

Вінницький національний технічний університет

(повне найменування вищого навчального закладу)

Факультет інформаційних технологій та комп'ютерної інженерії

(повне найменування інституту)

Кафедра обчислювальної техніки

(повна назва кафедри)

## **Пояснювальна записка**

до магістерської кваліфікаційної роботи

магістра

(освітньо-кваліфікаційний рівень)

на тему: «Інформаційна технологія формування аудіовідбитків музичних композицій у звукових файлах»

Виконала: студент 2 курсу, групи 1КІ-19м

напряму підготовки (спеціальності)

123 – «Комп'ютерна інженерія»

(шифр і назва напряму підготовки, спеціальності)

Щуровська А. М.

(прізвище та ініціали)

Керівник к.т.н., проф. Азарова А. О.

(прізвище та ініціали)

Вінниця

2020 р.

## АНОТАЦІЯ

Магістерська робота присвячена розробці інформаційної технології формування аудіовідбитків музичних композицій у звукових файлах.

В магістерській роботі розглянуті методи класифікації на основі змішаних Гаусівських моделей з використанням Мел-частотних кепстральних коефіцієнтів (MFCC) для аналізу музичних сигналів, розроблені алгоритм та програма для обробки звукових файлів для формування аудіовідбитків музичних композицій.

У роботі також виконані економічні розрахунки по обґрунтуванню доцільності розробки нового програмного продукту.

## **ANNOTATION**

The master's thesis is devoted to the development of information technology for forming audio prints of musical composition into the audio file.

The master's thesis discusses the methods of classification based on Gaussian mixture model using the Mel-frequency cepstral coefficients (MFCC) for analysis of the music signal, developed an algorithm and program for forming audio print of musical composition.

The master's thesis also contains economic calculations to justify the feasibility of developing a new software product.

## ЗМІСТ

<b>ВСТУП</b> .....	8
<b>1 АНАЛІЗ ТЕХНОЛОГІЙ ФОРМУВАННЯ АУДІОВІДБИТКІВ</b> .....	11
1.1 Особливості музичних сигналів .....	11
1.1.1 Складові музики.....	11
1.1.2 Властивості музики.....	13
1.2 Методи ідентифікації на основі цифрових відбитків.....	15
1.3 Ознаки звуку низького рівня.....	16
1.3.1 Швидкість перетину нуля .....	16
1.3.2 Середньоквадратичне значення.....	19
1.3.3 Спектральний центроїд .....	19
1.3.4 Спектральний перекис.....	20
1.4 Основні принципи отримання та використання цифрових відбитків .....	20
1.5 Delta Spectrum Magnitude .....	23
1.5.1 Кепстральні частотні коефіцієнти.....	24
1.5.2 Психоакустичні ознаки .....	25
1.5.3 Спеціальні ознаки .....	25
1.5.4 Психоакустичний крок шкали .....	26
1.5.5 Визначення ознак.....	26
<b>2 РОЗРОБКА ТЕХНОЛОГІЙ ФОРМУВАННЯ АУДІОВІДБИТКІВ МУЗИЧНИХ КОМПОЗИЦІЙ</b> .....	27
2.1 Розробка математичної моделі виділення аудіофрагментів.....	27
2.1.1 Обґрунтування вибору MFCC .....	27
2.1.2 Дослідження основних характеристик MFCC.....	28

					<i>08-23.МКР.018.00.000 ПЗ</i>			
					Програмний засіб для формування аудіовідбитків музичних композицій у звукових файлах.  Пояснювальна записка.	<i>Літ.</i>	<i>Маса</i>	<i>Масштаб</i>
<i>Змн.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>				
<i>Розробив</i>		<i>Щуровська А.М.</i>						
<i>Керівник</i>		<i>Азарова А. О.</i>						
<i>Рецензент</i>		<i>Куперштейн Л.М.</i>			<i>Арк.б</i>		<i>Аркушів</i>	
<i>Н. контроль</i>		<i>Швець С. І.</i>			<i>ВНТУ, гр. ІКІ-19м</i>			
<i>Затвердж</i>		<i>Азаров О. Д.</i>						

2.1.3 Загальна схема обчислення MFCC.....	31
2.2 Обґрунтування застосування гаусівських змішаних моделей .....	33
2.3 Формування картотеки аудіосигналів із застосуванням EM-алгоритму..	36
<b>3 ПРОГРАМНА РЕАЛІЗАЦІЯ ТА ЕКСПЕРИМЕНТАЛЬНЕ ДОСЛІДЖЕННЯ ТЕХНОЛОГІЇ.....</b>	<b>40</b>
3.1 Мова програмування.....	40
3.1.1 Мова Java .....	40
3.1.2 Мова C++ .....	42
3.1.3 Вибір мови програмування .....	45
3.2 Вибір засобів машинного навчання .....	45
3.2.1 Бібліотека Armadillo .....	46
3.2.2 Бібліотека mlpack .....	47
3.3 Умови проведення експериментального дослідження.....	48
3.4 Розробка програмного забезпечення.....	49
3.5 Застосування розробленого програмного забезпечення .....	51
<b>4 РОЗРАХУНОК ЕКОНОМІЧНОЇ ДОЦІЛЬНОСТІ .....</b>	<b>53</b>
4.1 Технологічних аудит розробки.....	53
4.2 Прогнозування витрат на виконання та впровадження .....	58
4.3 Прогнозування комерційних ефектів від реалізації розробки .....	62
4.4 Розрахунок ефективності вкладених інвестицій та періоду окупності....	63
4.5 Результати економічного ґрунтування .....	66
<b>ВИСНОВКИ .....</b>	<b>68</b>
<b>ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ.....</b>	<b>70</b>
<b>ДОДАТОК А</b> Технічне завдання .....	<b>74</b>
<b>ДОДАТОК Б</b> Лістинг програми .....	<b>78</b>
<b>ДОДАТОК В</b> Блок-схема системи ідентифікації.....	<b>96</b>
<b>ДОДАТОК Г</b> Залежність похибки кластеризації від розмірності MFCC .....	<b>97</b>
<b>ДОДАТОК Д</b> Схема отримання параметрів MFCC.....	<b>98</b>
<b>ДОДАТОК Е</b> Алгоритм формування аудіовідбитків .....	<b>99</b>
<b>ДОДАТОК Ж</b> Схема визначення кількості повторень .....	<b>100</b>

## ВСТУП

На сьогоднішній день, разом із швидким поширенням аудіо та мультимедійної інформації, зростає і кількість технологій, що використовуються для розпізнавання та відновлення мовлення. При розгляді лише музичних файлів, очевидною стає проблема їх індексації, адже вона полягає у змінних у часі характеристиках та досить нечіткому понятті схожості даних характеристик. Людина може з легкістю відрізнити один музичний твір від іншого, виділити мовлення серед музики, в той час як для програмного забезпечення це може стати суттєвою проблемою [1-2].

Широке впровадження цифрових методів ущільнення даних та надзвичайна простота копіювання музичних матеріалів у цифровому форматі призвели до значного та неконтрольованого розповсюдження несанкціонованих копій різноманітних ліцензованих музичних матеріалів.

Швидкий та неконтрольований ріст кількості нелегальних копій аудіовізуальної інформації в першу чергу пов'язаний із появою та поширенням мереж для обміну різними файлами та сервісів для здійснення обміну аматорськими аудіо та візуальними матеріалами, найбільш відомою із яких є мережа YouTube [3]. Мережа YouTube надає доступ до приблизно 40 мільйонів файлів та здійснює контроль за 45% ринку аматорських фільмів та музичних композицій у Інтернеті, серед яких досить часто зустрічаються також несанкціоновані копії музичних кліпів та різних композицій, фрагменти телевізійних програм, а також фільмів, які були незаконно відзняті із використанням різного типу мобільних приладів. Хоча й існують методи захисту та контролю за поширенням аудіовізуальної інформації, але вони мають суттєві недоліки та не забезпечують у повній мірі необхідний рівень надійності захисту та бажану продуктивність [4].

Одним із засобів боротьби із несанкціонованим розповсюдженням музичних копій є системи сегментації звукових файлів та автоматичного розпізнавання мовлення, що використовуються в різноманітних програмах та додатках для

пошуку потрібної вам композиції, автоматичного набору голосових команд. Одним з ефективних способів ідентифікації мультимедійних матеріалів є технологія, широко відома як «метод цифрових відбитків», названий аналогічно до методу ідентифікації людини по його відбитках пальців. Технологія аудіовідбитку лежить в основі міжнародних стандартів, включаючи MPEG-7. Велике зацікавлення до даної технології проявляють численні компанії, що надають сервіс в області доставки, пошуку і захисту мультимедійних матеріалів та інтелектуальної власності, зокрема з використанням мережі Інтернет.

Виходячи із розглянутого, завдання подальшого вдосконалення технології формування аудіовідбитків, що є кращими за критеріями швидкість та достовірність, є **актуальною задачею**.

**Метою дослідження** магістерської роботи є створення технології формування аудіовідбитків музичних композицій у звукових файлах.

Для досягнення поставленої мети система мусить мати певний перелік властивостей та вирішувати наступні **задачі дослідження**:

- провести аналіз існуючих програмних засобів з аналогічними функціями;
- вибрати математичні моделі для опису аудіовідбитків;
- розробити програмне забезпечення для формування аудіовідбитків музичних композицій у звукових файлах;
- провести тестування розробленого продукту та його порівняння з провідними аналогами;
- розрахувати економічну доцільність розробленого продукту.

**Об'єкт дослідження** магістерської роботи — процес формування аудіовідбитків музичних композицій у звукових файлах.

**Предмет дослідження** магістерської роботи — методи та технології формування аудіовідбитків у звукових файлах.

**Методи дослідження** магістерської роботи: використовувались методи класифікації на основі змішаних Гаусівських моделей з використанням Мел-частотних кепстральних коефіцієнтів (MFCC) для аналізу музичних сигналів та

мовлення. У роботі використано принципи об'єктно-орієнтованого програмування для реалізації запропонованого підходу.

**Наукова новизна отриманих результатів** магістерської роботи полягає у тому, що удосконалено технологію формування аудіовідбитків музичних композицій, що відрізняється від відомих підходів обчисленням мел-частотних кепстральних коефіцієнтів із використанням Гаусівської змішаної моделі, що дозволяє більш ефективно виділяти музику у звукових файлах.

**Практичне значення одержаних результатів** магістерської роботи:

— створено технологію формування аудіовідбитків музичних композицій у звукових файлах;

— розроблено програму обробки звукових файлів для формування аудіовідбитків музичних композицій.

**Апробація** результатів магістерської роботи — зроблено доповідь на молодіжній науково-практичній інтернет-конференції студентів, аспірантів та молодих науковців «Молодь в науці: дослідження, проблеми, перспективи (МН- 2021)».

По результатам роботи зроблені публікації [5].



# 1 АНАЛІЗ ТЕХНОЛОГІЙ ФОРМУВАННЯ АУДІОВІДБИТКІВ

Системи сегментації звукових файлів використовуються в різноманітних додатках та програмах для пошуку потрібної вам музичної композиції чи автоматичного набору голосових команд. Одним з ефективних способів ідентифікації мультимедійних файлів є технологія, відома під назвою «метод цифрових відбитків». Назва походить від методу ідентифікації людини по її відбитках пальців. Даний розділ магістерської роботи присвячений розгляду та аналізу існуючих методів сегментації звукових файлів та технології формування аудіовідбитків.

## 1.1 Особливості музичних сигналів

### 1.1.1 Складові музики

Практично будь-який звук може бути музичним з певними акустичними характеристиками, що відповідають естетиці та смакам різних епох, та може відтворюватись повторно. Джерелами такого звуку є безліч і вони досить різноманітні: людський голос, струни чи ударна поверхня музичних інструментів, електричні генератори тощо.

Мелодія — це провідний голос, що концентрує в собі естетичну цінність та художню думку музичного твору, підпорядковує собі інші компоненти твору [6].

Мелодія являє собою послідовність інтервалів, що спадають та зростають, повторюються та розвиваються за певним ритмом, що містить в собі певну гармонію та передає настрій, що не залежить від слів тексту, основних та супровідних голосів, абсолютної висоти звучання та інструментів. Для прикладу, у західній класичній музиці, композитори представляють початкову мелодію або тему, а далі створюють її варіації.

Тон — це основний компонент тональності, від якої він власне й отримує свою назву, центр, до якого прямує тяжіння всіх гармоній. Існує два види тональних структур: простий тон, до складу якого входить одна синусоїдальна форма сигналу та складний тон, що містить в собі більш ніж одну гармоніку.

Якість звуку напряму залежить від того, скільки негармонічних частотних компонентів він містить. Всі класичні музичні тони створюються з резонансної частоти рухомих частин музичних інструментів. Певні тони виходять з резонансної трубки, за винятком електронної музики, тон в якій створюється в залежності від електромагнітної сили.

Будь-які негармонічні частотні компоненти, що не є кратні основній частоті звучання, називають обертоном. Виробники музичних інструментів докладають зусиль аби зменшити кількість обертонів, створюючи широкосмугові тони, що можуть вмістити в себе всі звукові смуги. Не дивлячись на це, у всіх музичних інструментах звучать обертони, що не пов'язані з основною частотою гармоніки [7].

<b>A</b> <b>Hz</b>	<b>B</b> <b>Hz</b>	<b>C</b> <b>Hz</b>	<b>D</b> <b>Hz</b>	<b>E</b> <b>Hz</b>	<b>F</b> <b>Hz</b>	<b>G</b> <b>Hz</b>
A <sub>1</sub> 27.5	B <sub>1</sub> 30.863	C <sub>1</sub> 32.703	D <sub>1</sub> 36.708	E <sub>1</sub> 41.203	F <sub>1</sub> 43.654	G <sub>1</sub> 48.999
A <sub>2</sub> 55	B <sub>2</sub> 61.735	C <sub>2</sub> 65.406	D <sub>2</sub> 73.416	E <sub>2</sub> 82.407	F <sub>2</sub> 87.307	G <sub>2</sub> 97.999
A <sub>3</sub> 110	B <sub>3</sub> 123.47	C <sub>3</sub> 130.81	D <sub>3</sub> 146.83	E <sub>3</sub> 164.81	F <sub>3</sub> 174.61	G <sub>3</sub> 196
A <sub>4</sub> 220	B <sub>4</sub> 246.94	C <sub>4</sub> 261.63	D <sub>4</sub> 293.66	E <sub>4</sub> 329.63	F <sub>4</sub> 349.23	G <sub>4</sub> 3 92
A <sub>5</sub> 440	B <sub>5</sub> 493.88	C <sub>5</sub> 523.25	D <sub>5</sub> 587.33	E <sub>5</sub> 659.26	F <sub>5</sub> 698.46	G <sub>5</sub> 783.99
A <sub>6</sub> 880	B <sub>6</sub> 987.77	C <sub>6</sub> 1046.5	D <sub>6</sub> 1174.7	E <sub>6</sub> 1318.5	F <sub>6</sub> 1396.9	G <sub>6</sub> 1568
A <sub>7</sub> 1760	B <sub>7</sub> 1975.5	C <sub>7</sub> 2093	D <sub>7</sub> 2349.3	E <sub>7</sub> 2637	F <sub>7</sub> 2793	G <sub>7</sub> 3136
A <sub>8</sub> 3520	B <sub>8</sub> 3951.1	C <sub>8</sub> 4186				

Рисунок 1.1 — Шкала нотних частот

Музичний спектр вдвічі більший за пропускну здатність мовного спектру. В цілому, більша частина потужності звукової форми сигналу зосереджена в більш низьких частотах. Характеристики музики напряму залежать від розмірів та виду музичного інструменту.

Музиканти поділяють увесь музичний спектр на 8 частин, кожна з яких називається октавою та поділяється на 7 частин (тонів). Тони мають назву До(C), Ре (D), Мі (E), Фа (F), Соль (G), Ля (A) та Сі (H чи B). Цей поділ проводиться відповідно до частоти тону. В октаві кожен тон має відповідну частоту, тому музичний інструмент мусить створювати тон навколо цієї конкретної частоти [8].

Взявши до уваги спеціальну шкалу тонів та їх частот, що міститься на рис. 1.1, відзначимо, що найвищий тон C8 знаходиться на частоті 4186 Гц. Це найвища частота, що використовується в людському звучанні. Виробники музичних інструментів намагаються максимально прив'язати музичну частоту до звукових меж людини, хоча насправді музичні інструменти покривають більш виразну смугу (приблизно 20 кГц).

### 1.1.2 Властивості музики

Музику можна поділити на інструментальну й вокальну. Інструментальна музика є власне чистою музикою, в той час як вокальна представляє собою синтез музики та поетичного (інколи прозаїчного).

Музичний звук є основним елементом музичного мистецтва. Суб'єктивні особливості людського сприйняття звуку дозволяють виділити чотири головні властивості музичного звуку — висоту звуку, тембр, тривалість та гучність.

З точки зору акустики, висота звуку характеризує звук певною періодичністю акустичних коливань, чітко визначеними максимумами в спектрограмі основного тону та гармонічних обертонів. В той же час, невизначену частоту звуку характеризують неперіодичними коливаннями та відсутністю чітко означених максимумів в спектрограмі. Особливе місце відводять так званим мультифонічним звукам, що характеризуються як наявністю основного тону, так і сильних негармонічних обертонів, що надає цим звукам специфічної якості.

Незважаючи на те, що діапазон частот, який людський слух здатен розрізнати поширюється від 20 Гц до 18 кГц (за окремими даними від 16 Гц до 20 кГц), в музиці застосовуються лише тони частотою від 27 Гц (рідше від 22 Гц) до 4 кГц,

що можна пояснити тим фактом, що інтервальний компонент звуку сприймається лише в цих межах.

Весь доступний діапазон частот поділяється на октави. В свою чергу музичні звуки всередині октави організовані в музичну систему. В європейській музичній практиці виділяють 12 рівних півтонів всередині октави.

З точки зору акустики, тембр звуку можна визначити за формою акустичного сигналу, тобто за його спектром. Сукупність простих гармонічних коливань утворює спектр звуку. Будь-який акустичний сигнал можна розкласти на суму простих гармонічних коливань за допомогою перетворень Фур'є. Тембральна відмінність різних голосів визначається спектральним розподілом енергії звуку [9].

Музика є невичерпною в своєму тембровому розмаїті, оскільки тембральні відмінності містять не тільки звуки, що виконані на різних музичних інструментах чи різними голосами, але й звуки, які виконані різними виконавськими прийомами на одному й тому ж інструменті.

Гучність — це суб'єктивне уявлення про силу та інтенсивність звуку, яке виникає у свідомості людини, коли вона сприймає звук органами слуху. Гучність має пряму залежність від амплітуди та часткову від частоти коливань (тобто звуки однакові за інтенсивністю, проте різні за частотою сприймаються, як різні по гучності). Якщо інтенсивність однакова, найгучнішими здаються звуки середнього регістру (близько 500-1000 Гц). В загальному сприйнятті сили звуку підпорядковується загальному психофізіологічному закону Вебера-Фехнера.

Оскільки музика є мистецтвом, що залежить від часу, адже музичний твір сприймається у часовому просторі, музичні звуки чітко організовані у часі. Така організація послідовності в часі, групування тривалості музичних звуків і пауз носить назву музичний ритм.

В музичній практиці використовується шкала відносних нотних тривалостей, яка заснована на послідовному поділі цілої ноти — основної нотної тривалості — на два. У відповідності до даного поділу нотні тривалості мають наступні назви: «ціла нота», «половинна нота», «четвертна нота», «восьма нота» і т. д. За потреби основні нотні тривалості можна модифікувати додаванням спеціального знаку —

«крапки». Цей знак подовжує тривалість на половину. Також ще застосовують особливі види ритмічного поділу.

## 1.2 Метод ідентифікації на основі цифрових відбитків

Цифровий відбиток можна представити як компактну цифрову послідовність, що описує найбільш характерні та важливі компоненти аудіосигналу з точки зору психології сприйняття. Застосування методу цифрових відбитків дає змогу проводити порівняння внутрішнього вмісту матеріалів без їх модифікації, а також впровадження водяних знаків чи додаткових метаданих, що описують дані матеріали.

До найбільш важливої властивості цифрових відбитків можна віднести можливість їх використання для класифікації та порівняння сигналів, базуючись на їх перцепційній подібності. Такий підхід дозволяє замінити процес порівняння двох великих за обсягом і можливо досить спотворених сигналів на порівняння відповідних їм більш компактних та стійких до спотворення відбитків.

Під час обробки вихідний сигнал поділяється на послідовні фрагменти, потім для кожного з них генерується цифровий відбиток. Отримані в результаті відбитки зберігаються в базі даних, так само як і інформація про фрагмент, та можуть бути використані у вигляді контекстного індексу для пошуку метаданих.

Використання цифрових відбитків замість вихідних сигналів має ряд переваг, для прикладу, висуває менші вимоги відносно розміру необхідної пам'яті та сховища даних, а також дозволяє проводити більш ефективний пошук в базі даних.

Також досить значною особливістю цифрових відбитків є можливість застосування їх для порівняння сигналів, які були спотворені та містять шуми. Для прикладу, їх можна застосовувати для визначення ступеня спотворення та цілісності матеріалів або ж для виявлення певних модифікацій чи маніпуляцій, що були проведені над вихідним сигналом [10].

Найлегшим способом отримання відбитку медіафайла в такому випадку було би пряме порівняння цифрових послідовностей. Однак, ефективна реалізація даного методу полягає в виявленні хеш-функцій, для прикладу MD5 (Message

Digest), CRC (Cyclic Redundancy Checking) чи SHA-1 (Secure Hash Algorithm) для отримання компактного представлення файлу.

Основним джерелом проблем в процесі отримання якісних цифрових відбитків може виступати велика різниця характеристик сигналів для перцепційно досить схожих чи навіть ідентичних з точки зору людини матеріалів.

Відбиток, отриманий таким чином, описує формат, що містять дані та не може виступати критерієм для точної ідентифікації вмісту цього файлу. Даний спосіб не може використовуватись для ідентифікації навіть мало спотворених чи модифікованих сигналів [11].

### 1.3 Ознаки звуку низького рівня

#### 1.3.1 Швидкість перетину нуля

В рамках сигналів дискретного часу вважають, що перетин нуля відбувається тоді, коли послідовні зразки мають відмінні алгебраїчні ознаки. Вимірювання частотного змісту сигналу є швидкістю, з якою відбувається перетин нуля.

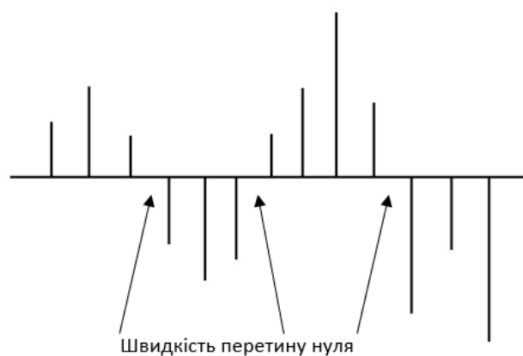


Рисунок 1.2 — Графічне зображення швидкості перетину нуля

Швидкість перетину нуля являє собою показник кількості разів, коли амплітуда мовленнєвих сигналів перетинає значення нуля за певний проміжок часу (рис. 1.2) або кадру (рис. 1.3).

Сигнали мовлення — це ширококутові сигнали, саме тому інтерпретація середньої швидкості перетину нуля менш точна.

Через те, що високі частоти передбачають високу швидкість перетинання нуля, в той час як низькі частоти передбачають зниження рівнів перетину нуля, можна зробити висновок, що існує сильна кореляція між швидкістю перетину нуля та розподілом енергії з частотою.

Таким чином, якщо швидкість перетину нуля перевищує межу, мовленнєвий сигнал не озвучується, а якщо швидкість перетину нуля не перевищує межу, то мовленнєвий сигнал озвучується.

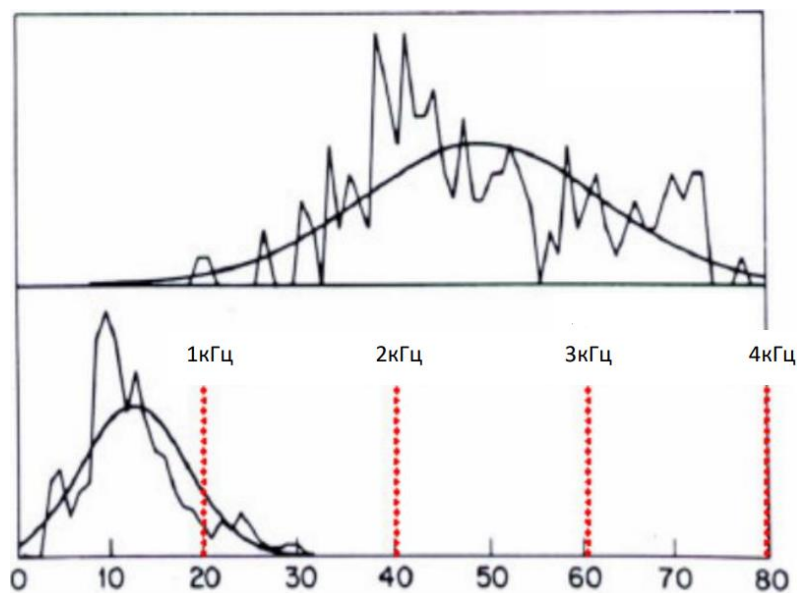


Рисунок 1.3 — Графік розподілу швидкості перетину нуля

Формула для визначення швидкості перетину нуля:

$$Z_n = a_0 + \sum_{m=-\infty}^{\infty} |\text{sgn}[x(m)] - \text{sgn}[x(m-1)]| w(n-m),$$

де  $\text{gn}[x(n)] = \begin{cases} 1, & x(n) \geq 0, \\ -1, & x(n) < 0, \end{cases}$  — функції, що вимірюються.

Модель мовленнєвого виробництва свідчить про те, що енергія вимовленої мови концентрується нижче 3 кГц через падіння спектру, що вводиться через голосову хвилю.

Для «неозвученої» мови більша частина енергії навпаки зустрічається на більш високих частотах [12].

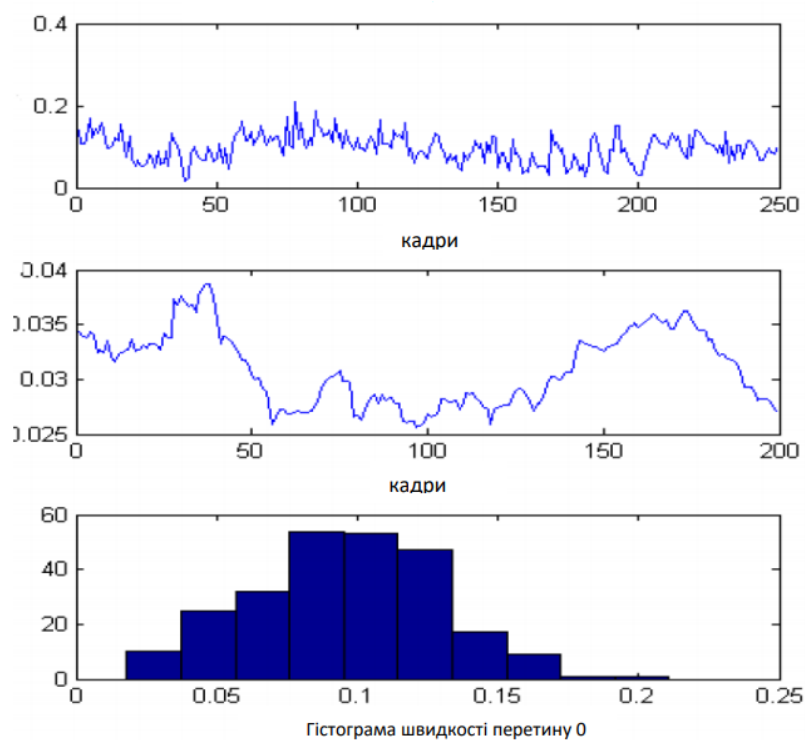


Рисунок 1.4 — Ознаки перетину нуля для музичного відрізка

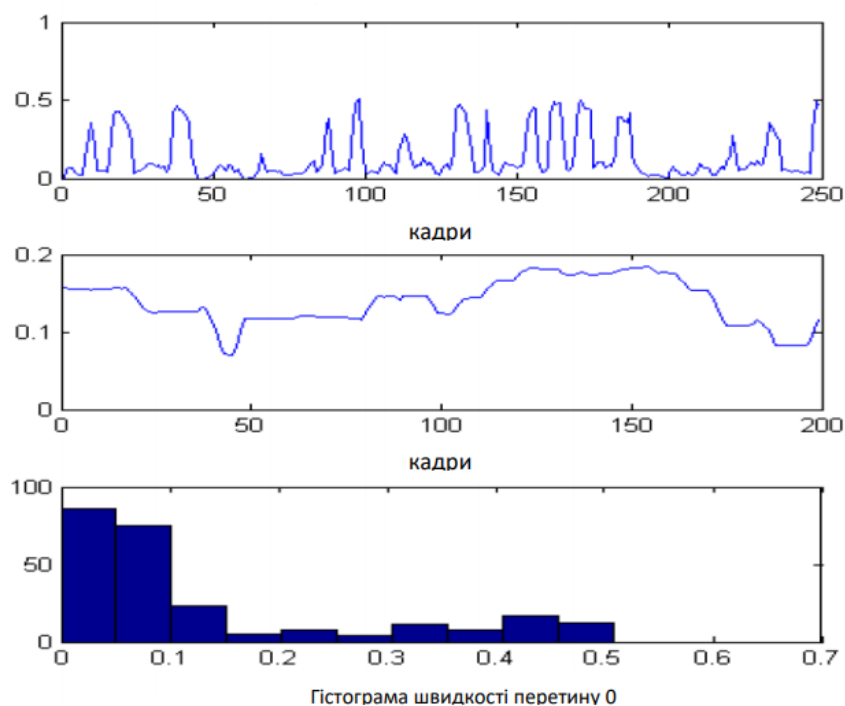


Рисунок 1.5 — Ознаки перетину нуля для мовлення

Хоча середня швидкість перетину нуля відносно однакова як для музики, так і для мовлення, на рис. 1.4 та рис. 1.5 яскраво видно, що розподіл нульових



переходів дуже різний. Музика має досить нормальний розподіл кадрів з більш низькими та високими курсами перетину нуля.

Однак мова відображає більш перекручений розподіл. Це відображено на досить довгих періодах, що містять низьку швидкістю перетину нуля, та відмінними періодами, що містять плавний перехід до значно вищого рівня перетину нуля. Як результат характеристик, середня дисперсія швидкості передачі нульових переходів сигналів мовлення вища, ніж у музичних сигналів.

### 1.3.2 Середньоквадратичне значення

Середньоквадратичне значення являє собою міру амплітуди звукової хвилі в одному вікні аналізу. Розрахувати середньоквадратичне можна за наступною формулою:

$$RSM = \sqrt{\frac{x_1^2 + x_2^2 + \dots + x_n^2}{n}},$$

де  $n$  — кількість зразків у вікні аналізу;

$x$  — значення вибірки.

Середнє та стандартне відхилення гучності запису використовуються як дискриптори розподілу гучності. Також з їх допомогою можна визначити чи фрейм безшумний, порівнюючи його гучність з порогом, що визначається на основі розподілу гучності всього аудіо файлу. В результаті виявлення мовчання можна обрахувати співвідношення тиші, шляхом співвідношення інтервалу мовчання відносно всього періоду. Це співвідношення відрізняється для кожного окремого аудіо файлу.

### 1.3.3 Спектральний центроїд

Дана ознака досить ефективна при описі спектральної форми звуку. Спектральний центроїд корелює з психоакустичними особливостями чіткості та

яскравості. Дана ознака розраховується як зважене середнє значення частот у ШПФ (швидке перетворення Фур'є) за формулою:

$$SC = \frac{\sum_{n=0}^{N-1} f(n)x(n)}{\sum_{n=0}^{N-1} x(n)},$$

де  $f(n)$  — центральну частоту даного відсіку;

$x(n)$  — величина відсіку  $n$ .

### 1.3.4 Спектральний переки

Так само як спектральний центроїд, спектральний роллоф являє собою представлення спектральної форми звуку. Дані ознаки достий сильно корельовані. Спектральний переки можна визначити як частоту, при якій 85% енергії у спектрі нижче даної частоти. Якщо  $K$  — відсік, то частота спектрального перекошу  $f(K)$  обраховується за формулою:

$$\sum_{n=0}^K x(n) = 0.85 \sum_{n=0}^{N-1} x(n),$$

де  $x(n)$  — величина відсіку під номером  $n$ ,

$f(n)$  — центральну частоту цього відсіку.

### 1.4 Основні принципи отримання та використання цифрових відбитків

До методів для отримання цифрових відбитків музичних композицій висувається ряд вимог, основними із яких є отримання максимальної кількості інформації про конкретний звуковий матеріал, нечутливість до спотворень та зручність і компактність при пошуку.

У цифровому відбитку повинен бути максимум значущої інформації, що сприймається людиною. Відбиток повинен мати достатній рівень унікальності для того, щоб відрізнитися при пошуку по базі даних, що містить велику кількість відбитків різних матеріалів, що мають схожі характеристики (для прикладу, музичні твори одного жанру) [13].

Забезпечення компактності, надійності та простоти обчислень. Отримані відбитки повинні бути відносно невеликі за розміром та забезпечувати швидкий пошук в базі даних, що містить значний об'єм інформації (для прикладу, мільйони записів загальною тривалістю більше мільярда секунд).

Нечутливість до різного типу спотворень. Відбитки, які були отримані при аналізі одного й того ж самого сигналу із різних джерел, до яких використовувались цифрові (фільтрація, ущільнення із втратами) та аналогові (запис на магнітну стрічку чи мікрофон, реверберація) спотворення, або при наявності різних завад повинні мати максимальну схожість.

Сама система обробки аудіо відбитків містить в собі два основних компоненти: блок генерації або ж формування відбитків та підсистему пошуку та розпізнавання музичної композиції [14]. Блок-схема системи ідентифікації цифрових відбитків музичних композицій відображена в Додатку В.

До складу системи ідентифікації музичних матеріалів входять такі блоки:

- блок генерації відбитку вхідного аудіо сигналу;
- блок індексації цифрових відбитків та пошуку кандидатів;
- блок перевірки гіпотези та перевірки співпадінь відбитків;
- блок пошуку метаданих та формування звіту про знайдений відбитки музичних композицій.

Як варіант побудови системи ідентифікації звукових матеріалів вона може бути розподіленою у просторі. У такому випадку формування відбитків музичних композицій може бути реалізовано на портативних пристроях у масштабі реального часу в процесі запису для аналізу сигналу на мобільний телефон.

Відбиток, що був створений, має відносно невеликий розмір, далі ці дані передаються від клієнта на сервер, де і здійснюється пошук відповідних метаданих, що можуть бути передані клієнту назад.

Під час пошуку аудіо відбиток співставляється із даними в наявній базі даних. В цьому випадку розраховується відстань між заданим відбитком та найближчим відбитком в базі даних. Оскільки кількість порівнянь відбитків може

бути досить великою, в той час як розрахунок відстані вимагає значних обчислювальних ресурсів, то сам пошук розбивається на декілька етапів [15].

При проведенні першого етапу, за допомогою індексації чи спрощеного методу порівняння відбитків проводиться швидкий пошук можливих кандидатів. Далі здійснюється більш детальний аналіз обраного набору алгоритмів. До систем пошуку та ідентифікації аудіо відбитків висувається ряд вимог:

- швидкість: система проводить пошук кандидатів в базах даних, які містять мільйони записів, тож повинна забезпечувати швидкий пошук;

- точність розпізнавання, яку підсистема порівняння кандидатів мусить забезпечувати та низький рівень помилок, що включають в себе як не ідентифіковані матеріали (FNR, false negative rate), так і такі матеріали, що ідентифіковані неправильно (FPR, false positive rate);

- мінімальні вимоги до задіяних обсягів пам'яті, оскільки більшість систем здійснюють первинний пошук кандидатів, використовуючи таблиці індексів, що займають весь доступний об'єм оперативної пам'яті. Саме тому об'єм пам'яті, який застосовується для цього етапу пошуку, значно впливає на ефективність пошуку та максимальний розмір бази даних.

Існуючі на теперішній час методи отримання аудіовідбитків музичних композицій можна розділити на три великі групи:

- методи, що враховують локальні характеристики звукового сигналу;
- методи, що аналізують довгострокові характеристики звукового сигналу;
- методи на основі прихованих Марківських моделей [16].

Методи, що враховують локальні характеристики звукового сигналу, ґрунтуються на формуванні послідовності векторів ознак (траєкторії в евклідовому просторі), що обчислюються через рівні проміжки часу.

Прикладом такого підходу може виступати величина спектральної рівномірності, яку використовують у стандарті MPEG-7.

Для обчислення відстані між елементами відбитків звукових композицій у багатьох випадках використовують відстань за Хеммінгом. Пошук та виконання операції порівняння отриманої послідовності елементів відбитків звукових

композицій може здійснюватися за допомогою методу динамічного програмування (Dynamic Time Warping, DTW).

Інший підхід, такий як довгострокові характеристики аудіо сигналу, використовує середнє значення та дисперсію параметрів, що отримані для всього музичного твору, або ж таких, як ритм або частота домінуючих тонів звукового файлу.

Такі характеристики не потребують точного позиціонування при виконанні операції порівняння, але мають досить низьку стійкість до завад та спотворень. Крім того, такий підхід не дозволяє порівнювати окремі фрагменти музичного твору, а також порівнювати звукові матеріали, що передаються у масштабі реального часу у вигляді неперервного потоку звукових даних.

Ще один підхід, це приховані Марківські моделі. Не зважаючи на високу ефективність та поширеність використання прихованих Марківських моделей у різних алгоритмах розпізнавання мовлення та здійснених спробах їх використання для розпізнавання випадкових аудіосигналів, на даний час не існує ефективних систем формування цифрових відбитків, що ґрунтуються на цьому методі.

Виходячи із проведеного аналізу розглянутих підходів до формування відбитків музичних композицій можна констатувати, що найкраще проблему розпізнавання аудіо матеріалів вирішують такі методи, що засновані на аналізі локальних параметрів звукових сигналів та визначають структуру процесу виділення набору ознак аналізованого аудіосигналу.

### 1.5 Delta Spectrum Magnitude

Величина зміни спектру або потік (Delta Spectrum Magnitude, DSM) — це показник швидкості, з якою спектральна форма змінюється чи коливається.

Його можна розрахувати шляхом підрахунку квадратів різниць величини спектра віконного перетворення Фур'є двох сусідніх кадрів за формулою:

$$F = \sum_{k=1}^{N/2} (|x_r[k]| - |x_{r-1}[k]|)^2,$$

де  $N$  — кількість точок FFT;

$x_r(k)$  — віконе перетворення Фур'є (STFT — Short-time Fourier transform) кадру  $r$  у відсіку  $k$ .

### 1.5.1 Кепстральні частотні коефіцієнти

Кепстральні частотні коефіцієнти (FCC — Frequency Cepstrum Coefficients) включають в себе Мел-частотні кепстральні коефіцієнти (MFCC — Mel Frequency Cepstral Coefficients) та логарифмічні кепстральні частотні коефіцієнти (LFCC — Logarithmic Frequency Cepstrum Coefficients). Все це ознаки представлення спектра потужності, розраховані з використанням різних частотних шкал.

Мел-частотний кепстральний аналіз застосовують для обробки звуку та розпізнавання мови. Мел — це психофізична одиниця висоти звуку, що пов'язана з частотою (в герцах) через:

$$mel = 2595 \log_{10} \left( 1 + \frac{f}{700} \right),$$

Кепстр є результатом взяття оберненого перетворення Фур'є від логарифма спектру сигналу, тобто нелінійним «спектром спектру».

Різниця між кепстром та мел-частотним кепстром полягає в тому, що в мел-частотному кепстрі смуги частот розміщені за шкалою Мела, що більше відповідає людській слуховій системі, ніж лінійно розміщені смуги частот, які використовуються у звичайному кепстрі

Коефіцієнти MFCC обчислюються шляхом прийому FFT кожного вікна аналізу, приведенням спектру до шкали Мела, беручи 10 основних логарифмів потужності, а потім застосування дискретної косинусової функції (DCT — Discrete Cosine Function) для декорелювання коефіцієнтів.

Загальна ефективність функцій MFCC трохи краща ніж SLL-функції.

Ефективність MFCC пов'язана з тим, що дана функція краще працює краще з поп- та рок-музикою, але дещо гірше з класичною музикою, яка містить дуже мало голосової інформації.

### 1.5.2 Психоакустичні ознаки

Дані ознаки більш тісно пов'язані з тим, як саме людина сприймає звук, тому мають назву психоакустичні.

Гучність — це відчуття сили сигналу та суб'єктивна людська міра, щоб оцінити звуки від слабких до більш сильних. Гучність може бути розрахована (Calculated Loudness), а далі виміряна в сонах (Sone). Один сон визначається як гучність чистого сигналу 1000 Гц при 40 дБ відносно 20 мкПа.

Шорсткість описується як "сприйняття тимчасової модуляції оболонки в діапазоні близько 20-150 Гц, максимальна при 70 Гц" та також вважається основною складовою музичного дисонансу.

Різкість — це міра високочастотної енергії, що пов'язана з низькочастотною енергетичною силою. Звуки з великою енергією у високих частотах та низькі енергетичні рівні в низьких частотах вважаються різкими.

### 1.5.3 Спеціальні ознаки

До спеціальних ознак можна віднести хроматичну ентропію, що є різновидом спектральної ентропії. Спектр спершу накладається на шкалу Мела, а далі поділяється на дванадцять піддіапазонів з центральними частотами, що співпадають з частотами хроматичної шкали. Енергія у кожному піддіапазоні потім нормалізується загальною енергією всіх піддіапазонів. Ентропія нормованої спектральної енергії розраховується за такою формулою:

$$E = - \sum_{i=0}^{L-1} n_i * \log_2(n_i),$$

де  $n_i$  — нормалізована енергія піддіапазону  $i$ ;

$L$  — кількість піддіапазонів.

Ознака модифікованого коефіцієнту низької енергії (MLER — Modified Low Energy Ratio) використовує той факт, що музика показує невелику зміну енергетичного контуру сигналу, в той час як мовлення показує великі варіації між звучанням і фрикацією. MLER визначається як частка кадрів з силою RMS,

меншою, ніж змінний поріг протягом однієї секунди. Для досягнення найкращої продуктивності пороговий показник повинен бути в інтервалі [0,05% — 0,12%].

Наступна ознака має назву Warped LPC Spectral Centroid (WLPC-SC). Частотний аналіз наноситься на шкалу Барка (Bark scale), а далі частота центроїда обчислюється однополіосним LPC-фільтром[17].

#### 1.5.4 Психоакустичний крок шкали

Психоакустичні ваги широко застосовуються в системах пошуку музичної інформації (MIR — Music Information Retrieval). Мовлення та музика часто добре підлаштовані під наш слух таким чином, щоб інформація знаходилась на частотах, де наш слух має найкращу роздільну здатність. Найбільш використана шкала — Мел, але іноді використовуються навіть Bark та еквівалентна прямокутна смуга пропускання (ERB — Equivalent Rectangular Bandwidth). Мел-частота визначається за формулою:

$$f = 1127.01048 * \log\left(\frac{f_1}{700} + 1\right),$$

де  $f_1$  — звичайна частота.

#### 1.5.5 Визначення ознак

Зазвичай вибраний набір ознак витягується з кожного кадру звуку. Далі ознаки часто нормалізуються шляхом визначення середнього значення і стандартного відхилення за більшу одиницю часу, а потім ознака зберігається у векторі.

Можна використовувати зміни у часі, які є більш точними, ніж просте використання абсолютних значень функцій. Також використовуються п'ять різних ознак: енергія, спектральна ентропія, ZCR та дві перші MFCC.

Виходячи із розглянутого, кращим методом формування аудіовідбитків вибираємо локальні методи, серед яких за основу буде взято метод Frequency Cepstral Coefficients.



## 2 РОЗРОБКА ТЕХНОЛОГІЇ ФОРМУВАННЯ АУДІОВІДБИТКІВ МУЗИЧНИХ КОМПОЗИЦІЙ

### 2.1 Розробка математичної моделі виділення аудіофрагментів

#### 2.1.1 Обґрунтування вибору MFCC

Параметри математичної моделі, що використовується для інформаційного пошуку аудіозаписів, повинні задовольняти наступним вимогам:

— компактність та некорельованість, що дозволить зменшити необхідні обсяги пам'яті для зберігання аудіозаписів та в подальшому впливатиме на швидкість пошуку в архівах;

— однакова дисперсія в середньому, тобто однорідність параметрів;

— підсилення корисного сигналу та стійкість до шумів за рахунок використання нерівномірної шкали частот та нелінійного сприйняття гучності (децибели), які актуалізують низькочастотну інформацію (корисний сигнал) і усереднюють високочастотні складові сигналу (де зосереджується шумова складова сигналу);

— можливість використання для аналізу різних звуків (музики, мовлення, оточуючих звуків), що дозволить забезпечити універсальність підходу при розв'язанні різних задач на базі визначення подібності аудіозаписів;

— висока точність розбиття на класи за категоріями (музика, мовлення, тиша);

— невелика обчислювальна складність отримання параметрів;

— можливість застосування простих метрик (евклідова метрика) для визначення близькості між наборами параметрів;

— можливість застосування кластерного аналізу;

— можливість застосування методів швидкого пошуку [18].

Враховуючи розглянуті вище вимоги та відповідно до проведеного в першому розділі аналізу було вирішено здійснювати параметризацію аудіозаписів з використанням мел-частотних кепстральних коефіцієнтів (MFCC — Mel Frequency Cepstral Coefficients).

#### 2.1.2 Дослідження основних характеристик MFCC

Для створення алгоритму обчислення коефіцієнтів MFCC в задачі пошуку аудіовідбитків музичних композицій у звукових файлах необхідно визначитися з набором основних характеристик, а саме: розміром перекриття фреймів, частотою дискретизації аудіофайлу, довжиною фрейму, для якого обчислюється вектор MFCC, кількістю трикутних фільтрів, а також кількістю коефіцієнтів MFCC.

Частота дискретизації. Підвищення частоти дискретизації збільшує якість опису аудіосигналу, що в подальшому може позитивно вплинути на повноту пошуку в корпусі, проте з іншого — збільшується кількість відліків, які описують аудіосигнал, тобто зростає розмір аудіофайлу [19]. Дана перевага має великий вплив на формування корпусу аудіозаписів високої якості; в той час як недолік не має впливу на розмір аудіозаписів корпусу чи час пошуку, за рахунок специфіки запропонованого підходу, описаного нижче в роботі (кластерний аналіз з однаковою кількістю кластерів та пошук аудіозапису за його фрагментом).

Підвищення якості подання аудіосигналу пов'язано зі збільшенням пропускнуої спроможності сигналу, в результаті чого використовуються більш високі частотні компоненти, ніж з низькою частотою дискретизації. Враховуючи це під час формування корпусу аудіозаписів з метою забезпечення належного рівня якості опису аудіосигналу було обрано частоту дискретизації 44,1 кГц.

Довжина фрейму. Вибір довжини фрейму обумовлений тим, що обчислення коефіцієнтів MFCC має сенс тільки на стаціонарних фрагментах. Стаціонарність процесу зберігається на невеликих фрагментах — не більше 30 мс (фрейми більшої довжини мають більше переходів і варіацій).

Довжина фрейму також не може бути малою, оскільки при цьому буде недостатньо статистики для виконання перетворень, необхідних для отримання параметрів. Тому, зазвичай, довжина фрейму становить 10-30 мс (зростання довжини фрейму в цих межах має позитивний, хоча і невеликий, вплив на якість подання аудіосигналу). В роботі було обрано середнє значення з цього діапазону, тобто довжина фрейму становить 20 мс.

Крім того, фрейми беруться з перекриттям в 50%, тобто крок обчислення параметрів складає 10 мс. Необхідність в перекритті фреймів полягає в тому, щоб запобігти втратам інформації про аудіосигнал на межах фреймів.

Кількість коефіцієнтів (розмірність) MFCC, розрахованих на кожному фреймі. Зменшення розмірності параметрів опису аудіозапису є найбільш простим способом скорочення обчислювальних витрат і складності системи для подальшої обробки.

Апроксимацією КЛ-перетворення (що дає найбільш адекватний (якісний) результат, але є обчислювально складним і трудомістким в реалізації), для зменшення розмірності параметрів, може служити дискретне косинусне перетворення. Оскільки в нашому випадку DCT є останнім етапом отримання MFCC-параметрів, не потрібно виконувати жодних додаткових операцій для зменшення розмірності коефіцієнтів MFCC [20].

В більшості випадків обчислюють лише молодші кепстральні коефіцієнти для отримання компактного опису аудіозапису. Зазвичай зберігають тільки 10-15 молодших коефіцієнтів MFCC, а інші відкидаються для того, щоб забезпечити незалежність отриманих параметрів від частоти основного тону, яка присутня в старших коефіцієнтах.

Чим більше коефіцієнтів MFCC застосовується, тим точніше описується спектр сигналу. Разом з тим посилюється вплив швидких змін спектра, які корелюють з частотою основного тону. Це створює небажану мінливість даних. Молодші коефіцієнти описують загальну форму спектра, тоді як інформація про частоту основного тону та спектральну структуру (spectral fine structure) міститься в старших коефіцієнтах.

Зауважимо, що перший з цих коефіцієнтів має фізичний зміст середньої енергії сигналу на взятому проміжку часу. Другий параметр MFCC є важливою характеристикою, цей параметр можна інтерпретувати як відносні рівні низько- і високочастотної енергії в сигналі. Коефіцієнти 6-13, які описують вхідний спектр на досить детальному рівні, можуть виявитися надлишковими. Для врахування

динамічних властивостей всієї спектральної обвідної використовують похідні кепстральних коефіцієнтів.

Отже, оскільки нас цікавить тільки спектральна обвідна, а не частота основного тону, а також компактність опису аудіозаписів, використання великої кількості коефіцієнтів MFCC не є доцільним.

Для формування бази параметрів аудіозаписів корпусу використовувався кластерний аналіз, що буде описано в розділі 2.2. В Додатку Г показано залежність похибки кластеризації від кількості коефіцієнтів MFCC.

Похибка кластеризації є початковою похибкою для кожного аудіозапису корпусу і чим меншою вона буде, тим більш точно буде поданий аудіозапис, відповідно вищою буде якість створеного корпусу.

Отримані результати, зображені в Додатку Г, показують, що спочатку похибка кластеризації аудіозаписів в корпусі зі збільшенням кількості коефіцієнтів MFCC помітно зменшується, але при досягненні розмірності тринадцять, зміни похибки кластеризації стають зовсім незначними (<1%).

З огляду на це під час формування корпусу аудіозаписів для виконання кластеризації з мінімальною початковою похибкою як базову розмірність простору було обрано 13 коефіцієнтів MFCC.

Таким чином, для 3 хв (180 с) аудіозапису з частотою дискретизації аудіофайлу 44,1 кГц, довжиною фрейму 20 мс з перекриттям 0,5 фрейму, маємо:

$$44100 \text{ відліків/с} * 180 \text{ с} = 7\,938\,000 \text{ відліків},$$

$$44100 \text{ відліків/с} * 20 \text{ мс/фрейм} = 882 \text{ відліків/фрейм},$$

$$7\,938\,000 \text{ відліків} / 882 \text{ відліків/фрейм} = 9000 \text{ фреймів}$$

$$(\text{або } 180 \text{ с} / 20 \text{ мс/фрейм} = 9000 \text{ фреймів}).$$

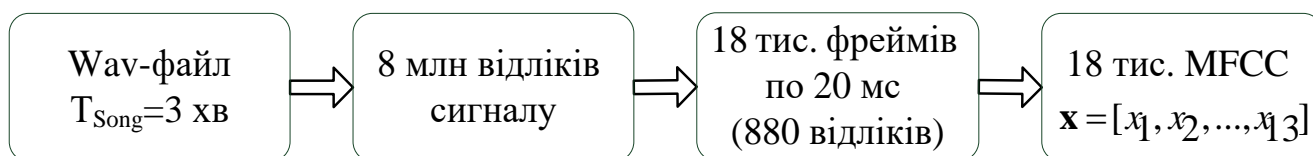


Рисунок 2.1 — Опис wav-файлу (3 хв)

Виходячи з наведених розрахунків аудіозапис (рис. 2.1) характеризують приблизно 8 млн відліків, або, з урахуванням перекриття фреймів, приблизно 18 тис. фреймів ( $9000 \text{ фреймів} * 2 - 1$ ), кожен з яких описується вектором параметрів MFCC розмірності 13.

Отже, параметризація дозволяє зменшити кількість інформації, необхідної для опису аудіозапису, в десятки разів:

$$\frac{7\,938\,000 \text{ відліків}}{17\,999 \text{ MFCC} * 13} = \frac{7\,938\,000}{233\,987} \approx 34 \text{ рази}$$

Таким чином, MFCC є компактним поданням спектральної обвідної, що дозволяє успішно замінити мільйони відліків аудіофайлу для виконання пошуку відповідного аудіозапису в корпусі.

### 2.1.3 Загальна схема обчислення MFCC

Схему отримання MFCC зображено в Додатку Д.

Спочатку вхідний аудіосигнал підсилюється, щоб згладити спектр. Попереднє підсилення сигналу зазвичай застосовується в системах розпізнавання мовлення, для таких сигналів, як музика, це не завжди може бути корисним, тому в Додатку Д цей крок відсутній.

Отже, обчислення параметрів MFCC відбувається в декілька етапів.

Поділ відліків аудіозапису на фрейми.

Множення кожного з  $x_0, \dots, x_{N-1}$  відліків фрейму на вагову функцію — вікно Хеммінга має вигляд:

$$w_n = 0.54 - 0.46 \cos\left(2\pi \frac{n}{N-1}\right), n = 0, \dots, N-1,$$

де  $N$  — кількість відліків, що припадає на фрейм.

Застосування вагової функції дозволяє послабити "розтікання" спектра на межі фреймів [21]. Виконання швидкого перетворення Фур'є, коефіцієнти якого (амплітуди спектра) обчислюють за формулою:

$$x_k = \sum_{n=0}^{N-1} x_n * w_n * e^{\frac{2\pi i}{N}kn}, k = 0, \dots, N - 1,$$

де  $k$  — індекс частоти.

Після застосування перетворення Фур'є в частотному поданні основну інформацію (корисний сигнал) несуть низькі частоти. Високі частоти описують шум і несуттєві деталі.

Видалення 50% високочастотної інформації спричинить видалення 5% корисної інформації, що міститься в аудіосигналі.

Частота  $k$ -го сигналу обраховується наступним чином:

$$f_k = \frac{F_s}{N}k, k = 0, \dots, N/2,$$

де  $F_s$  — частота дискретизації сигналу.

Переведення значень частот в мел-шкалу обраховується так:

$$m = 2595 \log_{10}\left(\frac{f}{700} + 1\right)$$

Шкала мел-частот є лінійною для частот нижче 1 кГц і логарифмічною шкалою для частот вище 1 кГц. Отже, MFCC актуалізують низькочастотну інформацію і усереднюють високочастотні складові сигналу.

Обчислення потужності спектра за формулою (2.1) відбувається шляхом обчислення квадрата абсолютних значень, отриманих після перетворення Фур'є, та накладення на обчислені значення амплітуд спектра набору з  $N_{FB}$  трикутних вікон мел-шкали, що перекриваються і рівномірно розміщуються на шкалі мел-частот.

$$S_j = \sum_{k=0}^{N-1} |x_k|^2 H_j(m_k), j = 0, \dots, N_{FB} - 1, \quad (2.1)$$

де  $N_{FB}$  — кількість трикутних фільтрів.

Перемноження спектра потужності з кожним трикутним фільтром і обчислення суми значень в кожному каналі, показане вище, дозволяє отримати значення спектральної енергії кожного каналу.

Логарифмування зміненого спектра з метою ущільнення його динамічного діапазону обчислюється:

$$S_j = \ln(S_j), j = 0, \dots, N_{FB} - 1,$$

Наступним етапом буде виконання дискретного косинусного перетворення для зменшення кількості вихідних параметрів та декореляції компонентів за формулою:

$$MFCC_i = \sum_{j=0}^{N_{FB}-1} S_j \cos\left(\frac{\pi i(j+0.5)}{N_{FB}}\right), i = 1, \dots, N_{MFCC},$$

Обравши за параметри MFCC, ми отримуємо опис аудіозапису у параметризованому вигляді, що проілюстровано на рис. 2.2.

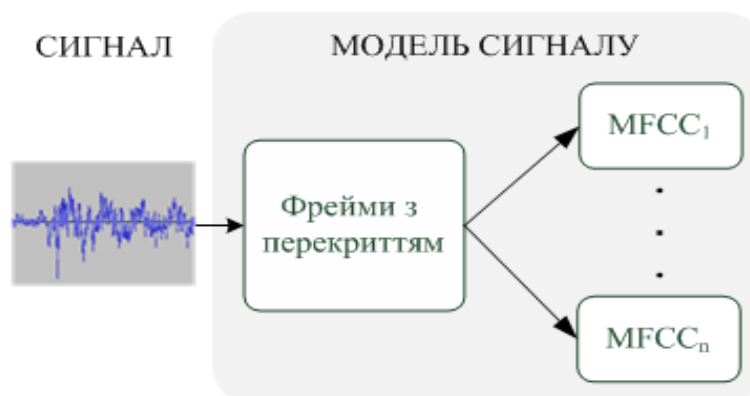


Рисунок 2.2 — Математична модель аудіосигналу MFCC

## 2.2 Обґрунтування застосування гаусівських змішаних моделей

Моделювання багатовимірних даних через опуклу суміш гаусіан, досить відоме як Гаусівські змішані моделі, має досить багато застосувань в різних областях, зокрема в обробці сигналів, економетриці, розпізнаванні образів та машинному навчанні. Гаусівські змішані моделі також досить широко застосовуються як розподіл ймовірності векторів ознак для прихованих моделей Маркова.

Гаусівські змішані моделі можуть використовуватись для багатоступінчатого вилучення ознак при розпізнаванні дій, неявного розсіяного

кодування для розпізнавання обличь, моделювання проміжних характеристик, ідентифікації особи на основі мовлення або ж ймовірнісної оцінки переднього плану для систем спостереження.

Вважається, що гаусова суміш це щось середнє між методом векторного квантування, в якому розподіл ознак представлений дискретною безліччю шаблонів, та одним гауссовим розподілом, що має єдиний вектор математичних сподівань та коваріаційну матрицю. Можна сказати, що вектори ознак незалежні один від іншого, тоді щільність спостереження векторів, які утворюють певний клас, можна назвати сумішшю гасівських розподілів.

Приклад графіку гаусівських змішаних моделей, що містять ваги компонентів, можна побачити на рис. 2.3.

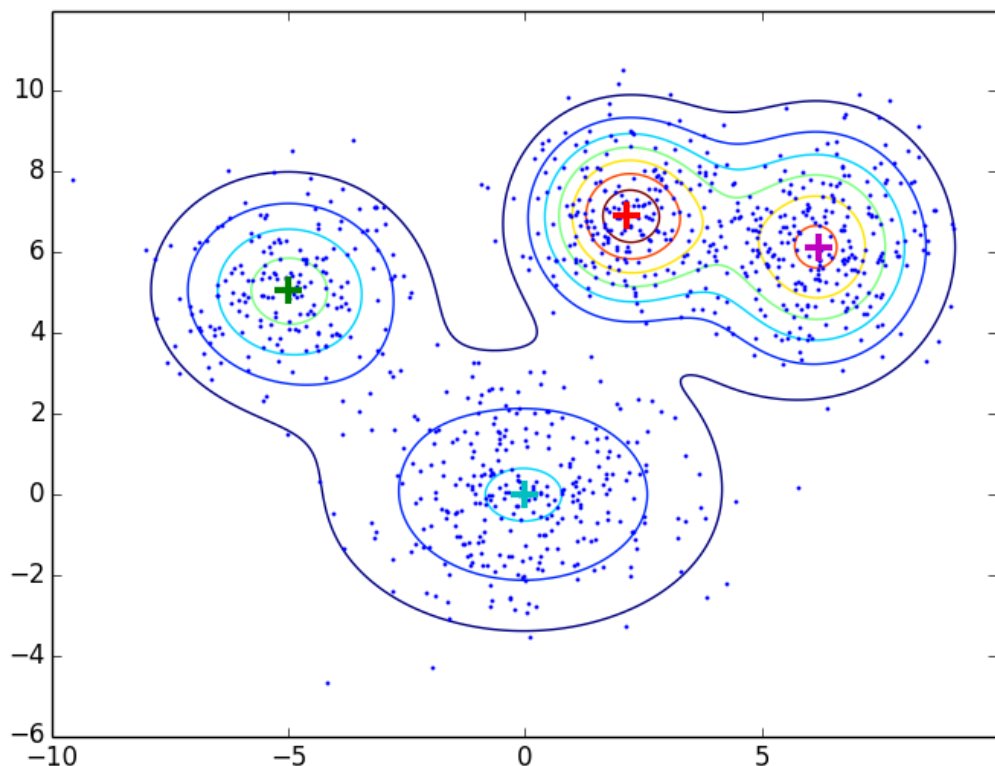


Рисунок 2.3 — Гаусівські змішані моделі

В підході змішаних гаусівських моделей розподіл векторів моделюється наступним чином:

$$p(x|\lambda) = \sum_{g=1}^{N_G} w_g N(x|\mu_g, \Sigma_g),$$



де  $x$  — це  $D$ -мірний вектор;

$w_g$  — вага компонента  $g$  (з обмеженням  $\sum_{g=1}^{N_G} w_g = 1, w_g \geq 0$ );

$N(x|\lambda, \Sigma)$  — функція (2.2)  $D$ -розміреної гаусівської щільності з середнім  $\mu$  та коваріаційною матрицею  $\Sigma$ .

$$N(x|\mu, \Sigma) = \frac{1}{(2\pi)^{\frac{D}{2}} |\Sigma|^{\frac{1}{2}}} \exp \left[ -\frac{1}{2} (x - \mu)^T \Sigma^{-1} (x - \mu) \right], \quad (2.2)$$

де  $|\Sigma|$  та  $\Sigma^{-1}$  — детермінант та інверсне  $\Sigma$  відповідно,

$x^T$  — переміщення  $x$ .

Повний набір параметрів можна компактно визначити як:

$$\lambda = \{w_g, \mu_g, \Sigma_g\}_{g=1}^{N_G},$$

де  $N_G$  — це число гаусіан.

Враховуючи набір початкових даних для тренування та значення  $N_G$ , оцінка  $\lambda$  зазвичай робиться за допомогою спеціального алгоритму максимізації очікувань (EM — Expectation Maximisation).

Можна виділити дві причини використання змішаних гаусівських моделей для виділення музики.

Окремі компоненти моделі мають змогу моделювати певну множину акустичних ознак чи подій.

Припустимо, що акустичний простір музики можна характеризувати великою кількістю класів, що являють деякі музичні події/звуки відповідно.

Спектр акустичного класу може бути представлений вектором математичного очікування, а зміна середнього спектру може бути представлена коваріаційною матрицею. Оскільки звуки, які використовуються для навчання та розпізнавання, не мають міток, то події «зашифровані» в класах акустичного простору. Вважаючи, що вектори ознак не залежать один від іншого, щільність спостереження векторів, що утворюють певний клас, можна вважати сумішню гаусівських розподілів [22].

До інших причин використання гаусівських змішаних моделей для виділення музики можна віднести емпіричне спостереження, що лінійна комбінація гаусівських розподілів може являти собою більше число класів акустичних ознак.

Одна із сильних сторін суміші гаусівських моделей — це те, що ці моделі можуть дуже точно апроксимувати випадкові розподіли. Модель змішаних гаусових розподілів являє собою гібрид між моделями векторного квантування та гаусівських розподілів, оскільки використовує дискретну множину гаусівських функцій. Кожна функція має власну величину вектора математичних очікувань і коваріаційну матрицю.

### 2.3 Формування картотеки аудіосигналів із застосуванням EM-алгоритму

EM-алгоритм, або як його ще називають алгоритм максимізації очікувань, широко застосовується як підхід до ітеративних обчислень оцінок максимальних ймовірностей (Maximum likelihood estimation — MLE), які корисні для різних неповних даних. Оцінка максимальної ймовірності та висновки, зроблені на основі даної ймовірності, займають центральне значення в статистичній теорії та аналізі даних [23].

EM-алгоритм являє собою алгоритм, який багатократно повторюється, та в кожному повторі існує фіксовані два кроки: крок очікування та крок максимізації. EM-алгоритм отримав своє ім'я від Демпстера в 1977 році.

Демпстер синтезувавши попередні формулювання алгоритму в більшості конкретних випадках та представив загальне формулювання даного методу знаходження оцінки максимальної ймовірності в різних задачах, а також надав початковий каталог, де цей метод можна застосовувати.

При використанні статистичних методів часто можна зустрітись із проблемою оцінки параметрів, у випадках коли функція правдоподібності ускладнена, що спричиняє складнощі з розрахунком задач максимізації.

Для прикладу можна навести деякі згруповані, переглянуті чи урізані дані, багатоканальні частотні дані, що містять складну вірогідну структуру елементів, багатоваріантні дані з частково відсутніми результатами спостережень або дані з

сумішей розподілів. Більшу частину з цих задач можна об'єднати в одну статистичну задачу, яка має схожі параметри з «розширеними даними», за допомогою яких можна розробити MLE більш просто.

Розширені дані також можна назвати «повними даними», тоді як наявні дані — «неповними даними», а відповідні ймовірності — «ймовірність повних даних» та «ймовірність неповних даних», тоді як оцінка максимальної ймовірності буде відповідно «задача повних даних» та «задача неповних даних».

У рамках роботи з неповними даними з EM-алгоритмом, позначимо  $x$  як вектор, що містить повні дані, та  $z$  як вектор, що містить відсутні дані. Навіть якщо спочатку здається, що задача не пов'язана з неповними даними, зазвичай обчислення MLE є значно простішим, якщо сформулювати задачу таким чином. Це можна пояснити тим, що EM-алгоритм використовує знижену складність оцінки максимальних ймовірностей, якщо враховувати повні дані [24].

Нехай  $g_c(x; \psi)$  позначає функцію щільності ймовірності для випадкового вектора  $X$ , що відповідає певному вектору даних  $X$ . Тоді функція ймовірностей повних даних може бути сформована для  $\psi$ , якщо  $x$  повністю спостережувана і задається:

$$\log L_C(\psi) = \log g_c(x; \psi).$$

EM-алгоритм наближається до задачі розв'язання рівняння ймовірності неповних даних побічно шляхом ітеративного проходження в межах  $\log L_C(\psi)$ . Оскільки її неможна спостерігати, вона замінюється на її умовне очікування задане  $u$ , використовуючи поточне значення для  $\psi$ . На  $k+1$  ітерації EM-алгоритму, крок очікування (E-крок) вираховується за формулою:

$$Q(\psi; \psi^{(k)}) = E_{\psi^{(k)}}\{\log L_C(\psi) | y\}.$$

Крок максимізації (M-крок) вибирає  $\psi^{(k+1)}$  для будь-якого значення  $\psi \in \Omega$ , що максимізує  $Q(\psi; \psi^{(k)})$  за формулою:

$$Q(\psi^{(k+1)}; \psi^{(k)}) \geq Q(\psi; \psi^{(k)}) \forall \psi \in \Omega.$$

Е- та М-кроки неодноразово чергуються, допоки не досягнуть конвергенції, яку можна визначити за допомогою відповідного правила зупинки, як наприклад,  $\|\psi^{(k+1)} - \psi^{(k)}\| < \varepsilon$  для деякого  $\varepsilon > 0$  з деякими відповідними нормами  $\|\cdot\|$  або різниця  $L(\psi^{(k+1)}) - L(\psi^{(k)})$ , що змінюється на досить малу величину у випадку збіжності послідовності значень ймовірності  $\{L(\psi^{(k)})\}$ .

Е- та М-кроки матимуть досить прості форми, коли  $g_c(x; \psi)$  належить експоненціальній сім'ї за виразом:

$$g_c(x; \psi) = \frac{b(x) \exp\{c^T(\psi)t(x)\}}{a(\psi)},$$

де  $t(x)$  — це  $k \times 1$  ( $k \geq d$ ) вектор повних даних достатньої статистики,

$c(\psi)$  — це  $k \times 1$  векторна функція  $d \times 1$  параметричного вектора  $\psi$ ,

$a(\psi)$  та  $b(\psi)$  — це скалярні функції.

Члени експоненціальної сім'ї включають в себе найбільш поширені розподіли, такі як багатовимірний нормальний розподіл, розподіл Пуассона і т. д. Для експоненціальної сім'ї Е-крок можна записати виразом [25]:

$$Q(\psi; \psi^{(k)}) = E_{\psi^{(k)}}(\log b(x)|y) + c^T(\psi)t^{(k)} - \log a(\psi),$$

де  $t^{(k)} = E_{\psi^{(k)}}\{t(X)|y\}$  — це оцінка достатньої статистики.

М-крок максимізує Q-функцію відповідно до  $\psi$ ; проте  $E_{\psi^{(k)}}(\log b(x)|y)$  не залежить від  $\psi$ .

Отже, достатньо буде вказати, що Е-крок обраховується за виразом:

$$t^{(k)} = E_{\psi^{(k)}}\{t(X)|y\},$$

М-крок рахується за формулою:

$$\psi^{(k+1)} = \arg \max_{\psi} [c^T(\psi)t^{(k)} - \log a(\psi)],$$

Набір параметрів  $\lambda$ , що відповідним чином моделює основний розподіл  $X$ , можна оцінити, використовуючи відповідний екземпляр ЕМ-алгоритму.

Параметри GGM, сформовані попередньою ітерацією ( $\lambda^{old}$ ), використовуються поточною ітерацією для створення нового набору параметрів ( $\lambda^{new}$ ), таких що  $p(X|\lambda^{new}) \geq p(X|\lambda^{old})$ .

В безпосередній реалізації EM-алгоритму з урахуванням GMM за формулою (2.3), розрахункова версія параметрів  $\widehat{w}_g$  (2.4),  $\widehat{\mu}_g$  (2.5),  $\widehat{\Sigma}_g$  (2.6) в межах однієї ітерації обчислюються наступним чином:

$$L_g = \sum_{i=1}^{N_V} l_{g,i} = l_{g,i} = \frac{w_g N(x_i|\mu_g, \Sigma_g)}{\sum_{k=1}^{N_G} w_k N(x_i|\mu_k, \Sigma_k)}, \quad (2.3)$$

$$\widehat{w}_g = \frac{L_g}{N_V}, \quad (2.4)$$

$$\widehat{\mu}_g = \frac{1}{L_g} \sum_{i=1}^{N_V} x_i l_{g,i}, \quad (2.5)$$

$$\widehat{\Sigma}_g = \frac{1}{L_g} \sum_{i=1}^{N_V} (x_i - \widehat{\mu}_g)(x_i - \widehat{\mu}_g)^T l_{g,i}, \quad (2.6)$$

Коли знайдені розрахункові параметри для всіх гаусіан, параметри оновлюються та ітерація починається спочатку. Процес зазвичай повторюється доти, поки кількість ітерацій не досягне попередньо визначеного числа, також загальна ймовірність збільшується після того, як кожна ітерація падає нижче попереднього визначеного порогу.

## **3 ПРОГРАМНА РЕАЛІЗАЦІЯ ТА ЕКСПЕРИМЕНТАЛЬНЕ ДОСЛІЖЕННЯ ТЕХНОЛОГІЇ**

### **3.1 Мова програмування**

Кожна із мов програмування не є об'єктивно кращою в порівнянні з іншими. Адже перед початком кожного нового проекту вирішується, яку з мов програмування використовувати. Мови низького рівня краще підходять для оптимізації, швидкості виконання або зменшення розміру програми, тоді як мови високого рівня гарні для створення ясного та добре структурованого коду.

В ході проведеного дослідження було виявлено дві основні мови програмування, що найкраще підходять для реалізації поставленого завдання — Java та C++.

#### **3.1.1 Мова Java**

Мова Java була випущена 1995 року компанією «Sun Microsystems». Java відноситься до об'єктно-орієнтованих мов програмування. Мова Java спочатку виникла як частина проекту, що був направлений на розробку передового програмного забезпечення для побутових пристроїв. Реалізація проекту була розпочата мовою програмування C++, та згодом утворилася група перешкод, найкращим способом боротьби з якою стала заміна самого інструмента, а саме мови програмування. Стало очевидно, що потрібна платформи-незалежна мова програмування, що дасть змогу розробляти програми, в яких не буде необхідності компілювати порізно для кожної окремої архітектури, а можна буде застосовувати на різноманітних процесорах під різними операційними системами [26].

Програмний код на Java транслюються в байт-код, що виконує віртуальна java-машина (JVM). Вона обробляє байтовий код та віддає інструкції обладнанню як інтерпретатор, лише з тією різницею, що байтовий код обробляється значно швидше, в порівнянні з текстом. Перевагою такого способу виконання програм є абсолютна незалежності байт-коду від операційної системи та устаткування, що

дає можливість запускати Java-додатки на будь-якому пристрої, що підтримує віртуальну машину.

Додатки можуть переноситись на безліч різних платформ. Додаток потрібно написати лише один раз і більше не доведеться переписувати під інші платформи: він буде здатний працювати без будь-яких змін на різноманітних операційних системах та апаратних архітектурах. Мова Java є об'єктно-орієнтованою, та водночас досить простою мовою програмування.

Розробнику не обов'язково тривалий час вивчати мову, до того як він зможе написати код, оскільки простота мови входить в її ключові властивості. Фундаментальні концепції мови Java легко схоплюються та розробники з самого початку вже можуть вести продуктивну роботу. Розробники Java врахували, що багато програмістів вже добре знайомі з мовою C++, саме тому Java, наскільки це можливо, наближена до мови C++.

Автоматична збірка сміття також була додана в Java, в результаті це спрощує процес програмування, оскільки не потрібно слідкувати за очисткою пам'яті, проте дещо ускладнює систему в цілому. Про проблеми, що виникали з управлінням пам'яттю в C та C++, в Java теж не доведеться багато піклуватися.

До переваг мови Java можна віднести:

- сувору статичну типізацію, яка значно скорочує чисельність помилок та поліпшує здійсненність підтримки коду, особливо при застосуванні статичного аналізатора;

- більш полегшена версія C#, адже Java вивчається на основі C++ та досить легко, та окрім того є елементарною, передбачуваною і мінімалістичною, а нові функції представлені досить ґрунтовно;

- мова Java має велику історію, оскільки існує на ринку з 1995 року, тому на сьогодні вона добре вивчена і має чудові пропозиції для сотень рішень та технологій;

- кросплатформеність, адже додатки працюють практично на будь-якій платформі: починаючи від суперкомп'ютерів та до смарт-карт, а з ОС добре виконуються на безкоштовному Linux та FreeBSD;

— значна кількість бібліотек, у Java існує близько 350 гігабайтів бібліотек, що доступні в Maven Repository, до того ж більша частина з них безкоштовна;

— повна зворотна сумісність Java з API та API ранніх версій, жодна інша мова не може похвалитися такою сумісністю, саме тому Java займає міцну позицію та ефективно застосовується в корпоративній сфері для розробки великих проектів.

### 3.1.2 Мова C++

Мова C++ являє собою універсальну мову програмування високого рівня з підтримкою кількох парадигм програмування: об'єктно-орієнтованої, узагальненої та процедурної. C++ була розроблена Б'ярном Страуструпом в 1979 році, коли він працював в AT&T Bell Laboratories, та спочатку отримала назву «Cі з класами», оскільки була розроблена на основі мови C. Свою сучасну назву мова програмування отримала вже аж в 1983 році.

У 90-х роках минулого століття мова C++ стала однією з найбільш уживаних мов програмування загального призначення. Її використовують для написання драйверів, потужних серверних та клієнтських програм, системного програмування, розробки програмного забезпечення, та для розробки розважальних програм, для прикладу, відеоігор. C++ мала суттєвий вплив на інші популярні на сьогоднішній день мови програмування, а саме C# та Java [27].

C++ являє собою мову загального призначення, яка була задумана таким чином, щоб програмісти отримували задоволення від самого процесу програмування. Якщо не зважати на другорядні деталі, C++ містить мову C як підмножину. Мова C була розширена введенням гнучких та ефективних засобів, що призначені для побудови нових типів. Програміст сам структурує свою задачу, означивши нові типи, які точно відповідають поняттям предметної області задачі. Даний метод побудови програми також називають абстракцією даних.

Інформація про нові типи міститься в деяких об'єктах типів, які визначені користувачем. З цими об'єктами можна працювати надійно та просто навіть тоді, коли їх тип не може бути встановлений на стадії трансляції. Програмування з застосуванням таких об'єктів ще називають об'єктно-орієнтованим. При



застосуванні даного методу правильно, програми стають більш короткими та зрозумілими, а супровід їх досить спрощується.

Ключовим поняттям в C++ є клас. Класом називається визначений самим користувачем тип. Класи можуть забезпечити приховування даних, їх ініціалізацію, динамічне задання типів, неявне перетворення користувальницьких типів, контрольоване користувачем управління пам'яттю та засоби для перевантаження операцій.

Крім того, в мові C++ концепції контролю типів та модульної побудови програм реалізовані більш повно. C++ містить удосконалення, які напряду з класами не пов'язані: функції-підстановки, символічні константи, стандартні значення параметрів функцій, операції керування вільною пам'яттю, перевантаження імен функцій та тип посилення. В C++ були збережені всі можливості C для більш ефективної роботи з основними об'єктами, які відображають апаратну "реальність" (розряди, байти, адреси і т.д.), що дозволяє досить ефективно реалізовувати різні користувацькі типи.

І мова, і стандартні бібліотеки C++ були спроектовані з урахуванням переносу. Наявні реалізації мови C++ працюють в більшості систем, які підтримують C. В програмах, написаних на C++ також можна застосовувати бібліотеки C. А більшість службових програм, що розраховані на C, можна використовувати в C++.

При створенні мови C++ одним з важливих критеріїв вибору була простота. При виникненні питання, що саме спростити: посібник з мови та документацію чи транслятор, вибір зробили на користь документації. Величезне значення надавалося також сумісності з мовою C, що не дозволило видалити її синтаксис.

В мові C++ немає типів даних та елементарних операцій високого рівня. Для прикладу, не існує типу матриця з операцією звернення чи типу рядок з операцією конкатенації. У випадку, коли користувачеві знадобляться подібні типи, він зможе визначити їх напряду в самій мові. Тож програмування на C++ по суті зводиться до створення універсальних чи залежних від галузі застосування типів.

Користувацький тип відрізняється від вбудованого типу даних лише способом визначення, однак не способом застосування.

Створення мови C++ відбувалося на основі мови C, тому, за невеликим винятком, мова C була збережена у вигляді підмножини C++. Базова мова C була спроектована так, що існує досить тісний зв'язок між типами, операторами, операціями та об'єктами, з якими напряду працює машина, тобто числами, символами та адресами. За винятком операцій `new`, `delete` та `throw`, а також операції перевірки блоку, виконання операторів та виразів C++ не потребує прихованої динамічної апаратної чи програмної підтримки.

Також в C++ використовується та сама послідовність команд для виконання функцій та повернення з них, що і в мові C. Тому в випадку, коли ці ефективні операції стають надто дорогими, виклик функції можна замінити підстановкою її тіла, при цьому буде збережено зручний функціональний запис без витрат на виклик функції.

Однією з найбільших переваг C++ є той факт, що змінна не може змінювати свій тип даних. Отже, при автодоповненні інтегроване середовище розробки буде пропонувати виключно ті методи, що застосовуються для даного типу даних.

Статична типізація також служить для значного зменшення кількості помилок, адже багато з них зникають на стадії компіляції програми.

Оскільки мова C++ розроблена з метою безпосередньої і всебічної підтримки безлічі стилів програмування (процедурне програмування, об'єктно-орієнтоване програмування, абстракцію даних та узагальнене програмування), розробник зможе спершу освоїти процедурне програмування, а лише потім перейти до вивчення ООП.

Ще однією вагомою перевагою є кросплатформність C++. Існує безліч компіляторів та IDE, які мають версії для Windows, MacOS і Linux (Code Blocks IDE призначена для Windows та Linux, CLion — для Windows, Linux та MacOS). Розробник може створювати додатки для різноманітних платформ, використовуючи «рідну» для системи IDE та компілятор.

### 3.1.3 Вибір мови програмування

До переваг Java можна віднести те, що вона послідовна, добре структурована та підходить для будь-яких платформ. Однак вона не є швидкою, тому що в більшості випадків вона працює на віртуальній машині Java (JVM), що лише інтерпретує код, але не виконує його.

Для подальшої роботи було обрано мову C++. Вона має найбільшу швидкість виконання. Мова C++ має більше можливостей, ніж більшість інших мов програмування.

Розширені функції, для прикладу, успадкування, поліморфізм, бібліотеки шаблонів, макроси та обробка винятків, дозволяють створювати досить структурований та повторно використовуваний код на високому рівні абстракції.

### 3.2 Вибір засобів машинного навчання

На сьогоднішній день машинне навчання набирає все більшу популярність серед розробників різних програмних засобів. Машинне навчання є підмножиною штучного інтелекту. Можна сказати, що все машинне навчання вважається ШІ, проте не увесь штучний інтелект вважається машинним навчанням.

Поняття штучного інтелекту має декілька визначень: 1) це галузь інформатики, яка займається моделюванням інтелектуальної поведінки в комп'ютерах; 2) це можливість машини імітувати розумну людську поведінку; 3) це комп'ютерна система, що здатна виконувати завдання, які зазвичай вимагають роботи людського інтелекту, наприклад візуальне сприйняття, розпізнавання мови, прийняття рішень та переклад з однієї мови на іншу.

Одним з аспектів, що відокремлює машинне навчання від експертних систем чи графіків знань — це його здатність до самомодифікації, коли він піддається впливу досить великої кількості даних.

Також машинне навчання є динамічним та не вимагає прямого втручання людини для внесення змін. Даний факт робить його менш крихким та менш залежним від людей.

Частина «навчання» у терміні машинне навчання означає, що алгоритми МН намагаються оптимізуватись. Тобто вони намагаються звести до мінімуму помилки

чи максимізувати ймовірність істинності їх роботи. Дана частина має три різних імені: функція втрати, функція помилки або ж цільова функція, оскільки даний алгоритм має певну ціль.

Беручи до уваги популярність машинного навчання, на сьогодні існує багато різних бібліотек, що написані на різних мовах програмування та призначені для різноманітних платформ. Кожна із бібліотек має свій перелік реалізованих алгоритмів, певних переваг та недоліків.

### 3.2.1 Бібліотека Armadillo

Armadillo являє собою програмну бібліотеку лінійної алгебри для мови програмування C++. Дана бібліотека спрямована на забезпечення ефективних та впорядкованих базових розрахунків, одночасно маючи досить простий у використанні інтерфейс. Цільовими користувачами можна назвати вчених та інженерів.

Дана бібліотека підтримує цілочисельні, з плаваючою комою (одинарна та подвійна точність) та комплексні числа, а також підмножини тригонометричних та статистичних функцій. Також підтримуються щільні та розріджені матриці.

Різне розкладання матриці забезпечується шляхом опційного впровадження з пакетом лінійної алгебри (LAPACK) та автоматично налаштованими бібліотеками програмного забезпечення лінійної алгебри (атласу), окрім того можуть використовуватися високопродуктивні бібліотеки заміни BLAS/LAPACK, а саме OpenBLAS та Intel MKL.

Бібліотека використовує під час компіляції підхід з відстроченою оцінкою з метою об'єднання декількох операцій в одну та скорочення потреби в тимчасових ресурсах.

У таких випадках порядок операцій оптимізується. Відстрочена оцінка та оптимізація досягаються з використанням метапрограмування шаблонів.

Armadillo пов'язаний з бібліотекою Boost Basic Linear Algebra Subprograms (uBLAS), що також використовує метапрограмування шаблонів. Armadillo, на

відміну від неї, спирається на бібліотеки ATLAS і LAPACK, забезпечуючи таким чином машинно-залежні оптимізації та функції, відсутні в uBLAS.

Дане програмне забезпечення містить відкритий вихідний код та розповсюджується по ліцензії Apache. Це робить його придатним для розробки програмного забезпечення з відкритим вихідним кодом та професійного програмного забезпечення.

### 3.2.2 Бібліотека mlprack

Бібліотека Mlprack являє собою швидку та гнучку бібліотеку машинного навчання, що написана на мові C++ та спрямована на забезпечення швидких, здатних до розширення реалізацій передових алгоритмів машинного навчання. В даній бібліотеці алгоритми представлені у вигляді простих програм командного рядка та класів C++, що в подальшому можуть бути інтегровані в масштабніші рішення машинного навчання.

Mlprack побудована на основі бібліотеки Armadillo. Бібліотека звертає основну увагу на масштабованість, швидкосте та простоту у застосуванні.

Мета даної бібліотеки полягає в тому, щоб машинне навчання стало більш простим для початківців з використанням простого, послідовного API, разом з тим використовуючи можливості мови C++, щоб забезпечити максимальну продуктивність та гнучкість для досвідченіших користувачів.

Бібліотека mlprack містить в собі реалізації різних алгоритмів машинного навчання, для прикладу, метод пошуку найближчих сусідів, тренувальне дерево оцінки щільності, методи лінійної регресії та регресії гребня, алгоритм пошуку k-середніх та інші алгоритми. Даний проект підтримується Технологічним інститутом Джорджії та вкладками з усього світу.

В mlprack є два основні класи нейронної мережі, які призначені для використання як контейнери для шарів нейронної мережі.

Кожен із класів підходить для різних налаштувань. Дана бібліотека поширюється з відкритим вихідним кодом під ліцензією BSD, що дозволяє

використовувати його для розробки програмного забезпечення з відкритим вихідним кодом, а також фірмового програмного забезпечення.

FFN: модель Feed Forward Network містить засоби для закріплення шарів у режимі повної взаємодії з подачею. Даний «стандартний» тип глибокої моделі навчання включає в себе згорткові нейронні мережі (CNN).

RNN: модель повторюваної нейронної мережі, яка забезпечує засоби для перегляду послідовних викликів для переадресації у вигляді різних етапів часу в послідовності. Досить часто застосовується для задач моделювання послідовностей часу, наприклад прогнозування наступного символу в послідовності.

FFN мережі дають сигналам можливість проходити лише в одну сторону. В мережі відсутній зворотній зв'язок, для прикладу, результат будь-якого шару впливає тільки на майбутній шар. Саме тому FFN мережі є прямими та ефективними. Вони досить широко використовуються в розпізнаванні образів та чудово підходять для моделювання взаємозв'язків між набором вхідних даних та однією чи кількома вихідними змінними.

RNN мережі, навпаки, дозволяють проходити сигналам в обох напрямках, вводячи цикли в мережу. Таким чином обчислення, які отримані з попередніх входів, подаються назад в мережу. Це дає рекурсивній мережі певну пам'ять. На сьогодні RNN застосовується для всіх видів послідовних завдань, для прикладу, прогнозування часових рядів, маркування послідовностей та класифікація послідовностей.

### 3.3 Умови проведення експериментального дослідження

Для проведення експериментального дослідження необхідно, щоб всі звуки мали однаковий формат. Було обрано звукові файли формату WAVMONO. WAVE базується на форматі RIFF, поширюючи його на інформацію про такі параметри аудіо, як застосований кодек, частота дискретизації та кількість каналів.

WAV — формат аудіофайла розроблений компаніями Microsoft та IBM.

WAVE файл складається з двох частин: заголовку файлу і області даних. Канонічний формат WAVE файлу починається з RIFF заголовку і двох підсекцій:

"fmt " і "data". Підсекція "fmt " описує параметри даних звукозапису. У підсекції даних "data" міститься розмір даних і фактичні дані звукозапису.

Незважаючи на те, що WAV файл може містити стислі аудіо, найбільш широко його застосовують для зберігання нестислого аудіо в форматі лінійної імпульсно-кової модуляції (LPCM). Стандартним форматом Audio-CD, наприклад, є аудіо в форматі LPCM, з 2 каналами, частотою дискретизації 44 100 Гц і 16 біт на семпл. Оскільки формат LPCM зберігає нестиснуте аудіо, яке абсолютно ідентично оригіналу, це дозволяє використовувати його професійним користувачам і аудіо експертам для отримання максимальної якості звучання.

Частотою дискретизації сигналу було обрано 44 100 Гц. Даної частоти достатньо, щоб адекватно передати форму сигналу, який сегментується. Тривалість одного фрейму складає 20 секунд.

Функція згладжування необхідна для видалення малозначущих за розміром сегментів. Ця функція проходить у дві послідовні фази. Лугове згладжування видаляє сегменти, довжиною менше 20 мс, оскільки вони незначні для мовлення та музики.

Наступний крок полягає в збереженні важливих зон (в плані розміру) музики. Згладжується близько 2000 мс для музики.

### 3.4 Розробка програмного забезпечення

Алгоритм роботи програми по виділенню кепстральних коефіцієнтів, формуванню картотеки та пошуку необхідних аудіо файлів наведено в Додатку Е.

Перевірку відкриття файлів організовано наступним чином:

```
int err = fopen_s(&First, pathToFile.c_str(), "rb");
if (err != 0)
{
    printf("Cannot open input file Sound.wav .\n");
    return 1;
}
```

Далі ми обраховуємо кепстральні коефіцієнти за способом запропонованим у другому розділі. Детальний лістинг коду буде наведено в Додатку Б.

Після цього нам необхідно навчити нашу модель. Для цього ми використаємо `mlpack_gmm_train` з пакету `mlpack`.

Структура моделі GMM буде мати наступний вигляд:

```
struct ModelGMM
{
    int firstFrame = 0;
    bool enoughData = false;
    list<double> listFrameProbab;
    double averageProbability = 0;
    int maxDurationFrame = 0;
    int minDurationFrame = 0;
    double idealprobabilities = 10;
    double edge = 10;
    std::string name;
    std::string locateFile;
    GMM* gmm;
    bool activeNow = false;
    list<double> listGoodFrame;
    bool enoughListGoodFrame = false;
};
```

Вхідні дані для тренування повинні бути задані параметром `--input_file (-i)`, а кількість гаусіан в моделі повинно бути задано параметром `--gaussians (-g)`. Кількість випробувань для запуску задається параметром `--trials (-t)`.

За замовчуванням запускається лише один випробування. Було взято для навчання 6 гаусіан та кількість випробувань 5.

У програмі це буде виглядати наступним чином:

```
string cmd5 = "mlpack_gmm_train --input_file C:\\CPS\\rifle\\rifle_train.txt --gaussians 6 --trials 5 --output_model_file C:\\CPS\\base\\base_rifle.txt";
```



Далі потрібно нам потрібно отримати еталонні значення. Для цього скористаємося `mlpack_gmm_probability` з пакету `mlpack`. Так ми обчислюємо ймовірність того, що дані точки надходять з даного GMM. Задаються параметри тестовий файл `--input_model_file (-m)`, тренувана база `--input_file (-i)`. Вихідні ймовірності можуть бути збережені через вихідний параметр `--output_file (-o)`. У програмі наступний вигляд:

```
string probability_silence = "mlpack_gmm_probability --input_file  
C:\\CPS\\silence\\silence_test.txt --input_model_file C:\\CPS\\base\\base_bird.txt --  
output_file C:\\CPS\\id\\id_silence.txt";
```

Отримані еталонні значення формують картотеку аудіовідбитків музичних композицій у звукових файлах.

### 3.5 Застосування розробленого програмного забезпечення

На сьогоднішній день досить широко використовуються різноманітні програми для розпізнавання та відновлення мовлення. Системи сегментації звукових файлів та автоматичного розпізнавання мовлення використовуються в різноманітних програмах та додатках, наприклад, для пошуку потрібної вам композиції.

Музичні твори є об'єктом авторських прав, що широко застосовуються на телебаченні та радіо. Використання музичних творів нерідко проводиться з порушенням авторських прав, у тому числі в передачах на радіо.

Саме тому з'являється задача відстежування трансляції музики на радіостанціях з метою захисту авторських прав. В Додатку Ж показано загальну схему системи, що відслідковує процес відтворення фонограм в ефірі.

Відповідно до наведеної в Додатку Ж схеми спочатку необхідно виконати розмічення записаного аудіоархіву на окремі треки та компактно представити кожен, тобто необхідно провести аналіз аудіо в записаному архіві і автоматично встановити межі музичних сегментів.

Суть задачі моніторингу трансляцій на радіостанціях (телебаченні) для відслідковування порушень авторських прав полягає в тому, що необхідно визначити скільки разів прозвучав той чи інший музичний твір.

Після виділення окремих музичних сегментів з суцільного аудіопотоку можна виконувати пошук відповідного треку в колекції аудіо.

Якщо в архіві містяться однакові треки, то, оскільки кластеризація виконується для кожного треку  $\tilde{Y}_q$  в загальній базі будуть міститися однакові центроїди. Відповідно при пошуку в числі найближчих сусідів будуть ідентичні результати для всіх однакових треків. При цьому кількість найближчих сусідів  $C = \{\tilde{Y}_1, \tilde{Y}_2, \dots, \tilde{Y}_k\}$ ,  $|C|=k$ , для яких фіксуються результати пошуку  $Distance_q$ , необхідно збільшити.

Далі ці результати (відстані) підсумовуються для кожного треку  $\tilde{Y}_q$ . Рішення приймаються за мінімальними сумами. Це дозволяє ідентифікувати кілька збігів з власним шаблоном.

Оцінка автоматичної класифікація проводилась шляхом порівняння з результатами ручної класифікації.

Затримка вимірюється між мітками, встановленими вручну та автоматично.

Точність класифікації вимірюється за формулою:

$$A = \frac{l_t - l_i - l_d}{l_t},$$

де  $l_t$  — довжина тестового аудіофайлу;

$l_i$  — довжина вставки;

Тож для музики відсоток затримки, що має значення менше 20 кодових сегментів, складає 91%. Вставки без музики відповідають сегментам, що містять дуже тиху музику. Точність класифікації музики складає 84%.

## 4 РОЗРАХУНОК ЕКОНОМІЧНОЇ ДОЦІЛЬНОСТІ

Проведення науково-дослідної роботи будь-якого типу завжди вимагає певних витрат. Витрати на виробництво та реалізацію продукту увесь час повинні зменшуватись, оскільки у цьому й полягає прогрес будь-якого виробництва. Якщо витрати не зменшуються, то ніяка науково-технічна розробка не буде реалізована на практиці, оскільки така розробка не буде більш ефективна, аніж існуючі на ринку аналоги.

Метою економічної частини магістерської кваліфікаційної роботи є довести економічну доцільність та ефективність впровадження розробки «Програмний засіб формування аудіовідбитків музичних композицій у звукових файлах».

На основі економічних розрахунків, можна довести економічну доцільність та ефективність впровадження результатів, тобто здійснити комерціалізацію наукових розробок. Саме цим завданням присвячений даний розділ магістерської кваліфікаційної роботи і він передбачає виконання таких етапів робіт (рис. 4.1):

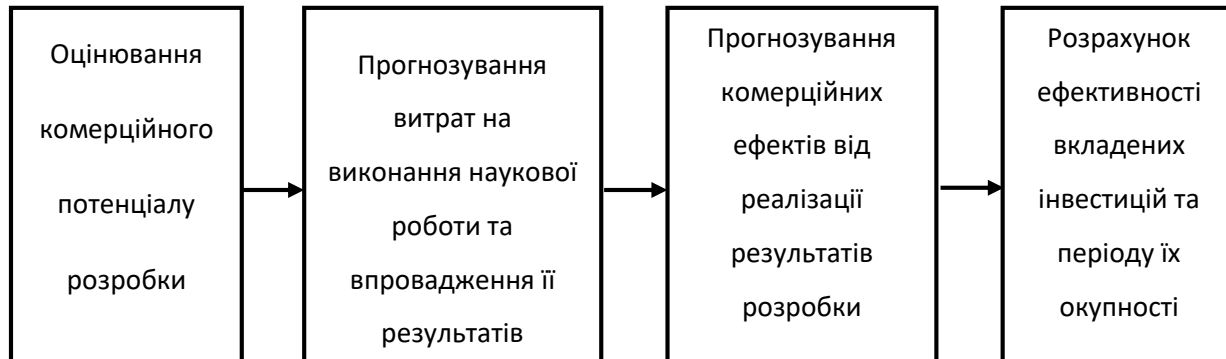


Рисунок 4.1 — Складові економічної частини магістерської роботи

На такі складові буде поділено економічну частину даної магістерської роботи. Усі подальші економічні розрахунки, будуть висвітлені у згаданих підрозділах економічної частини.

### 4.1 Технологічний аудит розробки

В даному підрозділі буде проведено оцінювання комерційного потенціалу розробки, створеної в результаті науково-технічного дослідження. За результатами

оцінювання робиться висновок щодо напрямів організації в майбутньому та її впровадження з врахуванням встановленого рейтингу.

Оцінювання комерційного потенціалу розробки будемо здійснювати за дванадцятьма критеріями, наведеними у таблиці 4.1.

Таблиця 4.1 — Оцінювання комерційного потенціалу розробки

Критерії оцінювання та бали (за 5-бальною шкалою)					
Критерій	0	1	2	3	4
Технічна здійсненність концепції:					
1	Достовірність концепції не підтверджена	Концепція підтверджена експертними висновками	Концепція підтверджена розрахунками	Концепція перевірена на практиці	Перевірено роботоздатність продукту в реальних умовах
Ринкові переваги (недоліки):					
2	Багато аналогів на малому ринку	Мало аналогів на малому ринку	Багато аналогів на великому ринку	Один аналог на великому ринку	Продукт не має аналогів на великому ринку
3	Ціна продукту значно вища за ціни аналогів	Ціна продукту дещо вища за ціни аналогів	Ціна продукту приблизно дорівнює цінам аналогів	Ціна продукту дещо нижче за ціни аналогів	Ціна продукту значно нижче за ціни аналогів
4	Технічні та споживчі властивості продукту значно гірші, ніж в аналогів	Технічні та споживчі властивості продукту трохи гірші, ніж в аналогів	Технічні та споживчі властивості продукту на рівні аналогів	Технічні та споживчі властивості продукту трохи кращі, ніж в аналогів	Технічні та споживчі властивості продукту значно кращі, ніж в аналогів
5	Експлуатаційні витрати значно вищі, ніж в аналогів	Експлуатаційні витрати дещо вищі, ніж в аналогів	Експлуатаційні витрати на рівні експлуатаційних витрат аналогів	Експлуатаційні витрати трохи нижчі, ніж в аналогів	Експлуатаційні витрати значно нижчі, ніж в аналогів
Ринкові перспективи					
6	Ринок малий і не має позитивної динаміки	Ринок малий, але має позитивну динаміку	Середній ринок з позитивною динамікою	Великий стабільний ринок	Великий ринок з позитивною динамікою

## Продовження таблиці 4.1

Практична здійсненність					
7	Активна конкуренція великих компаній на ринку	Активна конкуренція	Помірна конкуренція	Незначна конкуренція	Конкурентів немає
8	Відсутні фахівці як з технічної, так і з комерційної реалізації ідеї	Необхідно наймати фахівців або витратити значні кошти та час на навчання наявних фахівців	Необхідне незначне навчання фахівців та збільшення їх штату	Необхідне незначне навчання фахівців	Є фахівці з питань як з технічної, так і з комерційної реалізації ідеї
9	Потрібні значні фінансові ресурси, які відсутні.	Потрібні незначні фінансові ресурси. Джерела фінансування відсутні	Потрібні значні фінансові ресурси. Джерела фінансування є	Потрібні незначні фінансові ресурси. Джерела фінансування є	Не потребує додаткового фінансування
10	Необхідна розробка нових матеріалів	Потрібні матеріали, що використовуються у військово-промисловому комплексі	Потрібні дорогі матеріали	Потрібні досяжні та дешеві матеріали	Всі матеріали для реалізації ідеї відомі та давно використовуються у виробництві
11	Термін реалізації ідеї більший за 10 років	Термін реалізації ідеї більший за 5 років. Термін окупності інвестицій більше 10-ти років	Термін реалізації ідеї від 3-х до 5-ти років. Термін окупності інвестицій більше 5-ти років	Термін реалізації ідеї менше 3-х років. Термін окупності інвестицій від 3-х до 5-ти років	Термін реалізації ідеї менше 3-х років. Термін окупності інвестицій менше 3-х років
12	Необхідно регламентні документи та велика кількість дозвільних документів на виробництво продукту	Необхідно отримання великої кількості дозвільних документів на виробництво та реалізацію продукту	Процедура отримання дозвільних документів для виробництва та реалізації продукту вимагає незначних коштів та часу	Необхідно тільки повідомлення відповідним органам про виробництво та реалізацію продукту	Відсутні будь-які регламентні обмеження на виробництво та реалізацію продукту

На основі складеної таблиці ряд незалежних експертів, у нашому випадку керівник магістерської роботи та викладачі випускової кафедри поставили різні бали за кожен із критеріїв оцінювання. Результати цього оцінювання комерційного потенціалу занесено до таблиці 4.2. За даними таблиці 4.2, а також згідно із рекомендаціями, що наведені в таблиці 4.3, можна зробити висновок, щодо рівня комерційного потенціалу розробки.

Таблиця 4.2 — Результати оцінювання комерційного потенціалу розробки

Критерії	Прізвище, ініціали, посада експерта		
	1 Очкуров М. А., к.т.н., доц. кафедри ОТ	2 Ткаченко М., к.т.н., доц. кафедри ОТ	3 Семеренко В. П., к.т.н., доц. кафедри ОТ
	Бали, виставлені експертами:		
1	3	3	3
2	4	2	3
3	4	3	4
4	3	2	4
5	3	4	3
6	3	4	3
7	4	3	2
8	4	4	3
9	3	2	3
10	3	3	3
11	4	4	3
12	3	3	4
Сума балів	СБ <sub>1</sub> = 41	СБ <sub>1</sub> = 37	СБ <sub>1</sub> = 38
Середньо-арифметична сума балів $\overline{СБ}$	$\overline{СБ} = \frac{\sum_1^3 СБ_i}{3} = \frac{41 + 37 + 38}{3} = \frac{116}{3} = 38.7$		

Таблиця 4.3 — Рівні комерційного потенціалу розробки

Середньоарифметична сума балів $\overline{СБ}$ , розрахована на основі висновків експертів	Рівень комерційного потенціалу розробки
0 - 10	Низький
11 - 20	Нижче середнього
21 - 30	Середній
31 - 40	Вище середнього
41 - 48	Високий

Взявши до уваги, середньоарифметичну суму балів,  $\overline{СБ} = 38,7$ , що були виставлені експертами, можна стверджувати, що рівень комерційного потенціалу даної розробки — вище середнього.

Для порівняння властивостей було взято Shazam. Дана програма має більш широкий спектр використання. В свою чергу нова розробка є вузько спеціалізованою, через що більш ефективною. Аналог розповсюджується по платній підписці у той час, як плата за користування програмою, яка є продуктом розробки, є одноразовою.

Таблиця 4.4 — Порівняння характеристик розробки із аналогом

Показники	Розробка	Аналог
Функціонал	8	7
Швидкодія	8	9
Надійність	7	9
Метод розповсюдження	8	6
Інтерфейс, простота використання	9	8

Просування продукту здійснюватиметься шляхом його рекламування у соціальних мережах різного напрямку. За допомогою аналітики даних сервісів рекламу можна буде спрямувати на цільову аудиторію — музикантів та композиторів. Також продукт може застосовуватись у радіомовленні та телебаченні для контролю і фіксації несанкціонованого використання захищених авторським правом музичних творів.

Новизною розробки є використання принципіально нового методу формування ознак об'єктів, що дає змогу більш точно та швидко формувати та обробляти аудіо відбитки. Дана характеристика позитивно впливатиме на конкурентоспроможність продукту.

Виходячи з результатів даного порівняння можна зробити висновок, що нова розробка є конкурентоспроможною, оскільки по деяким параметрам переважає один з найкращих аналогів на ринку.

#### 4.2 Прогнозування витрат на виконання та впровадження

У магістерській кваліфікаційній роботі розглядається програмне забезпечення для формування аудіовідбитків музичних композицій у звукових

файлах, тому значна частина витрат — це витрати на розробку, а не на виробництво і відтворення. Звідси, й певна специфіка розрахунків. Основна заробітна плата розробників, які працюють над проектом визначається за формулою:

$$Z_o = \frac{M}{T_p} \cdot t \text{ [грн]},$$

де  $M$  — місячний посадовий оклад розробника;

$T_p$  — число робочих днів в місяці ( $T_p = 22$  дні);

$t$  — число днів роботи розробника.

Над створенням розробки працювали керівник проекту та інженер-програміст, отже, виконаємо для них всі необхідні розрахунки:

$$Z_o = \frac{12\,000}{22} \cdot 7 = 3\,818,20 \text{ (грн)}.$$

$$Z_o = \frac{10\,000}{22} \cdot 55 = 25\,000 \text{ (грн)}.$$

Додаткова заробітна плата ( $Z_d$ ) всіх розробників та робітників, які брали участь у виконанні даного етапу роботи, розраховується як (10...12)% від суми основної заробітної плати всіх розробників та робітників, тобто виразом:

$$Z_d = (10 \dots 12\%) \cdot Z_o \text{ [грн]},$$

де  $Z_o$  — основана заробітна плата.

$$Z_d = \frac{10 \cdot 28\,818,20}{100} = 2\,881,82 \text{ (грн)}.$$

Нарахування на заробітну плату ( $N_zp$ ) розробників та робітників, які брали участь у виконанні даного етапу роботи, розраховуються за формулою:

$$N_zp = 22\% \cdot (Z_o + Z_d) \text{ [грн]},$$
$$N_zp = \frac{22 \cdot (28\,818,20 + 2\,881,82)}{100} = 6\,974 \text{ (грн)}.$$

Амортизація обладнання, комп'ютерів та приміщень ( $A$ ), які використовувались під час виконання даного етапу роботи.



Дані відрахування розраховують по кожному виду обладнання, приміщенням тощо.

У спрощеному вигляді амортизаційні відрахування (А) в цілому бути розраховані за формулою:

$$A = \frac{Ц \cdot На}{100} \cdot \frac{T}{12} [\text{грн}],$$

де Ц — загальна балансова вартість всього обладнання, комп'ютерів, приміщень тощо, що використовувались для виконання даного етапу роботи, рахується в грн;

На — річна норма амортизаційних відрахувань, для нашого випадку можна прийняти, що На = (10...25)%;

T — термін, використання обладнання, приміщень тощо, місяці.

Всі розрахунки зводимо до таблиці 4.5.

Таблиця 4.5 — Заробітна плата

Найменування посади	Місячний посадовий оклад, грн.	Оплата за робочий день, грн.	Число днів роботи	Витрати на заробітну плату, грн.
1 Керівник	12 000	545,45	7	3 818,20
2 Старший інженер-програміст	10 000	454,55	55	25 000
Всього				$\sum Z_0 = 28\,818,20$

У магістерській кваліфікаційній роботі розробляється програмний продукт, який для споживача буде поширюватися через інтернет, а саме через веб-сторінку.

При розробці сторінки до послуг виробничого характеру сторонніх підприємств можна віднести надавання послуги «Хостинг», а також направлення обраного доменного імені на сервери хосту, тому на протязі всього існування проекту за ці послуги, щорічно потрібно сплачувати абонентську плату.

Таблиця 4.6 — Амортизація обладнання та приміщень

Найменування обладнання, приміщень	Балансова вартість, грн.	Норма амортизації, %	Термін використання, міс.	Величина амортизаційних відрахувань, грн.
ЕОМ	10 000	20%	3	250
Приміщення	120 000	15%	3	450
Всього				700

Таблиця 4.7 — Послуги, що використовуються при виготовленні програми

Найменування комплектуючих (робіт, послуг)	Кількість, шт.	Ціна за одиницю, грн.	Сума, грн.
1. Послуга «Хостинг», шт.	1	300	300
2. Послуга «Доменне ім'я», шт.	1	150	150
Всього			450 грн.

Витрати на силову електроенергію  $V_e$ , якщо ця стаття має суттєве значення для виконання даного етапу роботи, розраховуються за формулою:

$$V_e = V \cdot P \cdot \Phi \cdot K_{\Pi} [\text{грн}],$$

де  $V$  — вартість 1 кВт електроенергії, грн.;

$P$  — установлена потужність обладнання, кВт/год;

$\Phi$  — фактична кількість годин роботи обладнання, яке задіяне на виготовлення одного виробу, годин;

$K_{\Pi}$  — коефіцієнт використання потужності,  $K_{\Pi} \leq 1$ .

$$V_e = 3,35 \cdot 0,09 \cdot 230 \cdot 0,7 = 48,5 \text{ (грн).}$$

Інші витрати охоплюють: загально виробничі витрати (витрати управління організацією, ремонт та експлуатація основних засобів, витрати на опалення, освітлення тощо), адміністративні витрати (проведення зборів, оплата юридичних та аудиторських послуг, тощо), витрати на збут (витрати на рекламу,

перепідготовка кадрів) на інші операційні витрати (штрафи, пені, матеріальні допомоги, втрати від знецінення запасів тощо).

Інші витрати  $I_B$  можна прийняти як  $(100...300)\%$  від суми основної заробітної плати розробників та робітників, що виконували дану роботу, тобто:

$$I_B = (1..3) \cdot (Z_o + Z_p)$$

Отже, розрахуємо інші витрати:

$$I_B = 1 \cdot (28\,818,20 + 2\,881,82) = 31\,700 \text{ грн.}$$

Сума всіх попередніх статей витрат дає витрати на виконання даної частини (розділу, етапу) роботи —  $B$ .

$$B = 28\,818,20 + 2\,881,82 + 6\,974 + 700 + 450 + 113 + 31\,700 = 71\,637 \text{ (грн).}$$

Прогнозування загальних витрат  $ZB$  на виконання та впровадження результатів виконаної наукової роботи здійснюється за формулою:

$$ZB = \frac{B_{\text{заг}}}{\beta} [\text{грн}],$$

де  $\beta$  — коефіцієнт, який характеризує етап (стадію) виконання даної роботи.

Оскільки, розробка знаходиться на стадії впровадження, то  $\beta \approx 0,9$ ;

$$ZB = \frac{71\,637}{0,9} = 79\,596,7 \text{ (грн).}$$

$B_{\text{заг}}$  — загальна вартість всієї наукової роботи. У даному випадку  $B_{\text{заг}} = B$ .

Отже, розрахований кошторис витрат на розробку програмного забезпечення для формування ознак об'єктів у цифровому зображенні складає 79 596,7 грн.

#### 4.3 Прогнозування комерційних ефектів від реалізації розробки

У даному підрозділі здійснено прогнозування, яку вигоду можна отримати у майбутньому від впровадження результатів даної наукової роботи.

Передбачається, що виконання наукової роботи та впровадження результатів по розробці програмного забезпечення для формування аудіовідбитків музичних композицій у звукових файлах займе 1 рік.

Основні позитивні результати від впровадження розробки очікуються протягом 3 років після її впровадження. Саме зростання чистого прибутку за формулою (4.1) забезпечить підприємству (організації) надходження додаткових коштів, які дозволять покращити фінансові результати діяльності.

$$\Delta\Pi_i = \sum_1^n (\Delta\Pi_{\text{я}} \cdot N + \Pi_{\text{я}}\Delta N)_i, [\text{грн}], \quad (4.1)$$

де  $\Delta\Pi_{\text{я}}$  — покращення основного якісного показника від впровадження результатів розробки у даному році;

$N$  — основний кількісний показник, який визначає діяльність підприємства у даному році до впровадження результатів наукової розробки;

$\Delta N$  — покращення основного кількісного показника діяльності підприємства від впровадження результатів розробки;

$\Pi_{\text{я}}$  — основний якісний показник, який визначає діяльність підприємства у даному році після впровадження результатів наукової розробки;

$n$  — кількість років, протягом яких очікується отримання позитивних результатів від впровадження розробки.

В результаті впровадження результатів наукової розробки покращується якість програмного продукту, що дозволяє підвищити ціну його реалізації на першого року — на 1000 шт., протягом другого року — ще на 700 шт., протягом третього року — ще на 450 шт.

Орієнтовно: реалізація продукції до впровадження результатів наукової розробки складала 1 шт., а прибуток, що його отримувало підприємство на одиницю продукції до впровадження результатів наукової розробки — 250 грн.

Спрогнозуємо збільшення чистого прибутку підприємства від впровадження результатів наукової розробки у кожному році відносно базового.

Збільшення чистого прибутку підприємства  $\Delta\Pi_i$  протягом першого року складе:

$$\Delta\Pi_1 = 250 \cdot 1 + (250 + 150) \cdot 1000 = 400\,250(\text{грн}).$$

Збільшення чистого прибутку підприємства  $\Delta\Pi_i$  протягом другого року (відносно базового року, тобто року до впровадження результатів наукової розробки) складе:

$$\Delta\Pi_2 = 250 \cdot 1 + (250 + 150) \cdot 700 = 280\,250(\text{грн}).$$

Збільшення чистого прибутку підприємства  $\Delta\Pi_i$  протягом третього року складе:

$$\Delta\Pi_3 = 250 \cdot 1 + (250 + 150) \cdot 450 = 180\,250(\text{грн}).$$

#### 4.4 Розрахунок ефективності вкладених інвестицій та періоду окупності

Розрахований комерційний ефект можливого впровадження розробки однак не є гарантією, що розробка буде впроваджена. Якщо збільшення прогнозованого прибутку від впровадження результатів наукової розробки є вигідним для підприємства, то це ще не означає, що вона зацікавить інвесторів.

Основні показники, що визначають рентабельність фінансування розробки певним інвестором, є відносна і абсолютна ефективність вкладених інвестицій та термін їх окупності. Розрахунок ефективності вкладених інвестицій передбачає проведення наступного типу робіт.

Спочатку розраховуємо теперішню вартість інвестицій  $PV$ , що вкладаються в наукову розробку. Такою вартістю, можна вважати прогнозовану величину загальних витрат  $ZB$  на виконання та впровадження результатів НДДКР, розраховану нами раніше, тобто будемо вважати, що  $ZB = PV = 79\,596,7$ .

Розрахуємо очікуване збільшення прибутку  $\Delta\Pi_i$ , що його отримає підприємство (організація) від впровадження результатів наукової розробки, для кожного із років, починаючи з першого року впровадження.

Для спрощення подальших розрахунків побудуємо вісь часу, на яку нанесемо всі платежі (інвестиції та прибутки), що мають місце під час виконання науково-дослідної роботи та впровадження її результатів.

Платежі показуються у ті терміни, коли вони здійснюються. Рисунок, що характеризує рух платежів (інвестицій та додаткових прибутків) буде мати вигляд, наведений на рис. 4.2.



Рисунок 4.2 — Вісь часу з фіксацією платежів, що мають місце під час розробки та впровадження результатів НДДКР

Розрахуємо абсолютну ефективність вкладених інвестицій  $E_{абс}$ .

$$E_{абс} = (ПП - PV),$$

де ПП — приведена вартість всіх чистих прибутків, що їх отримає підприємство (організація) від реалізації результатів наукової розробки, грн;

PV — теперішня вартість інвестицій  $PV = 3B = 79\,596,7$ (грн).

У свою чергу, приведена вартість всіх чистих прибутків ПП розраховується за формулою:

$$ПП = \sum_1^T \frac{\Delta\Pi_i}{(1 + \tau)^t}, [\text{грн}],$$

де  $\Delta\Pi_i$  — збільшення чистого прибутку у кожному із років, протягом яких виявляються результати виконаної та впровадженої НДДКР, грн;

T — період часу, протягом якого виявляються результати впровадженої НДДКР, роки;

$\tau$  — ставка дисконтування, за яку можна взяти щорічний прогнозований рівень інфляції в країні; для України цей показник знаходиться на рівні 0,1;

$t$  — період часу (в роках) від моменту отримання чистого прибутку до точки «0».

Отримаємо:

$$\begin{aligned} \text{ПП} &= \frac{400\,250}{(1+0,1)^1} + \frac{280\,250}{(1+0,1)^2} + \frac{180\,250}{(1+0,1)^3} = 363\,863,64 + 231\,611,57 + 135\,424,5 \\ &= 730\,899,7 \end{aligned}$$

Тоді  $E_{\text{абс}} = (730\,899,7 - 79\,596,7) = 651\,303$ (грн).

Оскільки  $E_{\text{абс}} > 0$ , то результат від проведення наукових досліджень та їх впровадження може принести прибуток.

Розраховують відносну (щорічну) ефективність вкладених в наукову розробку інвестицій  $E_{\text{в}}$ . Для цього користуються формулою:

$$E_{\text{в}} = \sqrt[T_{\text{ж}}]{1 + \frac{E_{\text{абс}}}{\text{PV}}} - 1$$

де  $E_{\text{абс}}$  — абсолютна ефективність вкладених інвестицій, грн;

$\text{PV}$  — теперішня вартість інвестицій  $\text{PV} = \text{ЗВ}$ , грн;

$T_{\text{ж}}$  — життєвий цикл наукової розробки, роки.

$$E_{\text{в}} = \sqrt[3]{1 + \frac{651\,303}{79\,596,7}} - 1 = 2,09 - 1 = 109\%$$

Далі, розрахована величина  $E_{\text{в}}$  порівнюється з мінімальною (бар'єрною) ставкою дисконтування  $\tau_{\text{мін}}$ , яка визначає ту мінімальну дохідність, нижче за яку інвестиції вкладатись не будуть. У загальному вигляді мінімальна (бар'єрна) ставка дисконтування  $\tau_{\text{мін}}$  визначається за формулою:

$$t = d + f [\%],$$

де  $d$  — середньозважена ставка за депозитними операціями в комерційних банках;

$f$  — показник, що характеризує ризикованість вкладень; зазвичай, величина  $f = (0,1)$ , але може бути і значно більше.

$$t = 0,20 + 0,1 = 0,30 = 30\%$$

Величина  $E_B > \tau_{\text{мін}}$ , інвестор може бути зацікавлений у фінансуванні даної наукової розробки.

Розрахуємо термін окупності вкладених коштів у реалізацію наукового проекту за формулою:

$$T_{\text{ок}} = \frac{1}{E_B} [\text{років}],$$
$$T_{\text{ок}} = \frac{1}{1,09} = 0,9 \text{ роки.}$$

Оскільки  $T_{\text{ок}} = 0,9$  роки, то розробка «Програмного засобу формування аудіовідбитків музичних композицій у звукових файлах» являється доцільною.

#### 4.5 Результати економічного ґрунтування

У даному розділі магістерської кваліфікаційної роботи здійснено розрахунки, які доводять прибутковість та ефективність впровадження нового продукту.

Проведено оцінювання комерційного потенціалу розробки. На основі компетентної думки експертів було сформовано систему критеріїв та за 5-ти бальною шкалою, виставлено бали по кожному з них. Виставлені бали, показують, що рівень комерційного потенціалу є вище середнього.

Розраховано витрати на розробку. Розрахований кошторис витрат на розробку склав 79 596,7 грн.

Були спрогнозовані комерційні ефекти від реалізації розробки, тобто який дохід, можна отримати у майбутньому від впровадження виконаної наукової роботи. Доведено, що розробка отримує вигоду від впровадження.



Розраховано основні показники, які визначають доцільність фінансування наукової розробки інвестором. Такими показниками є абсолютна та відносна ефективність вкладених інвестицій, а також термін їх окупності.

Обрахована абсолютна ефективність становить 651 303 грн, що свідчить про те, що інвестор буде зацікавлений у фінансуванні даної розробки. Відносна (щорічна) ефективність становить 109%, що більше мінімальної ставки дисконтування, що ще раз підтверджує зацікавленість інвестора.

Термін окупності вкладених коштів у реалізацію наукового проекту становить 0,9 роки, що означає, що вкладені кошти повернуться, приблизно, через 11 місяців.

Отже, можна стверджувати, що фінансування даної розробки є доцільним.

## ВИСНОВКИ

Існуючі системи для сегментації звукових файлів використовуються в різноманітних програмах та додатках для пошуку необхідної композиції чи автоматичного набору голосових команд. До одних з ефективних способів ідентифікації мультимедійних матеріалів можна віднести технологію, що широко відома як «метод цифрових відбитків», названий аналогічно до методу ідентифікації людини по його відбитках пальців.

У першому розділі магістерської роботи було розглянуто музичні сигнали, способів їх утворення та представлення. На основі проведеного огляду різних існуючих методів класифікації для аналізу музичних сигналів було обрано метод класифікації на основі змішаних Гаусівських моделей з використанням Мел-частотних кепстральних коефіцієнтів (MFCC).

В наступному розділі було розглянуто розробку математичної моделі опису аудіосигналів та формування словника за допомогою EM-алгоритму на основі змішаних гаусівських сумішей.

Для розробки математичної моделі опису аудіосигналів застосовуються MFCC параметри. Було розглянуто переваги MFCC параметрів та обґрунтовано їх вибір як параметрів опису аудіосигналу. Також наведено загальну схему обчислення MFCC. Також в розділі було розглянуто формування словника за допомогою EM-алгоритму та обґрунтовано застосування гаусівських змішаних сумішей.

У третьому розділі для розробки програмного забезпечення було обрано бібліотеку машинного навчання mlpack, що базується на бібліотеці лінійної алгебри Armadillo та реалізована мовою програмування C++. Для створення програми використовувалась мова програмування C++. Також було розроблено структуру програмного продукту, розроблена програмне забезпечення та перевірена якість роботи програми.

У четвертому розділі магістерської кваліфікаційної роботи було проведено оцінювання комерційного потенціалу розробки, виконано розрахунок необхідних

економічних витрат для реалізації запропонованої технології формування аудіовідбитків музичних композицій у звукових файлах та визначено комерційні переваги від впровадження запропонованого програмного продукту в радіомовленні та телебаченні для контролю і фіксації несанкціонованого використання захищених авторським правом музичних творів.

Також було запропоновано способи застосування даного програмного забезпечення та наведено загальну схему системи, що відслідковує процес відтворення фонограм в ефірі з використанням програми для формування аудіовідбитків.

## ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Вахромеев В. Элементарная теория музыки / В. Вахромеев. — М.: Музыка, 2012 — 429 с.
2. Москаленко В. Лекции по музыкальной интерпретации /В. Москаленко // Учебное пособие. — Киев, 2013 — 271 с.
3. Batlle E. Automatic song identification in noisy broadcast audio / Batlle E., Masip J., Guaus E. // Proc. of the SIP — 2002.
4. Zwicker E. Psychoacoustics: Facts and Models /E. Zwicker, H. Fastl // Springer Verlag, 2nd ed. — 1999.
5. Щуровська А. М. Формування ознак музичних композицій із використанням методу цифрових відбитків// Електронна наукова конференція ВНТУ «Молодь в науці: дослідження, проблеми, перспективи (МН-2021)». Режим доступу: <https://conferences.vntu.edu.ua/index.php/mn/mn2021>
6. Михайловська Р. С. Послідовності інтервалів та акордів / Р. С. Михайловська — К.: Мелосвіт, 2013 — 341 с.
7. Афоніна О. С. Сольфеджіо на основі сучасної, джазової та естрадної музики / О. С. Афоніна, В. О. Баранова — Вінниця: Нова Книга, 2013 — 285 с.
8. Яцковський В. С. Алгоритм оцінювання темпу музикальних сигналів / В. С. Яцковський, Д. М. Бруслик // Електроніка та системи управління, 2012., №1
9. Haitsma J. A Highly Robust Audio Fingerprinting System /J. Haitsma, A. Kalker // International Symposium on Music Information Retrieval (ISMIR), 2002, pp. 107-115.
10. Cano P. Audio Fingerprinting: Concepts and Applications / Cano P., Gómez E., Batlle E., Gomes L., Bonnet M. // Proceedings of 2002 International Conference on Fuzzy Systems Knowledge Discovery — Singapore — 2002.
11. Ткаченко О. М. Ідентифікація музичного твору за його фрагментом / О. М. Ткаченко, О.Ф. Грійо Тукало, Ю. Л. Далекий//Міжнародна науково-практична конференція «Інформаційні технології та комп'ютерна інженерія». Вінниця. 2014. — с. 256-257.

12. Ткаченко А. Н. Метод быстрого поиска фрагмента аудиозаписи с усовершенствованной оценкой меры близости / А. Н. Ткаченко, О. Ф. Грийо Тукало, Ю. Л. Далекий // Международный научный журнал «Управляющие системы и машины». Киев, 2015. - №6. - с.51-58.
13. Philips.Audio Fingerprinting technology / [Электронный ресурс] — Режим доступа: <http://www.research.philips.com/initiatives/contentid/audiofp.html>.
14. Lawrence R. Rabiner Introduction to digital speech processing / Lawrence R. Rabiner, Ronald W. Schafer // Now Publishers Inc, 2007 — 200 с.
15. Cano P. A review of algorithms for audio fingerprinting. / Cano P., Batlle E., T. Kalker, Haitsma J. // Proc. IEEE Workshop on Multimedia Signal Processing. — 2002. — pp. 169-173.
16. Batlle E. Feature Decorrelation Methods in Speech Recognition. A Comparative Study / Batlle E., Nadeu C., Fonollosa J. // Proc. of International Conference on Speech and Language Processing. — 1998. — Sydney (Australia).— pp. 951-954.
17. Баклан І.В. Про деякі нові особливості використання прихованих марковських моделей для аналізу та прогнозування часових рядів / І.В. Баклан, Г.А. Степанкова // Штучний інтелект. 2010. № 4
18. Круг П.Г. Нейронные сети и нейрокомпьютеры: Учебное пособие по курсу «Микропроцессоры» / П.Г. Круг — М.: Издательство МЭИ, 2002. — 176 с.
19. Logan B. Mel Frequency Cepstral Coefficients for Music Modelling / B. Logan // Proc. International Symposium on Music Information Retrieval. — Plymouth (USA). — 2000.
20. Хайкин С. Б. Нейронные сети: полный курс /С. Б. Хайкин — 2-е изд. Пер. с англ. — М.: Издательский дом «Вильямс», 2006. — 1104 с.
21. Жирков А.О. Графический метод представления и нейросетевое распознавание частотно-временных векторов речевой информации / Жирков А.О., Корчагин Д.Н., Лукин А.С., Крылов А.С., Баяковский Ю.М. // Программирование — 2003.— №4.— с.41-52.
22. Carvajal J. Comparative evaluation of action recognition methods via Riemannian manifolds, Fisher vectors and GMMs: Ideal and challenging conditions / J.

Carvajal, A. Wiliem, C. McCool, B. Lovell, C. Sanderson // In Lecture Notes in Computer Science (LNCS), Vol. 9794, 2016. — pages 88-100

23. Seck M. Experiments on speech tracking in audio documents using Gaussian Mixture Modeling / M. Seck, I. Magrin-Chagnolleau, F. Bimbot // ICASSP'2001, Salt Lake City. may 2001. SPEECHP5.5

24. D. Reynolds Speaker verification using adapted Gaussian mixture models / D. Reynolds, T. Quatieri, and R. Dunn. // Digital Signal Processing. 2000. №10

25. Morishima M. Phonetically adaptive cepstrum mean normalization for acoustic mismatch compensation / Morishima M., Isobe T., Takahashi J. // Proc. Automatic Speech Recognition and Understanding. — Santa Barbara (USA). — 1997.— pp. 436-441.

26. LeCun Y. Deep learning. Nature / Y. LeCun, Y. Bengio, and G. Hinton // pp. 521:436—444, 2015.

27. Plomp, R. Tonal consonance and critical bandwidth / Plomp, R., & Levelt, W. J. M. // Journal of the Acoustical Society of America. — 1965. — Vol. 37, pp.548-560

28. G. McLachlan The EM Algorithm and Extensions/ G. McLachlan and T. Krishnan // John Wiley & Sons, 2nd edition, 2008.

29. Haitsma J. Speed-change resistant audio fingerprinting using autocorrelation // Conf. on Acoustics, Speech and Signal Processing. — 2003. — Vol.4, pp. 728-731.

30. Rhebergen, K. S. A Speech Intelligibility Index-based approach to predict the speech reception threshold for sentences in fluctuating noise for normal-hearing listeners / Rhebergen, K. S. // J. Acoust. Soc. Am. 117, — 2005, pp. 2181-2192

31. Hörnel D. Comprative Style Analysis with Neural Networks / Hörnel D., Frank Olbrich // ICMC Proceedings – 1999, pp. 433-436.

32. Білобров С. В. Метод ідентифікації аудіоматеріалів на основі аналізу спектра модуляції сигналу / С. В. Білобров // Вісник Донецького університету, Сер.А: Природничі науки. — 2005. — Вип.2.— с.387-391.

33. Julien Piquier Speech and music classification in audio documents / Julien Piquier, Christine Senac // Acoustics, Speech, and Signal Processing: International Conference, 2002
34. Abdullah I. Al-Shoshan Speech and Music Classification and Separation: A Review/ Abdullah I. Al-Shoshan// Engineering and science — 2006
35. Страуструп Б. Мова програмування C++ / Б. Страуструп — К.: Видавництво Addison — Wesley, 2013. — 369 с.
36. Sanderson C. Armadillo: C++ template metaprogramming for compile-time optimization of linear algebra / C. Sanderson, R. Curtin // In Platform for Advanced Scientific Computing (PASC) Conference, Switzerland, 2017
37. C. Sanderson and R. Curtin. Armadillo: a template-based C++ library for linear algebra. Journal of Open Source Software, 1:26, 2016.
38. Home page mlpack [Електронний ресурс] — Режим доступу: <https://www.mlpack.org/>
39. S. Rossignol Automatic characterisation of musical signals: feature extraction and temporal segmentation / S. Rossignol, X. Rodet, J. Soumagne, J.L. Collette and P. Depalle // Journal of New Music Research, 2000
40. Al-Atiyah A. Music and Speech Separation / Al-Atiyah A. // MS Thesis, King Saud University, Riyadh, Saudi Arabia, 2002.
41. Sound samples [Електронний ресурс] — Режим доступу: [http://www.philharmonia.co.uk/explore/sound\\_samples](http://www.philharmonia.co.uk/explore/sound_samples)
42. Кавецький В. В. Економічне обґрунтування інноваційних рішень: Практикум /В.В.Кавецький, В.О.Козловський, І.В.Причепа. — ВНТУ, 2013. — 110 с.
43. Адлер О.О. Методичні вказівки до підготовки та написання курсової роботи з дисципліни «Економічне обґрунтування інноваційних рішень» / Уклад. О.О.Адлер, І.В.Причепа, Н.М.Тарасюк. — Вінниця: ВНТУ, 2014. — 38 с.
44. Тарифи на електроенергію [Електронний ресурс]. — Режим доступу: <http://index.minfin.com.ua/tarif/electric.php>.