Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет Комп'ютерних наук
(повна назва)
Кафедра Штучного інтелекту
(повна назва)

КВАЛІФІКАЦІЙНА РОБОТА
Пояснювальна записка

рівень вищої освіти перший (бакалаврський)

Веб-орієнтована система для інтелектуального міксування вокалу та інструментального супроводу
(тема)

Виконав:
здобувач четвертого року навчання,
групи ІТШІ-21-1
Андрій Вороной
(власне ім'я, прізвище)
Спеціальність 122 Комп'ютерні науки
(код і повна назва спеціальності)
Тип програми освітньо-професійна
Освітня програма Штучний інтелект
(повна назва освітньої програми)
Керівник ас. Максим Поліїт
(посада, власне ім'я, прізвище)

Допускається до захисту

Завідувач кафедри ШІ _____ Олег ЗОЛОТУХІН
(підпис) (власне ім'я, прізвище)

2025 р.

Харківський національний університет радіоелектроніки

Факультет Комп'ютерних наук

Кафедра Штучного інтелекту

Рівень вищої освіти перший (бакалаврський)

Спеціальність 122 Комп'ютерні науки
(код і повна назва)

Тип програми освітньо-професійна

Освітня програма Штучний інтелект
(повна назва)

ЗАТВЕРДЖУЮ:
Зав. кафедри _____
(підпис)
«_____» _____ 20 ___ р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ

здобувачеві Вороному Андрію Станіславовичу
(прізвище, ім'я, по батькові)

1. Тема роботи Веб-орієнтована система для інтелектуального міксування вокалу та інструментального супроводу

затверджена наказом університету від 19 травня 20 25 р. № 265Ст
2. Термін подання студентом роботи до екзаменаційної комісії 24 червня 20 25 р.
3. Вихідні дані до роботи Науково-технічні публікації, дані Інтернет-джерел та відомих наукових проєктів, Spleeter документація, Librosa документація

4. Перелік питань, що потрібно опрацювати в роботі
1) Аналіз та постановка задачі
2) Огляд існуючих рішень
3) Розробка веб-орієнтованої системи
4) Демонстрація роботи веб-системи

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Строк / терміни виконання етапів роботи	Примітка
1	Отримання завдання на кваліфікаційну роботу	19.05.2025	виконано
2	Аналіз предметної галузі	23.05.2025	виконано
3	Дослідження цифрової обробки звуку	25.05.2025	виконано
4	Огляд існуючих аналогів	29.05.2025	виконано
5	Визначення основних вимог та обмежень до	30.05.2025	виконано
	веб-орієнтованої системи		
6	Порівняння методів реалізації відокремлення	03.06.2025	виконано
	вокалу		
7	Опис архітектури веб-орієнтованої системи	04.06.2025	виконано
8	Практична реалізація застосунку	07.06.2025	виконано
9	Написання пояснювальної записки	10.06.2025	виконано
10	Перевірка на академічний плагіат	12.06.2025	виконано
11	Нормоконтроль	13.06.2025	виконано
12	Підготовка презентації та доповіді	15.06.2025	виконано
13	Попередній захист	17.06.2025	виконано
14	Рецензування	18.06.2025	виконано
15	Захист перед ЕК	24.06.2025	

Дата видачі завдання 19 травня 2025 р.

Здобувач _____
(підпис)

Керівник роботи _____ ас. Максим Поліїт
(підпис) (посада, власне ім’я, прізвище)

РЕФЕРАТ

Пояснювальна записка: 72 с., 21 рис., 1 табл., 1 дод., 22 джерела.

АВТОМАТИЧНЕ МІКСУВАННЯ МУЗИКИ, ВІДОКРЕМЛЕННЯ ВОКАЛУ, РОЗДІЛЕННЯ ДЖЕРЕЛ ЗВУКУ, СИНХРОНІЗАЦІЯ ТЕМПУ, СПЕКТРАЛЬНИЙ АНАЛІЗ, LIBROSA, SPLEETER.

Об’єкт дослідження – веб-орієнтована система для інтелектуального міксування вокалу та інструментального супроводу.

Предмет дослідження – нейронні мережі для інтелектуального розділення пісень на складові частини, які складаються з вокалу та акомпанементу, технології та методи попередньої обробки пісень та їх міксування, технології створення та архітектура веб-орієнтованої інтелектуальної системи.

Мета роботи – розробка веб-системи, яка має можливість завантаження пісень та їх інтелектуального міксування із застосуванням моделей нейронних мереж, з подальшою можливістю завантаження результату на пристрій.

Методи дослідження – теоретичний (збір та структуризація теоретичного матеріалу), експериментальний (програмна реалізація веб-застосунку). Методи розробки базуються на мовах програмування Python та технологіях Spleeter, Librosa.

У ході виконання кваліфікаційної роботи розроблено веб-орієнтовану систему для інтелектуального міксування вокалу та інструментального супроводу двох пісень, які завантажує користувач із подальшої можливістю завантаження результату міксування на пристрій.

ABSTRACT

Bachelor’s thesis contains: 72 pp., 21 fig., 1 tabl., 1 ann., 22 references.

AUDIO SOURCE SEPARATION, AUTOMATIC MUSIC MIXING, LIBROSA, SPECTRAL ANALYSIS, SPLEETER, TEMPO SYNCHRONIZATION, VOCAL SEPARATION.

The object of research is a web-based system for intelligent mixing of vocals and instrumental accompaniment.

The subject of the research is neural networks for intelligent separation of songs into components consisting of vocals and accompaniment, technologies and methods of pre-processing songs and their mixing, technologies for creating and architecture of a web-based intelligent system.

The goal is to develop a web-based system that can download songs and mix them intelligently using neural network models, with the subsequent possibility of downloading the result to the device.

The research methods are theoretical (collection and structuring of theoretical material), experimental (software implementation of a web application). Development methods are based on Python programming language and Spleeter, Librosa technologies.

In the course of the qualification work, a web-based system for intelligent mixing of vocals and instrumental accompaniment of two songs was developed, which is uploaded by the user with the subsequent possibility of downloading the mixing result to the device.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

ДПФ – дискретне перетворення Фур’є;

КПФ – короткочасне перетворення Фур’є;

ШПФ – швидке перетворення Фур’є;

BPM – Beat Per Minute – ударів на хвилину;

DAW – Digital Audio Workstation – цифрові аудіо робоча станція.

ВСТУП

Цифрова трансформація музичної індустрії докорінно змінила спосіб створення, розповсюдження та споживання музики. Сучасне музичне виробництво все більше покладається на складні технології обробки звуку, які дозволяють творцям маніпулювати та комбінувати музичні елементи в інноваційний спосіб. Серед цих методів можливість відокремлювати та рекомбінувати компоненти з різних аудіоджерел стала потужним інструментом для реміксів, створення міксів та креативного аудіовиробництва.

Традиційні підходи до створення музичних міксів або реміксів вимагали доступу до багатодоріжкових записів або стем-файлів, які часто недоступні для комерційних релізів. Це обмеження історично обмежувало творчі можливості професійних продюсерів зі зв'язками в індустрії. Однак нещодавні досягнення в галузі машинного навчання та цифрової обробки сигналів демократизували доступ до технології розділення джерел, уможлививши виокремлення окремих компонентів із міксованих аудіозаписів з безпрецедентною точністю.

Конвергенція штучного інтелекту та обробки аудіо відкрила нові можливості для автоматизованого маніпулювання музикою. Моделі розділення джерел, зокрема підходи на основі глибокого навчання, тепер можуть виокремлювати вокал, барабани, бас та інші інструменти зі складних музичних сумішей. Водночас, складні алгоритми виявлення бітів і розтягування в часі дозволяють точно вирівняти в часі розрізнені музичні елементи. Ці технологічні розробки створюють можливості для інноваційних застосунків, які можуть автоматично об'єднувати компоненти з різних пісень, зберігаючи при цьому музичну цілісність.

Створення цілісних музичних композицій шляхом поєднання елементів з різних пісень пов'язане з кількома технічними проблемами. Основні труднощі включають:

– якість розділення джерел;

– вирівнювання в часі;

– узгодженість якості звуку;

– ефективність обробки.

Виділення чистого вокалу або акомпанементу з міксованого аудіо з мінімізацією артефактів і збереженням достовірності звуку залишається складним завданням, особливо для складних музичних аранжувань. Різні пісні зазвичай мають різні темпи, ритмічні структури та часові характеристики, які повинні бути точно вирівняні для створення музично приємних результатів. Вихідні матеріали часто мають різні рівні гучності, частотні характеристики та профілі шуму, які потребують нормалізації та покращення для безперешкодної інтеграції. Реальні програми вимагають розумного часу обробки при збереженні якості вихідного сигналу, що вимагає оптимізованих стратегій реалізації.

Існуючі рішення часто вимагають ручного втручання, спеціалізованих знань в області аудіоінженерії або дорогого комерційного програмного забезпечення. Існує потреба в автоматизованій, доступній системі, яка може виконувати ці складні завдання з обробки аудіо, зберігаючи при цьому результати професійної якості.

Розробка цілісного застосунку на основі Python, який об'єднає всі компоненти в ефективний конвеєр обробки з мінімальним втручанням користувача, дозволить зберігати моделі нейронних мереж та веб-систему в єдиній кодовій системі для спрощення підтримки та розвитку застосунку.

1 АНАЛІЗ ТА ПОСТАНОВКА ЗАДАЧІ

1.1 Аналіз предметної галузі

Цифрова обробка звуку є основою для сучасних програм маніпулювання та синтезу музики. У цьому розділі розглядаються основні принципи, що лежать в основі цифрового представлення аудіо, методи аналізу та фундаментальні характеристики музичних сигналів, які уможливлюють операції розділення джерел та об'єднання аудіо.

Цифровий звук виникає в результаті перетворення безперервних аналогових сигналів у дискретні цифрові представлення за допомогою двох фундаментальних процесів: дискретизації та квантування. Процес дискретизації фіксує миттєві значення амплітуди аналогового сигналу через регулярні часові інтервали, створюючи сигнал дискретного часу. Згідно з теоремою дискретизації Найквіста-Шеннона, частота дискретизації повинна щонайменше вдвічі перевищувати найвищу частотну складову вихідного сигналу, щоб уможливити ідеальну реконструкцію [1].

У практичних аудіо застосунках стандартні частоти дискретизації включають:

– 44,1 кГц (якість CD): захоплює частоти до 22,05 кГц;

– 48 кГц (професійний звук): стандарт для виробництва відео та мовлення;

– 96 кГц (аудіо високої роздільної здатності): розширена частотна характеристика для архівування.

Квантування перетворює безперервні значення амплітуди в дискретні рівні, вносячи похибку квантування, яка проявляється у вигляді шуму. Глибина розрядності визначає кількість можливих рівнів амплітуди. Таке відношення представлене у вигляді формули 1.1.

$$Q = 2^n, \tag{1.1}$$

де Q – кількість рівнів квантування;

n – розрядність.

Найпоширеніші розрядності – 16 біт (65 536 рівнів) для побутового аудіо і 24 біт (16 777 216 рівнів) для професійного застосування.

Відношення сигнал/шум (SQNR) безпосередньо пов'язане з бітовою глибиною та представлено у вигляді формули 1.2.

$$SQNR = 6.02n + 1.76\ dB. \quad (1.2)$$

Цифрове аудіо зберігається у вигляді послідовностей дискретизованих значень. Ці вибірки можна візуалізувати у вигляді форми сигналу (амплітуда проти часу) або спектрограми (частотний вміст проти часу). Наприклад, на рисунку 1.1 показано форму сигналу (вгорі) і спектрограму (посередині) людини, яка вимовляє речення «The sun began to rise», яке було отримано під час проведення дослідження у роботі [2].

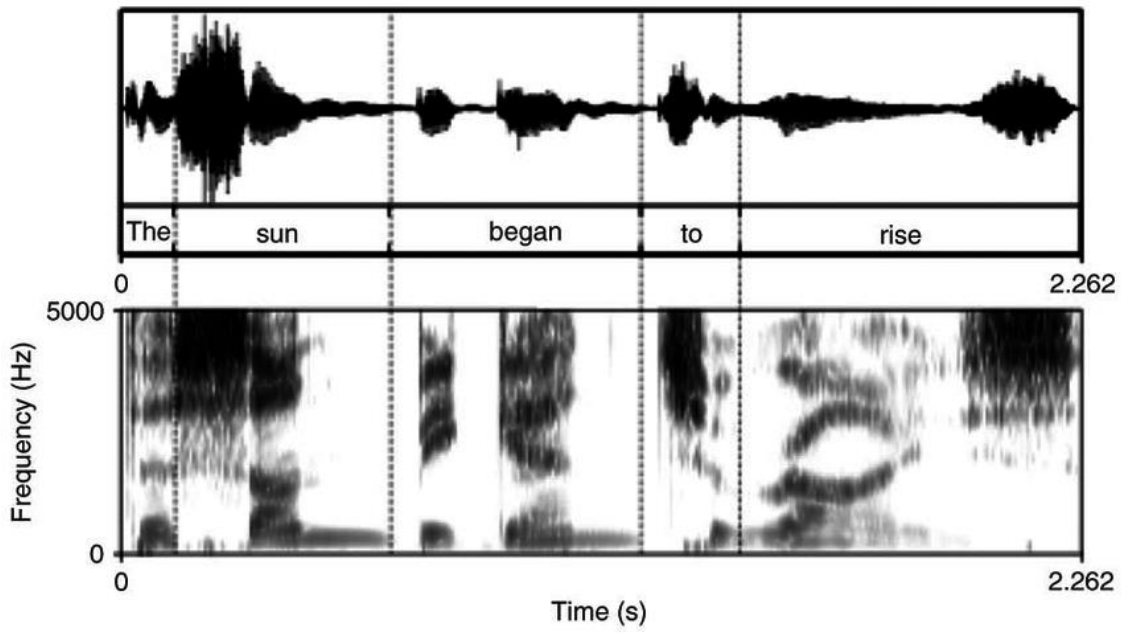

Рисунок 1.1 – Форма сигналу та спектограма речення

Графік форми сигналу відображає гучність звуку в часі, тоді як спектрограма використовує інтенсивність (яскравість), щоб показати, які частоти присутні в кожний момент. Ці представлення допомагають зрозуміти, які частини звукового файлу є важливими. Формати аудіофайлів відрізняються тим, як вони зберігають основні дані семплів – у вигляді сирих семплів або за допомогою різних схем стиснення – і це впливає на розмір файлу, якість і придатність для таких завдань, як міксування або сепарація.

Цифрові аудіосистеми використовують різні формати файлів, оптимізовані для різних застосувань:

а) нестиснуті формати;

– WAV (Waveform Audio File Format);

– AIFF (Audio Interchange File Format);

б) стиснення без втрат;

– FLAC (безкоштовний аудіокодек без втрат);

– ALAC (Apple Lossless Audio Codec);

в) стиснення з втратами;

– MP3 (MPEG-1 Audio Layer 3);

– AAC (Advanced Audio Coding);

– OGG Vorbis.

Нестиснуті формати зберігають необроблені семпли імпульсно-кодової модуляції (ІКМ) без стиснення, тому аудіо відтворюється точно так, як було записано. Поширеними прикладами є WAV та AIFF. Обидва можуть зберігати аудіо CD-якості (16-біт, 44,1 кГц стерео) або PCM з вищою роздільною здатністю. Оскільки вони містять кожен семпл, ці файли мають великий розмір – приблизно 10 МБ за хвилину для стерео CD-якості. Для порівняння, типовий MP3 має швидкість близько 1 МБ за хвилину, що на порядок менше.

Перевагами WAV/AIFF є ідеальна точність і простота, вони не мають артефактів, що призводять до втрат, і приймаються практично всіма

аудіопрограмами. Це робить їх ідеальними для запису, редагування та архівування майстер-копій.

Недоліками є розмір файлу та підтримка метаданих. Файли WAV та AIFF займають багато місця в пам'яті та мають обмежену кількість вбудованих тегів. Наприклад, WAV був розроблений Microsoft/IBM і є стандартним форматом для аудіо на компакт-дисках, але він пропонує лише базові метадані. AIFF – це еквівалент Apple (нестиснутий PCM на Mac) і додає трохи кращу підтримку метаданих, але він також має «великі розміри файлів».

Формати стиснення без втрат зменшують розмір файлу без втрати аудіоінформації. Іншими словами, оригінальний WAV/AIFF можна чудово відтворити. Два найпоширеніші кодеки без втрат – FLAC і ALAC. Вони використовують статистичну надмірність і передбачуваність аудіоданих для стиснення файлів, як правило, приблизно вдвічі менших за розмір еквівалентного WAV. Наприклад, 16-бітний/44,1 кГц WAV може мати швидкість ~10 МБ/хв, тоді як версія FLAC – ~4-6 МБ/хв, залежно від вмісту. Оскільки звук не видаляється, FLAC/ALAC забезпечують таку ж високу якість, як і оригінал. Вони також підтримують метадані (теги, обкладинки альбомів) набагато краще, ніж WAV/AIFF.

Формати з втратами досягають набагато вищого рівня стиснення за рахунок постійного відкидання аудіоінформації, яку вважають нечутною або неважливою. Для цього використовуються психоакустичні моделі, щоб видалити або спростити частини звуку. Результатом є невеликі файли з деякою втратою якості. Найпоширенішими кодеками з втратами є MP3, AAC та OGG Vorbis.

Ефективність сучасного стиснення звуку в основі своїй базується на психоакустичних принципах – науковому розумінні людського слухового сприйняття. У роботі [3] було наведено візуальне представлення психоакустичної моделі, яка зображена на рисунку 1.2.

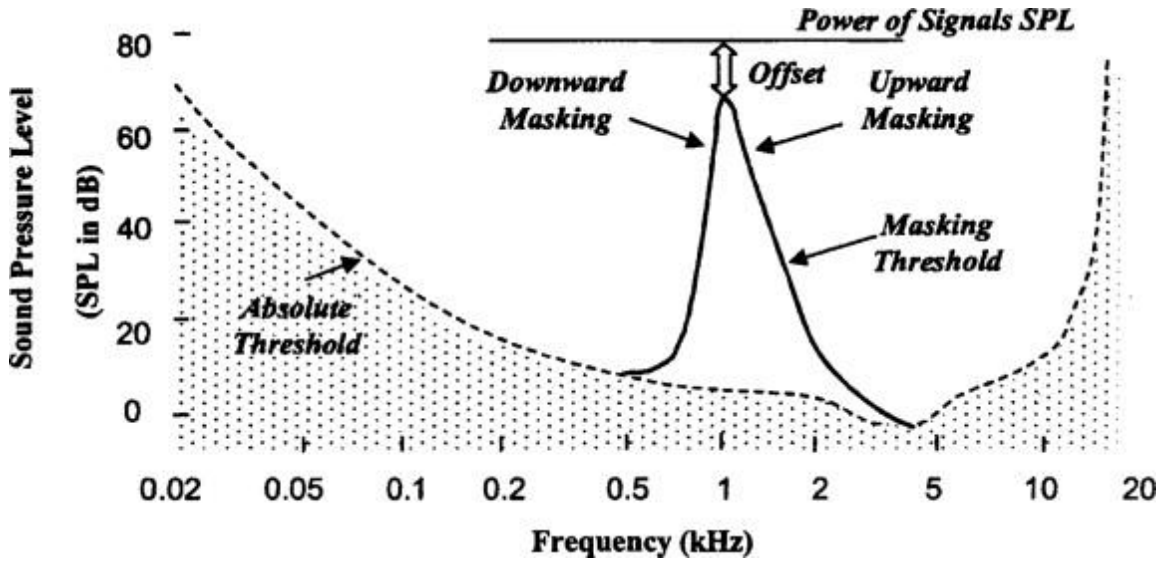

Рисунок 1.2 – Психоакустична модель

Алгоритми стиснення використовують кілька ключових феноменів людського слуху.

Тимчасове маскування відбувається, коли гучний звук робить нечутними близькі за часом звуки, що дозволяє компресорам виділяти менше бітів для аудіоподій, що відбуваються безпосередньо перед або після більш гучних фрагментів.

Частотне маскування використовує нездатність вуха сприймати тихіші звуки, які виникають одночасно на частотах, близьких до гучних.

Абсолютний поріг чутності – мінімальний рівень звукового тиску, необхідний для сприйняття по всьому частотному спектру – дозволяє алгоритмам повністю відкидати інформацію нижче цього порогу.

Критичні смуги представляють ще одну важливу концепцію, згідно з якою слухова система людини ділить частотний спектр на приблизно 25 областей, кожна з яких обробляється дещо незалежно. Згідно з роботою [4], розподіл такого спектру складається з сегментів, що представлені на рисунку 1.3.

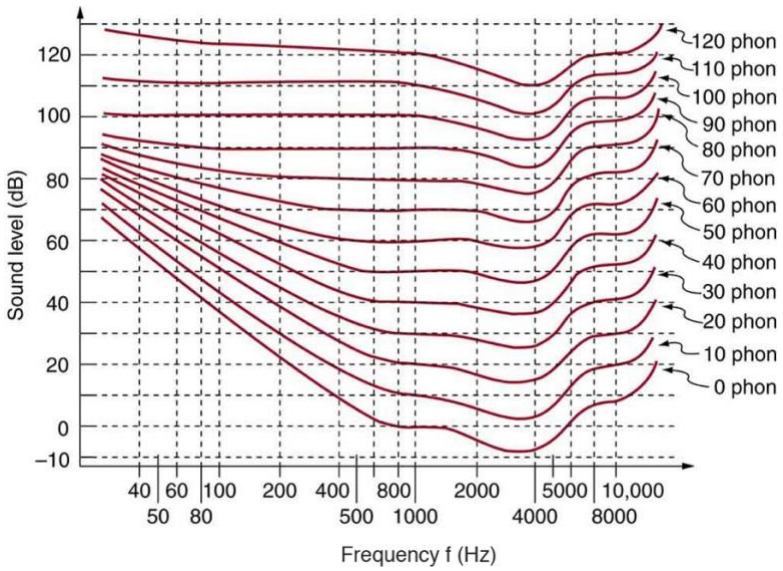

Рисунок 1.3 – Візуалізація спектру людського сприйняття звуку

Алгоритми стиснення розподіляють біти відповідно до їхньої важливості для сприйняття в межах кожної критичної смуги, замість того, щоб обробляти частотний спектр рівномірно. Крім того, різна чутливість вуха до різних частот (описана контурами рівної гучності) впливає на стратегії розподілу бітів, при цьому алгоритми зазвичай зберігають більше інформації в діапазонах частот, де людський слух найбільш чутливий (2-5 кГц). Систематично застосовуючи ці принципи, сучасні аудіокодеки можуть досягати чудових коефіцієнтів стиснення, зберігаючи при цьому прозорість сприйняття для більшості слухачів за типових умов прослуховування.

Нестиснуті файли або файли без втрат надають алгоритму розділення точний оригінальний сигнал. При цьому зберігаються всі деталі (гармоніки, перехідні процеси, фазова інформація), на які покладається Spleeter або інші моделі. На відміну від цього, файл з втратами вже втратив частину інформації через психоакустичне стиснення. Тонкі сигнали, які використовуються для розрізнення вокалу та інструментів, можуть бути розмиті або відсутні. Тому використання входів без втрат, як правило, дає кращу точність розділення. Як зазначено в одному з посібників, нестиснене аудіо дає «нульову втрату якості звуку». Для критичних завдань

обробки (зведення, мастеринг, навчання моделей машинного навчання) рекомендується використовувати формати без втрат.

Хоча файли без втрат займають більше місця на диску та пропускної здатності вводу/виводу, сучасні процесори можуть декодувати FLAC/ALAC на льоту досить ефективно. Librosa сама перетворює будь-який підтримуваний файл у форму сигналу з плаваючою комою, тому, якщо завантажити MP3, бібліотека спочатку декодує його (не вносячи додаткових артефактів, окрім початкового стиснення). Це означає, що Spleeter працює з WAV, навіть якщо завантажити MP3. Однак, декодована форма сигналу з MP3/AAC вже була змінена стисненням, тому будь-які помилки вже вбудовані. Для кращої точності, зазвичай, перед аналізом або розділенням аудіо конвертується у WAV/FLAC.

Формати з втратами можуть вносити незначні спотворення (наприклад, попереднє відлуння або аліасинг), які можуть вплинути на міксування або фільтрацію. У деяких випадках ці артефакти можуть заважати вилученню висоти тону або ритму. Використання чистих, нестиснутих джерел дозволяє уникнути цього. Тим не менш, якщо єдиний доступний запис у форматі MP3/AAC (як у багатьох онлайн-джерелах), Librosa і Spleeter все одно можуть обробити його, вони просто «бачать» стиснуту версію сигналу. Результати можуть бути майже такими ж хорошими, але слід пам'ятати про можливу деградацію.

Аналіз у частотній області є фундаментальною парадигмою в цифровій обробці аудіо, забезпечуючи математичну основу для розкладання складних форм сигналу на складові частотні компоненти. Хоча представлення в часовій області відображають амплітуду в залежності від часу, виявляючи тимчасові характеристики сигналу, вони часто приховують частотну інформацію, критично важливу для аналізу та маніпуляцій зі звуком. Частотна область трансформує цю перспективу, розкриваючи основний спектральний склад, який визначає тембральні якості та гармонійні співвідношення в аудіосигналах. Це перетворення є особливо

цінним у таких застосунках, як розділення звуку, де ідентифікація та ізоляція конкретних частотних компонентів стає необхідною.

Математичною основою цього перетворення є дискретне перетворення Фур'є (ДПФ), яке перетворює скінченну послідовність рівновіддалених відліків сигналу в часовій області в еквівалентне представлення в частотній області. Формально ДПФ визначається формулою 1.3:

$$X_k = \sum_{n=0}^{N-1} x_n \cdot e^{-j\frac{2\pi}{N}kn}, \quad k = 0,1,\dots,N-1, \tag{1.3}$$

де x_n – вхідний сигнал у часовій області;

X_k – результуюче представлення у частотній області;

N – загальна кількість відліків у послідовності;

n – індекс часу;

k – індекс частоти;

$e^{-j\frac{2\pi}{N}kn}$ – комплексна експоненціальна функція, що представляє базисні функції перетворення.

Наглядний приклад трансформації сигналів за допомогою ДПФ представлений на рисунку 1.4.

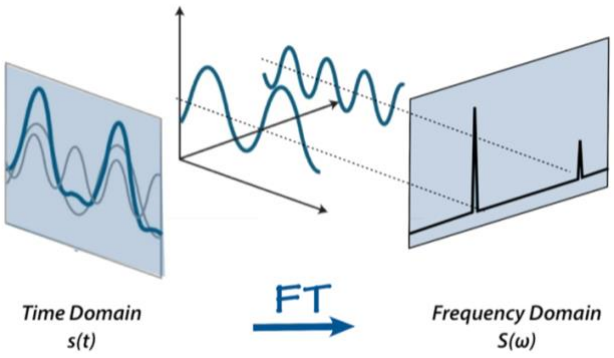

Рисунок 1.4 – Трансформація сигналу у частотне представлення за допомогою ДПФ

Хоча математично елегантний, безпосередня реалізація ДПФ вимагає $O(N^2)$ операцій, що робить його обчислювально неприйнятним для аудіо застосунків у реальному часі. Алгоритм швидкого перетворення Фур'є (ШПФ), популяризований Кулі і Тьюкі в 1965 році, хоча концептуально передував їхній роботі, значно зменшує це обчислювальне навантаження до $O(N\ log\ N)$ операцій. ШПФ досягає такої ефективності завдяки використанню симетрії та періодичності в складних експоненціальних членах, розкладаючи загальне обчислення на менші ДПФ за допомогою підходу «розділяй і володарюй». Найчастіше реалізації ШПФ використовують алгоритм radix-2, що вимагає довжини сигналу, яка є степенем двійки ($N = 2\hat{}m$), хоча варіанти зі змішаним радіксом дозволяють працювати з довільною довжиною, але за рахунок певної обчислювальної ефективності.

Частотно-специфічні енергетичні патерни показують часову еволюцію спектральних характеристик за допомогою таких технік, як спектрограма – візуальне зображення, що відображає залежність частоти від часу, а колір або яскравість вказує на величину енергії. Приклад спектограми представлений на рисунку 1.4.

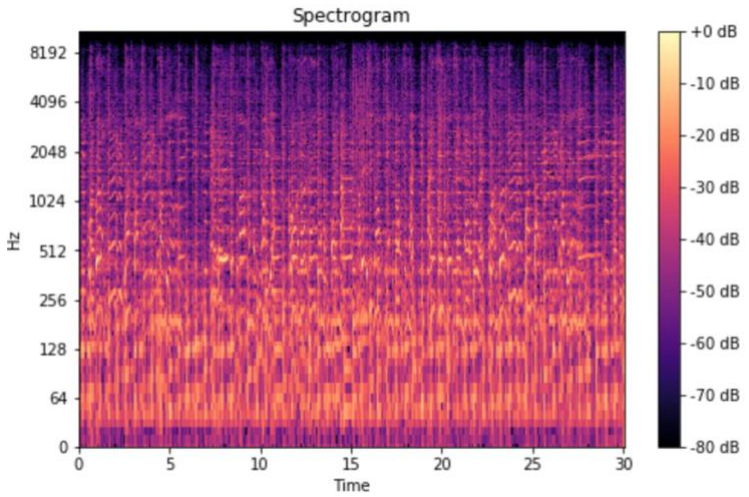

Рисунок 1.4 – Приклад спектограми

Різкі зміни спектрального вмісту часто відповідають початку нот або межам фонем, тоді як поступові трансформації можуть вказувати на виразні прийоми, такі як вібрато або тембральні переходи. Аналіз цих патернів дозволяє сегментувати аудіопотоки на значущі одиниці, ідентифікувати важливі структурні межі та виокремлювати музичні особливості вищого рівня.

Аналіз у частотній області виявляє кілька важливих аспектів аудіосигналів, які залишаються невидимими у часовому представленні. Спектральний розподіл вмісту дає уявлення про загальні тембральні характеристики, відрізняючи яскраві, різкі звуки (енергія сконцентрована у вищих частотах) від теплих, м'яких тонів (енергія сконцентрована в нижчих частотах). Гармонійні зв'язки стають легко помітними, оскільки музичні звуки, як правило, демонструють регулярні спектральні піки на цілих числах, кратних основній частоті. Відстань і відносні амплітуди цих гармонік однозначно характеризують різні інструменти і вокальні якості, полегшуючи завдання розділення та ідентифікації джерел.

Для багатьох аудіо застосунків стандартний аналіз Фур'є виявляється недостатнім через властивий йому компроміс між часовою і частотною роздільною здатністю. Це обмеження призвело до розробки короткочасного перетворення Фур'є (КПФ), яке застосовує ШПФ до послідовних сегментів сигналу, що перекриваються. Математично STFT виражається формулою 1.4.

$$\mathrm{STFT}\{x(t)\}(\tau,\omega) = \int_{-\infty}^{\infty} x(t)\; w(t-\tau)\, e^{-j\omega t}\, dt, \qquad (1.4)$$

де $x(t)$ – вхідний сигнал;

$w(t-\tau)$ – функція вікна, відцентрована за часом τ;

ω – змінна частоти.

Основні параметри STFT включають:

– розмір вікна;

– розмір стрибка;

– функція вікна.

Компроміс між часовою та частотною роздільною здатністю, який регулюється принципом невизначеності, вимагає ретельного вибору параметрів, виходячи з вимог застосування.

Алгоритм КПФ представлений у візуальній формі на рисунку 1.5.

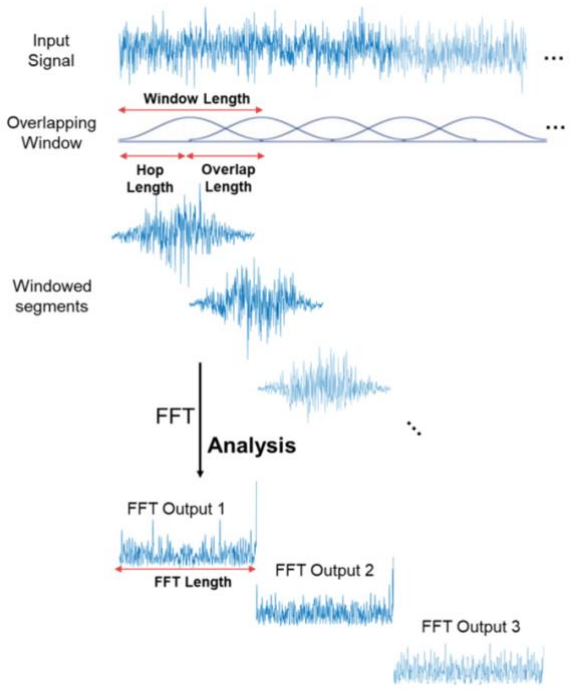

Рисунок 1.5 – Візуалізація КПФ алгоритму

Ця формула створює часо-частотне представлення з роздільною здатністю, що визначається довжиною вікна N і розміром стрибка H. Менші вікна покращують часову локалізацію за рахунок частотної роздільної здатності, тоді як більші вікна покращують частотну дискримінацію, але розмивають тимчасові події. Типові реалізації в обробці аудіо використовують довжину вікна від 512 до 4096 відліків з 50-75% перекриттям між послідовними кадрами, хоча ці параметри варіюються залежно від конкретних вимог програми та характеристик сигналу.

Для подолання обмежень КПФ з'явилися вдосконалені часо-частотні представлення, зокрема вейвлет-перетворення, які забезпечують аналіз з різною роздільною здатністю завдяки змінним розмірам вікон, і перетворення з постійною добротністю, які підтримують постійну відносну частотну роздільну здатність по всьому спектру [5]. Ці складні представлення відіграють дедалі важливішу роль у сучасних системах аналізу та обробки звуку, особливо в застосунках, що вимагають одночасного дослідження ознак, які охоплюють різні часові та частотні шкали.

Сучасний світ реміксування музики та створення міксів представляє складний набір технологічних можливостей поряд з постійними бар'єрами, які обмежують широке впровадження та творче самовираження. Хоча значний прогрес у цифровій обробці сигналів і машинному навчанні трансформував можливості маніпулювання аудіо, фундаментальні обмеження продовжують відокремлювати інструменти професійного рівня від доступних творчих платформ. Розуміння цих обмежень створює важливий контекст для розробки рішень, які можуть демократизувати створення музики, зберігаючи при цьому стандарти якості.

Традиційний підхід до реміксування музики значною мірою покладається на доступ до окремих елементів треку – зазвичай званих stem або багатодоріжковими записами – які забезпечують ізольований вокал, барабани, бас та інші інструментальні компоненти. Професійні реміксери зазвичай потребують таких відокремлених елементів, щоб досягти точного контролю, необхідного для безшовної музичної інтеграції. Однак багатодоріжкові записи є власністю лейблів звукозапису та артистів, які рідко випускають їх у відкритий доступ, що створює безпосередній бар'єр доступу для більшості творчих практиків.

Коли багатодоріжкові записи стають комерційно доступними, вони, як правило, продаються через спеціалізовані платформи на кшталт Splice Sounds або BeatStars, які часто вимагають передплати та ліцензійних угод,

що можуть забороняти певні види творчого використання. Обмежений каталог доступних stem ще більше обмежує творчі можливості, оскільки популярні пісні часто не мають багатодорожечних релізів, що змушує авторів працювати з неоптимальним вихідним матеріалом або взагалі закидати проєкти.

Така залежність створює фундаментальну нерівність в екосистемі створення музики, де професійні продюсери зі зв'язками в індустрії мають доступ до високоякісних вихідних матеріалів, тоді як незалежні творці та викладачі змушені покладатися на імпровізовані методи зі значно гіршими результатами. Економічні бар'єри ускладнюють технічні, оскільки ліцензування популярних треків може коштувати сотні й тисячі доларів, що робить ремікси професійної якості недоступними для більшості ентузіастів та навчальних закладів.

Провідні цифрові аудіо робочі станції (DAW), такі як Pro Tools, Logic Pro та Ableton Live, пропонують складні можливості для реміксів, але їхня складність вимагає значних витрат часу на навчання, яке може зайняти місяці або роки для ефективного освоєння. Ці платформи передбачають наявність глибоких попередніх знань принципів аудіоінженерії, технік зведення та музичної теорії, що створює безпосередні бар'єри для новачків, зацікавлених у творчих експериментах.

Фінансові інвестиції, необхідні для програмного забезпечення професійного рівня, ще більше обмежують доступність. Підписка на Pro Tools коштує 24-83 доларів щомісяця, тоді як Logic Pro вимагає 199 доларів авансового внеску та постійні вимоги до апаратного забезпечення macOS. У поєднанні з вартістю плагінів для розширеного розділення джерел (наприклад, iZotope RX вартістю 399-1 199 доларів), загальна сума інвестицій може легко перевищити 2000-5000 доларів за повне професійне налаштування.

Що ще важливіше, ці інструменти розглядають розділення джерел, вирівнювання темпу та міксування звуку як окремі етапи робочого процесу,

що вимагають ручної координації між кількома програмами та плагінами. Користувачі повинні розвивати навички роботи в різних програмних середовищах, керуючи складними системами організації файлів і підтримуючи якість звуку протягом багатоетапних конвеєрів обробки. Така фрагментація створює можливості для погіршення якості та неефективності робочого процесу, що перешкоджає експериментам і обмежує творчий потік.

Досягнення результатів професійної якості за допомогою наявних інструментів вимагає всебічного розуміння принципів аудіоінженерії, які виходять далеко за межі базової роботи з програмним забезпеченням. Користувачі повинні засвоїти такі поняття, як обробка в частотній області, фазові співвідношення, управління динамічним діапазоном і психоакустичні принципи, щоб уникнути поширених помилок, які погіршують якість звуку або створюють неприємні для сприйняття артефакти.

Процес вирівнювання темпу, необхідний для успішного зведення, вимагає ручного визначення бітів, квантування сітки та операцій з розтягування часу, які при неправильному застосуванні можуть призвести до значних артефактів. Розуміння компромісів між різними алгоритмами часового розтягування, вибір відповідних розмірів вікна аналізу та управління фазовою когерентністю між кількома обробленими доріжками вимагає спеціальних знань, якими володіють лише деякі непрофесійні користувачі.

Контроль якості протягом усього процесу реміксу вимагає тренованих слухових навичок для виявлення таких проблем, як спектральне маскування, гребінчаста фільтрація і гармонійні спотворення, які можуть бути не відразу очевидні для непідготовленого вуха, але суттєво впливають на професійну життєздатність кінцевого результату. Ітеративний характер поліпшення якості означає, що користувачам-початківцям часто важко визначити, коли їхні результати відповідають прийнятним стандартам, що

призводить або до неоптимальних результатів, або до надмірних витрат часу на незначні поліпшення.

Сучасні алгоритми розділення джерел, хоча і являють собою значний прогрес порівняно з традиційними методами, продовжують демонструвати фундаментальні обмеження, які впливають на практичне використання. Навіть найсучасніші моделі, такі як Spleeter і Demucs, створюють звукові артефакти, включаючи спектральне розмазування, коли частотні компоненти одного джерела забруднюють інші, створюючи каламутні або нечіткі результати розділення, які обмежують потенціал якості реміксу.

Виділення вокалу, що має вирішальне значення для створення міксу, часто страждає від залишкового інструментального вмісту, який створює конкуруючі гармонійні структури в поєднанні з новим акомпанементом. І навпаки, інструментальне відокремлення часто зберігає залишки вокалу, які стають особливо помітними під час тихих пасажів або при змішуванні з іншим вокальним контентом. Ці артефакти посилюються, коли користувачі намагаються обробити відокремлені доріжки далі, оскільки кожен додатковий етап обробки може посилити існуючі недоліки.

Підхід до обробки лише за амплітудою, який використовується багатьма популярними інструментами для розділення, створює проблеми з фазовою реконструкцією, що ставить під загрозу стереозображення та просторові характеристики. Хоча такі методи, як алгоритм Гріффіна-Ліма, дозволяють реконструювати звук за спектрограмами амплітуд, отримані оцінки фази часто не відповідають когерентності оригінального запису, створюючи відчуття штучної обробки, що відрізняє відокремлені доріжки від професійно записаного матеріалу.

Переважний підхід до розділення джерел за допомогою обробки амплітудної спектрограми створює невід'ємні обмеження у збереженні фази, які суттєво впливають на якість звуку. Коли алгоритми розділення відкидають фазову інформацію для спрощення обчислювальних вимог, процес реконструкції повинен оцінювати фазові співвідношення, які

можуть неточно відображати оригінальні просторові та часові характеристики вихідного матеріалу.

Ці помилки фазової оцінки проявляються у зменшенні ширини стерео, порушенні локалізації інструментів на звуковій сцені та тонких часових невідповідностях, які можуть створювати відчуття штучної обробки, навіть якщо якість розділення виглядає адекватною. Сукупний ефект фазових помилок стає особливо проблематичним при об'єднанні окремих джерел з різних пісень, оскільки невідповідність фазових характеристик може створювати руйнівні інтерференційні патерни, які знижують загальну когерентність міксу.

Професійні звукорежисери визнають фазову когерентність критичним фактором якості, який відрізняє аматорські записи від професійних. Нездатність сучасних доступних інструментів сепарації зберігати оригінальні фазові співвідношення створює стелю якості, яка обмежує професійну життєздатність результатів автоматизованої сепарації, незважаючи на покращення точності в амплітудній області.

Сучасні моделі розрізнення демонструють значні відмінності в результатах для різних музичних жанрів, що відображає упередженість, притаманну їхнім навчальним наборам даних. Моделі, навчені переважно на західній популярній музиці, як, наприклад, Spleeter на каталозі Deezer, часто не можуть впоратися з незахідними музичними традиціями, експериментальними жанрами або класичними аранжуваннями, які суттєво відрізняються від типових рок/поп-інструментів і технік виробництва.

Електронна музика створює особливі проблеми для алгоритмів розділення, оскільки синтезовані звуки можуть не відповідати гармонійним і спектральним моделям, які моделі вивчили на основі даних навчання акустичних інструментів. Щільні електронні аранжування з частотним перекриттям можуть заплутати алгоритми розділення, що призводить до низької якості ізоляції, яка обмежує потенціал реміксів для цих все більш популярних жанрів.

Часові припущення, закладені в навчальні дані, також створюють обмеження для музики з нетрадиційною структурою. Прогресивний рок, джаз-ф'южн та експериментальні композиції з нерегулярними змінами метру або нестандартною структурою пісень можуть не відповідати ритмічним патернам, які очікують моделі, що призводить до погіршення якості відокремлення саме для музичного контенту, який найбільше виграє від можливостей творчого реміксу.

Сучасна екосистема вимагає від користувачів координувати роботу з декількома спеціалізованими інструментами для створення повного міксу, причому кожен з них виконує лише частину загального робочого процесу. Розділення джерел зазвичай відбувається у спеціальних програмах, таких як Spleeter або Audacity, темп-аналіз вимагає окремого програмного забезпечення для виявлення бітів, вирівнювання може потребувати редагування часової шкали в DAW, а фінальне зведення вимагає ще одного набору інструментів і плагінів.

Така фрагментація створює численні можливості для погіршення якості, оскільки аудіо проходить через кілька циклів експорту/імпорту, кожен з яких може спричинити артефакти стиснення, помилки перетворення частоти дискретизації або зменшення бітової глибини, що в сукупності впливає на кінцевий результат. Керування організацією файлів у різних програмах стає складним і схильним до помилок, особливо для користувачів, які намагаються обробити кілька комбінацій пісень або повторювати різні творчі підходи.

Крива навчання збільшується з кожним інструментом, оскільки користувачам доводиться освоювати різні інтерфейси, вимоги до форматів файлів і парадигми робочого процесу. Когнітивні витрати, пов'язані з перемиканням між програмами та запам'ятовуванням специфічних для кожного інструмента процедур, створюють бар'єри для творчого потоку, які можуть стримувати спонтанне музичне експериментування.

Існуючі рішення, як правило, вимагають ручного узгодження бітів і регулювання темпу – процеси, які вимагають як технічних навичок, так і музичної інтуїції для ефективного виконання. Користувачі повинні визначати низькі долі, вимірювати варіації темпу та застосовувати операції розтягування в часі, одночасно відстежуючи артефакти, які можуть суттєво погіршити якість звуку.

Відсутність автоматичного визначення тональності та підбору гармоній ще більше ускладнює творчий процес, оскільки користувачі повинні покладатися на музичну підготовку для визначення сумісних пісень або приймати ризик гармонійних зіткнень, які можуть порушити музичну злагодженість кінцевого результату. Ця вимога обмежує доступність для користувачів без формальної музичної освіти, створюючи додаткові складнощі навіть для досвідчених музикантів.

Синхронізація декількох доріжок вимагає точності, яка кидає виклик ручним підходам, оскільки навіть невеликі помилки синхронізації стають очевидними при поєднанні вокалу з різними інструментальними доріжками. Ітеративна природа тонкого налаштування вимагає декількох циклів попереднього перегляду/коригування, які забирають багато часу і все одно можуть призвести до недосконалої синхронізації, що впливає на професійну життєздатність реміксу.

1.2 Постановка задачі

Фундаментальною проблемою, що розглядається в цій кваліфікаційній роботі, є розробка автоматизованої системи, здатної виокремлювати вокал з однієї пісні та акомпанемент з іншої, вирівнювати ці компоненти в часі та безперешкодно поєднувати їх для створення нової музичної композиції зі збереженням професійної якості звуку.

Традиційні ремікси та мікси зазвичай вимагають доступу до окремих багатодоріжкових записів або стем-файлів – окремих аудіофайлів, що

містять ізольовані інструменти та вокал. Однак, як було зазначено раніше, ці ресурси рідко доступні широкому загалу, оскільки комерційна музика зазвичай поширюється у вигляді зведених стереозаписів. Це обмеження є значним бар'єром для творчих експериментів і музичних досліджень для ентузіастів, викладачів і творців контенту.

Нещодавні досягнення в галузі машинного навчання та цифрової обробки сигналів дозволили виокремити достатньо чисті вокальні та інструментальні треки з міксованих записів. Однак створення цілісних музичних композицій з цих розрізнених компонентів пов'язане з численними технічними проблемами, які необхідно систематично вирішувати, щоб досягти результатів професійної якості.

Метою цієї кваліфікаційної роботи є розробка автоматизованої системи для об'єднання вокальних та акомпанементних компонентів з різних пісень, створення нових музичних композицій за допомогою інтелектуальної обробки звуку. Конкретні завдання для досягнення мети включають:

– реалізація розділення джерел;

– розробка вирівнювання бітів;

– конвеєр покращення звуку;

– системна інтеграція;

– оцінка продуктивності.

Першим важливим завданням є виділення високоякісного вокалу та акомпанементу з міксованого запису. Навіть найсучасніші моделі розділення, такі як Spleeter, стикаються з обмеженнями:

– артефакти розділення: недосконала сепарація призводить до появи таких артефактів, як спектральне розмиття, залишки вокалу в інструментальних доріжках та залишки музичних інструментів у вокальних доріжках;

– фазова когерентність: більшість алгоритмів розділення працюють зі спектрограмами амплітуд, втрачаючи або наближаючи інформацію про

фазу. Це може призвести до фазових неузгодженостей під час реконструкції, що спричиняє ефект гребінцевої фільтрації та зниження чіткості;

– стабільна якість в різних жанрах: ефективність сепарації значно відрізняється в залежності від музичних жанрів, складності інструментарію та стилів виробництва. Щільні оркестровки, сильно оброблений вокал та різні музичні гами створюють особливі проблеми;

– розділення низьких частот: басові частоти особливо важко відокремити через перекриття їх довжини хвилі і гармонійного складу, що часто призводить до нерівномірного звучання басів в окремих треках.

Злиття компонентів з різних пісень вимагає точного часового вирівнювання для збереження музичної цілісності:

– зміна темпу: пісні зазвичай мають різний темп, що вимагає розтягування в часі для синхронізації ритму. Однак надмірне розтягування в часі може призвести до появи артефактів і погіршення якості звуку;

– точність визначення бітів: точна ідентифікація позицій бітів має важливе значення для правильного вирівнювання, що особливо складно для пісень зі складним ритмічним малюнком, змінами темпу або нетрадиційними часовими сигнатурами;

– структурні відмінності: пісні часто мають різні структурні формати (куплет, приспів, тривалість мостів), що вимагає інтелектуальної сегментації або стратегій адаптації;

– виразний темп: людське виконання містить тонкі часові варіації (рубато, свінг, грув), які ускладнюють точне вирівнювання і можуть спотворюватися під час підстроювання темпу.

Підтримання однакової якості звуку в об'єднаних компонентах створює значні проблеми:

– різниця в гучності: різні пісні мають різні рівні гучності, динамічні діапазони та частотні баланси, що вимагає ретельної нормалізації для досягнення збалансованої інтеграції;

– невідповідність акустичного середовища: записи, зроблені в різних акустичних просторах, мають різні характеристики реверберації, що потенційно створює просторову невідповідність при об'єднанні;

– артефакти обробки: кожен крок обробки сигналу (розділення, розтягування в часі, нормалізація) може вносити кумулятивні артефакти, які погіршують якість кінцевого результату.

Окрім технічних проблем зі звуком, музична сумісність між об'єднаними компонентами представляє певні художні виклики:

– гармонійна сумісність: вокал з однієї пісні може гармонійно конфліктувати з акомпанементом з іншої, якщо вони слідують різним акордовим послідовностям або знаходяться в несумісних тональностях;

– стилістична узгодженість: злиття компонентів з різних музичних жанрів може створити стилістичний дисонанс без відповідної адаптації;

– емоційна узгодженість: емоційна якість вокалу може не відповідати настрою, що передається акомпанементом, що потенційно може створити художній розрив.

Автоматизована система зведення пісень повинна охоплювати широкий набір функціональних можливостей, призначених для полегшення безперешкодної інтеграції вокальних та інструментальних компонентів з різних аудіоджерел. Система обробки вхідного аудіосигналу повинна підтримувати безліч поширених аудіоформатів, включаючи WAV, MP3, FLAC і AAC, забезпечуючи широку сумісність з існуючими музичними бібліотеками та виходами для звукозапису. Ця система повинна включати в себе складні алгоритми обробки, здатні працювати з файлами з різною частотою дискретизації та глибиною розрядності, усуваючи необхідність ручної попередньої обробки. Крім того, система повинна реалізовувати суворі протоколи перевірки для оцінки вхідних параметрів якості та тривалості звуку, позначаючи потенційні проблеми, які можуть вплинути на кінцевий результат, і надаючи користувачам відповідні рекомендації для досягнення оптимальних результатів.

В основі системи лежить передова технологія розділення джерел, розроблена для вилучення вокальних компонентів з першої вхідної пісні, мінімізуючи при цьому артефакти і зберігаючи вокальні нюанси. Одночасно система ізолює акомпанемент з другої вхідної пісні з високою точністю, зберігаючи цілісність інструментальних текстур і тональних характеристик. Протягом усього процесу розділення система зберігає стереозображення і просторові характеристики, притаманні оригінальним записам, гарантуючи, що розмірні якості як вокальних, так і інструментальних елементів залишаються недоторканими. Таке збереження просторової інформації робить значний внесок у природність та ефект занурення кінцевого зведеного результату.

Система включає в себе складний аналіз бітів і можливості вирівнювання, необхідні для створення музично цілісних комбінацій. Вдосконалені алгоритми визначення темпу точно визначають позиції бітів і ритмічні патерни в обох вхідних піснях, створюючи основу для синхронізації. Система реалізує адаптивні механізми часового розтягування, які інтелектуально модифікують темпові характеристики, щоб вирівняти вокальні та інструментальні компоненти без внесення чутних артефактів або спотворень. Ці алгоритми надають пріоритет збереженню ритмічної цілісності під час регулювання темпу, підтримуючи природне відчуття обох елементів, забезпечуючи точну синхронізацію між витягнутим вокалом і доріжками акомпанементу.

Для покращення якості сприйняття об'єднаного вихідного сигналу система використовує декілька процесів покращення звуку. Алгоритми нормалізації гучності аналізують і регулюють рівні сигналу для досягнення однакової гучності між вокальними та інструментальними компонентами, запобігаючи недоречному домінуванню одного з них у міксі. Складні методи шумозаглушення цілеспрямовано виявляють і мінімізують артефакти, що з'являються під час процесу розділення, що призводить до більш чистого і професійного звучання на виході. Крім того, система

застосовує спектральне балансування для забезпечення узгодженої частотної характеристики у всьому аудіоспектрі, гармонізуючи тональні характеристики розділених компонентів і створюючи природний, уніфікований звуковий почерк на кінцевому виході.

Можливості міксування та виведення звуку системи надають користувачам гнучкий контроль над кінцевою композицією. Регульовані коефіцієнти міксування забезпечують точне балансування між вокалом і елементами акомпанементу, що дозволяє налаштовувати композицію відповідно до художніх уподобань або характеристик вихідного матеріалу. Протягом усього ланцюжка обробки система надає пріоритет збереженню якості звуку, гарантуючи, що кінцевий об'єднаний результат відповідає або перевищує точність оригінальних вхідних файлів завдяки ретельному управлінню траєкторією сигналу та алгоритмам обробки з високою роздільною здатністю.

2 ОГЛЯД ІСНУЮЧИХ РІШЕНЬ

За останнє десятиліття сфера автоматизованого розділення джерел звуку та створення музичних міксів зазнала значного розвитку завдяки прогресу в машинному навчанні та зростаючому попиту на доступні інструменти для виробництва музики. У цьому розділі розглядаються сучасні рішення на різних платформах і моделях розгортання, аналізуються їхні можливості, обмеження та цільові аудиторії.

Професійні аудіо робочі станції поступово включали в себе можливості розділення джерел, хоча і з різним ступенем складності та доступності для користувача.

iZotope RX Suite представляє галузевий стандарт для професійного редагування та покращення звуку. Його модуль Music Rebalance використовує алгоритми машинного навчання для відокремлення вокалу, басу, барабанів та інших інструментів із зведених записів. Програмне забезпечення вирізняється високою точністю редагування, інтерфейс якого представлений на рисунку 2.1, та пропонує широкі можливості ручного керування параметрами розділення.

Рисунок 2.1 – Інтерфейс застосунку iZotope RX Suite

Однак RX вимагає значних технічних знань для досягнення оптимальних результатів і має преміальну ціну у понад 1000$ згідно з джерелом [4], що обмежує його доступність для звичайних користувачів. Робочий процес в першу чергу призначений для пост-продакшну аудіо, а не для створення творчих міксів, йому бракує інтегрованого вирівнювання темпу та можливостей автоматизованого міксування.

Steinberg SpectraLayers Pro підходить до розділення за допомогою розширеного спектрального редагування, дозволяючи користувачам візуально ідентифікувати і витягувати аудіокомпоненти. Його перевага полягає в хірургічній точності для складних завдань розділення та інтеграції з Cubase DAW. Тим не менш, крива навчання досить крута, що вимагає значних витрат часу на освоєння інтерфейсу і методів, які представлені на рисунку 2.2. Програма зосереджена на детальному редагуванні, а не на спрощеному створенні міксів, що робить її менш придатною для швидких творчих експериментів.

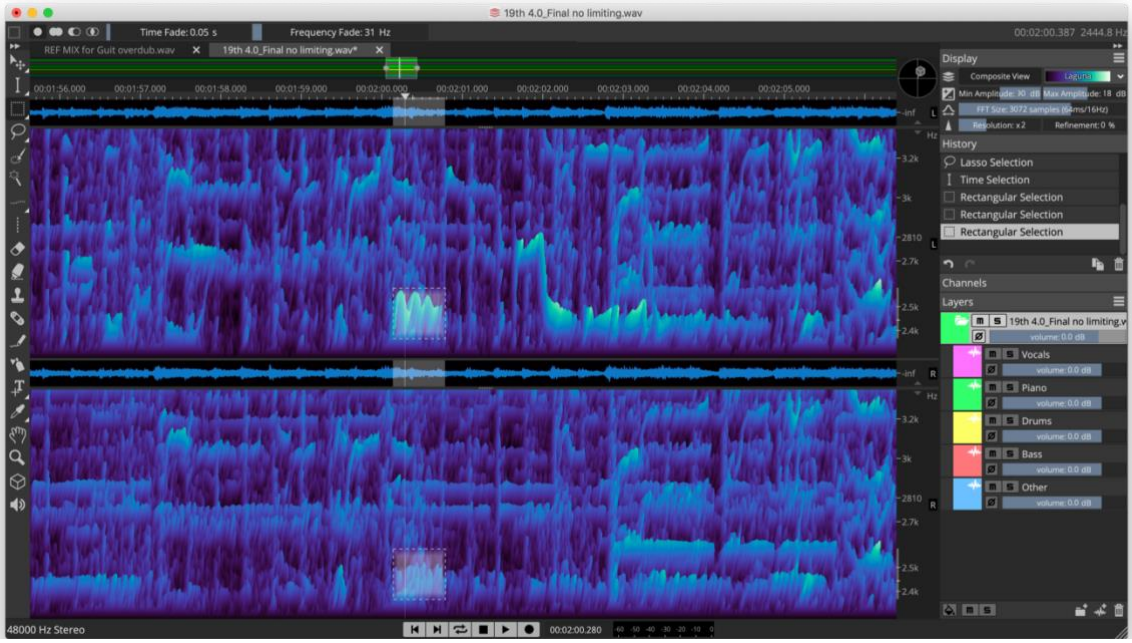

Рисунок 2.2 – Інтерфейс та методи застосунку Steinberg SpectraLayers Pro

Adobe Audition включає в себе інструменти відображення спектральних частот і виділення центрального каналу, які можуть ізолювати вокал за певних умов міксування. Хоча цей інструмент є частиною широко використовуваного Creative Suite, його можливості виділення обмежені традиційними методами на основі ширини стерео, які не працюють з сучасним моноцентричним вокалом і складними аранжуваннями. Інструменту бракує витонченості сучасних підходів машинного навчання і він не надає інтегрованої функції вирівнювання бітів.

Можливості Adobe Audition представлені на рисунку 2.3.

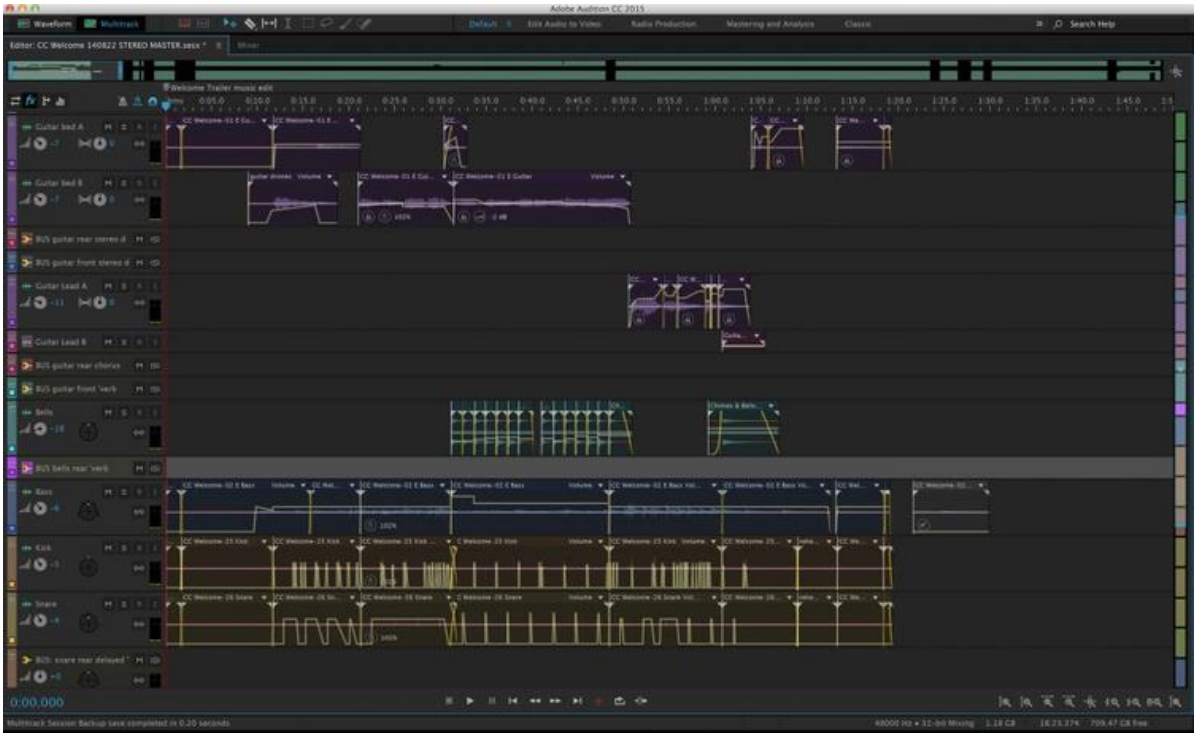

Рисунок 2.3 – Приклад демонстрації можливостей Adobe Audition

Веб-сервіси поширили доступ до технології розділення джерел, хоча часто зі значними компромісами в якості та функціональності.

LALAL.AI став відомим онлайн-сервісом, який обробив понад 10 мільйонів треків за допомогою свого механізму розділення на основі машинного навчання. Платформа пропонує виділення стемів вокалу,

акомпанементу, барабанів, бас-гітари, фортепіано та інших інструментів за допомогою зручного інтерфейсу, що не вимагає технічних знань, який представлений на рисунку 2.4.

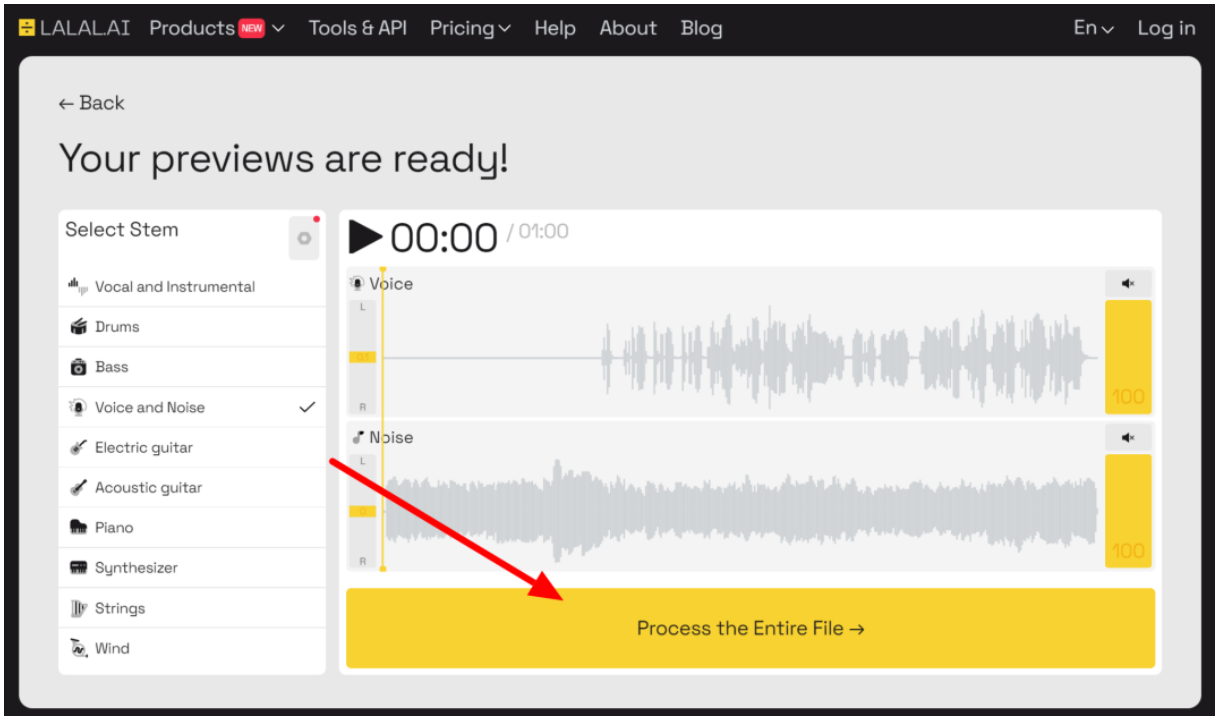

Рисунок 2.4 – Інтерфейс виділення різних інструментів та вокалу з треку сервісом LALAL.AI

Користувачі просто завантажують аудіофайли і отримують відокремлені треки протягом декількох хвилин. Однак сервіс працює на основі кредитної платіжної системи, не має опцій налаштування параметрів розділення і не надає інтегрованих інструментів для міксування чи вирівнювання. Якість значно варіюється в залежності від музичних жанрів і стилів виробництва, з особливими проблемами при роботі зі складними оркестровими аранжуваннями.

Moises позиціонує себе як комплексний практичний інструмент для музикантів, що поєднує розділення джерел зі зміною висоти тону, регулюванням темпу та визначенням акордів. Платформа підтримує як веб,

так і мобільні інтерфейси, що робить її доступною на всіх пристроях, та має вигляд, представлений на рисунку 2.5.

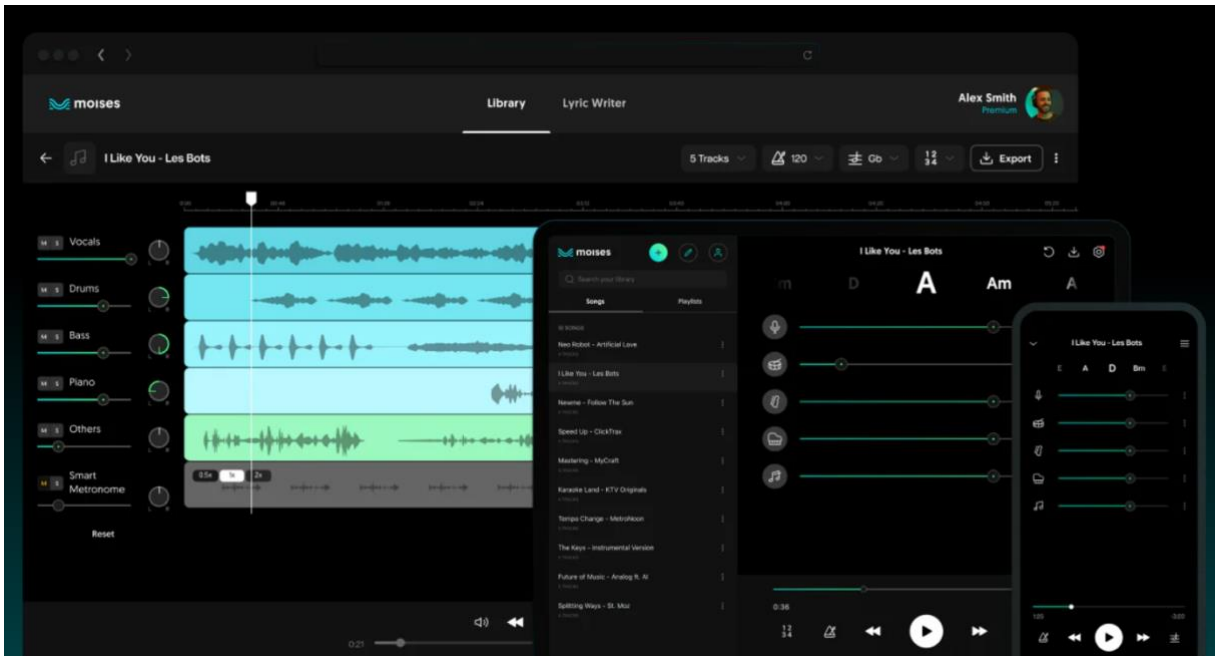

Рисунок 2.5 – Вигляд платформи Moises на різних пристроях

Її сильна сторона полягає в освітніх застосунках, які допомагають музикантам вивчати пісні, ізолюючи окремі інструменти. Однак Moises орієнтована на практику та навчання, а не на творче виробництво міксів, що обмежує якість експорту та не має складних алгоритмів вирівнювання для поєднання різних джерел звуку.

VocalRemover.org пропонує базові послуги з вилучення вокалу за допомогою простого веб-інтерфейсу, переважно з використанням традиційних методів фазової компенсації.

Хоча сервіс є безкоштовним і миттєво доступним, він страждає від значних обмежень у якості розділення, особливо з сучасними методами виробництва музики. Платформа не має розширених функцій для підбору темпу або професійного міксування, що робить її непридатною для створення високоякісних міксів.

Інтерфейс VocalRemover.org максимально простий та представлений на рисунку 2.6.

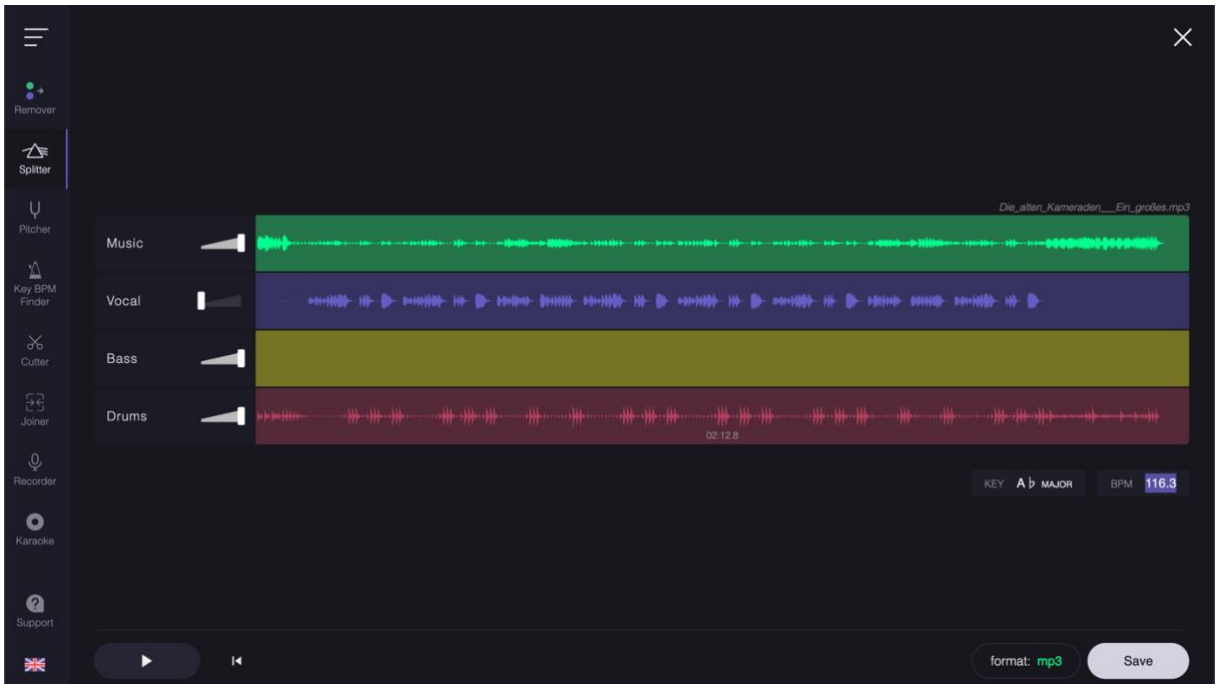

Рисунок 2.6 – Інтерфейс VocalRemover.org

Додатки для смартфонів принесли розділення джерел звичайним користувачам, хоча і з істотними компромісами в обчислювальній потужності і глибині функцій.

Мобільний додаток Moises розширює функціональність веб-платформи для смартфонів, дозволяючи практикувати сепарацію на ходу. Зручність мобільної обробки приваблює музикантів і творців контенту, хоча обчислювальні вимоги обмежують якість сепарації та швидкість обробки. Додатку бракує точних елементів керування та можливостей пакетної обробки, необхідних для серйозної виробничої роботи.

Програми для розділення музики зі штучним інтелектом (різних розробників) наповнюють магазини додатків базовими функціями розділення, зазвичай використовуючи спрощені алгоритми, придатні для мобільних процесорів. Хоча ці програми зручні і часто безкоштовні, вони,

як правило, дають низькоякісні результати і пропонують мінімальні можливості для кастомізації. Більшість з них орієнтовані на караоке, а не на творче виробництво музики.

Академічні ініціативи та ініціативи з відкритим вихідним кодом створили потужні інструменти для екосистеми розділення джерел, хоча часто з обмеженими зручними інтерфейсами.

Spleeter від Deezer революціонізував доступне відокремлення джерел завдяки відкритому коду попередньо навчених моделей, здатних виокремлювати вокал, барабани, бас-гітару, фортепіано та інші інструменти. Інструмент забезпечує вражаючу якість розділення і працює локально, гарантуючи конфіденційність і необмежену обробку. Однак Spleeter вимагає роботи з командним рядком і налаштування середовища Python, що створює бар'єри для нетехнічних користувачів. Інструмент надає лише функцію розділення без інтегрованих можливостей змішування, вирівнювання або покращення.

Перелік параметрів Spleeter для контролю розділення треку представлені у таблиці 2.1.

Таблиця 2.1 – Перелік параметрів для контролю розділення треку

Параметр	Опис
audio_descriptor	Дескриптор аудіо, який ідентифікує аудіоресурс для відокремлення
destination	Шлях до каталогу, в який буде записано окремий вихідний код
audio_adapter	(Необов'язково) Аудіоадаптер для вводу/виводу звуку
offset	(Необов'язково) Параметр зміщення завантаження
duration	(Необов'язково) Параметр тривалості завантаження
codec	(Необов'язково) Збереження параметра кодека
bitrate	(Необов'язково) Параметр збереження бітрейту

Продовження таблиці 2.1

Параметр	Опис
filename_format	(Необов'язково) Форматний рядок для імені вихідного файлу
synchronous	(Необов'язково) Булевий прапор для керування частковою асинхронною обробкою

Побудований на 12-шаровій архітектурі U-Net, що обробляє спектрограми амплітуд, Spleeter досягає 100-кратної швидкості обробки в реальному часі на графічному процесорі, зберігаючи при цьому конкурентну якість. Сама архітектура Spleeter схематично наведена на рисунку 2.7.

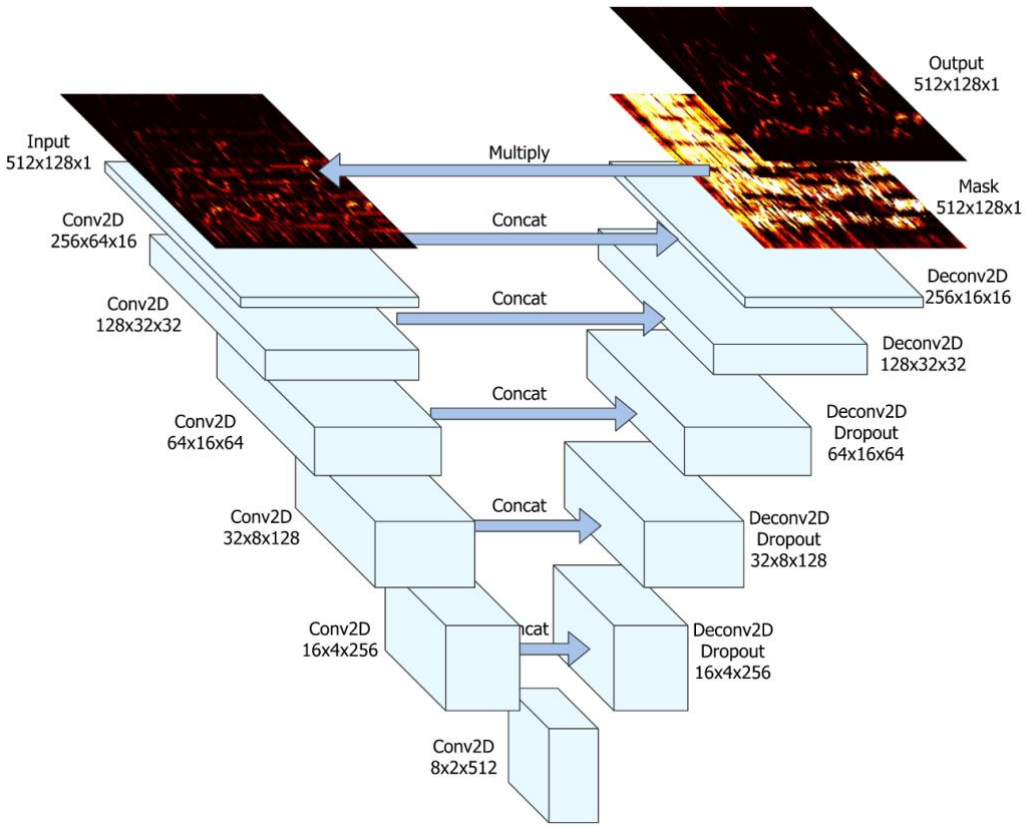

Рисунок 2.7 – Архітектура Spleeter

Інструмент підтримує розділення на 2, 4 і 5 доріжок з попередньо навченими моделями, досягаючи 6,55 дБ SDR для вокалу і 5,93 дБ для барабанів на MUSDB18 згідно з ресурсом [5].

Попередньо навчені моделі пропонують можливість негайного розгортання з трьома конфігураціями: 2-стовпчастий для ізоляції вокалу, 4-стовпчастий для повного розділення груп (вокал, барабани, бас, інше) і 5-стовпчастий з додаванням розділення фортепіано. Моделі за замовчуванням обробляють аудіо до 11 кГц (можна налаштувати до 16 кГц), використовуючи підходи м'якого маскування, які помножують вивчені маски на вхідні спектрограми для генерації оцінок джерела.

Архітектурні обмеження інструменту включають максимально досяжну якість. Робота виключно зі спектрограмами амплітуд створює проблеми з фазовою реконструкцією, а гранична частота 11 кГц знижує точність вихідних даних порівняно з підходами на основі повної смуги, наслідки якого можна побачити на спектограмі, що представлена на рисунку 2.8.

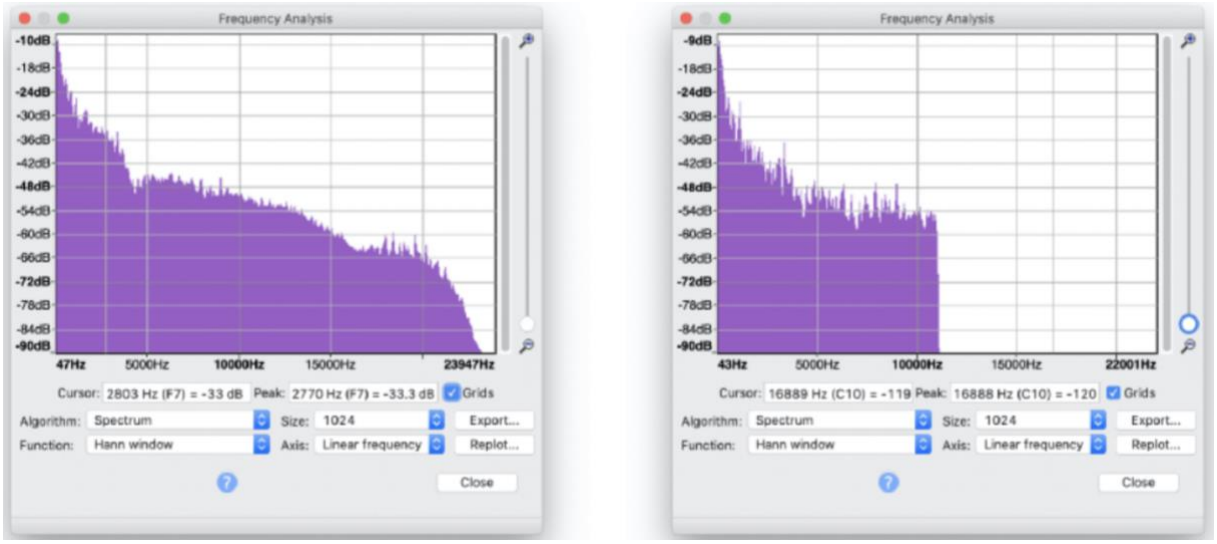

Рисунок 2.8 – Спектограма оригінального треку зліва та після обробки Spleeter зправа

Схильність навчальних даних до західної популярної музики впливає на виконання різних музичних жанрів і культурних традицій. Незважаючи на значний обсяг навчальної бази Deezer, їй бракує різноманітності, необхідної для досягнення оптимальних результатів у виконанні світових музичних традицій, експериментальних жанрів або класичних аранжувань з нетрадиційним інструментарієм.

Основна перевага Spleeter полягає в готовності до практичних завдань та простоті розгортання. На відміну від дослідницьких інструментів, що вимагають складної конфігурації, Spleeter ефективно працює з мінімальним налаштуванням, що робить його доступним для розробників без глибоких знань в області обробки аудіо. Вичерпна документація, широка підтримка спільноти та перевірений досвід комерційної інтеграції значно знижують ризики при впровадженні.

Успіх комерційної інтеграції демонструє практичну цінність у різних сферах застосування. Основне аудіо програмне забезпечення, включаючи iZotope RX 8, функцію стемів від Native Instruments та численні мобільні додатки, використовують технологію Spleeter. Ліцензія MIT дозволяє необмежене комерційне використання з єдиними вимогами до атрибуції, хоча користувачі повинні отримати окремі дозволи на обробку комерційної музики.

Можливості інтеграції охоплюють використання бібліотеки Python через офіційні API та контейнеризацію Docker для розгортання мікросервісів. Сторонні проекти включають плагіни для DAW (Spleeter4Max), десктопні графічні інтерфейси та обгортки REST API, створюючи комплексну екосистему навколо основної технології.

Open-Unmix з'явився в результаті співпраці між Inria (Французьким національним інститутом досліджень у галузі цифрової науки і технологій) та корпорацією Sony, і став основним еталонним реалізацією для досліджень у сфері розділення музичних джерел. Проєкт є першою реалізацією з відкритим вихідним кодом, яка відповідає найсучаснішому

рівню продуктивності, зберігаючи при цьому повну відтворюваність, що робить його незамінним для академічних досліджень і розробки алгоритмів.

Open-Unmix пропонує орієнтований на дослідження підхід до розділення джерел з прозорими, відтворюваними алгоритмами. Хоча цей інструмент є цінним для академічних застосувань і розробки алгоритмів, він вимагає значного технічного досвіду і йому бракує відшліфованості, необхідної для широкого використання користувачами.

Вибір дизайну, зроблений для Open-Unmix, був спрямований на досягнення двох дещо суперечливих цілей. Першою метою є забезпечення найсучаснішої продуктивності, а другою – залишатися легко зрозумілим, щоб він міг слугувати основою для досліджень, які дозволять покращити показники у майбутньому.

Архітектура базується на тришаровій двонаправленій LSTM-мережі, що обробляє спектрограми амплітуд STFT, та схематично представлена на рисунку 2.9. На відміну від згорткового підходу Сплітера, рекурентна архітектура фіксує часові залежності на довших часових горизонтах, що дозволяє більш складне моделювання музичної структури та динаміки.

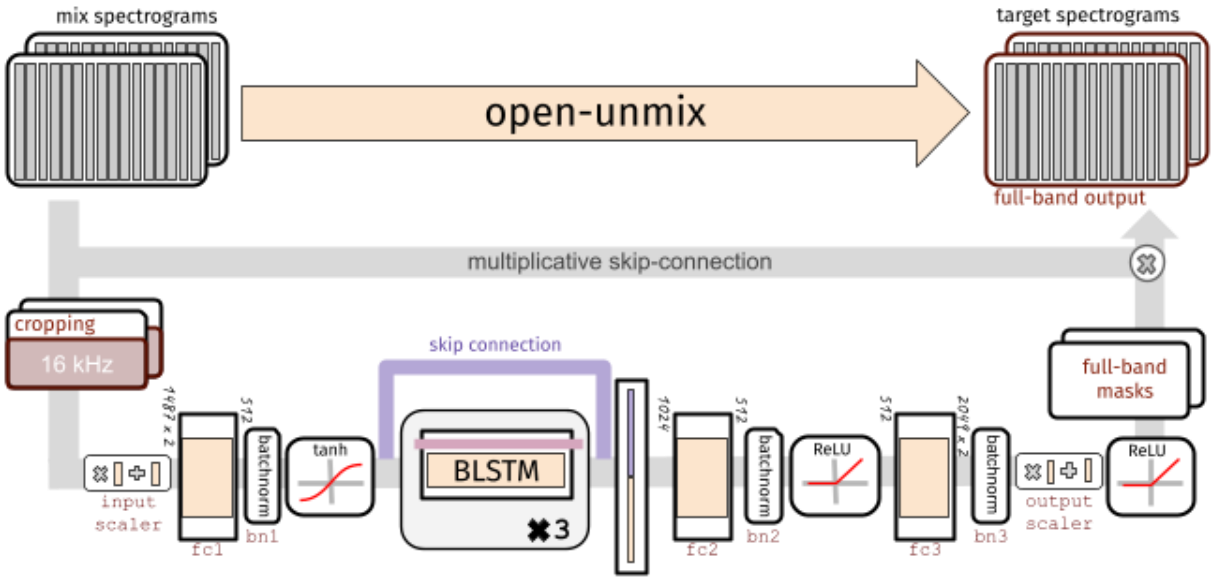

Рисунок 2.9 – Архітектура Open-Unmix

Конвеєр обробки включає кілька інноваційних елементів:

– багатоканальна фільтрація вінера для остаточного розділення джерел, що забезпечує теоретично оптимальне лінійне розділення з урахуванням оцінок амплітуди;

– складна нормалізація, що включає глобальне коригування середнього значення/стандартного відхилення для кожного частотного діапазону та пакетну нормалізацію для інваріантності коефіцієнта підсилення;

– двонаправлена обробка, що охоплює як минулий, так і майбутній часовий контекст, але не дає змоги працювати в реальному часі.

Підхід на основі STFT використовує вікна з 4096 відліків з кроком 1024 відліки, що забезпечує високу частотну роздільну здатність, необхідну для відокремлення джерел гармонік. Обробка лише амплітуд усуває фазову складність, забезпечуючи при цьому ефективне навчання на обмежених обчислювальних ресурсах.

Три основні конфігурації моделі відповідають різним вимогам до якості та обчислень:

– UMX;

– UMXHQ;

– UMXL.

Продуктивність бенчмарку, яка отримана у дослідженні [5], демонструє конкурентоспроможні результати у всіх категоріях джерел, з особливою силою в розділенні вокалу, де двонаправлена архітектура LSTM ефективно захоплює мелодійні контури і гармонійні прогресії.

Найбільша перевага інструменту полягає в його доступності для освіти та досліджень. Навмисно спрощена «MNIST-подібна» архітектура дозволяє легко модифікувати інструмент для проведення власних експериментів, а вичерпна документація та модульні компоненти полегшують розуміння фундаментальних принципів розділення. Інтеграція

з ширшою екосистемою sigsep (MUSDB18, museval, norbert) забезпечує повний робочий процес дослідження від завантаження даних до їх оцінки.

Успіх академічної інтеграції відображає потужну інституційну підтримку з боку Inria та Sony, з регулярними оновленнями сумісності та виправленнями помилок. Інструмент слугує стандартною базовою лінією для порівняння продуктивності в сучасній науковій літературі, забезпечуючи постійну актуальність і підтримку.

Але у цього інструменту існують також певні обмеження. Двонаправлена архітектура LSTM вимагає повних аудіопослідовностей для оптимальної продуктивності. Хоча обробка залишається прийнятною для офлайн-застосунків, вимоги часової залежності перешкоджають потоковому або інтерактивному використанню.

Компроміси між якістю та швидкістю стають очевидними у порівнянні з новими архітектурами. Хоча підхід на основі LSTM був конкурентоспроможним під час його випуску в 2019 році, його випередили трансформаторні та гібридні архітектури, які досягають вищої продуктивності при аналогічних обчислювальних вимогах.

Обробка лише за амплітудою успадковує подібні проблеми фазової реконструкції, як і метод Spleeter, хоча підхід фільтрації Вінера забезпечує більш складну постобробку для зменшення артефактів.

Facebook Demucs представляє найсучасніші дослідження в галузі розділення форм хвиль і доменів, досягаючи найвищих показників якості в академічних тестах. Незважаючи на вражаючу технічну продуктивність, інструмент залишається переважно дослідницьким з обмеженими практичними можливостями розгортання для кінцевих користувачів.

Остання ітерація поєднує обробку в часовій і частотній областях за допомогою складних механізмів перехресної уваги. Цей дводоменний підхід одночасно обробляє як сирі форми хвиль через часові енкодери, так і спектрограми через частотні енкодери, використовуючи взаємодоповнюючі

сильні сторони кожного представлення, схематично представлений на рисунку 2.10.

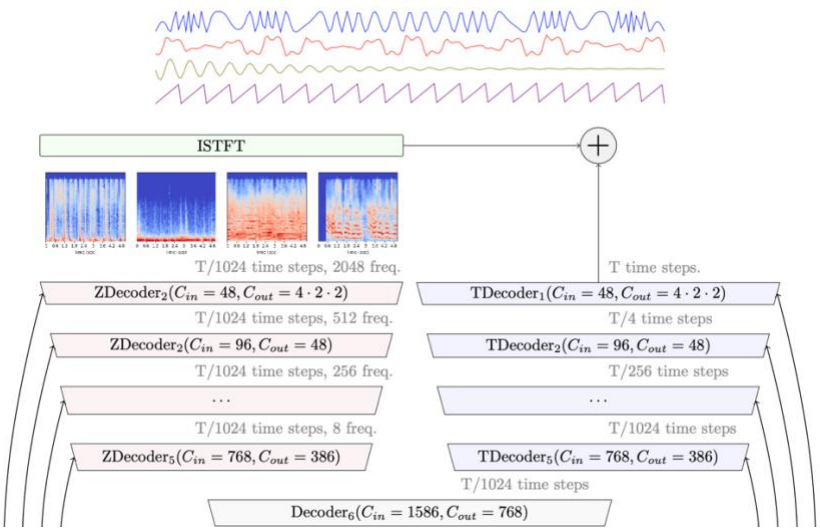

Рисунок 2.10 – Архітектура Facebook Demucs

Гібридний підхід усуває фундаментальні обмеження однодоменних методів, оскільки обробка в часовій області краще зберігає деталі перехідних процесів, в той час як аналіз в частотній області більш ефективно фіксує гармонійні зв'язки.

Високі вимоги до обчислювальної потужності значно обмежують впровадження. Вимога 7 ГБ пам'яті для графічного процесора виключає багато споживчих систем, а швидкість обробки унеможливлює використання додатків у реальному часі, окрім як за допомогою спеціалізованих реалізацій, таких як VST плагіни.

Складність навчання для кастомних моделей вимагає значного досвіду в глибокому навчанні та обробці аудіо, що робить комерційне розгортання складним для організацій, які не мають спеціалізованих технічних команд.

Сучасні рішення демонструють чітку дихотомію між професійними інструментами, що пропонують комплексний контроль, але вимагають

значного досвіду, і споживчими сервісами, що забезпечують доступність, але обмежують функціональність і якість. Професійне програмне забезпечення, таке як iZotope RX, забезпечує виняткові результати, але вимагає значних витрат часу і навичок. Онлайн-сервіси роблять процес розділення треку доступним, але обмежують користувачів у межах фіксованих конвеєрів обробки та моделей підписки.

З цього аналізу випливає кілька критичних прогалин:

– обмеження інтеграції;

– бар'єри доступності;

– обмежені можливості кастомізації;

– фрагментація робочого процесу.

Більшість існуючих рішень розглядають розділення джерел як ізольоване завдання, а не як частину комплексного робочого процесу створення музики. Користувачі повинні вручну комбінувати кілька інструментів, щоб досягти повного створення міксу, що ускладнює роботу і може призвести до погіршення якості.

Професійні інструменти вимагають значних технічних знань і фінансових інвестицій, тоді як спрощені користувацькі інструменти жертвують якістю та контролем. Небагато рішень ефективно долають цей розрив.

Онлайн-сервіси зазвичай пропонують мінімальний контроль параметрів, що змушує користувачів приймати стандартні налаштування обробки, які можуть не відповідати їхнім конкретним аудіоматеріалам або творчим цілям.

Створення міксів наразі вимагає знання кількох спеціалізованих інструментів – програмного забезпечення для розділення, DAW для вирівнювання та міксерних платформ для остаточного виробництва. Така фрагментація збільшує складність і зменшує творчий потік.

3 РОЗРОБКА ВЕБ-ОРІЄНТОВАНОЇ СИСТЕМИ

Розроблена система реалізує комплексний конвеєр обробки аудіо, який перетворює два окремі музичні записи на цілісний мікс за допомогою автоматизованого розділення джерел, часового вирівнювання та методів покращення звуку.

Реалізація використовує модульну архітектуру, яка розділяє проблеми між окремими етапами обробки, уможливлюючи незалежну оптимізацію та налагодження кожного компонента, зберігаючи при цьому чіткий потік даних у конвеєрі. Система спирається на вже існуючі бібліотеки з відкритим кодом – насамперед Spleeter для розділення джерел та librosa для розширеної обробки аудіо – замість того, щоб реалізовувати алгоритми розділення з нуля, що відображає прагматичний підхід, який використовує перевірені інструменти дослідницького рівня, зосереджуючи інновації на інтеграції та оптимізації робочого процесу.

Повний перелік запозичений бібліотек, які використовуються для обробки аудіо, наведений у лістингу 3.1.

Лістинг 3.1 – Програмний код запозичених бібліотек

```
import spleeter
from spleeter.separator import Separator
from spleeter.audio.adapter import AudioAdapter
import librosa
import soundfile as sf
```

Ця філософія проектування надає перевагу надійності та ремонтопридатності над алгоритмічною новизною, визнаючи, що основний внесок полягає у створенні доступного, інтегрованого робочого процесу, а не в удосконаленні окремих методів обробки. Модульна структура дозволяє замінювати компоненти в майбутньому, коли стануть доступними

вдосконалені алгоритми, забезпечуючи довговічність і адаптивність системи.

Система реалізує стандартизовані процедури завантаження аудіо, які нормалізують вхідні характеристики, зберігаючи якість звуку протягом усього конвеєра обробки. Реалізація використовує аудіоадаптер Spleeter для узгодженої обробки частоти дискретизації та сумісності форматів для різних джерел вхідного сигналу. Реалізація завантаження аудіо наведена у лістингу 3.2.

Лістинг 3.2 – Програмний код завантаження аудіо

```
    # Standardized audio loading with quality preservation
    sr = 44100  # Sample rate optimization for quality/speed
balance
    audio_loader = AudioAdapter.default()
    acc_unseparated, _ = audio_loader.load(acc_path,
sample_rate=sr)
    vocal_unseparated, _ = audio_loader.load(vocal_path,
sample_rate=sr)
```

Вибір частоти дискретизації 44,1 кГц – це ретельно продуманий компроміс між ефективністю обробки та збереженням якості звуку. Хоча вища частота дискретизації теоретично може забезпечити кращу частотну характеристику, стандарт 44,1 кГц забезпечує сумісність зі споживчими аудіоформатами, зберігаючи при цьому обчислювальну ефективність на стандартному обладнанні. Це рішення відображає орієнтацію системи на доступність, а не на аудіофільські вимоги до обробки.

Етап попередньої обробки включає в себе функцію перетворення моно, яка інтелектуально обробляє як стерео, так і моно джерела вхідного сигналу та наведена у лістингу 3.3.

Лістинг 3.3 – Програмний код попередньої обробки аудіо

```
def mono(audio):
    if len(audio.shape) == 2:
        return audio.mean(axis=1)
    else:
        return audio
```

Ця реалізація використовує просте усереднення для перетворення стерео на моно, що зберігає характеристики гучності, усуваючи просторову складність, яка може заважати подальшим алгоритмам виявлення біту і вирівнювання. Хоча більш складні методи, такі як обробка середніх частот, можуть зберігати додаткову просторову інформацію, підхід усереднення забезпечує узгоджені результати на різних вхідних матеріалах, не вносячи артефактів обробки.

Система використовує 4-х доріжкову модель Spleeter для розділення джерел, обрану завдяки балансу якості розділення, швидкості обробки та надійної роботи в різних музичних жанрах. Реалізація розділяє кожну вхідну доріжку на вокал, барабани, бас та «інші» компоненти, а потім стратегічно рекомбінує ці елементи на основі запланованої структури міксу. Процес розділення треку на окремі інструменти наведено у лістингу 3.4.

Лістинг 3.4 – Програмний код розділення треку на інструменти

```
    # Strategic source separation using proven algorithms
    separator = Separator('spleeter:4stems')
    vocal_separated = separator.separate(vocal_unseparated)
    acc_separated = separator.separate(acc_unseparated)

    # Intelligent component recombination for mashup creation
    vocal = mono(vocal_separated['vocals']).T
    acc = mono(acc_separated['other'] + acc_separated['drums']
+ acc_separated['bass']).T
```

Рішення об'єднати барабани, бас та інші інструментальні компоненти в одну доріжку акомпанементу відображає орієнтацію системи на вокально-інструментальні мікси, а не на повне реміксування стемів. Такий підхід спрощує процес вирівнювання та зведення, зберігаючи при цьому музичну цілісність, оскільки ці інструментальні елементи зазвичай мають спільні часові характеристики, що полегшує синхронізовану обробку.

Стратегія рекомбінації компонентів враховує, що більшість застосувань міксування передбачає заміну вокалу зі збереженням ритмічної та гармонійної основи доріжки акомпанементу. Розглядаючи невокальні елементи як єдиний акомпанемент, система може застосовувати послідовні методи обробки, уникаючи складнощів незалежного вирівнювання декількох інструментальних джерел.

Реалізація також включає в себе складну функцію нормалізації гучності, що наведена у лістингу 3.5, яка вирішує один з найважливіших факторів якості при створенні аудіо-мешапу – забезпечення однакової гучності між вокальними та інструментальними компонентами.

Лістинг 3.5 – Програмний код функції нормалізації гучності

```
    def normalize_loudness(acc, vocal, frame_length=512,
hop_length=512):
        vocal_normalized = vocal.copy()
        # Calculate RMS loudness for dynamic matching
        acc_loudness = librosa.feature.rms(y=acc,
frame_length=frame_length, hop_length=hop_length)[0]
        vocal_loudness =
librosa.feature.rms(y=vocal_normalized,
frame_length=frame_length, hop_length=hop_length)[0]
        # Apply global loudness normalization
        vocal_normalized = vocal_normalized *
acc_loudness.mean() / vocal_loudness.mean()
        return vocal_normalized, loudness
```

Ця реалізація використовує аналіз середньоквадратичного відхилення (RMS) для обчислення сприйнятої гучності на коротких часових інтервалах, забезпечуючи більш точне вимірювання, ніж простий аналіз пікової амплітуди. Довжина кадру в 512 відліків (приблизно 11,6 мс при 44,1 кГц) забезпечує достатню часову роздільну здатність для фіксації динамічних змін, зберігаючи при цьому обчислювальну ефективність.

Алгоритм включає положення для покадрового вирівнювання гучності, яке могло б забезпечити більш точне динамічне вирівнювання, але було вимкнено, щоб запобігти надмірній обробці, яка може призвести до появи артефактів або неприродного динамічного стиснення. Підхід глобальної нормалізації зберігає оригінальні динамічні характеристики вокального виконання, забезпечуючи при цьому відповідне співвідношення гучності з акомпанементом.

Функція повертає як нормалізоване аудіо, так і дані детального аналізу гучності, що дозволяє перевіряти якість і оптимізувати параметри за допомогою візуального зворотного зв'язку – важливої функції для розуміння поведінки алгоритму і усунення проблем з обробкою.

Реалізація вирівнювання ритму являє собою найбільш технічно складний компонент системи, що вирішує фундаментальну проблему синхронізації музичного контенту з різними темпами і ритмічними характеристиками. Реалізація функції наведена у лістингу 3.6.

Лістинг 3.6 – Програмний код функції вирівнювання ритму

```
    def align_beat(vocal, vocal_unseparated, acc,
acc_unseparated, sr=44100, hop_length=1024, verbose=False):
        def beat(song):
            return librosa.beat.beat_track(y=song, sr=sr,
hop_length=hop_length, units='samples')

        def calc_stretch(beat_times_from, beat_times_to):
```

Продовження лістингу 3.6

```
            return ((beat_times_to[1:] - beat_times_to[:-
1]).mean() /
                    (beat_times_from[1:] - beat_times_from[:-
1]).mean())
```

Алгоритм використовує функцію відстеження бітів librosa, яка використовує методи динамічного програмування для визначення місцезнаходження бітів з точністю до субдискретизації. Довжина кроку в 1024 відліки забезпечує баланс між часовою точністю та обчислювальною ефективністю, дозволяючи точно виявляти біти, зберігаючи при цьому розумну швидкість обробки.

Розрахунок коефіцієнта розтягування, що наведений у лістингу 3.7, порівнює середні інтервали між доріжками, щоб визначити коефіцієнт розтягування в часі, необхідний для вирівнювання темпу. Цей підхід виявляється більш надійним, ніж просте порівняння BPM, оскільки враховує варіації темпу та ритмічну складність, які можуть заплутати прості методи вирівнювання на основі темпу.

Лістинг 3.7 – Програмний код розрахунку коефіцієнту розтягування треку

```
    # Symmetric time stretching for optimal quality
preservation
    mult = calc_stretch(beat_samples_vocal, beat_samples_acc)
    mult_vocal = 1. / (mult ** 0.5)
    mult_acc = mult ** 0.5
```

Реалізація використовує симетричне розтягування в часі, коли обидві доріжки отримують взаємодоповнюючі коригування, які збігаються в проміжному темпі. Такий підхід мінімізує максимальний коефіцієнт розтягування, що застосовується до однієї з доріжок, зменшуючи артефакти

розтягування в часі, забезпечуючи при цьому оптимальне збереження якості. Обчислення квадратного кореня гарантує, що середнє геометричне значення коефіцієнтів розтягування дорівнює одиниці, зберігаючи математичну елегантність та оптимізуючи якість сприйняття.

Функція вирівнювання ритму включає складне вирівнювання фаз, щоб забезпечити одночасність настання відповідних музичних подій, та наведена у лістингу 3.8.

Лістинг 3.8 – Програмний код функції вирівнювання ритму

```
    # Phase-coherent beat alignment with automatic shift
correction
    shift = beat_samples_song_speedup[1] -
beat_samples_other_speedup[0]
    song_speedup_shifted = song_speedup[shift if shift >= 0
else 0:]
```

Цей розрахунок зсуву вирівнює перший виявлений біт обробленої вокальної доріжки з початком доріжки акомпанементу, гарантуючи, що музичні партії починаються синхронно. Умовна обробка запобігає помилкам індексації масиву, зберігаючи точність вирівнювання в різних музичних структурах.

Система включає в себе додаткову функцію шумозаглушення, яка усуває артефакти, що виникають на етапах розділення та обробки джерел. Вона наведена у лістингу 3.9.

Лістинг 3.9 – Програмний код функції шумозаглушення

```
    def denoise(audio, sr=44100, hop_length=1024, n_fft=2048,
top_db=20):
        stft = librosa.stft(audio, n_fft=n_fft,
hop_length=hop_length)
        mask = np.abs(stft) > librosa.power_to_db(top_db)
        stft_denoised = mask * stft
```

Продовження лістингу 3.9

```
        audio_denoised = librosa.istft(stft_denoised,
hop_length=hop_length)
        return audio_denoised
```

Цей спектральний підхід визначає і пригнічує частотні компоненти нижче певного порогу, ефективно видаляючи низькорівневі артефакти і фоновий шум, які можуть погіршити кінцеву якість вихідного сигналу. Поріг 20 дБ забезпечує агресивне шумозаглушення зі збереженням музичного контенту, хоча ця функція залишається необов'язковою, щоб запобігти надмірній обробці, яка може видалити тонкі музичні деталі.

ШПФ на 2048 відліків забезпечує достатню частотну роздільну здатність для точного спектрального аналізу, зберігаючи при цьому обчислювальну ефективність. Реалізація зберігає фазову інформацію протягом усього процесу шумозаглушення, гарантуючи, що просторові характеристики та деталі перехідних процесів залишаються недоторканими.

Система завершується інтелектуальним зведенням, яке поєднує оброблені вокальні доріжки та доріжки акомпанементу, використовуючи емпірично оптимізовані співвідношення рівнів, що наведено у лістингу 3.10.

Лістинг 3.10 – Програмний код інтелектуального зведення вокалу та акомпанементу

```
# Optimized mixing with level balancing
merged = acc_adjusted * 0.8 + vocal_adjusted
```

Масштабний коефіцієнт 0,8 для доріжки акомпанементу відображає типові практики зведення, коли вокалу відводиться незначна роль у фінальному міксі. Це співвідношення було отримано в результаті емпіричного тестування в різних музичних стилях і являє собою компроміс, який ефективно працює в різних жанрах, залишаючись при цьому пристосованим до конкретних творчих вимог.

Реалізація включає комплексну обробку вихідних даних у професійних аудіоформатах, що наведена у лістингу 3.11.

Лістинг 3.11 – Програмний код обробки вихідних даних

```
# Professional-quality output with metadata preservation
sf.write(output_path, merged, sr)
```

Використання бібліотеки звукових файлів забезпечує високоякісний експорт аудіо з належною обробкою метаданих і сумісністю форматів у професійних аудіопрограмах.

Інтегровані в систему компоненти візуалізації слугують важливими діагностичними інструментами, які перетворюють абстрактні концепції обробки звуку на зрозумілі візуальні уявлення. Ці аналітичні засоби виконують подвійну функцію: дозволяють оцінювати якість у реальному часі під час розробки та надають навчальну інформацію про фундаментальні принципи обробки звуку, що лежать в основі успішного створення міксу. Розуміння того, чому було обрано саме ці візуалізації і як вони розкривають характеристики продуктивності системи, є важливим як для технічної перевірки.

Однією з таких візуалізацій є оцінка ефективності нормалізації гучності треку. Графіки порівняння гучності представляють, мабуть, найбільш практичний компонент візуалізації, вирішуючи одну з найпоширеніших проблем якості в аматорському аудіовиробництві – неузгодженість сприйняття гучності між різними джерелами. У якості демонстраційних треків було обрано N.Y. State of Mind репера Nas для відокремлення вокалу та трек Mass Appeal групи Gang Starr для відокремлення акомпонементу.

Результати функції нормалізації гучності продемонстровані на рисунку 3.1.

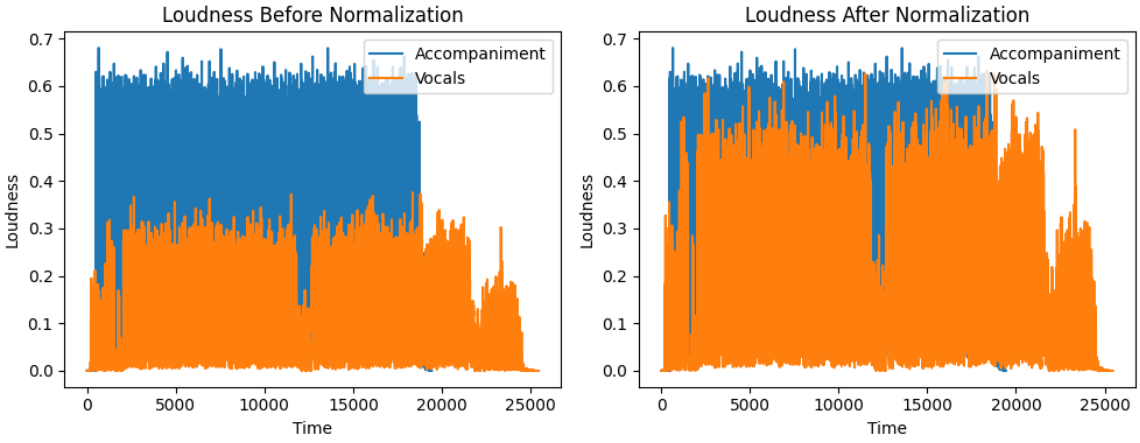

Рисунок 3.1 – Результати функції нормалізації гучності між двома треками

Візуалізація середньоквадратичного значення гучності за часом, забезпечує релевантну для сприйняття міру звукової енергії, яка тісно корелює з людським сприйняттям гучності. На відміну від простих вимірювань амплітуди, які можуть вводити в оману через пікові перехідні процеси, RMS-аналіз показує постійний вміст енергії, який визначає, наскільки гучно звук насправді звучить для слухачів.

Порівняння «до та після» відразу показує ефективність алгоритму нормалізації. У типових сценаріях необроблені вокальні доріжки часто демонструють різко відмінні профілі гучності порівняно з їхнім запланованим акомпанементом – вокал може бути значно тихішим через різні умови запису або значно голоснішим через агресивне стиснення, застосоване під час первинного мастерингу. Ці розбіжності створюють очевидні артефакти у фінальному зведенні, які одразу ж ідентифікують контент як штучно сконструйований.

Візуалізація різниці квадратів дає кількісне уявлення про якість часового вирівнювання між вокалом і доріжками акомпанементу, виявляючи ефективність алгоритму синхронізації бітів за допомогою математичної, а не суто суб'єктивної оцінки.

Графік різниці квадратів до та після функції вирівнювання біту продемонстрований на рисунку 3.2.

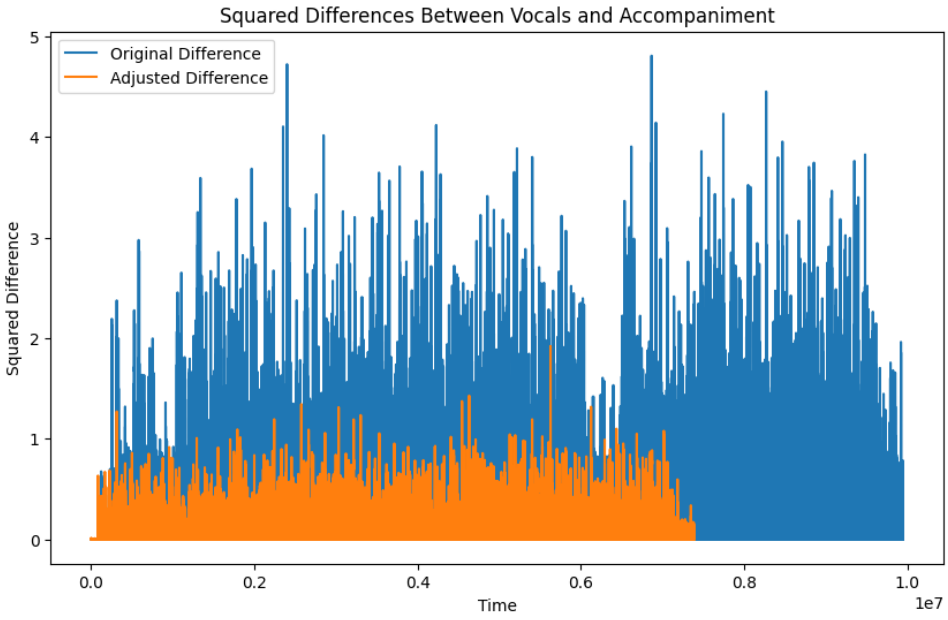

Рисунок 3.2 – Графік різниці квадратів до та після функції вирівнювання біту

Обчислення різниці квадратів виявляє кореляційні патерни між двома аудіопотоками, які безпосередньо пов'язані з якістю часової синхронізації. Коли вокал і акомпанемент правильно вирівняні, їхні ритмічні елементи повинні демонструвати взаємодоповнюючі патерни – сильні вокальні партії можуть відповідати відносно рідкісним інструментальним секціям, тоді як інструментальні акценти можуть збігатися з вокальними паузами або витриманими нотами.

Метрика різниці L2 (у квадраті) підсилює більші розбіжності, применшуючи дрібні варіації, ефективно висвітлюючи значні розбіжності та фільтруючи нормальні музичні варіації, які повинні існувати між різними елементами виконання. Цей математичний підхід забезпечує об'єктивну оцінку, яка доповнює суб'єктивну оцінку прослуховування.

Спектрограма за шкалою Мела дає уявлення про частотну область, яка розкриває спектральні характеристики, що мають вирішальне значення для

розуміння якості розділення джерел і гармонійної сумісності між об'єднаними елементами.

Спектограма за шкалою Мела для створеного з демонстраційних треків міксу наведена на рисунку 3.3.

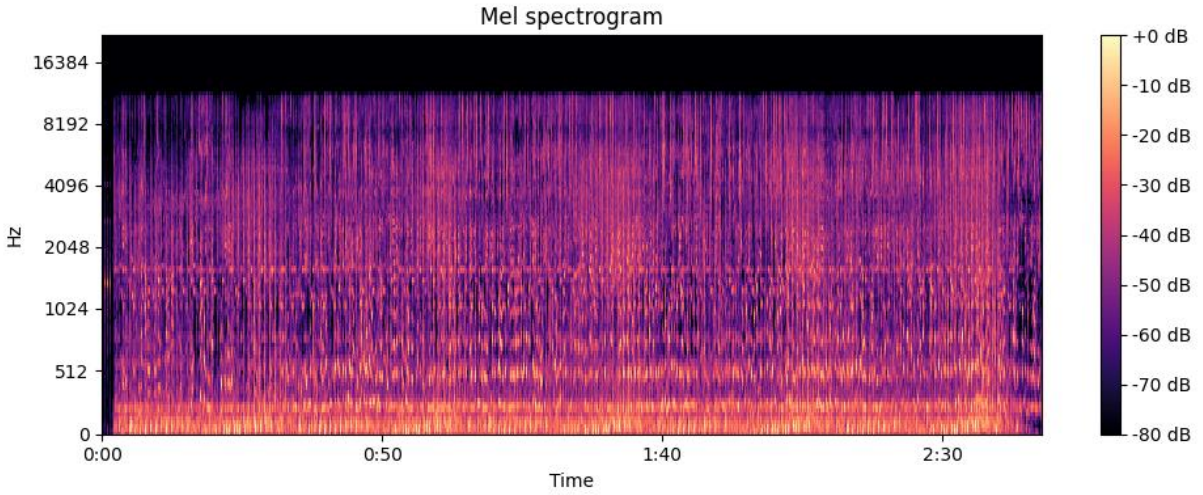

Рисунок 3.3 – Спектограма за шкалою Мела створеного міксу

Цей гнучкий інструмент візуалізації виявляється неоціненним під час розробки системи для виявлення артефактів обробки, перевірки поведінки алгоритмів і розуміння спектральних характеристик різних джерел звуку. Логарифмічна вісь частот підкреслює важливі для сприйняття діапазони частот, забезпечуючи при цьому достатню роздільну здатність для технічного аналізу.

Можливість візуалізувати будь-який аудіосигнал на будь-якому етапі обробки дозволяє систематично налагоджувати ланцюжки алгоритмів – розробники можуть досліджувати вхідні сигнали, проміжні результати обробки та кінцеві результати, щоб точно визначити, де відбувається погіршення якості або де алгоритми обробки можуть потребувати коригування.

Окрім технічних діагностичних функцій, ці візуалізації слугують важливим освітнім цілям, які покращують розуміння фундаментальних

концепцій обробки звуку. Багато користувачів підходять до створення аудіоміксів без глибоких технічних знань, а ці візуальні представлення перетворюють абстрактні поняття на відчутні, спостережувані явища.

Розроблена система заповнює виявлені прогалини на ринку за допомогою декількох ключових інновацій, які відрізняють її від існуючих рішень:

– інтегрований дизайн робочого процесу;

– інтелектуальна автоматизація з користувацьким контролем;

– локальна обробка за допомогою сучасних алгоритмів;

– конвеєр вдосконалення, орієнтований на якість;

– освітня та творча спрямованість;

– відкрита архітектура для розширення.

Система поєднує в собі автоматизовану обробку зі значущими можливостями налаштування користувача. Надаючи розумні налаштування за замовчуванням для нетехнічних користувачів, вона розкриває ключові параметри, які дозволяють досвідченим користувачам точно налаштувати результати відповідно до свого творчого бачення.

На відміну від існуючих інструментів, які зосереджені виключно на розділенні або міксуванні, цей проект реалізує повний наскрізний конвеєр від вхідних аудіофайлів до готових міксів. Користувачі можуть досягати професійних результатів за допомогою єдиного, цілісного інтерфейсу, а не керувати кількома програмними застосунками.

Впроваджуючи найсучасніші моделі розділення, такі як Spleeter, у локальному додатку, проєкт забезпечує конфіденційність користувачів, необмежену обробку та стабільну якість, зберігаючи при цьому доступність завдяки інтуїтивно зрозумілому дизайну інтерфейсу.

Замість того, щоб розглядати розділення як завершальний етап, проект включає комплексне покращення звуку, включаючи нормалізацію гучності, зменшення шуму та спектральне балансування, щоб забезпечити професійну якість на виході незалежно від вхідного вихідного матеріалу.

Система орієнтована як на освітні програми (допомагаючи студентам зрозуміти структуру музики та методи виробництва), так і на творче самовираження (дозволяючи швидко експериментувати з музичними комбінаціями), заповнюючи прогалину між професійними виробничими інструментами та простими споживчими програмами.

Модульна конструкція дозволяє в майбутньому інтегрувати вдосконалені моделі розділення, додаткові ефекти обробки та альтернативні алгоритми вирівнювання, забезпечуючи довговічність та адаптивність, оскільки галузь продовжує розвиватися.

Такий диференційований підхід позиціонує проєкт як такий, що слугує користувачам, які наразі недостатньо охоплені існуючими рішеннями – творчим особистостям, які потребують більше можливостей, ніж надають базові споживчі інструменти, але не мають ресурсів чи досвіду, необхідних для програмного забезпечення професійного рівня. Поєднуючи доступність і витонченість, проєкт відкриває нові можливості для музичних досліджень, освіти і творчого самовираження в умовах зростаючого музичного продакшну за допомогою штучного інтелекту.

4 ДЕМОНСТРАЦІЯ РОБОТИ ВЕБ-СИСТЕМИ

Веб-реалізація перетворює конвеєр обробки звуку в командному рядку на доступний, зручний інтерфейс, який демократизує створення складних аудіоміксів для користувачів без технічних знань.

Веб-застосунок на основі Flask реалізує триступеневу модель взаємодії з користувачем, яка проводить користувачів через весь процес створення міксу: завантаження файлів і вибір параметрів, автоматизовану обробку та отримання результатів. Така лінійна структура робочого процесу мінімізує складність рішень, забезпечуючи при цьому достатній контроль над критично важливими параметрами обробки.

Архітектура системи надає перевагу простоті та надійності, а не складності функцій, визнаючи, що основна база користувачів складається з творців контенту та викладачів, які цінують функціональні результати, а не широкі можливості кастомізації. Оптимізований підхід зменшує когнітивне навантаження, зберігаючи при цьому професійні стандарти якості результатів.

Стратегія управління файлами використовує унікальні ідентифікатори сеансів для запобігання конфліктам між одночасними користувачами, зберігаючи при цьому безпеку системи завдяки санітарній обробці імен файлів та обмеженій перевірці типів файлів, що дозволяє завантажувати основні музичні формати як MP3, WAV, OGG, FLAC. Такий підхід забезпечує багатокористувацьку роботу без складних систем автентифікації, які могли б створити бар'єри для випадкового використання.

Основний інтерфейс представляє користувачам чисту, інтуїтивно зрозумілу форму, яка абстрагується від технічної складності обробки аудіо, водночас розкриваючи основні творчі елементи керування. Філософія дизайну підкреслює ясність і доступність, гарантуючи, що користувачі можуть зрозуміти призначення і функцію кожного елемента інтерфейсу без попередніх технічних знань.

Основна сторінка представлена на рисунку 4.1.

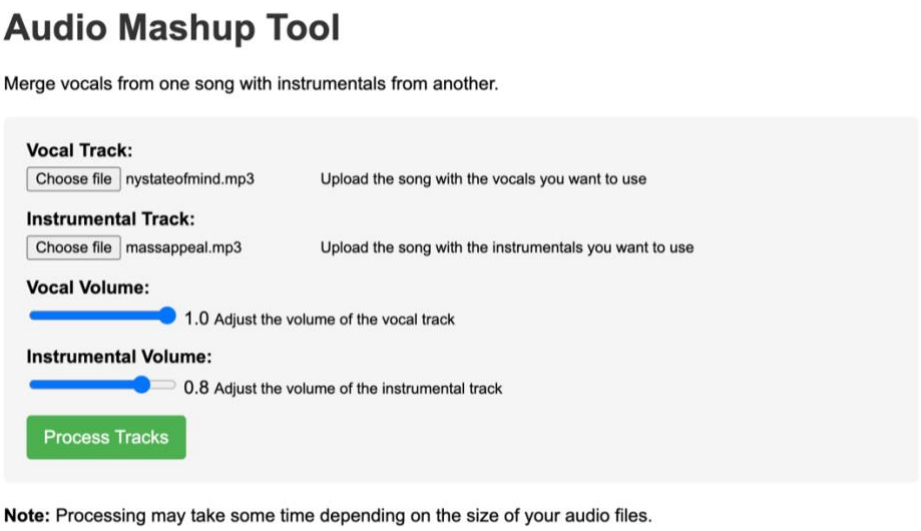

Рисунок 4.1 – Інтерфейс основної сторінки

Система подвійного завантаження файлів чітко розмежовує вокальний матеріал та інструментальний супровід, використовуючи описові мітки та довідковий текст для вибору відповідного контенту.

Інтерфейс обмежує завантаження файлів поширеними аудіоформатами, водночас надаючи чіткі вказівки щодо очікуваних типів контенту. Текст довідки слугує як для навчальних, так і для роз'яснювальних цілей, зменшуючи плутанину користувачів щодо того, який файл повинен містити які музичні елементи. Атрибут HTML5 accept забезпечує негайний зворотній зв'язок щодо підтримуваних типів файлів, а перевірка на стороні клієнта запобігає надходженню явно недійсних файлів.

Реалізація управління гучністю демонструє продуманий дизайн інтерфейсу, який балансує між простотою і функціональною точністю. Візуальний зворотний зв'язок у реальному часі покращує розуміння користувачем налаштувань параметрів, запобігаючи поширеним помилкам міксування.

Налаштування гучності за замовчуванням (вокал: 1.0, інструментальні: 0.8) відображають професійну практику зведення, де

вокалу зазвичай відводиться незначна роль у фінальному міксі. Цей інтелектуальний вибір за замовчуванням дозволяє користувачам досягати прийнятних результатів без розуміння складних принципів зведення, водночас забезпечуючи можливість регулювання для творчих експериментів.

Маршрут обробки process реалізує комплексну обробку помилок і зворотній зв'язок з користувачем, який перетворює потенційно заплутані технічні збої на чіткі, дієві вказівки. Реалізація надає пріоритет користувацькому досвіду завдяки інтелектуальним повідомленням про помилки та плавному відновленню після збоїв.

Система реалізує багаторівневу валідацію, яка захищає як безпеку системи, так і якість користувацького досвіду. Система перевірки забезпечує негайний зворотній зв'язок щодо сумісності файлів, запобігаючи вразливостям безпеки за допомогою санітарної обробки імен файлів та обмеження типів. Повідомлення про помилки вказують як на проблему, так і на її вирішення, зменшуючи розчарування користувачів і сприяючи успішному виконанню завдань. Стан інтерфейсу з помилкою продемонстрований на рисунку 4.2.

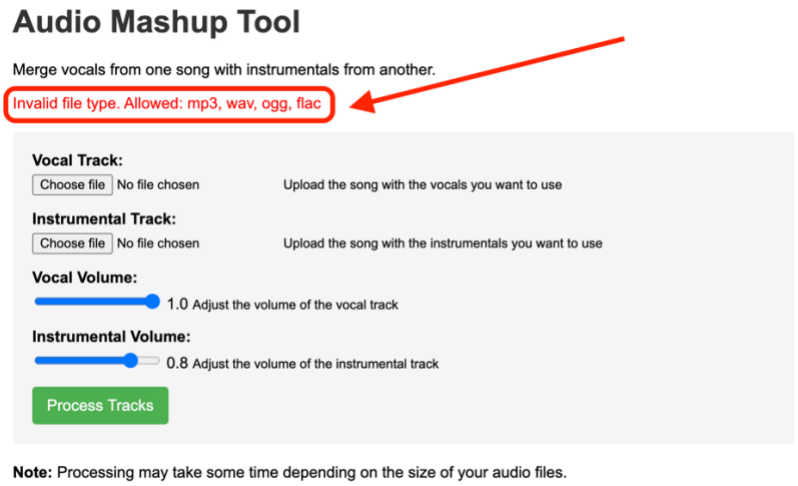

Рисунок 4.2 – Інтерфейс системи з повідомленням про помилку

Унікальна система ідентифікаторів сеансів гарантує, що одночасні користувачі можуть працювати незалежно, без конфліктів файлів і забруднення результатів. Такий підхід усуває необхідність автентифікації користувачів, зберігаючи при цьому операційну цілісність при одночасному виконанні декількох запитів на обробку. Автоматичне очищення тимчасових файлів запобігає накопиченню даних у сховищі, зберігаючи результати до завершення завантаження.

Інтерфейс завантаження результатів, що наведений на рисунку 4.3, забезпечує чітке підтвердження успіху та прямий доступ до результатів обробки. Дизайн святкує успішне завершення, пропонуючи негайний доступ як до створеного контенту, так і можливість для додаткових експериментів.

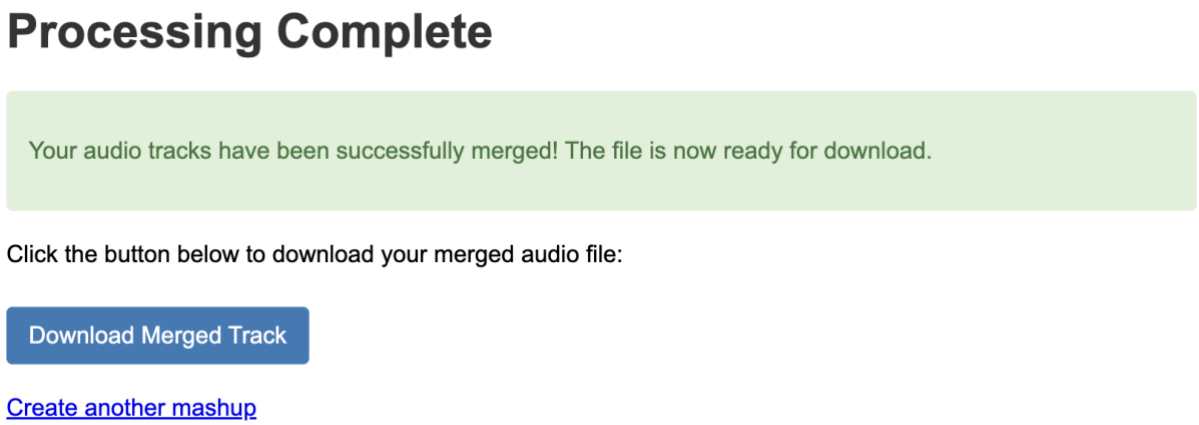

Рисунок 4.3 – Інтерфейс завантаження результатів

На сторінці результатів реалізовано дизайн позитивного підкріплення, який підтверджує успішну обробку та надає чіткі подальші кроки. Повідомлення про успіх завершує операцію обробки, а кнопка завантаження забезпечує негайний доступ до результатів. Помітне посилання «Create another mashup» заохочує до подальшої участі та експериментів, підтримуючи освітні та творчі цілі системи.

Веб-інтерфейс успішно абстрагується від технічної складності, зберігаючи творчий контроль над основними параметрами. Користувачі можуть досягати результатів професійної якості без розуміння алгоритмів, що лежать в основі, тоді як контроль параметрів дозволяє творчо експериментувати, що може призвести до унікальних мистецьких результатів.

Додатково система забезпечує неявне навчання концепціям аудіовиробництва через організацію параметрів і значення за замовчуванням. Користувачі поступово розвивають розуміння взаємозв'язків між гучністю, вибором вихідного матеріалу та оцінкою якості через багаторазову взаємодію з системою.

Негайний зворотний зв'язок, що забезпечується результатами обробки, уможливлює експериментальне навчання, яке може мотивувати користувачів до вивчення більш досконалих технік аудіовиробництва. Цей освітній аспект відрізняє систему від суто функціональних інструментів, підтримуючи зростання та творчий розвиток користувачів.

Поточна реалізація надає перевагу простоті розробки та доступності розгортання над максимальною оптимізацією продуктивності. Модель синхронної обробки забезпечує передбачуване використання ресурсів, а автоматичне очищення запобігає накопиченню даних у сховищі, що з часом може погіршити продуктивність системи.

ВИСНОВКИ

Ця кваліфікаційна робота успішно розробила та продемонструвала комплексну систему створення аудіо-міксу, яка перетворює складні алгоритми обробки сигналів на доступний творчий інструмент для освітніх та мистецьких застосувань. Дослідження спрямоване на подолання значного розриву між можливостями професійного аудіовиробництва та технічними знаннями, необхідними для реалізації складних методів розділення вокалу та вирівнювання звуку, надаючи як теоретичні уявлення, так і практичні рішення, що сприяють розвитку галузі автоматизованої обробки аудіосигналів..

Реалізована система демонструє кілька помітних технічних досягнень, які сприяють сучасному розумінню автоматизованого створення аудіо міксування. Інтеграція архітектури нейронної мережі Spleeter з користувацькими алгоритмами вирівнювання бітів являє собою новий підхід до вирішення проблем часової синхронізації, які історично обмежували якість автоматизованих систем зведення аудіо.

Методологія інтелектуального вирівнювання бітів, розроблена в цій роботі, спеціально вирішує поширену проблему несумісності темпів між вихідними матеріалами за допомогою адаптивного вибору інтервалів. Замість того, щоб застосовувати жорстке узгодження темпів, яке може призвести до появи звукових артефактів, система використовує аналіз геометричних співвідношень для визначення оптимальних патернів вибору бітів, які зберігають музичну цілісність і водночас досягають часової когерентності. Цей підхід виявляється особливо ефективним для поєднання матеріалів із суттєво різними ритмічними характеристиками, розширюючи діапазон вдалих комбінацій для міксування за межі практичних обмежень, що існували раніше.

Система нормалізації гучності, реалізована за допомогою аналізу середньоквадратичного значення енергії, забезпечує релевантне для

сприйняття узгодження гучності, що перевершує прості методи амплітудного масштабування, які зазвичай застосовуються в аматорському аудіовиробництві. Підхід на основі фреймового аналізу забезпечує динамічне регулювання, яке реагує на зміну енергетичних характеристик протягом усього звукового часу, що призводить до більш природного звучання інтеграції між вокальними та інструментальними елементами.

Створена система відповідає реальним практичним потребам освітніх і творчих спільнот, які не мають доступу до дорогого професійного програмного забезпечення для створення аудіозаписів. Можливість створювати високоякісні вокальні мікси з використанням легкодоступних вихідних матеріалів відкриває нові можливості для музичної освіти, творчого самовираження та створення мультимедійного контенту.

Освітній потенціал системи виходить за межі її безпосередніх функціональних можливостей. Компоненти візуалізації забезпечують конкретне представлення абстрактних концепцій обробки звуку, потенційно слугуючи навчальними інструментами для ознайомлення з принципами обробки сигналів, методами часового вирівнювання та перцептивною оцінкою якості звуку.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Müller M. Fundamentals of Music Processing: Audio, Analysis, Algorithms, Applications. *Springer*, 2016. 516 с.

2. Weber A., Scharenborg O. Models of spoken-word recognition. *Wiley Interdisciplinary Reviews: Cognitive Science*. 2012. Т. 3, № 3. С. 387–401. URL: https://doi.org/10.1002/wcs.1178 (дата звернення: 24.05.2025).

3. Multimedia Communications. *Elsevier*, 2001. URL: https://doi.org/10.1016/b978-0-12-282160-8.x5000-2 (дата звернення: 24.05.2025).

4. RX 11 Pricing Options. *iZotope | Plugins for Audio Restoration, Mixing, Mastering and More*. URL: https://www.izotope.com/en/products/rx/pricing-options.html?srsltid=AfmBOoqFuUu49_aJ8hxDxdtzfDuxEHc3hSdaXd5ObbruLMNFp5vH3D7f (дата звернення: 04.06.2025).

5. Spleeter: a fast and efficient music source separation tool with pre-trained models / R. Hennequin та ін. *Journal of Open Source Software*. 2020. Т. 5, № 50. С. 2154. URL: https://doi.org/10.21105/joss.02154 (дата звернення: 04.06.2025).

5. Konstantinovsky T. Wavelet Transform: A Practical Approach to Time-Frequency Analysis. *Medium*. URL: https://medium.com/pythoneers/wavelet-transform-a-practical-approach-to-time-frequency-analysis-662bdadeb08b (дата звернення: 24.05.2025).

6. Spleeter: a fast and efficient music source separation tool with pre-trained models / R. Hennequin та ін. *Journal of Open Source Software*. 2020. Т. 5, № 50. С. 2154. URL: https://doi.org/10.21105/joss.02154 (дата звернення: 10.05.2025).

7. Music Source Separation with Generative Adversarial Network and Waveform Averaging / R. Tanabe та ін. *2019 53rd Asilomar Conference on Signals, Systems, and Computers*, м. Pacific Grove, CA, USA, 3–6 листоп. 2019

p. 2019. URL: https://doi.org/10.1109/ieeeconf44664.2019.9048852 (дата звернення: 10.05.2025).

8. Librosa: Audio and Music Signal Analysis in Python / B. McFee та ін. *Python in Science Conference*, м. Austin, Texas. 2015. URL: https://doi.org/10.25080/majora-7b98e3ed-003 (дата звернення: 10.05.2025).

9. Dagstuhl ChoirSet: A Multitrack Dataset for MIR Research on Choral Singing / S. Rosenzweig та ін. *Transactions of the International Society for Music Information Retrieval*. 2020. Т. 3, № 1. С. 98–110. URL: https://doi.org/10.5334/tismir.48 (дата звернення: 10.05.2025).

10. Diversity in Music Corpus Studies / N. Shea та ін. *Music Theory Online*. 2024. Т. 30, № 1. URL: https://doi.org/10.30535/mto.30.1.8 (дата звернення: 10.05.2025).

11. Harmonic/Percussive Separation Using Kernel Additive Modelling / D. FitzGerald та ін. *25th IET Irish Signals & Systems Conference 2014 and 2014 China-Ireland International Conference on Information and Communities Technologies (ISSC 2014/CIICT 2014)*, м. Limerick, Ireland. 2014. URL: https://doi.org/10.1049/cp.2014.0655 (дата звернення: 10.05.2025).

12. Kernel Additive Models for Source Separation / A. Liutkus та ін. *IEEE Transactions on Signal Processing*. 2014. Т. 62, № 16. С. 4298–4310. URL: https://doi.org/10.1109/tsp.2014.2332434 (дата звернення: 10.05.2025).

13. Ellis D. P. W. Beat Tracking by Dynamic Programming. *Journal of New Music Research*. 2007. Т. 36, № 1. С. 51–60. URL: https://doi.org/10.1080/09298210701653344 (дата звернення: 10.05.2025).

14. Dynamic Time Warping. *Information Retrieval for Music and Motion*. Berlin, Heidelberg, 2007. С. 69–84. URL: https://doi.org/10.1007/978-3-540-74048-3_4 (дата звернення: 10.05.2025).

15. Sakoe H., Chiba S. Dynamic programming algorithm optimization for spoken word recognition. *IEEE Transactions on Acoustics, Speech, and Signal Processing*. 1978. Т. 26, № 1. С. 43–49. URL: https://doi.org/10.1109/tassp.1978.1163055 (дата звернення: 10.05.2025).

16. Ewert S., Muller M., Grosche P. High resolution audio synchronization using chroma onset features. *ICASSP 2009 - 2009 IEEE International Conference on Acoustics, Speech and Signal Processing*, м. Taipei, Taiwan, 19–24 квіт. 2009 р. 2009. URL: https://doi.org/10.1109/icassp.2009.4959972 (дата звернення: 10.05.2025).

17. 40 Days With Jesus: Services And Video Clips (Igniting Worship). Abingdon Press, 2006. 112 с.

18. A tutorial on onset detection in music signals / J. P. Bello та ін. *IEEE Transactions on Speech and Audio Processing*. 2005. Т. 13, № 5. С. 1035–1047. URL: https://doi.org/10.1109/tsa.2005.851998 (дата звернення: 10.05.2025).

19. Bell A. P. Mastering the Multitrack. *Oxford University Press*, 2018. URL: https://doi.org/10.1093/oso/9780190296605.003.0008 (дата звернення: 10.05.2025).

20. Salamon J., Gomez E. Melody Extraction From Polyphonic Music Signals Using Pitch Contour Characteristics. *IEEE Transactions on Audio, Speech, and Language Processing*. 2012. Т. 20, № 6. С. 1759–1770. URL: https://doi.org/10.1109/tasl.2012.2188515 (дата звернення: 10.05.2025).

21. Cutting Music Source Separation Some Slakh: A Dataset to Study the Impact of Training Data Quality and Quantity / E. Manilow та ін. *2019 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, м. New Paltz, NY, USA, 20–23 жовт. 2019 р. 2019. URL: https://doi.org/10.1109/waspaa.2019.8937170 (дата звернення: 10.05.2025).

22. Perraudin N., Balazs P., Sondergaard P. L. A fast Griffin-Lim algorithm. *2013 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA 2013)*, м. New Paltz, NY, 20–23 жовт. 2013 р. 2013. URL: https://doi.org/10.1109/waspaa.2013.6701851 (дата звернення: 10.05.2025).