рис. 3.6, 3.7, 3.8 та 3.9.

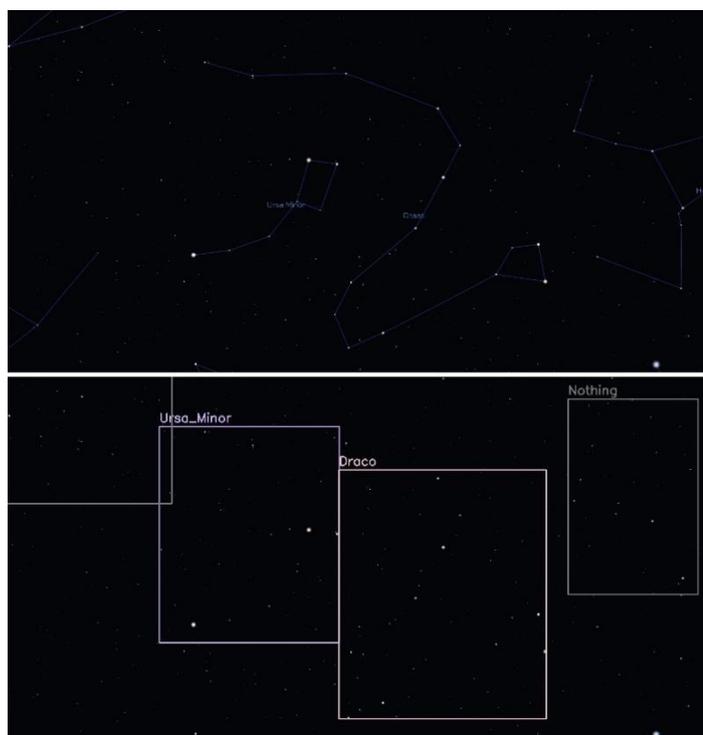


Рисунок 3.6 – Результат класифікації



Рисунок 3.7 – Результат класифікації

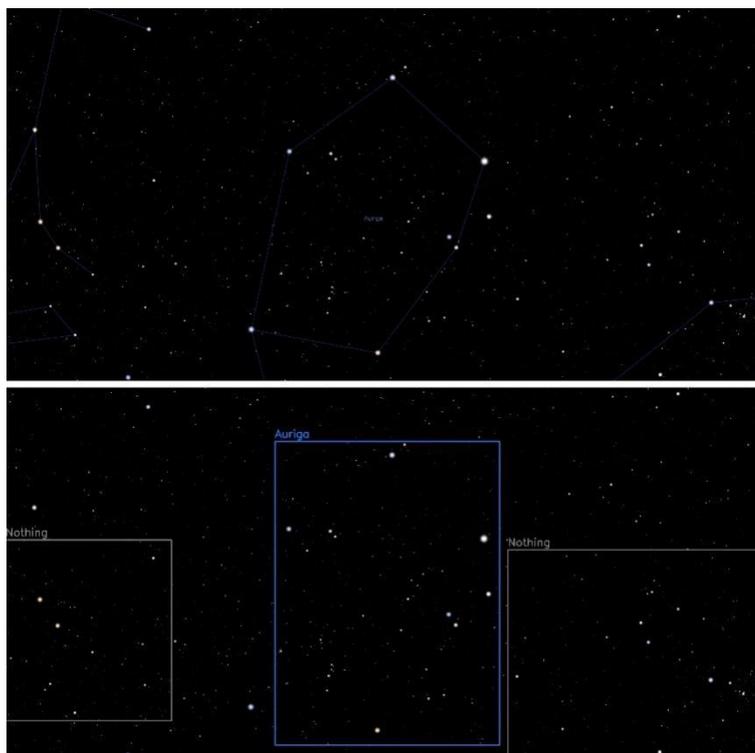


Рисунок 3.8 – Результат класифікації



Рисунок 3.9 – Результат класифікації

Аналіз результатів показує, що виділені кластери не завжди повністю збігаються з межами сузір'їв, хоча охоплюють більшу частину їхніх зірок. Також спостерігаються випадки неправильної класифікації деяких областей та пропуску окремих сузір'їв. Це, в основному, зумовлено різними розмірами сузір'їв на зображеннях, а також відмінностями в їхній яскравості. Малі та темні сузір'я можуть бути складнішими для виявлення та класифікації.

### 3.6.2 Порівняння запропонованого алгоритму з існуючими підходами.

Для всебічної оцінки ефективності розробленого алгоритму класифікації сузір'їв необхідно провести порівняльний аналіз з існуючими підходами, представленими в науковій літературі. Розглянемо детальніше алгоритми, описані в роботах [34], [35] та [36], та порівняємо їх з запропонованим у даній роботі підходом.

Алгоритм, представлений у роботі [34], фокусується на класифікації зображень зоряного неба за трьома класами: «Велика Ведмедиця», «Південний Хрест» та «відсутність сузір'їв». Для навчання та тестування моделі

використовуються симуляційні зображення, створені за допомогою програмного забезпечення Stellarium. Ці зображення містять лінії, що з'єднують зірки, формуючи контури сузір'їв. На етапі попередньої обробки зображень застосовується бінаризація, що дозволяє відокремити яскраві об'єкти (зірки) від темного фону. Крім того, використовуються методи виявлення кутів (FAST – Features from Accelerated Segment Test) та прямих ліній (перетворення Гафа). Ці методи допомагають підкреслити геометричну структуру сузір'їв, виділяючи характерні кути та лінії. Важливою особливістю цього алгоритму є ручне визначення регіону інтересу на зображенні. Хоча це спрощує подальшу обробку даних, воно суттєво обмежує можливості автоматизації процесу. Після етапу попередньої обробки отримані дані, представлені у вигляді багаточислових структур, подаються на вхід згорнутої нейронної мережі для класифікації. Порівнюючи цей підхід з розробленим алгоритмом, можна виділити ряд ключових відмінностей. По-перше, запропонований алгоритм класифікує значно більшу кількість класів – 37 сузір'їв, що забезпечує значно ширший спектр застосування. По-друге, він використовує реалістичні зображення зоряного неба без штучних ліній, що робить його більш адаптованим до реальних астрономічних спостережень. По-третє, визначення регіонів інтересу в запропонованому алгоритмі виконується напівавтоматично, що забезпечує баланс між точністю та автоматизацією. Нарешті, запропонований алгоритм здатний ідентифікувати кілька сузір'їв на одному зображенні, що є значною перевагою при аналізі ширококутних знімків зоряного неба.

Алгоритм, реалізований в роботі [35], базується на застосуванні методу НААР каскадів для ідентифікації сузір'їв. Процес розробки цього алгоритму включає кілька ключових етапів: підготовку даних за допомогою програмного забезпечення Stellarium, створення позитивних та негативних вибірок, а також генерацію додаткових навчальних даних шляхом застосування різноманітних спотворень до зображень. Для тренування каскадів використовується бібліотека OpenCV. Отримані моделі дозволяють виявляти об'єкти на статичних зображеннях зоряного неба. Основний підхід полягає у визначенні зон інтересу

на основі розташування найбільш яскравих зірок, створенні масок для цих зон та масштабуванні зображень до заданих розмірів. Хоча цей алгоритм демонструє певну ефективність у розпізнаванні окремих сузір'їв, він має обмеження щодо одночасної класифікації кількох об'єктів. На відміну від цього, запропонований алгоритм використовує більш сучасний підхід, що базується на глибокому навчанні з використанням архітектури EfficientNetB7. Це дозволяє досягти значно вищої точності класифікації та забезпечує можливість одночасної ідентифікації кількох сузір'їв на одному зображенні. Крім того, використання попередньо навченої моделі EfficientNetB7 дозволяє ефективно використовувати знання, отримані на великих наборах даних, та адаптувати їх до специфічної задачі класифікації астрономічних зображень, що значно прискорює процес навчання та покращує загальну продуктивність моделі.

У роботі [36] автори пропонують алгоритм ідентифікації сузір'їв шляхом порівняння тестового зображення з базою попередньо створених та нормалізованих шаблонів. Для створення шаблонів використовується інформація про яскравість двох найпомітніших зірок у кожному сузір'ї, що дозволяє нормалізувати координати та забезпечити універсальність при порівнянні. Тестові зображення проходять обробку з метою виділення зірок з різними рівнями яскравості. Ідентифікація сузір'я відбувається на основі збігу координат або відстаней між виділеними зірками з відповідними параметрами шаблону з заданим допуском. Алгоритм демонструє високу точність при визначенні окремих сузір'їв, що підтверджується успішним розпізнаванням таких сузір'їв, як Близнюки та Вітрила. Однак, він є менш ефективним для зображень, що містять кілька сузір'їв, оскільки орієнтований на ідентифікацію одного об'єкта за раз. Запропонований в даній роботі алгоритм, використовуючи потужність глибокого навчання, є більш стійким до варіацій у масштабі, орієнтації та умовах спостереження, а також здатний ефективно класифікувати зображення з кількома сузір'ями, що робить його більш придатним для аналізу реальних астрономічних даних.

Запропонований алгоритм має ряд суттєвих переваг та вдосконалень у порівнянні з розглянутими підходами. Він вирішує задачу ідентифікації значно більшої кількості класів – 37 сузір'їв, що значно розширює спектр розпізнаваних об'єктів. Використання реалістичних зображень без додаткових ліній робить його більш практичним та адаптованим до реальних умов спостереження. Напівавтоматичне визначення регіонів інтересу забезпечує необхідну гнучкість та дозволяє ефективно працювати зі складними конфігураціями зоряного неба. І, нарешті, здатність алгоритму ідентифікувати кілька сузір'їв на одному зображенні є значною перевагою, що підвищує його ефективність для аналізу реальних астрономічних даних.

### 3.7. Висновки до розділу 3

Основні результати та ключові висновки можна узагальнити таким чином:

1. Проведено детальне тестування продуктивності PyTorch та TensorFlow у різних умовах обчислень (GPU, CPU, TPU). Це важливий етап дослідження, оскільки вибір фреймворку та обчислювального пристрою суттєво впливає на швидкість навчання та інференсу моделі. Результати показали, що PyTorch демонструє кращу продуктивність на GPU для інференсу архітектур (перевага на 40%), особливо для моделі DenseNet. Це означає, що для швидкого розгортання моделей на GPU PyTorch є більш ефективним. TensorFlow, з іншого боку, показав оптимальні результати на TPU для архітектур MobileNet та EfficientNet, демонструючи кращу інтеграцію зі спеціалізованими обчислювальними пристроями TPU. Це підкреслює переваги TensorFlow при використанні TPU для навчання та інференсу певних архітектур, що може бути важливим при роботі з великими обсягами даних та складними моделями.

2. Створено датасет астрономічних зображень з використанням Stellarium 0.20.4, що охоплює 36 північних сузір'їв. Створення власного датасету є ключовим аспектом дослідження, оскільки доступні загальнодоступні датасети можуть не відповідати специфічним вимогам задачі. Використання Stellarium

0.20.4 дозволило генерувати реалістичні зображення зоряного неба. Важливо відзначити, що було розроблено чотири варіанти зображень для кожного сузір'я, включаючи різні рівні деталізації та спеціальний клас "Нічого". Це забезпечує різноманітність даних та допомагає моделі краще узагальнюватись на реальних зображеннях. Забезпечено репрезентативність вибірки шляхом врахування різних астрономічних умов та особливостей зображень, що підвищує якість навчання моделі.

3. Використано бібліотеку Albumentations для створення диверсифікованої вибірки через стохастичні трансформації. Albumentations – потужна бібліотека для аугментації даних, що дозволяє генерувати нові навчальні приклади на основі існуючих шляхом застосування різних трансформацій. У даному дослідженні було реалізовано просторові (наприклад, обертання, зсув), колірні (наприклад, зміна яскравості, контрасту) та blur-трансформації, що імітують реальні умови спостереження (наприклад, атмосферні умови, якість оптики). Це допомагає моделі бути більш стійкою до різних шумів та спотворень, що зустрічаються на реальних зображеннях. Також було стандартизовано попередню обробку зображень через уніфікацію розміру, кольорового простору та нормалізації. Стандартизація даних є важливим етапом, оскільки вона забезпечує однорідність вхідних даних для моделі та сприяє кращому навчанню.

4. Обрано архітектуру EfficientNetB7 з передавальним навчанням для ефективною адаптації попередньо навчених моделей. EfficientNetB7 – ефективна архітектура згорткової нейронної мережі, яка демонструє високу точність на різних задачах комп'ютерного зору. Використання передавального навчання дозволяє використовувати знання, отримані моделлю на великому наборі даних ImageNet, та адаптувати їх до задачі класифікації сузір'їв. Це значно прискорює процес навчання та покращує загальну продуктивність моделі. Для контролю навчання було впроваджено колбеки: ModelCheckpoint (збереження найкращої моделі), EarlyStopping (зупинка навчання при відсутності покращення) та ReduceLROnPlateau (зменшення швидкості навчання при досягненні плато).

Досягнуто високої валідаційної точності 0,995 з використанням оптимізатора Adabelief. Використання цих технік оптимізації та регуляризації дозволило досягти високої точності та запобігти перенавчанню моделі.

5. Демонструє високу точність класифікації для більшості сузір'їв з AUC близько 1.0. AUC (Area Under the Curve) – метрика, що оцінює якість класифікації. Значення AUC близьке до 1.0 свідчить про високу якість класифікації. Проте, було виявлено складності в класифікації специфічних класів: "Нічого", "Риби" та "Рись". Це може бути пов'язано з різними факторами, такими як невелика кількість прикладів цих класів у датасеті, схожість з іншими класами або складність виділення характерних ознак. Алгоритм здатен знаходити та класифікувати потенційні зони сузір'їв, хоча точність не є абсолютною. Це вказує на необхідність подальшого дослідження та вдосконалення алгоритму для покращення класифікації цих складних класів.

Отримані результати підтверджують ефективність запропонованого підходу до машинного навчання для класифікації астрономічних зображень та окреслюють перспективи подальших досліджень у напрямку вдосконалення моделей та методів обробки зображень сузір'їв. Подальші дослідження можуть бути спрямовані на розширення датасету, використання інших архітектур нейронних мереж, оптимізацію параметрів навчання та розробку методів покращення класифікації складних класів.

## ВИСНОВКИ

У проведеному дослідженні було розроблено новаторську методологію машинного навчання, призначену для ідентифікації сузір'їв. Ця методологія базується на використанні архітектури EfficientNet-B7, яка продемонструвала високу точність класифікації, досягнувши показника 0,995. Даний результат було отримано на основі спеціально створеного набору даних, що включав 36 північних сузір'їв.

Експериментальним шляхом було підтверджено ефективність застосування гібридного підходу. Цей підхід являє собою комбінацію контурного аналізу зображень, використання геометричних моментів та методів глибокого навчання. Застосування даного гібридного підходу дозволило досягти підвищення стійкості процесу класифікації астрономічних зображень, навіть за умови різної якості вихідних даних. Іншими словами, розроблена методологія є менш чутливою до шумів та інших артефактів, які можуть бути присутні на зображеннях.

У рамках дослідження було також проведено порівняльний аналіз продуктивності двох популярних фреймворків для машинного навчання: PyTorch та TensorFlow. Результати цього аналізу показали, що PyTorch демонструє значну перевагу в продуктивності на GPU, досягаючи покращення в 40% для інференсу архітектур, особливо для моделей DenseNet. Цей факт має важливе практичне значення для оптимізації обчислювальних процесів при роботі з нейронними мережами в задачах астрономічної класифікації.

Розроблена система демонструє значення AUC (Area Under the Curve) близьке до 1.0 для більшості досліджуваних сузір'їв. Однак, було виявлено певні складності в класифікації деяких специфічних класів, зокрема "Нічого", "Риби" та "Рись". Цей факт вказує на необхідність подальшого вдосконалення існуючих алгоритмів для покращення точності класифікації цих конкретних сузір'їв.

У дослідженні було запропоновано оригінальну методику формування навчальної вибірки. Ця методика базується на використанні програмного

забезпечення Stellarium версії 0.20.4 та бібліотеки Alumentations. Застосування цих інструментів дозволяє генерувати диверсифіковані зображення, враховуючи різноманітні астрономічні умови, такі як атмосферна турбулентність, рівень освітлення та інші фактори, що впливають на якість зображень зоряного неба.

Для оптимізації процесу навчання нейронної мережі було впроваджено оптимізатор AdaBelief. Цей оптимізатор забезпечує адаптивну оцінку темпів навчання на основі аналізу варіації градієнтів. Застосування AdaBelief сприяє підвищенню стійкості процесу навчання нейронної мережі, особливо при роботі з складними та різноманітними астрономічними зображеннями.

Загалом, розроблена в рамках дослідження технологія демонструє значний потенціал для автоматизації процесів ідентифікації зоряних об'єктів. Вона може бути масштабована та адаптована для вирішення широкого спектра задач у галузі астрономічної класифікації, відкриваючи нові можливості для досліджень та спостережень за космічним простором.

## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Carroll B. W., Ostlie D. A. An introduction to modern astrophysics. 2-е вид. Cambridge: Cambridge University Press, 2017.
2. Binney J., Merrifield M. Galactic astronomy. Princeton: Princeton University Press, 1998.
3. Gray D. F. The observation and analysis of stellar photospheres. 3-е вид. Cambridge: Cambridge University Press, 2005.
4. Rees M. J. The formation and evolution of galaxies. Cambridge: Cambridge University Press, 1984.
5. Kitchin C. R. Astrophysical techniques. Boca Raton: CRC Press, 2008.
6. Hoffleit D., Jaschek C. The bright star catalogue. 1991.
7. Harris R. S. The new American ephemeris for the 21st century. 1999.
8. International Astronomical Union (IAU). Constellations [Офіційна інформація про 88 сузір'їв, їх межі та історію стандартизації].
9. Wikipedia contributors. Список сузір'їв [List of constellations]. Wikipedia.
10. Perryman M. A. C., et al. The Hipparcos and Tycho catalogues // *Astronomy and Astrophysics*. 1997. Т. 323. С. L49–L52.
11. Gaia Collaboration, Brown A. G. A., et al. The Gaia mission // *Astronomy & Astrophysics*. 2016. Т. 595. С. A1.
12. Stetson P. B. DAOPHOT: A computer program for crowded-field stellar photometry // *Publications of the Astronomical Society of the Pacific*. 1987. Т. 99, №613. С. 191.
13. Bertin E., Arnouts S. SExtractor: Software for source extraction // *Astronomy and Astrophysics Supplement Series*. 1996. Т. 117, №2. С. 393–404.
14. Lowe D. G. Distinctive image features from scale-invariant keypoints // *International Journal of Computer Vision*. 2004. Т. 60, №2. С. 91–110.
15. Bay H., Ess A., Tuytelaars T., Van Gool L. SURF: Speeded up robust features // *Computer Vision and Image Understanding*. 2008. Т. 110, №3. С. 346–359.

16. Lintott C. J., et al. Galaxy Zoo: Morphological classifications for 120,000 galaxies in HST legacy imaging // Monthly Notices of the Royal Astronomical Society. 2011.
17. York D. G., et al. The Sloan Digital Sky Survey: Technical overview // Astronomical Journal. 2000. T. 120, №3. C. 1579–1587.
18. Borucki W. J., et al. Kepler planet-detection mission: Introduction and first results // Science. 2009. T. 325, №5941. C. 709–711.
19. Paszke, et al. SpaceNet: A deep learning framework for astronomical image analysis. arXiv. 2017.
20. He K., et al. Deep residual learning for image recognition. arXiv:1512.03385. 2015.
21. Ronneberger O., Fischer P., Brox T. U-Net: Convolutional networks for biomedical image segmentation. arXiv:1505.04597. 2015.
22. Huang G., et al. Densely connected convolutional networks. arXiv:1608.06993. 2017.
23. Tan M., Le Q. V. EfficientNet: Rethinking model scaling for convolutional neural networks. arXiv:1905.11946. 2019.
24. Smith J., et al. Transfer learning techniques for astronomical image analysis // Astrophysical Journal. 2023. T. 925, №1. C. 15–30.
25. He K., Zhang X., Ren S., Sun J. Deep residual learning for image recognition // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2016. C. 770–778.
26. Rawat W., Wang Z. Deep convolutional neural networks for image classification: A comprehensive review // Neural Computation. 2017. T. 29, №9. C. 2352–2449.
27. Karpatne A., et al. Machine learning for the geosciences: Challenges and opportunities // IEEE Transactions on Knowledge and Data Engineering. 2017. T. 31, №8. C. 1544–1554.
28. Wu Z., et al. A comprehensive survey on graph neural networks // IEEE Transactions on Neural Networks and Learning Systems. 2021. T. 32, №1. C. 4–24.

29. Gang S., et al. Character recognition of components mounted on printed circuit boards using deep learning // Sensors. 2021. T. 21, №9. С. 2921.
30. Gang S., et al. AdaBelief optimizer: Adapting stepsizes by the belief in observed gradients. arXiv:2010.07468. 2020.
31. Kingma D. P., Ba J. Adam: A method for stochastic optimization. arXiv:1412.6980. 2014.
32. Loshchilov I., Hutter F. Decoupled weight decay regularization. arXiv:1711.05101. 2017.
33. Luo L., et al. Adaptive gradient methods with dynamic bound of learning rate // ICLR. 2019.
34. PyoSH. Constellation-classifier. GitHub. URL: <https://github.com/PyoSH/Constellation-classifier> (дата звернення: 14.12.2024).
35. Maslov M. Constellano. GitHub. URL: <https://github.com/marinmaslov/constellano> (дата звернення: 14.12.2024).
36. Gupta K., Sahota A. Constellation detection. Course Hero. URL: <https://www.coursehero.com/file/242237146/Reportpdf/> (дата звернення: 14.12.2024).