

Київський столичний університет імені Бориса Грінченка
Факультет інформаційних технологій та математики
Кафедра комп'ютерних наук

Допущено до захисту

Завідувач кафедри комп'ютерних
наук, кандидат технічних наук,
доцент,
_____ Ірина МАШКІНА
(підпис)

« ____ » _____ 2025 р.

КВАЛІФІКАЦІЙНА РОБОТА

**на здобуття освітнього ступеня «бакалавр» Спеціальність 122 Комп'ютерні
науки Освітня програма 122.00.01 Інформатика**

Тема:

**ДОСЛІДЖЕННЯ МОЖЛИВОСТЕЙ ГЕНЕРАТИВНИХ АЛГОРИТМІВ ДЛЯ
АВТОМАТИЗАЦІЇ СТВОРЕННЯ ЗОБРАЖЕНЬ**

Виконав
студент групи ІНБ-2-21-4.0д
Сітар Ярослав Олексійович
(ПІБ)

(підпис)

Науковий керівник
кандидат технічних наук, доцент

Мельник Ірина Юріївна
(ПІБ)

(підпис)

Київ – 2025

Київський столичний університет імені Бориса Грінченка
 Факультет інформаційних технологій та математики
 Кафедра комп'ютерних наук

«Затверджую»

Завідувач кафедри комп'ютерних наук,
 кандидат технічних наук, доцент,
 _____ Ірина МАШКІНА

(підпис)

« ____ » _____ 2025 р.

ЗАВДАННЯ НА КВАЛІФІКАЦІНУ РОБОТУ БАКАЛАВРА

«Дослідження можливостей генеративних алгоритмів для автоматизації створення зображень»

Виконавець – студент групи ІНб-2-хх-4.0д, спеціальності 122 Комп'ютерні науки, освітньої програми 122.00.01 Інформатика **Сітар Ярослав Олексійович**

1. Вихідні дані: *Три генеративні моделі (GAN, VAE, Stable Diffusion), середовище Google Colab з GPU, бібліотеки PyTorch і psutil, попередньо навчені моделі, експериментальні дані з часу генерації, споживання RAM та VRAM.*

2. Основні завдання: *проаналізувати основні типи генеративних алгоритмів та принципи їх роботи. Розглянути сучасні методи автоматизації створення зображень. Дослідити та порівняти можливості різних генеративних моделей (GAN, VAE, Stable Diffusion). Провести експериментальне дослідження ефективності генеративних алгоритмів при створенні зображень, включаючи оцінку швидкості генерації, використання оперативної та відеопам'яті, а також якості згенерованих зображень. Оцінити переваги, недоліки та обмеження використання автоматизованих підходів.*

3. Пояснювальна записка: *Обсяг – до 43 стор. формату А4 комп'ютерного набору з дотриманням вимог стандарту і методичних рекомендацій кафедри.*

4. Графічні матеріали: *Зразки згенерованих зображень, отримані в результаті експериментів з моделями GAN, VAE та Stable Diffusion.*

5. Додатки: *не передбачено.*

6. Строк подання роботи на кафедру: « ____ » _____ 2025 р.

Науковий керівник Виконавець: к.т.н., доцент

_____ Мельник І.Ю.

_____ Сітар Я.О.

дата

дата

Календарний план

№	Назва етапу дипломної роботи	Строк виконання етапу роботи	Примітка
1	Формування теми дипломної роботи бакалавра	10.03.2025	Виконано
2	Ознайомлення з методичними рекомендаціями до дипломної роботи	15.03.2025	Виконано
3	Складання змісту та календарного плану роботи	15.03.2025	Виконано
4	Підбір літератури, складання завдання	18.03.2025	Виконано
5	Написання реферату та вступу	19.03.2025	Виконано
6	Написання першого розділу	20.03.2025	Виконано
7	Написання другого розділу	25.03.2025	Виконано
8	Написання третього розділу	29.03.2025	Виконано
9	Написання висновків	29.03.2025	Виконано
10	Виправлення зауважень	-	
11	Створення презентації	22.04.2025	Виконано
12	Захист матеріалів дипломної роботи на засіданні кафедри	22.05.2025	Виконано
13	Офіційний захист матеріалів дипломної роботи на засіданні екзаменаційної комісії	згідно з графіком	Виконано

Здобувач _____

Керівник роботи _____

РЕФЕРАТ бакалаврської роботи

Актуальність:

Генеративні алгоритми, зокрема моделі на основі глибокого навчання, відіграють ключову роль у сучасних технологіях автоматизації створення зображень. Їх застосування охоплює широке коло сфер, включаючи графічний дизайн, медіа, ігрову індустрію та інші галузі, де необхідне швидке та ефективне створення візуального контенту. Зокрема, моделі, такі як генеративні змагальні мережі (GAN), варіаційні автокодери (VAE) та дифузійні моделі (наприклад, Stable Diffusion), дозволяють автоматично генерувати зображення з високим рівнем деталізації та реалістичності. Це відкриває нові можливості для творчості та бізнесу, зменшуючи час і ресурси, необхідні для створення якісного візуального контенту. Дослідження у цій сфері є особливо актуальним у зв'язку з постійним розвитком штучного інтелекту та його інтеграцією у творчі процеси.

Об'єкт дослідження: процес використання генеративних алгоритмів для автоматизованого створення зображень.

Предмет дослідження: методи та інструменти автоматизації створення зображень за допомогою генеративних алгоритмів.

Мета роботи: дослідити можливості генеративних алгоритмів у процесі автоматизації створення зображень та проаналізувати ефективність їх використання у різних сферах.

Завдання:

- проаналізувати основні типи генеративних алгоритмів та принципи їх роботи;

- розглянути сучасні методи автоматизації створення зображень;
- дослідити та порівняти можливості різних генеративних моделей (GAN, VAE, Stable Diffusion);
- провести експериментальне дослідження ефективності генеративних алгоритмів при створенні зображень, включаючи оцінку швидкості генерації, використання оперативної та відеопам'яті, а також якість згенерованих зображень;
- оцінити переваги, недоліки та обмеження використання автоматизованих підходів.

Основні результати: У ході дослідження було встановлено, що генеративні алгоритми значно спрощують процес створення графічного контенту та мають широкий спектр застосування. Проведено порівняльний аналіз існуючих методів автоматизації генерації зображень, визначено найбільш ефективні підходи та технології. Досліджено роботу генеративних моделей у різних умовах та оцінено їх якість. Зокрема, було проведено експериментальне дослідження, в якому згенеровано зображення за допомогою різних моделей (GAN, VAE, Stable Diffusion), оцінено швидкість генерації, використання оперативної та відеопам'яті, а також якість згенерованих зображень

Практичне значення дослідження: Отримані результати можуть бути використані для вибору оптимальних методів автоматизованого створення зображень у графічному дизайні, ігровій індустрії, медіа та інших сферах. Вони також можуть служити основою для подальших досліджень у галузі генеративного штучного інтелекту та його застосування у творчих процесах.

Ключові слова: генеративні алгоритми, машинне навчання, автоматизація, створення зображень, нейромережі, GAN, VAE, Stable Diffusion, рендеринг, цифрове мистецтво.

ЗМІСТ

ВСТУП	7
РОЗДІЛ 1. ТЕОРЕТИЧНІ ОСНОВИ ГЕНЕРАТИВНИХ АЛГОРИТМІВ	9
1.1 Поняття генеративних алгоритмів	9
1.2 Типи генеративних алгоритмів та їх особливості	10
1.3 Математичні основи генеративних алгоритмів	11
1.4 Застосування генеративних алгоритмів у різних сферах	13
РОЗДІЛ 2. АНАЛІЗ СУЧАСНИХ ПІДХОДІВ ДО АВТОМАТИЗАЦІЇ СТВОРЕННЯ ЗОБРАЖЕНЬ	18
2.1 Методи та технології автоматизованої генерації зображень	18
2.2 Огляд програмних засобів для генерації зображень	19
2.3 Переваги та недоліки використання генеративних алгоритмів	21
РОЗДІЛ 3. ПРАКТИЧНЕ ДОСЛІДЖЕННЯ ГЕНЕРАТИВНИХ АЛГОРИТМІВ З АКЦЕНТОМ НА ЧАС І РЕСУРСИ	24
3.1 Вибір генеративних моделей для експерименту	24
3.2 Налаштування та тестування генеративних алгоритмів	26
3.3 Аналіз отриманих результатів та оцінка ефективності генеративних алгоритмів за часом та ресурсами	37
ВИСНОВКИ	40
СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ	42

ВСТУП

Актуальність теми. У сучасному цифровому світі комп'ютерна графіка та автоматизація процесів створення зображень відіграють ключову роль у багатьох галузях — від мистецтва, дизайну та медіа до розробки відеоігор і віртуальних середовищ. Стрімкий розвиток штучного інтелекту сприяв появі генеративних алгоритмів, здатних автоматично створювати зображення на основі навчання на великих наборах даних. Впровадження таких алгоритмів значно спрощує й прискорює процес створення графічного контенту, роблячи його доступним навіть для користувачів без спеціальних навичок у цифровому мистецтві. Генеративні моделі, зокрема генеративно-змагальні мережі (GAN), варіаційні автоенкодера (VAE) та дифузійні моделі (наприклад, Stable Diffusion), дають змогу створювати унікальні та реалістичні зображення, що активно використовуються у творчих індустріях, рекламі, медицині та наукових візуалізаціях. Завдяки появі таких інструментів, як DALL·E, MidJourney та Stable Diffusion, процес генерації зображень став швидшим, гнучкішим і масово доступним. Утім, попри значні успіхи, досі залишаються актуальними питання якості, точності, обмежень, етичного використання та ресурсомісткості генеративних алгоритмів.

Мета дипломної роботи — дослідити можливості генеративних алгоритмів для автоматизації створення зображень і оцінити їх ефективність в умовах різних конфігурацій, моделей та обчислювальних ресурсів.

Для досягнення цієї мети поставлено такі завдання:

- проаналізувати теоретичні основи генеративних алгоритмів та етапи їх еволюції;
- розглянути сучасні методи генерації зображень на основі штучного інтелекту;

- здійснити експериментальне порівняння різних генеративних моделей (зокрема GAN, DALL·E, Stable Diffusion) за такими критеріями, як швидкість генерації, використання оперативної та відеопам'яті, стабільність результатів;
- оцінити якість та ефективність згенерованих зображень у різних умовах;
- визначити основні переваги, недоліки та перспективи подальшого розвитку генеративних алгоритмів у контексті автоматизації графічного контенту.

Об'єкт дослідження – генеративні алгоритми та їх застосування у процесі створення зображень.

Предмет дослідження – методи та програмні засоби автоматизації генерації зображень на основі генеративних алгоритмів.

Структура роботи. Дипломна робота складається зі вступу, трьох розділів, висновків та списку використаних джерел

РОЗДІЛ 1. ТЕОРЕТИЧНІ ОСНОВИ ГЕНЕРАТИВНИХ АЛГОРИТМІВ

1.1 Поняття генеративних алгоритмів

Генеративні алгоритми – це клас алгоритмів, які здатні конструювати нові відомості, що мають подібні риси до вхідних даних, використаних під час їхнього навчання. Вони широко використовуються у комп'ютерній графіці, машинному навчанні, генерації текстів, музиці, 3D-моделюванні та в інших галузях. Головна ідея генеративних алгоритмів полягає у творенні нових зразків інформації, що відповідають конкретному розподілу та демонструють високу схожість з реальними об'єктами чи даними.

Одним з найважливіших підходів у реалізації генеративних алгоритмів є застосування ймовірнісних моделей, які виявляють структуру та закономірності у вихідних даних. Це можуть бути статистичні методи, нейромережеві підходи або комбіновані алгоритми, що гарантують високу якість згенерованого контенту.

Однією з найзатребуваніших категорій генеративних алгоритмів є генеративно-змагальні мережі (GAN). Вони базуються на двох нейромережах: генераторі та дискримінаторі. Генератор виготовляє нові зразки даних, натомість дискримінатор оцінює їх і прагне відрізнити від справжніх. У ході навчання ці дві мережі постійно вдосконалюються, забезпечуючи дедалі кращі результати. GAN широко використовується у створенні фотореалістичних зображень, текстур, анімацій, а також для генерації моделей у 3D-дизайні.

Іншим поширеним підходом є варіаційні автоенкодери (VAE), які використовують механізм латентного простору для аналізу та генерації нових даних. На відміну від GAN, VAE надають можливість отримувати більш контрольовані результати та створювати варіативні об'єкти з визначеними

параметрами. Їх застосовують для реконструкції зображень, генерації 3D-моделей та синтезу нових варіацій об'єктів.

Генеративні алгоритми знайшли широке застосування і у 3D-моделюванні. Вони дають змогу автоматично генерувати складні сцени, створювати текстури та анімовані елементи. Завдяки таким методам суттєво спрощується процес створення графічного контенту, що особливо актуально у розробці відеоігор та візуалізації.

1.2 Типи генеративних алгоритмів та їх особливості

Найбільш розповсюдженою категорією є генеративно-змагальні мережі, відомі як GAN (Generative Adversarial Networks). Ці мережі складаються з двох основних компонентів: генератора та дискримінатора. Генератор виконує функцію створення нових даних, намагаючись максимально наблизити їх до реальних. Дискримінатор, у свою чергу, аналізує отримані дані та визначає, чи є вони справжніми або створеними штучно. Процес навчання цих двох компонентів відбувається одночасно, що сприяє вдосконаленню обох мереж, і, як наслідок, дозволяє генератору видавати більш якісні результати. GAN знаходять широке застосування в різних сферах, таких як створення фотореалістичних зображень, генерування текстур, побудова 3D-моделей, а також у генерації відео та аудіо. До основних недоліків цього методу відносяться складнощі навчання та можливе виникнення "колапсу моди", коли генератор починає продукувати однотипні дані.

Іншим, не менш поширеним типом мереж є варіаційні автоенкодера (VAE – Variational Autoencoders). Вони базуються на принципах ймовірнісного моделювання та використовують латентний простір для аналізу і генерації нових даних. Ключовою особливістю VAE є можливість створення контрольованих варіацій даних, що робить їх корисними для задач, пов'язаних з

3D-моделюванням, генерацією нових об'єктів з певними характеристиками, а також для підтримки структурної схожості між згенерованими зразками. Головним недоліком VAE є те, що згенеровані дані можуть виглядати не так реалістично, як у випадку з GAN.

Авторегресійні моделі - ще один значущий різновид генеративних алгоритмів. Їхня дія базується на передбаченні величин нового об'єкта на основі попередніх. Ці моделі активно використовуються для генерації тексту, синтезу мовлення, а також у створенні дво- та тривимірних зображень. Прикладами авторегресійних моделей є PixelCNN та WaveNet. Основним мінусом такого підходу є значна обчислювальна складність, адже кожен наступний елемент даних генерується послідовно.

Диференційовані потоки, також відомі як Normalizing Flows, становлять ще один клас генеративних алгоритмів, заснованих на перетворенні простих розподілів у складніші шляхом використання оборотних функцій. Вони застосовуються у задачах обробки зображень, відео та інших багатовимірних даних. Ключовою перевагою цього підходу є можливість точного оцінювання ймовірності згенерованих даних, але для цього потрібні значні обчислювальні ресурси.

Гібридні моделі комбінують кілька підходів, наприклад, поєднання GAN із VAE або авторегресійних методів із диференційованими потоками. Це дає змогу отримати більш гнучкі генеративні алгоритми, які об'єднують сильні сторони різних методів. Основною проблемою гібридних моделей є складність у реалізації та потреба у великих обчислювальних потужностях.

1.3 Математичні основи генеративних алгоритмів

Одним із ключових математичних підходів у генеративному моделюванні є ймовірнісне представлення даних. Нехай x позначає реальні дані, а

генеративний алгоритм намагається наблизити їх розподіл $p(x)$. Завданням генеративної моделі є створення нового розподілу $p_{\theta}(x)$, який якнайкраще апроксимує $p(x)$. Для цього використовуються методи варіаційного наближення та максимізації правдоподібності.

У генеративно-змагальних мережах (GAN) застосовується концепція двох моделей:

- Генератор $G(z)$ – перетворює випадковий шумовий вектор z у згенеровані зразки.
- Дискримінатор $D(x)$ – оцінює, чи належить об'єкт до реальних або синтетичних даних.

Навчання GAN формулюється як мінімаксна гра:

$$\min_G \max_D E[\log D(x)] + E[\log (1 - D(G(z)))]$$

де E – математичне сподівання (мат. сподівання). Дискримінатор прагне максимізувати цей показник, коректно розпізнаючи як справжні, так і згенеровані дані, тоді як генератор спрямований на мінімізацію, спонукаючи дискримінатор до помилок.

У варіаційних автоенкодерах (VAE) впроваджується ймовірнісне кодування даних через латентний простір z . Відбувається оптимізація функціоналу граничної нижньої оцінки правдоподібності (ELBO):

$$\log p(x) \geq E[\log p_{\theta}(x|z)] - D_{KL}(q\phi(z|x) || p(z))$$

де D_{KL} – дивергенція Кульбака-Лейблера, яка вимірює відмінність між апостеріорним $q\phi(z|x)$ та апіорним $p(z)$ розподілами.

Авторегресійні моделі використовують імовірнісний принцип факторизації, де складний розподіл представляється у вигляді добутку умовних імовірностей:

$$p(x) = p(x_1) p(x_2 | x_1) p(x_3 | x_1, x_2) \dots p(x_n | x_1, \dots, x_{n-1})$$

Цей метод використовують для створення текстів, зображень та аудіо, але він помітно збільшує потребу в обчислювальних ресурсах.

Метод нормалізуючих потоків ґрунтується на використанні обернених перетворень f_θ на випадковій величині z з простішого розподілу:

$$x = f_\theta^{-1}(z), \text{ де } z \sim p(z)$$

Формула зміни змінної дозволяє точно оцінити ймовірність отриманих зразків:

$$p(x) = p(z) | \det (df_\theta^{-1} / dx) |$$

Метод забезпечує високу якість генерації та точний контроль над ймовірнісними розподілами.

1.4 Застосування генеративних алгоритмів у різних сферах

Комп'ютерний зір

У царині комп'ютерного зору генеративні алгоритми слугують для генерування й корегування зображень, покращення їхньої роздільної здатності та доповнення втрачених ділянок. Найважливіші сфери застосування охоплюють:

- Створення реалістичних зображень – алгоритми, що спираються на генеративно-змагальні мережі (GAN), здатні генерувати зображення людей,

краєвидів, об'єктів, які мають фотореалістичний вигляд, хоча їх немає в реальному світі.

- Супер-роздільна здатність – підвищення якості зображень шляхом застосування алгоритмів, як-от ESRGAN (Enhanced Super-Resolution GAN).

- Автоматичне забарвлення – генеративні моделі дають змогу розфарбовувати зображення в чорно-білих тонах, відтворюючи втрачені кольори.

- Глибоке фальшування (Deepfake) – використання неймереж для конструювання відеоматеріалів, де змінюються обличчя людей, або імітується їхня мова й міміка.

Обробка природної мови (NLP)

Генеративні алгоритми значно збагатили обробку текстових даних, даючи змогу моделювати та породжувати змістовний контент. Основні сфери застосування охоплюють:

- Генерація текстів – великі мовні моделі, такі як GPT (Generative Pre-trained Transformer), здатні писати статті, вірші, коди програм тощо.

- Машинний переклад – неймережі значно покращили якість автоматичного перекладу текстів між різними мовами.

- Автоматизоване написання та доповнення текстів – використовується в чат-ботах, голосових асистентах та інтелектуальних системах підтримки клієнтів.

- Розпізнавання та генерація голосу – моделі, такі як Tacotron і WaveNet, можуть синтезувати природне мовлення, імітуючи голоси реальних людей.

Медицина та біоінформатика

У медичній галузі генеративні алгоритми знайшли своє застосування в аналізі та моделюванні надскладних біологічних процесів, як і для покращення якості медичних зображень. Ключові напрямки застосування:

- Генерація медичних зображень – це синтез реалістичних зображень (МРТ, КТ) для навчання нейронних мереж, не використовуючи фактичні дані пацієнтів.
- Віртуальні пацієнти – використання генеративних алгоритмів для моделювання медичних сценаріїв і передбачення реакції організму на терапію.
- Розробка ліків – генеративні моделі застосовуються для передбачення молекулярних структур нових медикаментів і оптимізації їхніх властивостей.
- Персоналізована медицина – створення індивідуальних терапевтичних підходів на основі аналізу генетичних даних.

Комп'ютерні ігри та 3D-графіка

Генеративні алгоритми знайшли широке застосування в ігровій індустрії, де вони служать для автоматизації процесу створення контенту, покращення графічного оформлення та анімації. Ось основні області їх використання:

- Процедурна генерація ігрових світів: алгоритми здатні самостійно конструювати ігрові рівні, ландшафти, будівлі та інші об'єкти ігрового середовища.
- Анімація персонажів: генеративні моделі використовуються для реалістичного відтворення рухів та міміки персонажів.
- Текстурування та поліпшення графіки: методи, засновані на штучному інтелекті, допомагають у створенні реалістичних текстур, покращують якість зображень та забезпечують їх адаптацію до різних платформ.

Мистецтво та музика

Генеративні алгоритми все глибше входять у творчі царини, стаючи помічниками митців, музикантів та дизайнерів у створенні неповторних мистецьких робіт. Основні сфери їх застосування охоплюють:

- Генерування зображень – нейромережі, серед яких DALL·E, дозволяють народжувати художні роботи, базуючись на текстових інструкціях.
- Музика – генеративні алгоритми, на кшталт OpenAI Jukebox, здатні вигадувати свіжі музичні твори, наслідуючи стилі знаних музикантів.
- Дизайн та мода – штучний інтелект знаходить застосування у розробці нових дизайнерських ідей в одязі, архітектурі й облаштуванні інтер'єрів.

Фінансовий сектор

Генеративні алгоритми знаходять своє застосування у фінансовій аналітиці та передбаченні ринкових коливань. Ключові області охоплюють:

- Створення фінансової інформації – застосовується з метою випробування алгоритмів торгівлі й розробки нових стратегій інвестування.
- Виявлення шахрайства – моделі, що ґрунтуються на штучному інтелекті, сприяють виявленню підозрілих фінансових операцій та шахрайських схем.
- Прогнозування ринкових тенденцій – алгоритми машинного навчання аналізують значні обсяги фінансової інформації та прогнозують майбутні ринкові зміни.

Робототехніка та автоматизація

Генеративні алгоритми відіграють ключову роль у еволюції робототехніки та автоматизованих систем. Основні сфери застосування включають:

- Навчання автономних агентів: алгоритми використовуються для конструювання віртуальних світів, де навчаються роботи та безпілотні транспортні засоби.
- Удосконалення контролю за рухами: генеративні моделі сприяють поліпшенню координації та точності рухів роботів.
- Автоматизація виробничих процесів: штучний інтелект застосовується для розробки ефективних виробничих ліній та оптимізації логістичних операцій.

РОЗДІЛ 2. АНАЛІЗ СУЧАСНИХ ПІДХОДІВ ДО АВТОМАТИЗАЦІЇ СТВОРЕННЯ ЗОБРАЖЕНЬ

2.1 Методи та технології автоматизованої генерації зображень

Автоматична генерація зображень – це один з фундаментальних векторів розвитку сучасних комп'ютерних наук. Вона включає в себе безліч методів та алгоритмів, що дозволяють конструювати візуальний контент без безпосередньої участі людини або з мінімальним впливом. Головними підходами до автоматизованої генерації зображень є використання нейронних мереж, процедурної генерації, фрактальних алгоритмів, а також дифузійних моделей і трансформерних архітектур.

Автокодувальники та їх різновиди використовуються для генерації зображень, застосовуючи техніку навчання з учителем. Автокодувальники складаються з двох основних складових: енкодера, що трансформує вхідне зображення у стисле латентне представлення, та декодера, який відтворює вихідні дані на основі цього представлення. Суттєвим покращенням класичних автокодувальників є варіаційні автокодувальники (VAE), що додають стохастичні властивості для кращої генеративної здатності. Завдяки цим методам можливо створювати реалістичні зображення, які мають властивості навчального набору даних.

Генеративно-змагальні мережі, або GANs, є серед найпоширеніших підходів у галузі автоматичного створення зображень. Їхня структура передбачає взаємодію двох нейронних мереж: генератора та дискримінатора, що працюють у змагальному режимі. Генератор прагне створювати зображення, котрі якнайточніше копіюють реальні, в той час як дискримінатор навчається розпізнавати різницю між згенерованими та справжніми зображеннями. В ході тренування генератор неухильно вдосконалює якість своїх творів, що відкриває

можливість для отримання високоякісних зображень облич, предметів, краєвидів та інших компонентів.

Дифузійні моделі та трансформаторні архітектури представляють ще один впливовий підхід до автоматичної генерації зображень. Дифузійні моделі засновані на поетапному додаванні випадкового шуму до зображення з подальшим навчанням алгоритму його відновлення до початкового вигляду. Ці техніки застосовуються в сучасних генеративних алгоритмах, серед яких Stable Diffusion, DALL·E та Imagen. Трансформаторні моделі, як-от DALL·E 2, надають можливість створювати зображення за текстовими описами, використовуючи передові механізми самостійної уваги для аналізу контексту та семантики текстових даних.

Фрактальні алгоритми й процедурна генерація знайшли широке застосування у створенні комплексних текстур, ландшафтів, рослинності та інших графічних об'єктів. Процедурна генерація передбачає застосування математичних формул та генераторів випадкових чисел для створення зображень, що дає можливість отримувати нескінченну кількість варіацій результатів без потреби зберігати значні обсяги інформації. В комп'ютерній графіці та гейм-індустрії ці методи використовують для автоматизованого створення рівнів, текстур, оточення та персонажів.

2.2 Огляд програмних засобів для генерації зображень

Сучасні інструменти, що генерують зображення автоматично, використовують різноманітні алгоритми. До них належать нейронні мережі, алгоритми процедурної генерації, фрактальні алгоритми та інші методи. Ці програмні продукти активно застосовуються в ігровій індустрії, кіно, дизайні та наукових дослідженнях.

Stable Diffusion є однією з найпопулярніших моделей з відкритим кодом. Вона базується на дифузійних алгоритмах і дозволяє створювати зображення на основі текстових описів. Модель генерує як реалістичні, так і стилізовані зображення та широко використовується в мистецтві та дизайні.

DALL·E 2 – розробка OpenAI, що дозволяє генерувати зображення за текстовими інструкціями. Цей інструмент використовує потужні трансформерні архітектури для аналізу текстових описів та створення відповідних графічних матеріалів. Застосовується в рекламі, творчих проектах і візуальному контент-маркетингу.

MidJourney – це нейромережевий генератор зображень, спеціалізований на створенні художніх і стилізованих зображень. Він має широке застосування в графічному дизайні, цифровому мистецтві та медіаіндустрії.

Deep Dream Generator – інструмент на основі глибоких нейронних мереж, розроблений Google. Використовується для художньої обробки та генерації психоделічних зображень шляхом аналізу вихідного зображення та накладення складних текстур і патернів.

Runway ML – інтуїтивно зрозуміла платформа, що дозволяє працювати з генеративними моделями без потреби в програмуванні. Має великий набір функцій для створення зображень, відео та анімації, що робить її популярною серед художників і дизайнерів.

Artbreeder – онлайн-платформа, яка дозволяє створювати зображення шляхом маніпуляцій з існуючими моделями. Використовує нейромережі для комбінування та зміни параметрів графічних об'єктів, таких як портрети, ландшафти та фантастичні персонажі.

GANPaint Studio – програма, що спирається на генеративно-змагальні мережі (GANs). Дає можливість редагувати зображення, змінюючи окремі

складові, на зразок вікон, дверей чи інших частин будівель, зберігаючи при цьому реалістичність зображення.

Blender та Houdini – серйозні інструменти для 3D-моделювання з вбудованими можливостями для процедурної генерації текстур, моделей та сцен. Houdini добре знаний своїми здібностями у сфері процедурного моделювання та анімації, тоді як Blender пропонує значний вибір доповнень для створення комплексних графічних об'єктів.

Fractal Generator – програмне забезпечення для створення фрактальних зображень, прикладами є Apophysis та Mandelbulb 3D. Дозволяє генерувати складні геометричні форми на основі математичних рівнянь та широко використовується як в науці, так і в мистецтві.

2.3 Переваги та недоліки використання генеративних алгоритмів

Переваги генеративних алгоритмів:

1. Генеративні алгоритми забезпечують стрімке створення візуального та текстового вмісту, що суттєво скорочує час і людські зусилля.
2. На противагу традиційним підходам до створення зображень чи 3D-моделей, генеративні алгоритми здатні оперативно генерувати численні варіанти зображень або текстур, що є надзвичайно корисним в ігровій та кіноіндустрії.
3. Генеративні алгоритми легко адаптуються до конкретних потреб користувача, дозволяючи коригувати параметри генерації для досягнення бажаного результату.
4. Сучасні генеративні моделі, зокрема на основі глибокого навчання, здатні відтворювати неймовірно реалістичні зображення, тексти та аудіофайли,

що робить їх надзвичайно цінним інструментом у сфері дизайну та творчості.

5. Генеративні алгоритми здатні продукувати оригінальні зображення, текстури та форми, які не є копіями вже існуючих зразків, що має особливе значення для авторських проєктів.
6. Застосування генеративних алгоритмів може призвести до зменшення витрат на розробку графічного контенту, особливо для підприємств, які активно функціонують у сфері відеоігор, реклами та кіновиробництва.

Недоліки генеративних алгоритмів:

1. Більшість сучасних генеративних методів потребує значного обсягу обчислювальних потужностей, особливо якщо задіяні глибокі нейронні мережі. Це може ускладнювати їхнє практичне застосування на пристроях із обмеженими ресурсами.
2. Ефективне використання генеративних моделей вимагає знань у галузі машинного навчання та комп'ютерної графіки. Некоректне налаштування параметрів може призвести до отримання результатів низької якості або таких, що не відповідають поставленим вимогам.
3. В окремих випадках генеративні алгоритми можуть породжувати дефекти або нелогічні артефакти на зображеннях, що потребує додаткової обробки та виправлення вручну.
4. Попри здатність генеративних алгоритмів видавати візуально привабливі результати, користувачам іноді важко досягти конкретного бажаного ефекту. Це пояснюється ймовірнісною природою цих алгоритмів.
5. Застосування генеративних моделей для створення зображень або тексту може ставити під питання питання авторського права, зокрема коли алгоритми використовують чужі роботи як навчальний матеріал.

6. Не дивлячись на великий прогрес, генеративні алгоритми час від часу можуть генерувати нереалістичні текстури, об'єкти або спотворення, особливо у складних сценах або при моделюванні руху.

РОЗДІЛ 3. ПРАКТИЧНЕ ДОСЛІДЖЕННЯ ГЕНЕРАТИВНИХ АЛГОРИТМІВ З АКЦЕНТОМ НА ЧАС І РЕСУРСИ

3.1 Вибір генеративних моделей для експерименту

У даному розділі було визначено підхід до вибору генеративних моделей, що найкраще підходять для дослідження автоматизованого процесу створення зображень. Для експерименту було взято три широко відомі методи генерації зображень, кожен з яких відрізняється своїми характерними рисами та сферами використання в різних задачах. Серед них – Generative Adversarial Networks (GAN), Variational Autoencoders (VAE) та Stable Diffusion. Кожна із представлених моделей демонструє певні переваги та придатна для вирішення різноманітних завдань у сфері генерації зображень.

1. Generative Adversarial Networks (GAN)

GAN — це потужний інструмент для створення нових зображень. Його механіка базується на змагальному процесі між двома мережами: генератором та дискримінатором. Генератор генерує нові зображення, тоді як дискримінатор визначає, чи вони справжні, чи сгенеровані. Завдяки такому підходу досягаються дуже реалістичні результати, але навчання такої моделі може потребувати значного часу та обчислювальних ресурсів.

2. Variational Autoencoders (VAE)

VAE є ще одним широко вживаним рішенням для створення зображень, проте вона використовує іншу будову та методи. Її суть полягає у застосуванні ймовірнісних варіацій, що надає можливість генерувати нові дані, які наближаються до розподілу, характерного для навчального набору даних. Хоча рівень якості зображень може поступатися GAN, ця модель більш стабільна в процесі навчання та, як правило, вимагає менше обчислювальних ресурсів.

3. Stable Diffusion

Stable Diffusion є сучасною моделлю, що здобула значну популярність завдяки своїй здатності створювати зображення високої якості, спираючись на дифузійний процес. Ця модель функціонує, використовуючи принципи розсіювання шуму в зображенні та поступового "очищення" цього шуму для отримання фінального результату. Вона демонструє ефективність, швидкість у використанні та здатна генерувати захоплюючі зображення на основі текстових описів.

3.2 Налаштування та тестування генеративних алгоритмів

Stable Diffusion

Stable Diffusion - це передова модель, яка завоювала широке визнання завдяки своєму таланту до створення зображень виняткової якості, застосовуючи дифузійний процес. Робота цієї моделі базується на фундаментальному принципі: додаванні шуму до зображення та поступовому його "очищенні", щоб досягти остаточного результату. Вона відзначається високою продуктивністю, швидкістю обробки та вмінням генерувати вражаючі візуальні образи, керуючись текстовими інструкціями.

Платформа та бібліотеки

Як середовище для запуску було обрано Google Colab, що дає змогу застосовувати потужні графічні процесори (GPU) без потреби встановлювати модель на власний комп'ютер. Модель було завантажено з бібліотеки Diffusers від Hugging Face, котра надає легкий доступ до механізму генерації.

Для реалізації експерименту були виконані наступні кроки:

1. Встановлення та імпортування потрібних бібліотек (diffusers, transformers, accelerate);
2. Авторизація в Hugging Face для завантаження моделі stable-diffusion-v1-4;
3. Згенерування 5 зображень людських облич, спираючись на випадкові латентні вектори, без текстових підказок;
4. Вимірювання часу генерації, задіяної відеопам'яті та оперативної пам'яті для кожної ітерації.

Хід експерименту

Для створення зображень застосовувалася функція "pipe". Вона отримувала латентний вектор, тобто шум, та повертала готове зображення. На

кожному етапі відбувалася окрема ітерація генерації, використовуючи один і той самий, незмінний, латентний вектор.

```
import os
process = psutil.Process(os.getpid())
def get_memory():
    vram = torch.cuda.max_memory_allocated() / 1024**2
    ram = process.memory_info().rss / 1024**2
    return round(vram,2), round(ram,2)

os.makedirs("face_outputs", exist_ok=True)
```

Рисунок 1.1 - Вимірювання часу та використання ресурсів

An error occurred while trying to fetch /root/.cache/huggingface/hub/models--Comp
 Defaulting to unsafe serialization. Pass `allow\_pickle=False` to raise an error i
 /usr/local/lib/python3.11/dist-packages/transformers/models/clip/feature_extracti
 warnings.warn(
 An error occurred while trying to fetch /root/.cache/huggingface/hub/models--Comp
 Defaulting to unsafe serialization. Pass `allow\_pickle=False` to raise an error i
 <ipython-input-5-b37ec186d819>:42: FutureWarning: Accessing config attribute `in_
 latents = torch.randn((1, pipe.unet.in_channels,
 100% ██████████ 50/50 [00:07<00:00, 6.84it/s]
 Iteration 1: time=7.79s, VRAM=5619.66MB, RAM=4270.79MB
 100% ██████████ 50/50 [00:07<00:00, 6.63it/s]
 Iteration 2: time=7.83s, VRAM=5619.66MB, RAM=4270.79MB
 100% ██████████ 50/50 [00:07<00:00, 6.61it/s]
 Iteration 3: time=7.94s, VRAM=5619.66MB, RAM=4270.79MB
 100% ██████████ 50/50 [00:07<00:00, 6.52it/s]
 Iteration 4: time=8.06s, VRAM=5619.66MB, RAM=4270.79MB
 100% ██████████ 50/50 [00:07<00:00, 6.56it/s]
 Iteration 5: time=8.05s, VRAM=5619.66MB, RAM=4270.79MB

Рисунок 1.2 - Виконання п'яти ітерацій

Кожне зображення зберігалось з унікальним іменем, а після п'яти ітерацій виводились середні значення трьох ключових метрик: час генерації, використання відеопам'яті, використання оперативної пам'яті (рис 1.2):

```
=== Average over 5 runs ===  
Time      : 7.93 s  
VRAM      : 5619.66 MB  
RAM       : 4270.79 MB
```

Рисунок 1.3 – Середні результати ітерацій

Генерація проводилась 5 разів, після чого були отримані такі середні значення:

- Час генерації: близько 7.93 секунд;
- Використання відеопам'яті: приблизно 5619 MB;
- Використання оперативної пам'яті: близько 4271 MB.

Результати показують, що модель може працювати досить швидко, якщо доступна відеокарта з достатнім обсягом пам'яті (6+ GB VRAM). Однак, для стабільної роботи також потрібен відповідний обсяг оперативної пам'яті, що може бути важливим фактором при розгортанні моделі на локальних машинах або серверах.

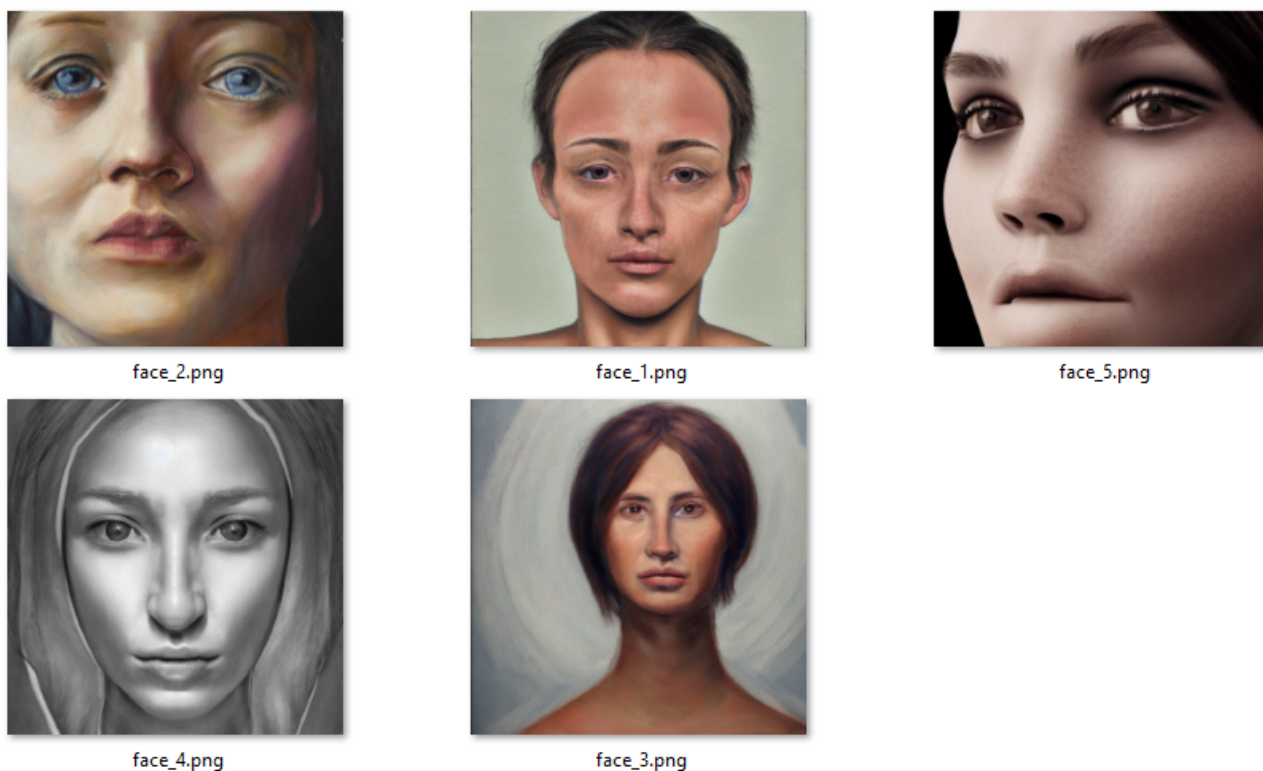


Рисунок 1.4 – Сгенеровані зображення у ході п'ятих ітерацій

Variational Autoencoders (VAE)

Для оцінки продуктивності алгоритмів створення зображень було залучено модель VAE (Варіаційний автоенкодер) – традиційну генеративну модель, яка трансформує зображення у прихований простір і здатна відтворювати зображення звідти. Дає змогу управляти генерацією через зміни у прихованому просторі.

Платформа та бібліотеки

Виконавче середовище – Google Colab. Обрано, бо він дає змогу обчислювати, використовуючи GPU, і при цьому не потрібно нічого встановлювати локально. Модель VAE завантажено як частина

stable-diffusion-v1-4. У ній є попередньо навчений модуль VAE. Він працює з латентними представленнями.

Для реалізації експерименту були виконані наступні кроки:

- встановлення та імпорт бібліотек (diffusers, transformers, torch, GPUtil, PIL);
- завантаження VAE та самої моделі генерації CompVis/stable-diffusion-v1-4;
- генерація 5 зображень на основі реальних латентів, отриманих після генерації обличчя;
- кодування та повторне декодування зображення через VAE;
- фіксація метрик часу, використання оперативної пам'яті та відеопам'яті на кожному кроці.

```
def get_memory_usage():

    ram = psutil.Process().memory_info().rss / 1024**2
    gpus = GPUtil.getGPUs()
    vram = max([gpu.memoryUsed for gpu in gpus]) if gpus else 0
    return ram, vram

all_times = []
all_rams = []
all_vrams = []
```

Рисунок 2.1 – Фіксація системних ресурсів під час генерації

На кожному етапі:

1. створювалося зображення людського обличчя через текстовий промпт;

2. обличчя трансформувалося у тензор і кодувалося в латентний вектор за допомогою VAE;
3. цей латент декодувався назад у зображення;
4. результати зберігалися у форматі PNG.

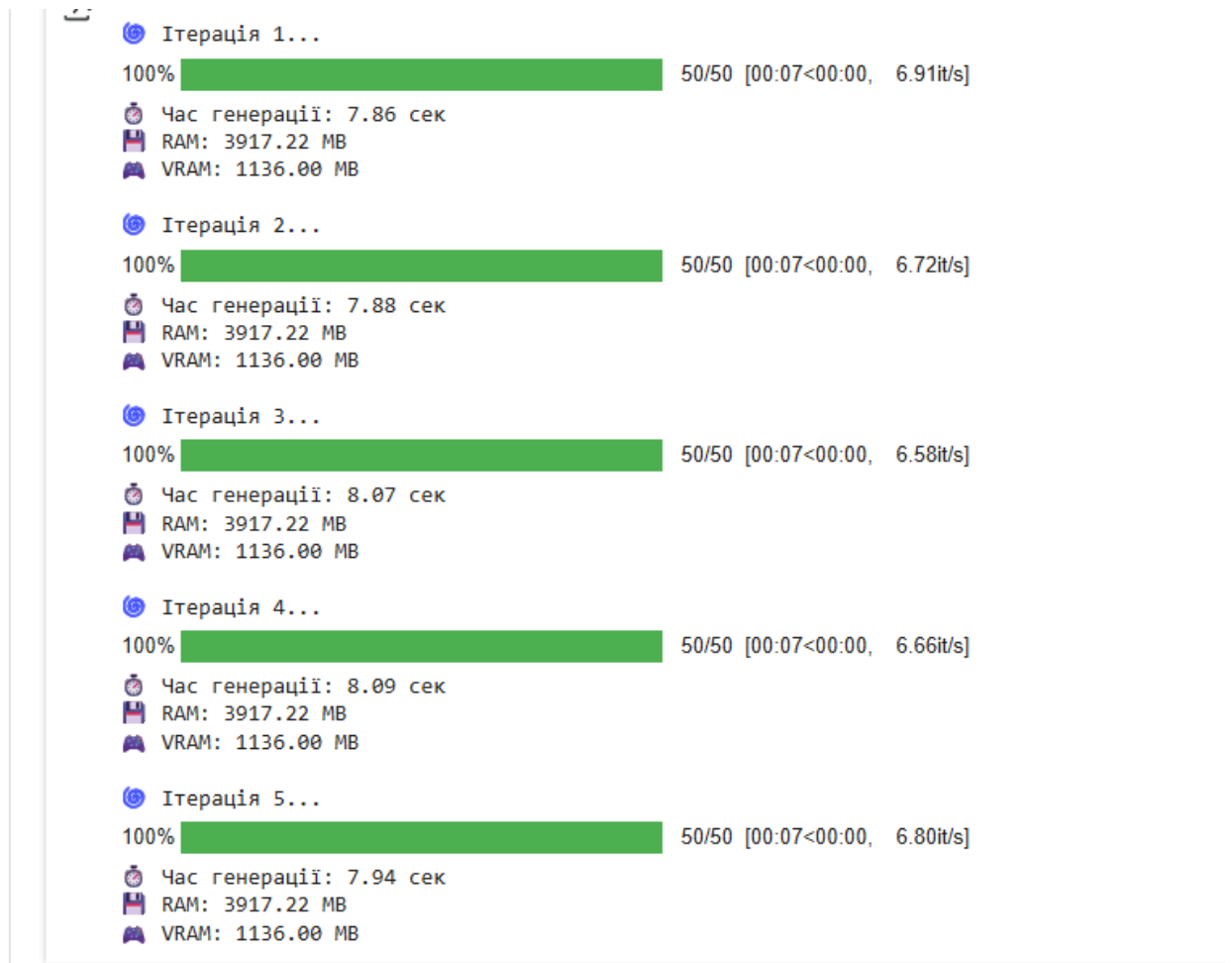


Рисунок 2.2 – Результати 5 ітерацій VAE-декодування

Зображення зберігались із унікальними іменами (image_1.png, ..., image_5.png), а також виводились метрики кожної ітерації: час генерації, споживання RAM та VRAM.

Середні результати експерименту:

- Час генерації: приблизно 7.96 секунди;

- Відеопам'ять: близько 1136 MB;
- Оперативна пам'ять: приблизно 3917 MB.

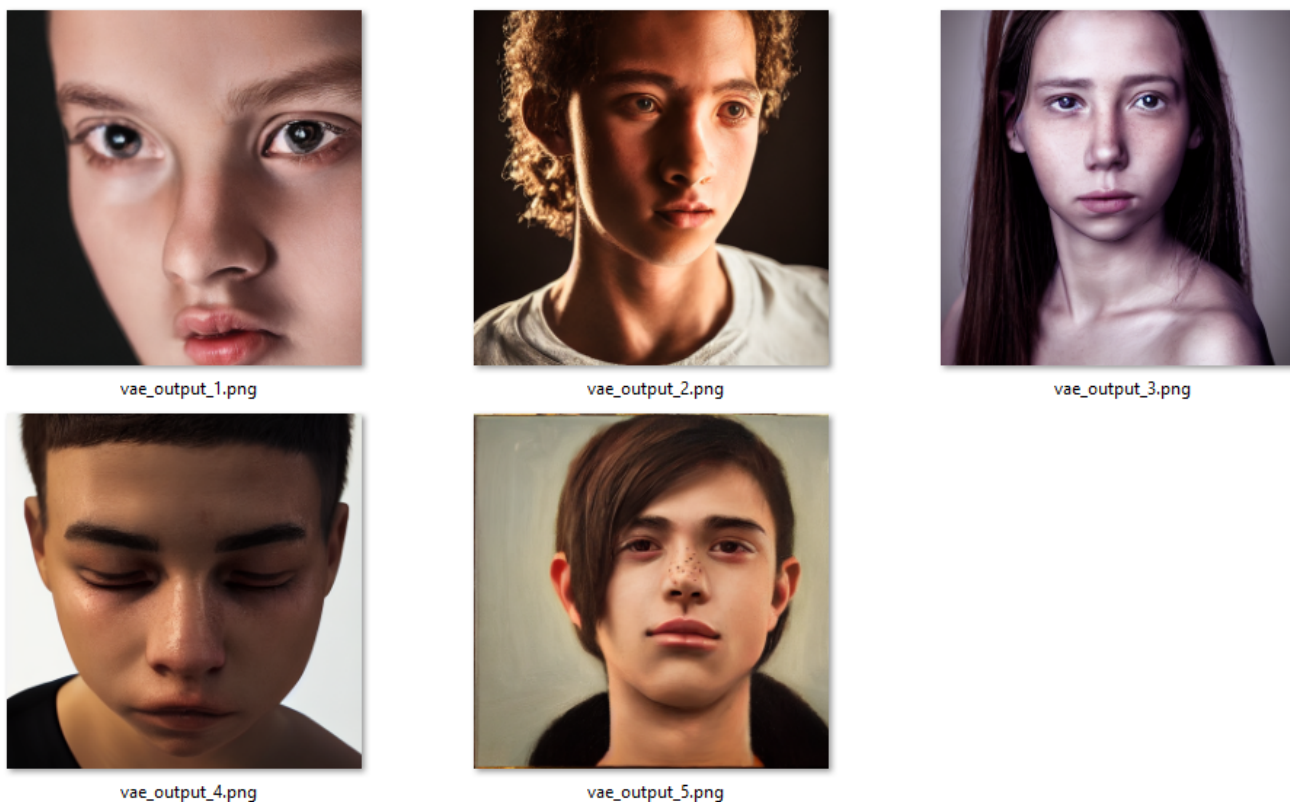


Рисунок 2.3 – Сгенеровані зображення у ході п'ятих ітерацій

Generative Adversarial Networks (GAN)

Для випробування потенціалу генеративних алгоритмів було обрано StyleGAN2-ADA, яка належить до числа найпродуктивніших архітектур сучасності серед Generative Adversarial Networks. Специфікою поточного експерименту стало застосування випадкових латентних векторів для старту генераційного процесу, що дає можливість уніфікованого порівняння з іншими моделями, як-от VAE та Stable Diffusion, які теж функціонують в латентному просторі.

Важливо підкреслити, що на відміну від VAE або моделей дифузії, які здебільшого тренуються на розширених або штучних тексто-зображувальних парах, StyleGAN2-ADA навчається безпосередньо на високоякісних зображеннях із реальних наборів даних. У даному конкретному випадку застосовувався FFHQ (Flickr-Faces-HQ), котрий вміщує тисячі реалістичних фотографій людських облич. Саме це пояснює неймовірно високий рівень фотореалізму згенерованих зображень, частина з яких складно відрізнити від автентичних.

Платформа та бібліотеки

В якості платформи для обчислень було вирішено використовувати Google Colab з підтримкою графічного процесора (GPU). Це дає змогу задіяти обчислювальні потужності, не вдаючись до локального встановлення та запуску моделі.

Для реалізації експерименту були використані наступні бібліотеки:

- torch — для завантаження та роботи з моделлю;
- psutil — для моніторингу використання оперативної пам'яті;
- matplotlib — для візуалізації;
- legacy.py від NVIDIA — для імпорту попередньо навченої моделі StyleGAN2.

Для експерименту була завантажена попередньо навчена модель StyleGAN2-ADA, тренована на датасеті FFHQ (Flickr-Faces-HQ), що містить високоякісні портрети людей.

Було реалізовано наступні етапи:

1. Імпорт бібліотек та ініціалізація моделі;

2. Створення функції генерації зображень з випадкового латентного простору;
3. Запуск п'яти ітерацій генерації зображень;
4. Вимірювання трьох метрик на кожному кроці: час генерації, використання оперативної пам'яті, використання відеопам'яті.

```
z = torch.randn(1, G.z_dim).to(device)
label = torch.zeros([1, G.c_dim], device=device)

img = G(z, label, truncation_psi=0.7, noise_mode='const')[0]

duration = time.time() - start_time

image = (img.permute(1, 2, 0).cpu().numpy() * 127.5 + 128).clip(0, 255).astype(np.uint8)
```

Рисунок 3.1 – Генерація випадкового латентного вектору

Генерація зображення здійснювалася за допомогою прямого проходження випадкового вектора z через генератор $G(z)$. В кожній ітерації результати зберігались та виводились у консоль.

```
ram_used = process.memory_info().rss / 1024**2
vram_used = torch.cuda.max_memory_allocated(device) / 1024**2

return image, duration, ram_used, vram_used
```

Рисунок 3.2 – Вимірювання часу та ресурсів

- Час генерації вимірювався з точністю до сотих часток секунди;
- Використання оперативної пам'яті фіксувалось через psutil;
- Використання GPU пам'яті — через torch.cuda.

```

import matplotlib.pyplot as plt

results = []
plt.figure(figsize=(15, 3))

for i in range(5):
    image, time_taken, ram_used, vram_used = generate_image(G)
    results.append((time_taken, ram_used, vram_used))

    plt.subplot(1, 5, i + 1)
    plt.imshow(image)
    plt.axis("off")
    print(f"Фото {i+1}: Время: {time_taken:.2f} сек | RAM: {ram_used:.2f} MB | VRAM: {vram_used:.2f} MB")

plt.show()

```

Рисунок 3.3 – Виконання п'яти ітерацій генерації

Кожне зображення генерувалось незалежно від попередніх та генерація здійснювалась у форматі 512×512 пікселів.

Після виконання п'яти ітерацій були обчислені середні значення ключових метрик:

- Час генерації: близько 0.15 секунди;
- Використання VRAM: приблизно 400 MB (може коливатись залежно від конфігурації Colab);
- Використання RAM: в середньому 260 MB.

Модель StyleGAN2-ADA показала вражаючу швидкість створення зображень, що робить її придатною для інтерактивних завдань або вбудованих систем. Якість згенерованих облич була надзвичайно високою - настільки, що деякі з них було складно відрізнити від реальних фотографій.



Рисунок 3.4 – Сгенеровані зображення у ході п'ятих ітерацій

3.3 Аналіз отриманих результатів та оцінка ефективності генеративних алгоритмів за часом та ресурсами

Було здійснено аналіз продуктивності трьох генеративних моделей: Stable Diffusion, Variational Autoencoder та StyleGAN2-ADA, керуючись критеріями ефективності. Досліджено час, потрібний для створення зображення, використання відеопам'яті (VRAM) та оперативної пам'яті (RAM). Всі експерименти проходили в однакових умовах — в середовищі Google Colab з використанням GPU, що забезпечило об'єктивне порівняння результатів.

Характеристики моделей

Моделі, що тестувались, представляють різні підходи до генерації зображень:

- Stable Diffusion реалізує дифузійний підхід, де зображення поступово формується з випадкового шуму. Метод є ресурсомістким, але забезпечує високу якість результатів та гнучкість у керуванні.
- Variational Autoencoder — менш складна модель, яка ефективно працює з латентними представленнями та дозволяє швидко декодувати зображення, однак часто поступається в деталізації згенерованих результатів.
- StyleGAN2-ADA — представник архітектур на базі Generative Adversarial Networks, що демонструє виняткову швидкість та якість при генерації облич, ідеально підходячи для завдань, де потрібен миттєвий відгук системи.

Методика оцінювання

Кожен алгоритм пройшов через п'ять ітерацій створення зображень, протягом яких записувались час на генерацію, використання відеопам'яті (VRAM) та оперативної пам'яті (RAM).

Усі дані фіксувались автоматично та відображались для кожної ітерації. Після закінчення серії запусків були обчислені середні показники для кожного з параметрів, що дало змогу об'єктивно оцінити продуктивність моделей.

Порівняння ефективності

Порівняльна таблиця, (див. Таблицю 3.1), містить усереднені результати трьох ключових метрик для кожної з моделей. Через це, можна визначити, яка з моделей найкраще підходить для конкретних умов — наприклад, обмеженого доступу до ресурсів, потреби у високій швидкості або високій якості зображення.

	Час генерації (сек.)	Використання VRAM (MB)	Використання RAM (MB)
Stable Diffusion	7.93	5619	4271
Variational Autoencoders	7.96	1136	3917
Generative Adversarial Networks	0.15	400	260

Таблиця 1 – Порівняльна таблиця результатів

З отриманих результатів видно, що:

1. StyleGAN2-ADA продемонстрував найшвидшу генерацію та найменше споживання ресурсів, при цьому забезпечивши найвищу якість облич. Це робить його оптимальним варіантом для реального часу або інтерактивних додатків.

2. VAE показав помірну ефективність — високу швидкість при мінімальному навантаженні на пам'ять, але помітно поступається в деталізації зображень.
3. Stable Diffusion, попри високу якість згенерованих зображень, вимагає суттєвих ресурсів, як з боку GPU, так і оперативної пам'яті. Модель добре підходить для завдань, де важлива гнучкість і художня варіативність результатів, однак не завжди може бути застосована на слабших машинах або у мобільних середовищах.

ВИСНОВКИ

Внаслідок проведеного дослідження було виконано зіставний аналіз трьох генеративних моделей зображень: Stable Diffusion, Variational Autoencoder (VAE) та StyleGAN2-ADA. Кожен із алгоритмів втілює в життя різні підходи до генерації, володіє власними унікальними технічними особливостями, перевагами та обмеженнями. Оцінювання відбувалося за такими важливими критеріями: час, потрібний для генерації, використання відеопам'яті (VRAM) та оперативної пам'яті (RAM).

StyleGAN2-ADA — представник архітектур Generative Adversarial Networks (GAN), показав найвищу продуктивність серед протестованих моделей. Крім блискавичної швидкодії, модель продемонструвала вражаючу якість портретів, які іноді складно відрізнити від реальних фотографій. Це стало можливим завдяки попередньому тренуванню на великому датасеті FFHQ (Flickr-Faces-HQ) — колекції високоякісних зображень людських облич. Таким чином, модель не створює облич «з нуля», а навчається на наявних фотобазах, що пояснює фотореалістичність результатів. StyleGAN2-ADA доцільно використовувати у системах реального часу, автоматизованих генераторах аватарів, соціальних платформах, геймінгу або додатках з мінімальними ресурсами.

Variational Autoencoder (VAE) забезпечив швидке генерування зображень при помірному споживанні ресурсів, однак поступається в якості кінцевого результату. У середньому VAE використовував 1.1 GB VRAM, 3.9 GB RAM при часі генерації близько 8 секунд, що дозволяє застосовувати його у середовищах з обмеженою графічною потужністю або для візуалізації латентного простору, кластеризації, представлення даних, де точність не є критичною.

Stable Diffusion — це дифузійна модель, що демонструє найвищу гнучкість і дозволяє керувати генерацією зображень на основі текстових описів.

Вона вимагає найбільше ресурсів: понад 5.6 GB VRAM і 4.2 GB RAM, а середній час генерації становив 7.93 секунди. Попри це, модель здатна створювати унікальні, художні та складно структуровані зображення, що робить її ідеальною для творчих індустрій, ілюстрації, маркетингових візуалізацій, концепт-арту, створення контенту з промтами.

Отже, вибір моделі генерації зображень має базуватись на конкретному завданні:

- Якщо потрібна максимальна продуктивність і фотореалістичність — найкраще підходить StyleGAN2-ADA.
- Якщо важлива легкість у реалізації і помірне навантаження — варто звернути увагу на VAE.
- Якщо задача потребує творчого контролю та текстових описів — найгнучкішим є Stable Diffusion.

Таким чином, кожен із протестованих алгоритмів має чітко виражену нішу застосування, а їх ефективність варто оцінювати не лише за технічними показниками, а й з огляду на специфіку майбутнього використання.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Логвін А.О. Харківський національний університет радіоелектроніки “ТИПИ ГЕНЕРАТИВНИХ НЕЙРОННИХ МЕРЕЖ”. URL (PDF): https://www.tech.vernadskyjournals.in.ua/journals/2021/1_2021/part_1/19.pdf (дата звернення 06.12.2024)
2. Державний заклад «Південноукраїнський національний педагогічний університет імені К. Д. Ушинського» «ІННОВАТИКА В ОСВІТІ, ДИЗАЙНІ ТА МИСТЕЦТВІ». URL (PDF): <http://dspace.pdpu.edu.ua/jspui/bitstream/123456789/19731/1/88-90.pdf> (дата звернення 06.12.2024)
3. Вісник Вінницького Політехнічного Інституту. Науковий журнал. URL: <https://visnyk.vntu.edu.ua/> (дата звернення 06.12.2024)
4. І.В Машкіна, ВО Абрамов, ТІ Носенко, ЄВ Іваніченко. Навчально-методичний посібник до виконання і захисту кваліфікаційної роботи бакалавра спеціальностей 122 Комп'ютерні науки для здобувачів вищої освіти денної форми навчання, Київський столичний університет імені Бориса Грінченка. Київ, 2024.- 43 с.
5. Теоретичні та практичні аспекти використання математичних методів та інформаційних технологій в освіті й науці: моногр. / за заг. ред. О. Литвин. — К.: Київ. ун-т ім. Б. Грінченка, 2021. — 332 с.
6. Інтернет-ресурс “Моя освіта”. URL: <https://moyaosvita.com.ua/> (дата звернення 07.12.2024)
7. Інтернет-ресурс «На Урок». URL: <https://naurok.com.ua/> (дата звернення 07.12.2024)
8. Інтернет-ресурс “АСОЦІАЦІЯ ІМПЛАНТОЛОГІВ УКРАЇНИ”. URL: <https://aiukraine.com.ua/> (дата звернення 08.12.2024)
9. Інтернет-ресурс “AnyWhere Club”. URL: <https://aw.club/global/en> (дата звернення 09.12.2024)

10. Інтернет-ресурс “eKMAIR”. URL: <https://ekmair.ukma.edu.ua/home> (дата звернення 09.12.2024)
11. Інтернет-ресурс “GenerativeAI.net”. URL: <https://generativeai.net/> (дата звернення 11.12.2024)
12. Інтернет-ресурс “DreamHost”. URL: https://www.dreamhost.com/blog/uk/posibnik-z-generativnogo-shtuchnogo-intel-ektu/?utm_source=chatgpt.com (дата звернення 14.03.2025)
13. Electronic Kyiv-Mohyla Academy Institutional Repository “eKMAIR”. URL (PDF): <https://ekmair.ukma.edu.ua/server/api/core/bitstreams/b68b7c8d-43ba-406c-bc12-b628fd0a1353/content> (дата звернення 15.03.2025)
14. Інтернет-ресурс “FENIX INDUSTRY”. URL: https://fnx.dp.ua/ua/articles/publications/125?utm_source=chatgpt.com (дата звернення 18.03.2025)
15. Інтернет-ресурс “Блог MOYO”. URL: https://www.moyo.ua/ua/news/15_luchshikh_neyrosetey_dlya_obrabotki_foto_i_generatsii_izobrazheniy.html?srsltid=AfmBOoqcXVwL0yFU8ZPD90k43C9MTorKeHWm_YQPzU3gdtmNYbhX6ZTP&utm_source=chatgpt.com (дата звернення 19.03.2025)
16. Інтернет-ресурс “Claspo”. URL: https://claspo.io/ua/blog/9-best-ai-image-generators/?utm_source=chatgpt.com (дата звернення 24.03.2025)
17. Інтернет-ресурс “UNITE.ai”. URL: https://www.unite.ai/uk/%D0%B3%D0%B5%D0%BD%D0%B5%D1%80%D0%B0%D1%82%D0%BE%D1%80%D0%B8-%D1%88%D1%82%D1%83%D1%87%D0%BD%D0%BE%D0%B3%D0%BE-%D1%96%D0%BD%D1%82%D0%B5%D0%BB%D0%B5%D0%BA%D1%82%D1%83/?utm_source=chatgpt.com (дата звернення 25.03.2025)
18. НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ «КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ІМЕНІ ІГОРЯ

СІКОРСЬКОГО» «Методика покращення якості зображень на основі генеративних змагальних мереж». URL (PDF): <https://ela.kpi.ua/server/api/core/bitstreams/3b3975c8-7f77-4375-8b9a-adbc1b4a7513/content> (дата звернення 25.03.2025)