


Вінницький національний технічний університет  
Факультет інформаційних технологій та комп'ютерної інженерії  
Кафедра програмного забезпечення

## МАГІСТЕРСЬКА КВАЛІФІКАЦІЙНА РОБОТА

на тему:

«Методи та програмні засоби оброблення та синхронізації контенту з елементами візуалізації»

Виконав: студент II курсу  
групи ЗП-22м спеціальності  
121 – Інженерія програмного забезпечення

 Базалицький М. Р.

Керівник: д.т.н., проф. Романюк О. Н.

«Н» грудня 2023 р.

Оponent: к.т.н., доц. каф. ОТ Черняк О. І.

«Н» грудня 2023 р.

**Допущено до захисту**

Завідувач кафедри ПЗ

д.т.н., проф. Романюк О. Н.

(прізвище та ініціали)

«Н» грудня 2023 р.

ВНТУ – 2023

Вінницький національний технічний університет  
Факультет інформаційних технологій та комп'ютерної інженерії  
Кафедра програмного забезпечення  
Рівень вищої освіти II-й (магістерський)  
Галузь знань 12 – Інформаційні технології  
Спеціальність 121 – Інженерія програмного забезпечення  
Освітньо-професійна програма – Інженерія програмного забезпечення

ЗАТВЕРДЖУЮ

Завідувач кафедри ПЗ

Романюк О. Н.



«19» вересня 2023 р.

## ЗАВДАННЯ НА МАГІСТЕРСЬКУ КВАЛІФІКАЦІЙНУ РОБОТУ СТУДЕНТУ

Базалицький Максим Романович

1. Тема роботи – Методи та програмні засоби оброблення та синхронізації контенту з елементами візуалізації. Керівник роботи: Романюк Олександр Никифорович, к.т.н., професор кафедри ПЗ, затверджені наказом вищого навчального закладу від “18” вересня 2023 року № 247.

2. Строк подання студентом роботи

5 грудня 2023 р.





3. Вихідні дані до роботи: Кількість фонем – 39 од.; Кількість форм положень вуст – 15 од. Анімаційний редактор MAYA; Мова програмування Python 3.0; Формат тривимірного об'єкта – .ma; Кількість анімаційних точок на поверхні об'єкта – 75 од.

4. Зміст розрахунково-пояснювальної записки: вступ; аналіз предметної області; розробка методу оброблення та синхронізації контенту з елементами візуалізації ; розробка програмних засобів оброблення та синхронізації контенту з елементами візуалізації; економічна частина; висновки; список використаних джерел; додатки.

5. Перелік графічного матеріалу: мета; об'єкт та предмет дослідження; завдання дослідження; аналіз стану питання; порівняння з аналогами; використані технології при розробці системи; тестування системи; наукова новизна одержаних результатів; практична цінність одержаних результатів; апробація та публікації.



6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада Консультанта	Підпис, дата	
		завдання видав	викона прийняв
1-4	Романюк О. Н., д. т. н., професор кафедри ПЗ	19.09.23 	05.12.23 
5	Причепя І. В, к.е.н., доцент кафедри ЕПВМ	27.11.23 	29.11.23 

7. Дата видачі завдання \_\_\_\_\_ 19 вересня 2023 р.

**КАЛЕНДАРНИЙ ПЛАН**

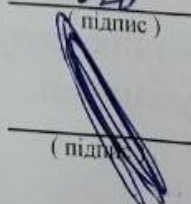
№ з/п	Назва етапів магістерської кваліфікаційної роботи	Строк виконання етапів роботи	Примі
1	Аналіз стану питання	19.09.23 – 25.09.23	<i>вик</i>
2	Аналіз предметної області	26.09.23 – 28.09.23	<i>вик</i>
3	розробка методу оброблення та синхронізації контенту з елементами візуалізації	29.09.23 – 7.10.23	<i>вик</i>
4	розробка програмних засобів оброблення та синхронізації контенту з елементами візуалізації	8.10.23 – 24.11.23	<i>вик</i>
5	Тестування роботи програмного модулю	25.11.23 – 26.11.23	<i>вик</i>
6	Економічна частина	27.11.23 – 29.11.23	<i>вик</i>
7	Оформлення матеріалів до захисту МКР	24.11.23 – 5.12.23	<i>вик</i>

Студент

  
(підпис)

**Базалицький А**  
(прізвище та ініціал)

Керівник магістерської кваліфікаційної роботи

  
(підпис)

**Романюк О. Н.**  
(прізвище та ініціал)

## АНОТАЦІЯ

УДК 004.92

Базалицький М. Р. Методи та програмні засоби оброблення та синхронізації контенту з елементами візуалізації. Магістерська кваліфікаційна робота зі спеціальності 121 – Інженерія програмного забезпечення, освітня програма – Інженерія програмного забезпечення. Вінниця: ВНТУ, 2023. 54 с.

На укр. мові. Бібліогр.: 28 назв; рис.: 18; табл. 9.

Магістерська кваліфікаційна робота присвячена методам та програмним засобам обробки та синхронізації контенту та відповідних зображень. Розроблено блок-схеми, діаграми прецедентів та алгоритми програмного модуля для обробки полігональних моделей анімованих зображень. Також розроблено метод створення анімації обличчя. Розроблений програмний модуль призначений для створення анімації обличчя з використанням бібліотеки мов обробки природної мови NLTK.

Для реалізації проекту використовувався редактор анімації MAYA та мова програмування Python. Для створення користувацького інтерфейсу використано бібліотеку PyQt. Розроблений додаток прискорює роботу над анімацією обличчя на 75% та повністю автоматизує процес.

Графічна частина складається з десяти плакатів із результатами проектування та реалізації.

Ключові слова: фонемі, форми вуст, полігональна модель, лінгвістичні вирази.

## **ABSTRACT**

Bazalytskyi M.R. methods and software tools for processing and synchronizing content and its corresponding image. Master's thesis on specialty 121 - Software engineering, educational program - Software engineering. Vinnytsia: VNTU, 2023. 54 p.

In Ukrainian language. Bibliography: 19 titles; fig.: 18; table 9.

The master's thesis is devoted to methods and software tools for processing and synchronizing content and corresponding images. Flow charts, case diagrams and software module algorithms for processing polygonal models of animated images have been developed. A method of creating facial animation has also been developed. The developed software module is designed to create facial animation using the library of natural language processing languages. The animation editor MAYA and the Python programming language were used to implement the project. PyQt was used to create the user interface. The developed application accelerates work on facial animation by 75% and fully automates the process.

The graphic part consists of five posters with the results of design and implementation.

Key words: phonemes, shapes of lips, polygonal model, linguistic expressions.

## ЗМІСТ

ВСТУП.....	4
1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ.....	8
1.1 Аналіз методів та засобів оброблення та синхронізації контенту з елементами візуалізації .....	8
1.2 Аналіз аналогів програмної реалізації оброблення та синхронізації контенту з елементами візуалізації .....	13
1.3 Лінгвістика та особливості трансформації текстового файлу в анімацію обличчя .....	18
1.4 Висновки.....	20
2 РОЗРОБКА МЕТОДУ ОБРОБЛЕННЯ ТА СИНХРОНІЗАЦІЇ КОНТЕНТУ З ЕЛЕМЕНТАМИ ВІЗУАЛІЗАЦІЇ.....	21
2.1 Обґрунтування вибору тривимірного редактору для розробки програмного модуля та інструментів його роботи .....	21
2.2 Розробка методу розбиття слова на фонетичні звуки .....	32
2.3 Розробка математичної моделі анімаційного зображення.....	35
2.4 Розробка структурної схеми функціонування програмного засобу .....	39
2.5 Розробка інтерактивних анімацій обличчя для програмного засобу оброблення та синхронізації контенту з елементами візуалізації.....	42
2.6 Розробка алгоритму функціонування програмного засобу оброблення та синхронізації контенту з елементами візуалізації.....	46
2.7 Висновки .....	47
3 ПРОГРАМНА РЕАЛІЗАЦІЯ МОДУЛЯ ОБРОБЛЕННЯ ПОЛІГОНАЛЬНИХ МОДЕЛЕЙ АНІМАЦІЙНИХ ЗОБРАЖЕНЬ.....	49
3.1 Обґрунтування вибору мови програмування та особливості платформи реалізації.....	49
3.2 Аналіз середовища розробки .....	53
3.3 Програмна реалізація засобу оброблення та синхронізації контенту з елементами візуалізації.....	55

3.4 Висновки.....	63
4. ТЕСТУВАННЯ ПРОГРАМНОГО ЗАСОБУ ОБРОБЛЕННЯ ТА СИНХРОНІЗАЦІЇ КОНТЕНТУ З ЕЛЕМЕНТАМИ ВІЗУАЛІЗАЦІЇ.....	64
4.1 Аналіз підходів до тестування програмного засобу .....	64
4.2 Тестування програмного забезпечення.....	65
4.3 Розробка інструкції користувача .....	69
4.4 Висновки .....	70
5 ЕКОНОМІЧНА ЧАСТИНА.....	71
5.1 Проведення комерційного та технологічного аудиту науково-технічної розробки .....	72
5.2 Розрахунок витрат на проведення науково-дослідної роботи.....	75
5.3 Розрахунок економічної ефективності науково-технічної розробки при її можливій комерціалізації потенційним інвестором.....	86
5.5 Висновки .....	90
ВИСНОВКИ .....	91
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	94
ДОДАТОК А (обов'язковий). Технічне завдання.....	97
ДОДАТОК Б (обов'язковий). Протокол перевірки навчальної роботи....	100
ДОДАТОК В (довідниковий). Лістинг коду.....	101
ДОДАТОК Г (обов'язковий). Ілюстративна частина .....	112

## ВСТУП

**Обґрунтування вибору теми дослідження.** Сьогодні швидка анімація необхідна для створення автентичних віртуальних середовищ [1]. Для автоматизації процесу анімації 3D-персонажів у мультимедійних продуктах та покращення якості анімованих поверхонь у заданих точках координат поверхні доцільним є використання підходу розбиття на фонетичні звуки текстових файлів.

Лицьова анімація [2] відноситься, насамперед, до області комп'ютерної графіки і включає в себе методи та прийоми створення і анімації зображень і моделей облич персонажів. Персонажами можуть бути люди, гуманоїди, тварини або міфологічні істоти. Залежно від тематики та типу виводу, персонажі безпосередньо пов'язані з багатьма іншими науковими та мистецькими об'єктами людської діяльності, починаючи від психології і закінчуючи традиційною анімацією.

Ця технологія має потенціал змінити спосіб спілкування з віртуальними персонажами, робить інтеграції більш природними та імерсійними. Зростання популярності віртуальної реальності та розширеної реальності робить цю тему ще більш актуальною, оскільки реалістичні обличчя стають ключовим елементом для імерсійного взаємодії з цими технологіями.

Крім того, можливість використання технології автоматизованої анімації обличчя в освіті та навчанні мов може відкривати нові шляхи для покращення навчального процесу та розвитку мовленнєвих навичок. Таким чином, тема створення анімації обличчя на основі текстових файлів визначається не лише технічними можливостями, але і соціокультурним та освітнім впливом, що робить її особливо актуальною у контексті сучасних тенденцій.

З використанням текстових файлів для програмування анімації обличчя відкривається можливість більш гнучкого та ефективного управління



емоційним виразом персонажів. Це дозволяє легко змінювати та адаптувати анімації, щоб вони відповідали різним сценаріям та контекстам використання, будь то у відеоіграх, фільмах чи навчальних програмах.

Застосування такої технології також дозволяє розширити можливості автоматизації процесу анімації. Завдяки текстовим файлам, можна розробляти скрипти та програми для генерації анімації автоматично, що прискорює та оптимізує робочий процес.

Особливо важливим є потенціал використання цієї технології для створення персоналізованих та інтерактивних віртуальних асистентів. Здатність передавати емоційні вирази через анімацію обличчя робить їх більш доступними та природними в спілкуванні.

Необхідність дослідження та розвитку цього напрямку визначається швидким розвитком сучасних технологій та попитом на інноваційні рішення у галузі комп'ютерної графіки та віртуальної реальності.

Таким чином розробка програмного модулю оброблення полігональних моделей анімаційних зображень є актуальною задачею.

**Зв'язок роботи з науковими програмами, планами, темами.** Робота виконувалась відповідно до плану науково-дослідних робіт кафедри програмного забезпечення.

**Мета та завдання дослідження.** Метою роботи є зменшення часу проектування анімацій обличчя тривимірного персонажа за рахунок розробки програмних засобів оброблення та синхронізації контенту з елементами візуалізації.

Для досягнення поставленої мети необхідно розв'язати такі задачі:

1) Проаналізувати предметну галузь розробки анімацій для тривимірних персонажів.

2) Обґрунтувати вибір тривимірного редактору для розробки методів та програмних засобів оброблення та синхронізації контенту з елементами візуалізації.

3) Розробити методи оброблення та синхронізації контенту і

відповідного йому зображення.

4) Розробити алгоритм функціонування програмних засобів оброблення синхронізації контенту та відповідного йому зображення;

5) Розробити програмний засіб оброблення та синхронізації контенту з елементами візуалізації.

б) Провести тестування розробленого програмного засобу оброблення і синхронізації контенту ти відповідного йому зображення.

**Об'єкт дослідження** – процес розробки методів і програмних засобів автоматизації створення анімацій обличчя шляхом перенесення інформації з тестового файлу в анімаційний редактор та трансформації її в фонему для відображення динамічних форм вуст.

**Предмет дослідження** – методи та програмні засоби оброблення та синхронізації контенту з елементами візуалізації.

**Методи дослідження.** У процесі досліджень використовувались: теорія розпізнавання образів, артикуляторна фонетика та лінгвістика, теорія комп'ютерної графіки та анімації, методи скелетної анімації для розробки моделей та методів синхронізації контенту і анімації руху губ; комп'ютерне моделювання для аналізу та перевірки отриманих теоретичних положень.

#### **Наукова новизна одержаних результатів.**

1. Вперше запропоновано математичну модель лицьової анімації, особливістю якої є використання текстових файлів для генерації анімації обличчя тривимірних персонажів, що дозволило скоротити час створення контенту на 75%.
2. Подальшого розвитку отримав метод фонетичного морфінгу для анімації обличчя, у якому на відміну від існуючих використано обробка природної мови NLTK, що дозволило підвищити реалістичність анімації з урахуванням вимови та інтонації слів.

**Практична цінність отриманих результатів.** Практична цінність одержаних результатів полягає в тому, що на основі отриманих в магістерській кваліфікаційній роботі теоретичних положень запропоновано

алгоритми та розроблено програмний засіб оброблення та синхронізації контенту з елементами візуалізації, який автоматично створює анімацію обличчя тривимірного персонажу на основі текстового файлу, а також вбудовано цей програмний засіб у тривимірний редактор MAYA.

**Особистий внесок здобувача.** Усі наукові результати, викладені у магістерській кваліфікаційній роботі, отримані автором особисто. У роботах, опублікованих у співавторстві, здобувачу належать такі результати: розробка математичної моделі розбиття слова на фонетичні звуки та коректна їх анімація [1]; методи та програмні засоби оброблення полігональних моделей анімаційних зображень [2].

**Апробація матеріалів магістерської кваліфікаційної роботи.** Основні положення магістерської кваліфікаційної роботи доповідалися та обговорювалися на XVI міжнародній науково-практичній конференції «Інформаційні технології і автоматизація» – Одеса 2023, на Міжнародній науково-практичній Інтернет-конференції «Електронні інформаційні ресурси: створення, використання, доступ» – Вінниця 2023.

**Публікації.** Основні результати досліджень опубліковано в 2 наукових працях у матеріалах конференцій.

## 1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

### 1.1 Аналіз методів та засобів оброблення та синхронізації контенту з елементами візуалізації

Програмний засіб [3] – це комплекс програмних та апаратних засобів, призначений для вирішення конкретних завдань або виконання певних функцій у галузі програмування та інформатики. Це може включати в себе інструменти розробки програм, компілятори, інтерфейси користувача, бібліотеки, середовища виконання та інші компоненти, які сприяють розробці, виконанню та управлінню програмами. Програмні засоби використовуються для різноманітних цілей, включаючи створення веб-додатків, наукових досліджень, систем управління базами даних, графічного дизайну та інше.

Програмний модуль [4] - це фрагмент програмного забезпечення, який виконує конкретну функцію чи завдання. Він представляє собою логічно об'єднаний блок коду, який може бути використаний у складі більшої програми або самостійно. Модулі допомагають розділити великі програми на менші, більш керовані частини, що полегшує розробку, тестування та підтримку програмного забезпечення. Вони можуть містити функції, класи, змінні та інші елементи, які взаємодіють для досягнення конкретної мети. Програмні модулі можна використовувати у різних проектах або обмінювати ними між розробниками для полегшення роботи.

Тривимірна графіка [5] – це розділ комп'ютерної графіки, що спеціалізується на методах створення зображень і переміщення зображень шляхом моделювання об'єктів у тривимірному просторі [4]. Основний процес моделювання полягає у з'єднанні набору точок лініями або полігонами для створення каркасної моделі (рис. 1.1).

Вертакси [6] , у тривимірній графіці, це точки або вершини, які

утворюють основну структуру тривимірного об'єкта. Кожен вертекс має свої координати в тривимірному просторі ( $x, y, z$ ) і визначається їхньою позицією у просторі. Група вертексів утворює меш (сітку), яка визначає форму об'єкта. Рух, трансформації та властивості об'єкта у тривимірному просторі обчислюються на основі координат вертексів.

Едж [5] в тривимірній графіці визначається як лінія або відрізок, який з'єднує дві вершини (вертекси або точки) у тривимірному просторі. Еджі визначають границі і форму об'єкта, а їхня довжина та напрямок впливають на вигляд об'єкта у тривимірному просторі. Група еджів утворює полігон, який може бути видимою поверхнею на 3D-моделі. Еджі важливі для визначення структури та деталей об'єктів у тривимірній графіці.

Грань [5] в тривимірній графіці представляє собою плоску поверхню або полігон, обмежений еджами (відрізками або лініями), які з'єднують вершини (вертекси або точки). Грань визначає форму та розміри об'єкта в просторі. Вона є важливим елементом для побудови поверхонь 3D-моделі, і її властивості включають колір, текстури, матеріали та інші атрибути, які впливають на зовнішній вигляд об'єкта. Група граней утворює поверхню об'єкта, яку можна відобразити та взаємодіяти з нею у тривимірному просторі.

Полігон [5] в тривимірній графіці є плоскою геометричною фігурою, що обмежується трьома чи більше точками (вертексами) і еджами (відрізками або лініями), які з'єднують ці точки. Полігон може мати три, чотири, або більше вертексів, і він визначає плоску поверхню в тривимірному просторі.

Полігональна сітка у тривимірній графіці представляє собою мережу, складену з полігонів, які утворюють поверхні тривимірних об'єктів. Це мережа з'єднаних еджів та вертексів, яка визначає форму та структуру об'єкта у тривимірному просторі. Гранями зазвичай є трикутники, чотирикутники або інші прості опуклі багатокутники (полігони), так як це спрощує рендеринг, але сітки можуть також складатися з найбільш загальних



увігнутих багатокутників, або багатокутників з отворами [4].

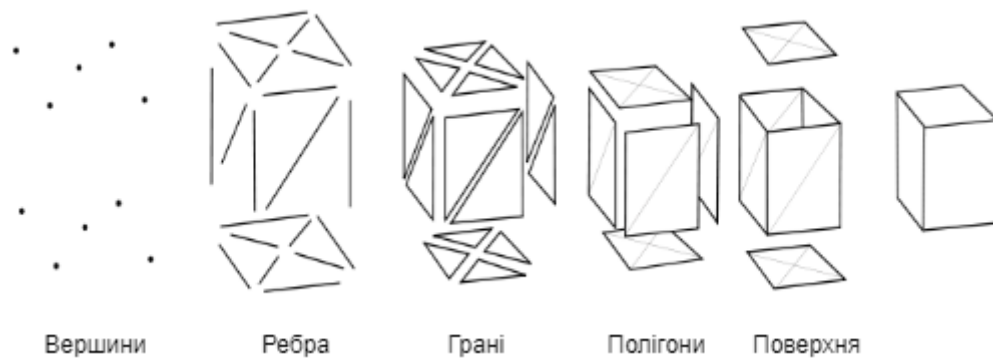


Рисунок 1.1 – візуальний приклад вершин, ребр, полігонів, поверхонь

Тривимірна анімація [7] - це процес створення рухомого зображення в тривимірному просторі. Вона включає в себе створення і рух тривимірних об'єктів, які змінюють своє положення, форму та вигляд в часі. Така анімація може бути використана у відеоіграх, фільмах, рекламі, наукових візуалізаціях та інших галузях .

Процес тривимірної анімації включає в себе створення моделей об'єктів, їх анімаційних рухів, встановлення освітлення та текстур, а також віртуальну камеру для перегляду сцени. За допомогою комп'ютерних програм і спеціалізованих технологій, тривимірні об'єкти стають живими, а їхні рухи і взаємодії можуть бути створені та контрольовані художниками чи аніматорами.

Скелетна анімація [8] — це метод тривимірної анімації, в якому модель персонажа або об'єкта представлена у вигляді скелета, що складається з кісток і з'єднаних між ними сегментів. Кожна кістка відповідає частині тіла чи об'єкту, і за допомогою анімаційних рухів кісток створюється реалістичний рух моделі.

У скелетній анімації для керування рухом моделі використовуються ключові кадри, які визначають положення і обертання кісток у певний момент часу. Спеціальні програми або редактори дозволяють аніматорам встановлювати ці ключові кадри, а решта руху обчислюється автоматично.

Цей підхід робить анімацію більш ефективною та легкою для контролю над рухами персонажа чи об'єкта.

Масиви кісток можуть варіюватися від декількох кісток до майже 100 кісток, щоб врахувати всі найтонші вирази обличчя. Основна перевага кісткової анімації полягає в тому, що ту саму анімацію можна використовувати для різних персонажів. Кісткова анімація найширше підтримується тривимірними (3D) ігровими рушіями. Кісткова анімація може використовуватися як з двовимірною (2D), так і з тривимірною (3D) графікою. Зокрема, такелаж відноситься до процесу створення кісткової структури 3D-моделі. Ця кісткова структура використовується для управління 3D-моделлю як маріонеткою для анімації. Створення скелета для анімації називається ригінг.

Ригінг [8] в тривимірній анімації та графіці — це процес створення цифрового скелета чи арматури, яка надає моделі гнучкість та можливість анімації. Ригінг включає в себе призначення контрольних точок (які називаються бонами або джойнтами) на моделі та встановлення взаємозв'язків між цими точками.

Після завершення ригінгу аніматор отримує можливість контролювати та анімувати модель так, ніби вона обладнана внутрішнім скелетом. Це дозволяє створювати природні, плавні та реалістичні анімації, які відображають рухи та деформації об'єкта в тривимірному просторі.

Цей процес дозволяє аніматорам керувати рухами та деформацією моделі, як це робиться в скелетній анімації. Ригінг є важливою частиною підготовки тривимірної моделі для анімації, забезпечуючи можливість створювати природні та реалістичні рухи об'єктів чи персонажів. Приклад кістяної ієрархії зображено на рисунку 1.2.

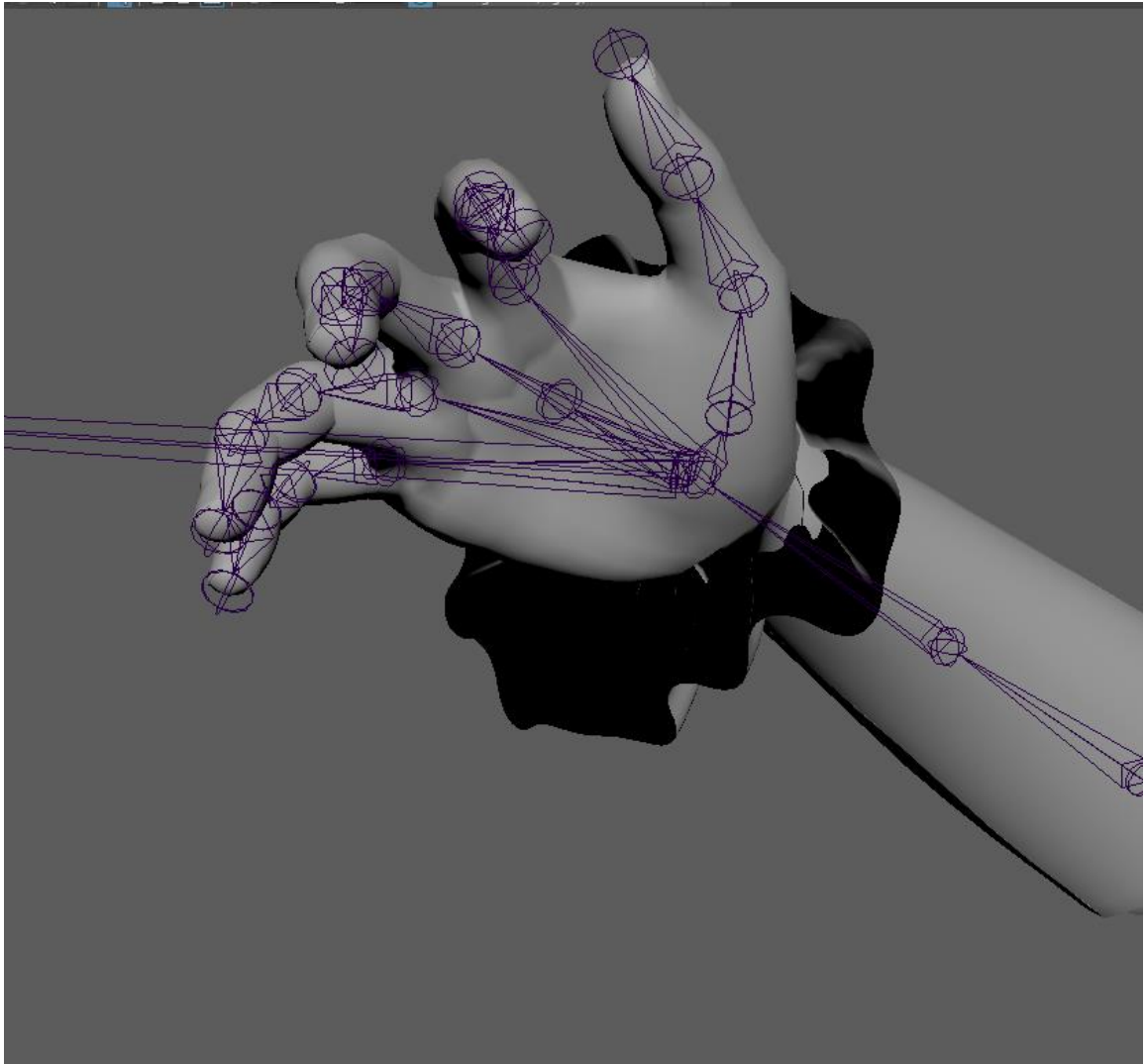


Рисунок 1.2 – Схема кістяної ієрархії пальців

Скінінг [9] в тривимірній графіці — це процес прив'язки поверхні тривимірного об'єкта до його скелета або ріга. Цей процес дозволяє об'єкту деформуватися та змінювати свою форму відповідно до рухів скелета під час анімації.

Скінінг включає в себе призначення ваг для кожного вертекса (точки) поверхні моделі, які визначають, як сильно кожен вертекс буде піддаватися впливу того чи іншого бону або джойнта скелета. Цей процес робить можливим відтворення природних рухів та деформацій об'єкта під час анімації.

Скінінг є важливою частиною підготовки моделі до анімації, сприяючи створенню реалістичних та еластичних рухів в тривимірному просторі.

Анімація обличчя складається з послідовності змін форми обличчя для

відтворення мовленнєвих елементів та вираження емоцій. У цьому процесі кожен лінгвістичний склад слів відтворюється через динамічні зміни положення та виразів різних частин обличчя, таких як губи, очі, брови і інші. Лицьова анімація також враховує паузи між словами та використання знаків пунктуації для створення природного вигляду мовленнєвого виразу.

Для системи імітації мовлення достатньо п'ятнадцяти різних форм людського рота. У людській мові різні слова містять приголосні літери, які змушують щелепи закриватися і відкриватися. Цей принцип можна побачити в таких поширених словах, як "параплан" і "таран", де щелепи відкриваються і закриваються три-чотири рази.

Ще одним важливим елементом в анімації обличчя є час анімованої зміни форми губ. До кожного слова додається двокадрова пауза, а до розділових знаків, таких як коми та крапки, - п'ятикадрова пауза [1].

## **1.2 Аналіз аналогів програмної реалізації оброблення та синхронізації контенту з елементами візуалізації**

Існують три методи роботи зі скелетною анімацією:

### 1) Ручна анімація:

Опис: У цьому методі аніматор вручну встановлює ключові кадри для кожного моменту анімації. Він обирає конкретні положення, обертання та масштабування кожної частини скелета чи об'єкта на різних етапах анімації.

Переваги: Повний контроль над кожним елементом анімації, можливість створення унікальних та креативних рухів.

Недоліки: Великий обсяг ручної роботи, часово інтенсивний, не завжди ефективний для великих об'ємів анімації.

### 2) Захоплення руху:

Опис: Цей метод використовує датчики для захоплення рухів актора в реальному часі та передачі їх на тривимірну модель. Такий підхід дозволяє аніматорам реалізувати натуральні та автентичні рухи, оскільки вони

базуються на реальних рухах живого об'єкта [10].

Переваги: Реалістичний рух, особливо важливий для анімації людського обличчя, ефективний для анімації динамічних сцен.

Недоліки: Вимагає спеціального обладнання, обмежений контроль над деталями анімації, не завжди підходить для абстрактних чи надмірно фантастичних анімацій.

### 3) Автоматична анімація:

Опис: Цей метод використовує алгоритми або готові бібліотеки для автоматичного генерування анімації. Аніматор встановлює певні параметри або використовує ключові слова, а програма генерує відповідні рухи.

Переваги: Швидше створення анімацій, особливо для повторюваних дій, менше ручної роботи.

Недоліки: Менше контролю над деталями, може виглядати менш унікальним, не завжди підходить для складних або високоінтенсивних сцен.

На сьогоднішній день існує достатня кількість програмних реалізацій оброблення та синхронізації контенту з елементами візуалізації. Найвідоміші з них це: MetaHuman, AUDIO2FACE, Animaze.

MetaHuman [11]– це інноваційний інструмент, розроблений компанією Epic Games, який дозволяє створювати вражаючі тривимірні людські обличчя та персонажі для використання у відеоіграх, анімації, фільмах та інших областях інтерактивного контенту. Цей інструмент визначається високоякісною графікою, реалістичними анімаціями та великою кількістю параметрів для налаштування вигляду персонажів. MetaHuman дозволяє розробникам швидко та ефективно створювати персоналізованих цифрових акторів з різноманітними виразами обличчя, стилізованими волоссям, реалістичною текстурою шкіри та іншими деталями, що сприяє створенню вражаючого та унікального контенту в галузі розважальної індустрії. Приклад роботи MetaHuman зображена на рисунку 1.3.



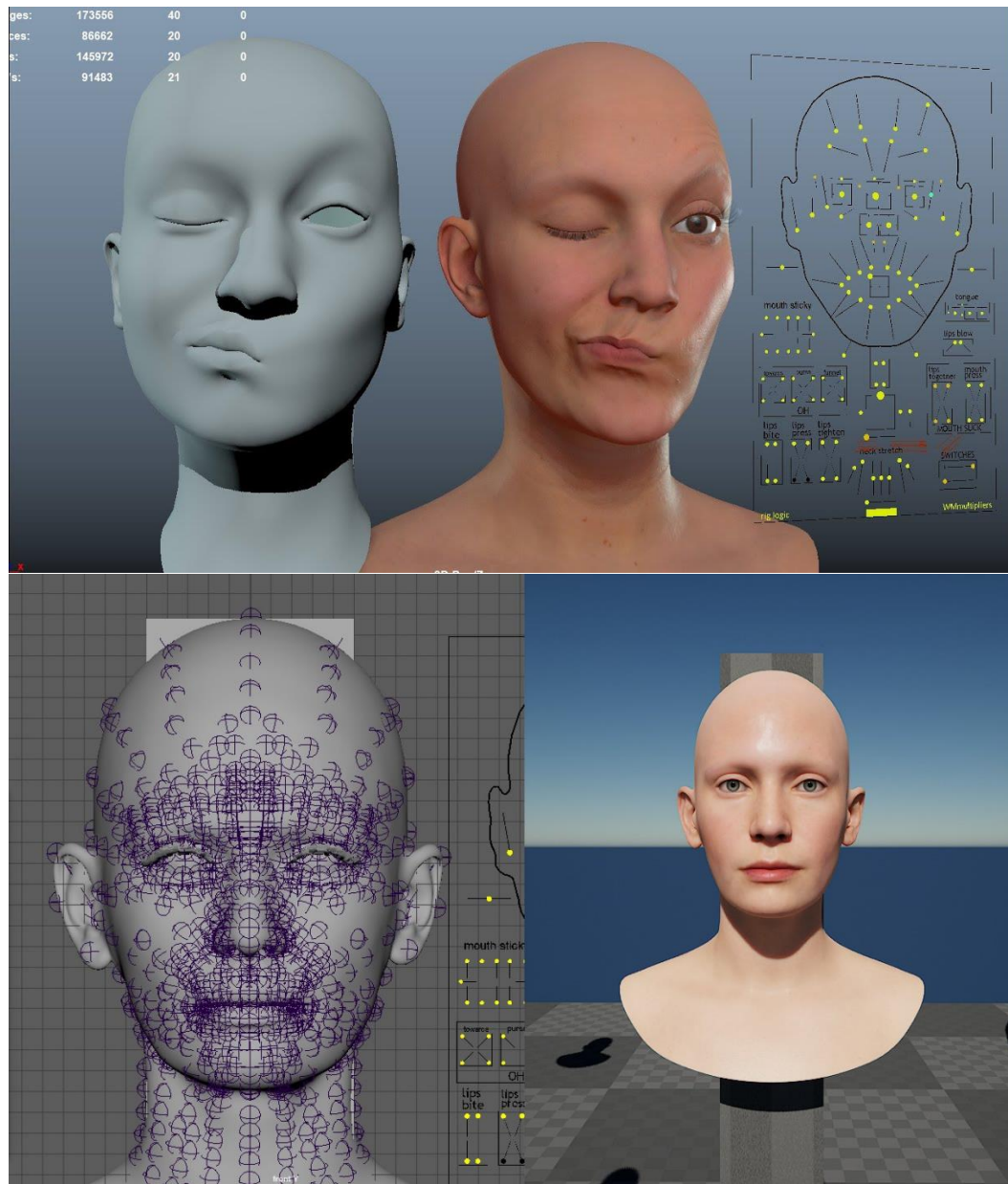


Рисунок 1.3 – Загальний вигляд роботи MetaHuman

AUDIO2FACE [12]– це технологія, розроблена NVIDIA, яка дозволяє створювати автентичну анімацію обличчя для тривимірних персонажів на основі аудіозаписів. Ця технологія використовує машинне навчання для синхронізації рухів обличчя зі звуковими сигналами, надаючи реалістичну і виразну анімацію обличчя.

AUDIO2FACE може бути використаний у відеоіграх, фільмах, віртуальній реальності та інших областях, де потрібна автоматизована синхронізація рухів обличчя з аудіо контентом. Ця технологія спрощує

процес створення реалістичної анімації, роблячи її більш доступною для розробників та творців контенту. Приклад роботи AUDIO2FACE зображено на рисунку 1.4.

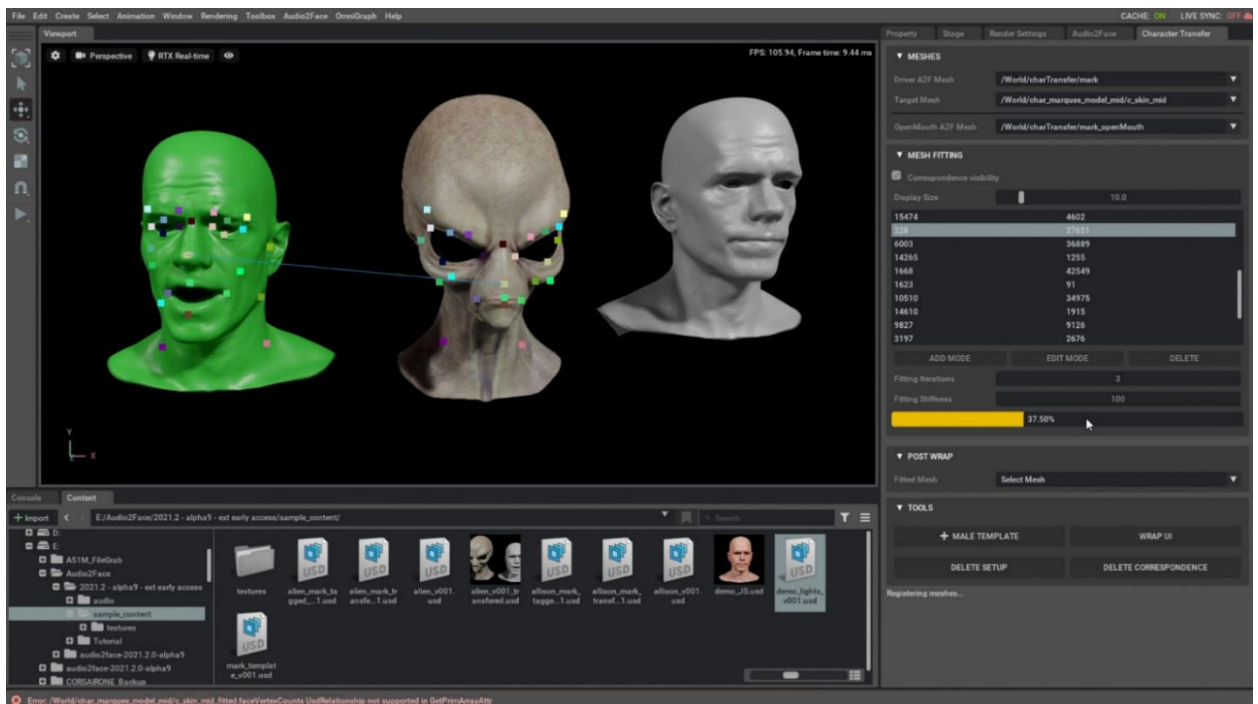


Рисунок 1.4 – Загальний вигляд роботи AUDIO2FACE

Animaze [13]– це програмний продукт, який дозволяє користувачам створювати анімаційні 3D-моделі обличчя та взаємодіяти з ними у режимі реального часу. Цей інструмент призначений для створення живої та експресивної анімації обличчя, яку можна використовувати для онлайн-спілкування, стрімінгу, чату та інших форм віртуальної взаємодії.

Animaze використовує технологію захоплення рухів обличчя та передачі їх на тривимірну модель, дозволяючи в режимі реального часу відтворювати вирази обличчя та рухи. Це може бути використано для створення персональних анімованих персонажів, які відображають емоції та вирази користувача. Animaze широко використовується в інтерактивних медіа, соціальних мережах та інших платформах, де важлива віртуальна комунікація. Приклад роботи Animaze зображено на рисунку 1.5.

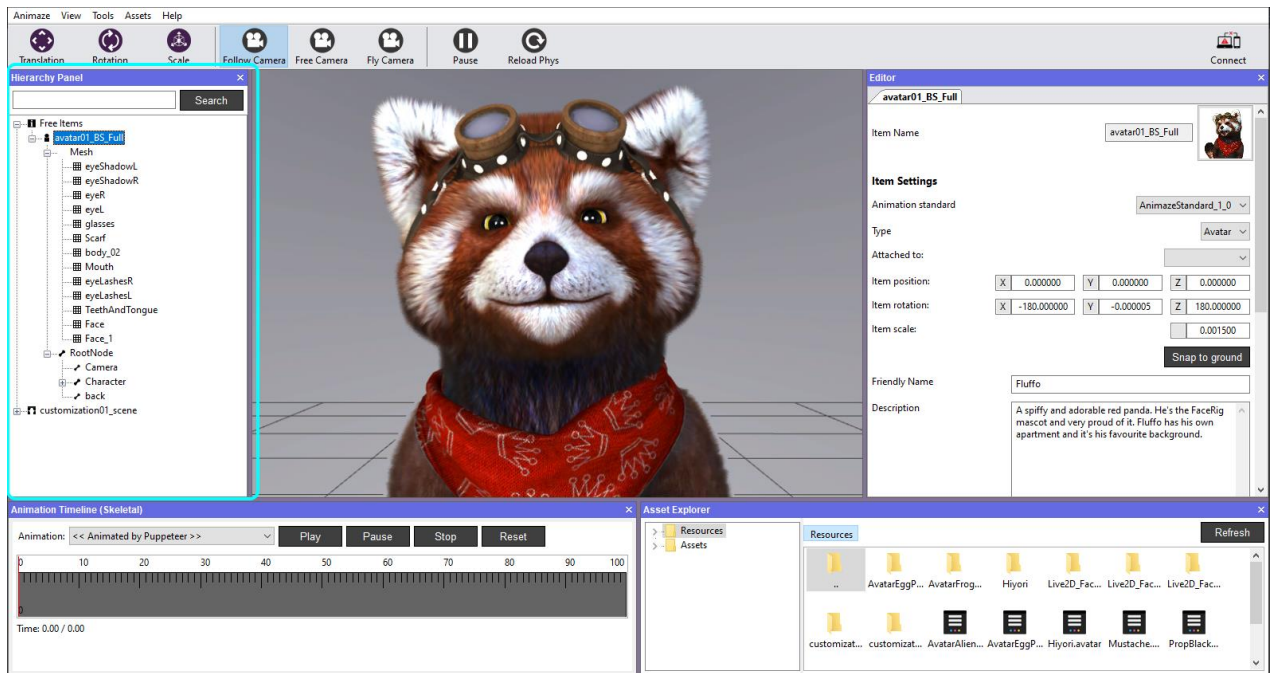


Рисунок 1.5 – Загальний вигляд роботи Animaze

Розглянувши вищезазначені подібності, можна зробити висновок, що для повної автоматизації виробництва анімації цифрових 3D-персонажів необхідно створювати анімацію не тільки з аудіо- та відеоджерел, а й у форматі текстового файлу. Існує також необхідність подальшого використання цієї анімації для створення високоякісних медіапродуктів.

### **1.3 Лінгвістика та особливості трансформації текстового файлу в анімацію обличчя**

Лінгвістика [14]- це наука, що вивчає мову, її структуру та функції. У контексті створення анімації обличчя, лінгвістика грає ключову роль у розумінні, як перетворити текстовий файл на послідовність фонетичних звуків. Сучасні методи використовують фонетичний розбір слів, щоб визначити правильні анімаційні параметри для кожного фонетичного елементу.

Важливо розуміти, як звукова структура слів впливає на рухи губ та обличчя. Різні звуки активують різні м'язи, що визначає форму та рух обличчя. Лінгвістика допомагає створити відображення мовної інформації в анімаційних параметрах, забезпечуючи натуральність та виразність анімації.

Процес трансформації текстового файлу в анімацію обличчя включає кілька ключових етапів. Починаючи зі зчитування тексту, важливо розбити його на слова та подальше на фонетичні склади. Цей розбір використовує лінгвістичні методи для визначення фонетичної структури слів.

Далі відбувається перетворення фонетичних елементів у відповідні анімаційні параметри. Наприклад, кожний фонетичний звук може визначати рухи губ, позицію язика та інші обличчєві аспекти. Тут лінгвістика взаємодіє з тривимірною графікою та анімаційною обробкою для створення повноцінної лицьової анімації.

Сучасні технології використовують фонетичну розбудову слів для створення реалістичних анімацій обличчя. Важливим елементом є час анімаційної зміни форми губ, де кожний лінгвістичний склад слів відтворюється через кілька кадрів анімації. Знаки пунктуації додають паузи для реалістичності.

У реалізації цих концепцій використовуються бібліотеки, такі як Natural Language Toolkit (NLTK) у мові програмування Python. Це дозволяє автоматизувати процес трансформації фонетичних елементів у параметри

анімації. Такий підхід робить анімацію обличчя більш деталізованою та виразною, а лінгвістичний аналіз тексту стає ключовим етапом у створенні реалістичних персонажів у тривимірному середовищі.

Міміка, або мова обличчя, вкладається в ефективність анімації обличчя під час говоріння. Рухи губ визначають звуковий спектр мови і, таким чином, стають ключовим елементом анімації. Під час говоріння, різні фонетичні елементи викликають різноманітні рухи губ, відкриваючи можливість виразно відображати звучання слів у тривимірному просторі.

Розрізняючи між великою кількістю рухів губ, анімаційна система вдається відтворити не лише звуковий вихід, але і емоційний фон тексту. Таким чином, міміка грає важливу роль у покращенні природності та виразності анімованого персонажа.

Під час говоріння не лише рухи губ є важливими. Рухи голови, язика, замикання та відмикання рота також вносять суттєвий внесок у виразність анімації обличчя. Головний рух може підсилити вираз слова або вказати на зміну настрою. Рухи язика додають реалістичності та показують динаміку мовлення.

Замикання та відмикання рота дозволяє показати важливі аспекти голосового вираження. Ці рухи можуть бути суттєвими для правильної передачі приголосних звуків та інтонації. У відповідь на різні фонетичні елементи, система анімації відтворює відповідні рухи, створюючи гармонійну та реалістичну анімацію обличчя.

Ці особливості додають глибину та автентичність створеним персонажам, роблячи їх більш схожими на реальних осіб. Отже, врахування різноманітних рухів обличчя є ключем до успішної трансформації текстового файлу в живу та виразну анімацію.

Кліпання очей може змінюватися залежно від настрою та емоційного стану персонажа. Наприклад, активне кліпання може вказувати на збудженість чи радість, в той час як менше інтенсивне кліпання може вказувати на спокій чи концентрацію. Врахування цих деталей важливо для створення природного



та реалістичного образу персонажа.

Загальна координація рухів губ, голови, язика та очей узгоджена для відтворення повного спектру мовленнєвих та емоційних виразів. Такий підхід не лише покращує відтворення мови, а й робить анімацію обличчя більш живою та виразною.

#### **1.4 Висновки**

1. Аналіз методів обробки полігональних моделей анімованих зображень показав, що найбільш прийнятним для подальшого аналізу та використання є метод скелетної анімації, оскільки цей метод характеризується меншими обчислювальними затратами порівняно з аналогами, що дає можливість його реалізації чисто програмним способом.

2. Розглянуто особливості реалізації лінгвістичних засобів оброблення та синхронізації контенту з елементами візуалізації, виділено основні особливості міміки обличчя, які потребують реалізації в програмному модулі.

3. Показано, що відомі програмні реалізації фонетичного морфінгу для анімації обличчя вимагають участі оператора на різних етапах створення ліпсингової анімації, тому актуальною є задача автоматизації процесу анімації обличчя.

## 2 РОЗРОБКА МЕТОДУ ОБРОБЛЕННЯ ТА СИНХРОНІЗАЦІЇ КОНТЕНТУ З ЕЛЕМЕНТАМИ ВІЗУАЛІЗАЦІЇ

### 2.1 Обґрунтування вибору тривимірного редактору для розробки програмного модуля та інструментів його роботи

Для вибору тривимірного редактора і порівняв між собою три конкуруючі програми такі як: Maya, Blender та 3DMax.

Autodesk Maya [15] - це професійний 3D-редактор, розроблений компанією Autodesk. Цей програмний продукт використовується для створення тривимірної графіки, анімації, візуалізації та спеціальних ефектів у різних галузях, таких як відеоігри, кіно, телебачення та дизайн.

Autodesk Maya надає широкі можливості для моделювання 3D-об'єктів, анімації персонажів, створення візуально захопливих сцен та спеціальних ефектів. Він використовується професіоналами у галузі комп'ютерної графіки, анімації та візуальних ефектів для реалізації творчих ідей та проектів. Maya також підтримує розширення та скриптовання, що дозволяє користувачам адаптувати програму під свої потреби. Загальний вигляд інтерфейсу редактора анімації Maya зображений на рисунку 2.1.

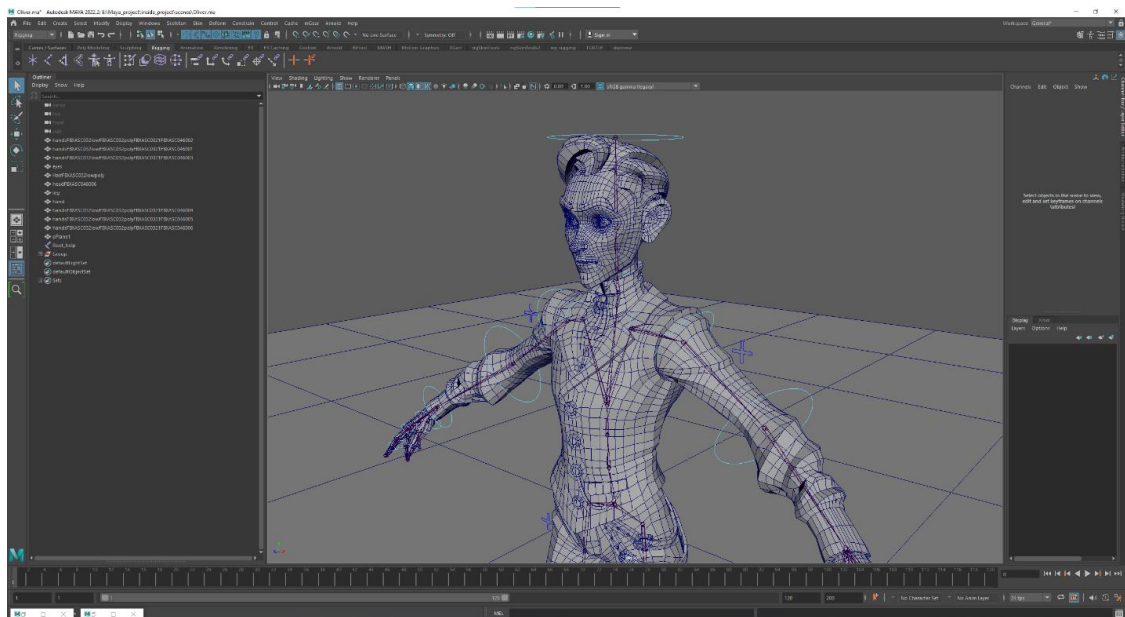


Рисунок 2.1 – Загальний вигляд інтерфейсу редактора анімації Maya.

Перевага редактора анімації Maya полягає в тому, що це неймовірно гнучкий інструмент. Вбудована мова сценаріїв MEL (Maya Embedded Language) може використовуватися для створення власних інструментів. Крім того, скрипти на Python - популярна мова програмування. В інтернеті є багато корисних скриптів, які значно полегшують життя аніматора і позбавляють його від непотрібної рутинної роботи. Багато з цих скриптів тепер є частиною нової версії редактора анімації Maya, яка має зручний редактор скриптів, що дозволяє писати програмні модулі в редакторі анімації Maya. (рис. 2.2).

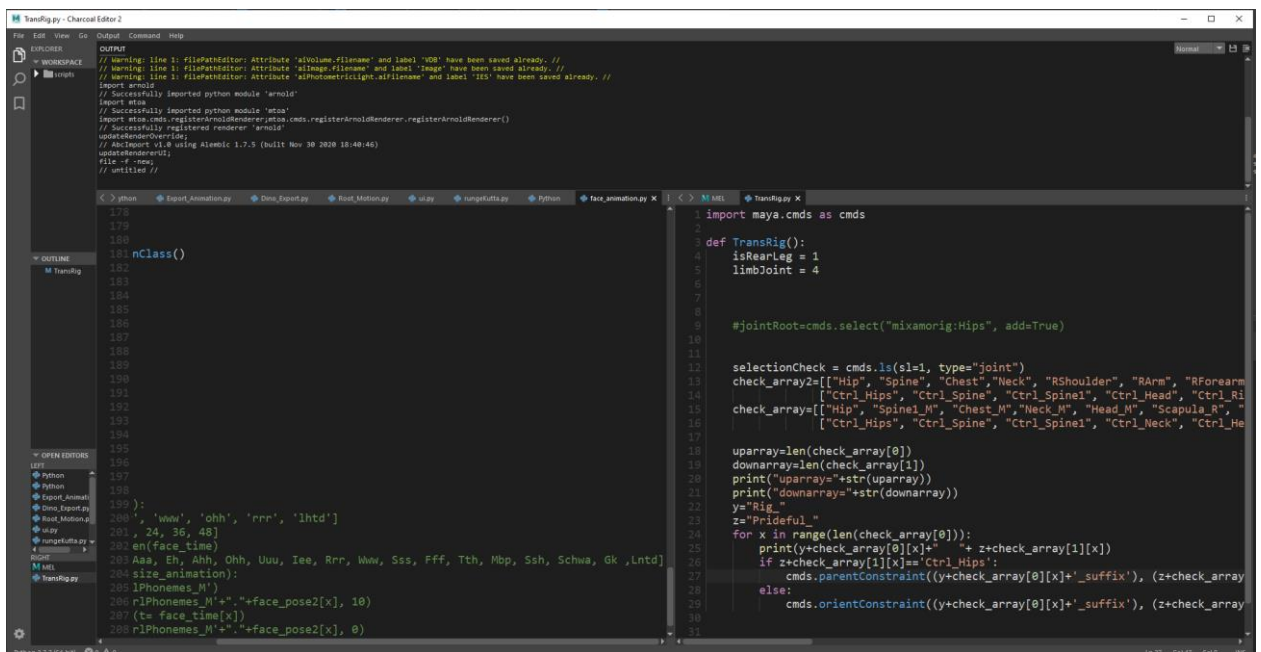


Рисунок 2.2 – Загальний вигляд інтерфейсу редактору коду Script Editor

Анімаційний редактор Maya – стандарт індустрії у сфері тривимірного моделювання, анімації та візуалізації. Розроблений компанією Autodesk, цей програмний продукт володіє величезними можливостями та визнаним впливом на професіоналів у галузі комп'ютерної графіки.

Maya використовується в різних галузях, включаючи відеоігри, кіноіндустрію, телебачення та дизайн. Він надає інструменти для створення складних 3D-моделей, анімації персонажів, симуляції різних об'єктів та застосування вражаючих візуальних ефектів.

Мауа вирізняється великою гнучкістю, можливістю розширення та широким спектром інтегрованих інструментів, що робить його невід'ємною частиною творчого процесу в індустрії 3D-графіки.

Для роботи з тривимірною анімацією та підготовкою її до роботи в анімаційному редакторі МАУА існують такі інструменти:

Advanced Skeleton [16] – це плагін для анімаційного редактора Мауа, який надає продвинуті засоби для створення скелетної структури та ригування тривимірних персонажів. Цей інструмент дозволяє аніматорам та художникам зручно створювати складні скелетні системи для персонажів, що використовуються у відеоіграх, кіно, анімаційних фільмах та інших проектах. Загальний вигляд інтерфейсу Advanced Skeleton зображено на рисунку 2.3

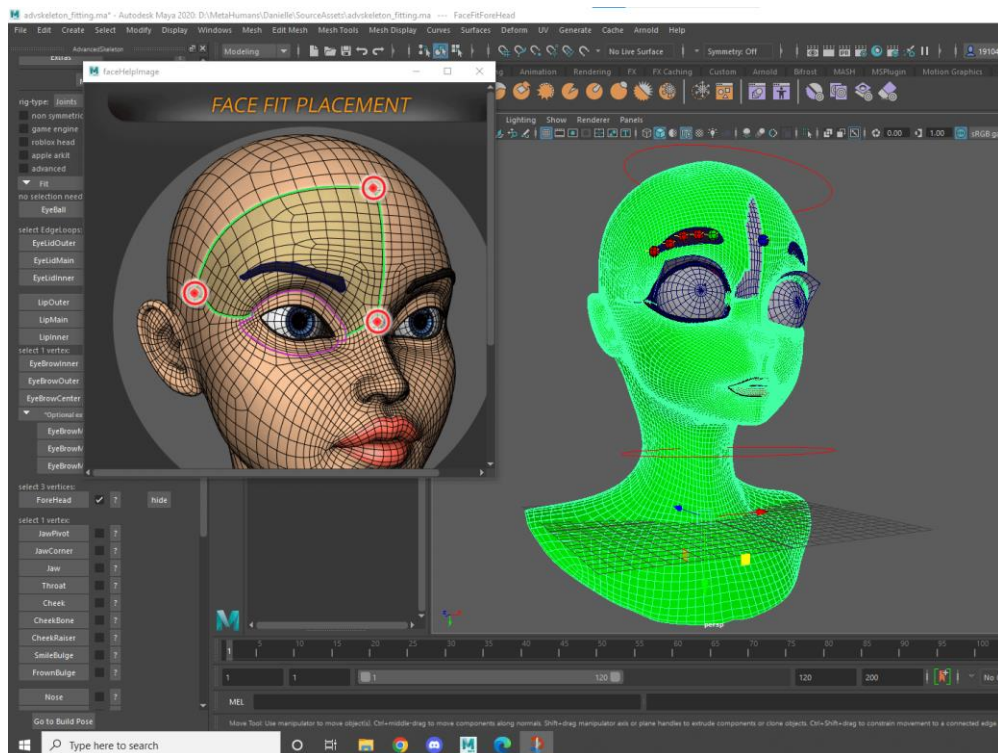


Рисунок 2.3 – інтерфейс Advanced Skeleton під час створення ригу обличчя

Основні функції Advanced Skeleton включають автоматизоване створення скелету, інтеграцію з іншими інструментами Мауа, можливість редагування та вдосконалення структури кісток, а також підтримку різних видів ригування, таких як IK/FK (inverse kinematics/forward kinematics).

Цей плагін дозволяє швидко та ефективно створювати готові до анімації персонажі з складною структурою кісток, що робить його популярним серед професіоналів у галузі тривимірної графіки.

Paint Skin Weights Tool [9] – це інструмент, який використовується в анімаційному редакторі Maya для редагування і визначення ваг кісток (Skin Weights) у тривимірних моделях персонажів. Цей інструмент дозволяє художникам і аніматорам точно налаштовувати ваги кожної кістки на поверхні моделі, що впливає на спосіб, як об'єкт реагує на рухи скелета під час анімації.

За допомогою Paint Skin Weights Tool можна визначити, яка частина поверхні моделі прикріплена до конкретної кістки, контролювати гнучкість та реалістичність анімації. Це особливо важливо при створенні реалістичних персонажів, де правильна настройка ваг кісток грає важливу роль у відтворенні природних рухів та деформацій.

Інструмент Paint Skin Weights Tool надає користувачеві кілька ключових можливостей:

**Визначення ваг:** За допомогою інструменту можна точно визначити вагу кожної кістки на поверхні моделі. Це дозволяє контролювати, яка частина об'єкта повинна реагувати на рухи конкретної кістки.

**Гнучкість анімації:** Налаштовуючи ваги кісток, аніматор може впливати на гнучкість та м'якість деформацій моделі під час руху. Це забезпечує природний вигляд анімованого об'єкта.

**Точні налаштування:** Інструмент дозволяє працювати з вагами на місцевому рівні, надаючи аніматорам точний контроль навіть над найменшими деталями.

**Можливість виправлення помилок:** Якщо під час анімації виникають артефакти чи неправильні деформації, Paint Skin Weights Tool дозволяє легко виправляти та оптимізувати ваги кісток для отримання більш досконалої анімації.



Робота в реальному часі: Інструмент забезпечує можливість редагування ваг кісток в реальному часі, що спрощує та прискорює процес анімації.

Paint Skin Weights Tool є необхідним інструментом для аніматорів та моделювальників, які працюють з тривимірною графікою у Maya. (рис. 2.3).

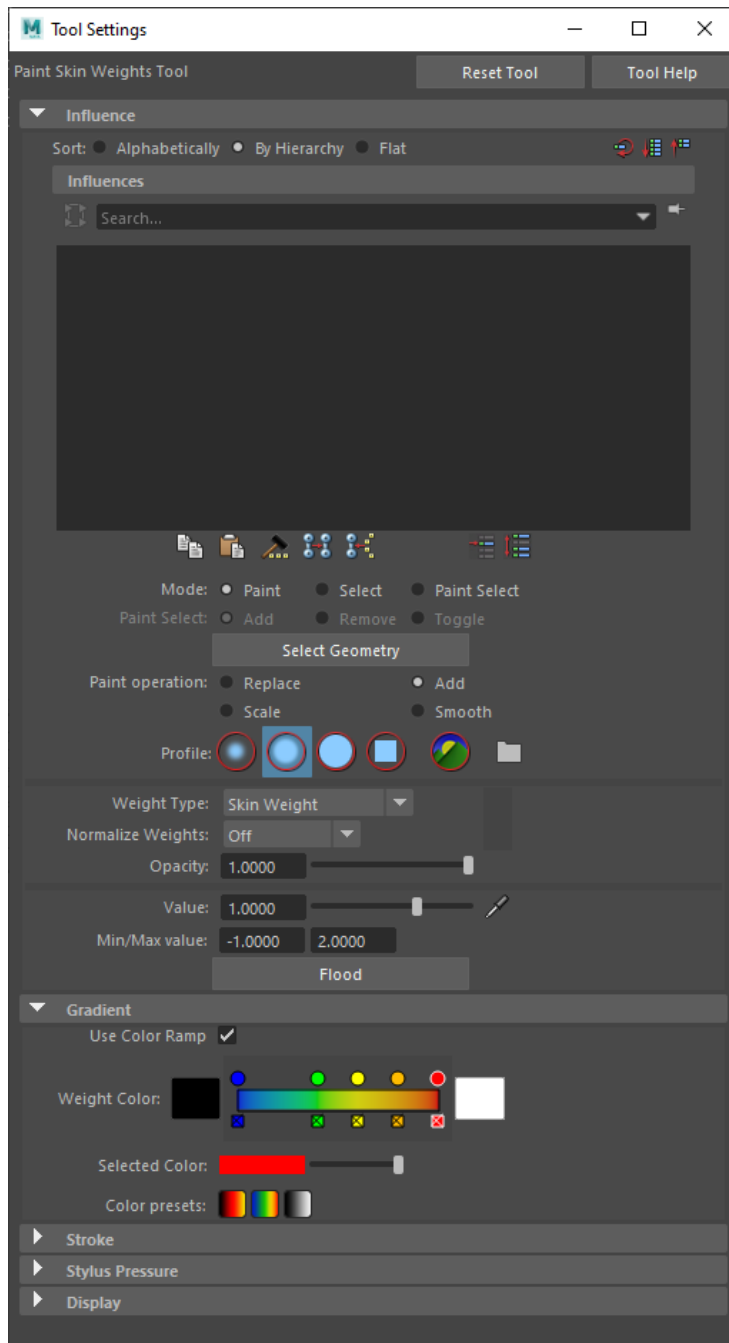


Рисунок 2.4 – Загальний вигляд інтерфейсу інструмента Paint Skin Weight Tool.

Blender [17] - це безкоштовний та потужний 3D редактор, який набув широкої популярності серед художників, дизайнерів та аніматорів. Давайте розглянемо його основні особливості та переваги для роботи з анімацією обличчя:

- Відкритий Код: Blender доступний для завантаження безкоштовно, що робить його доступним для широкого кола користувачів. Відкритий код дозволяє спільноті розробників вносити внески та вдосконалювати програму. На рисунку 2.5 зображено редактор коду Blender.

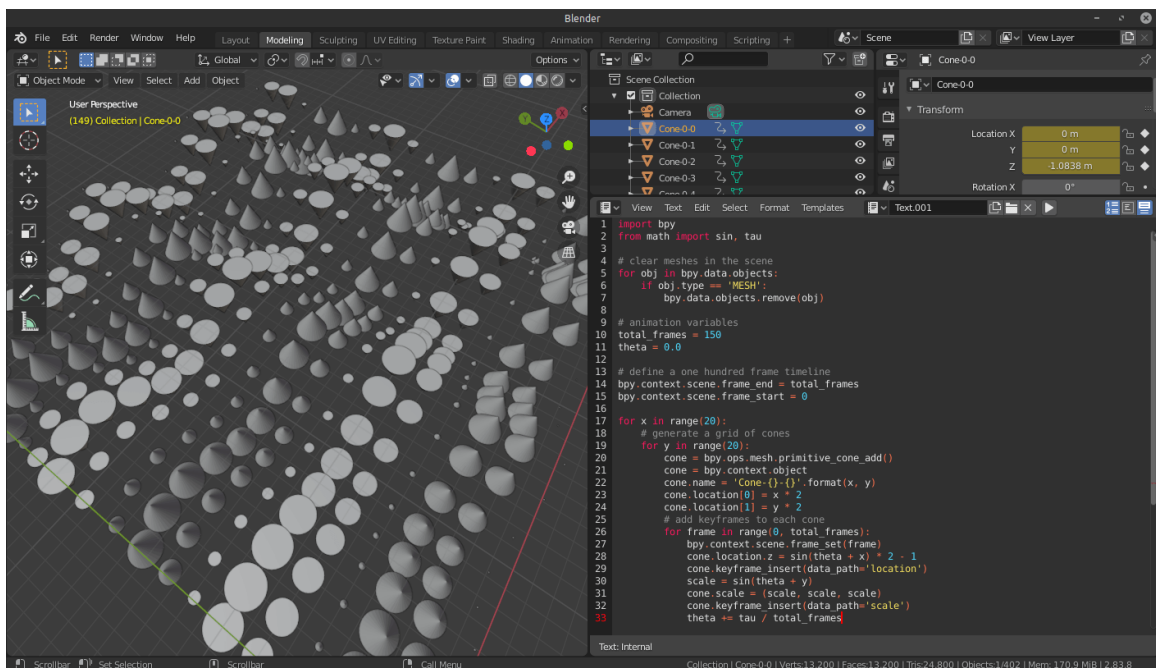


Рисунок 2.5 – Редактор коду Blender

- Моделювання та Анімація: Blender надає широкий функціонал для створення 3D-моделей та їх анімації. Інструменти для скелетної анімації, ріггінгу та текстурної роботи роблять його потужним інструментом для аніматорів. Blender пропонує різноманітні інструменти для 3D-моделювання, включаючи полігональне, NURBS- та скульптурне моделювання. Це надає артистам можливість виражати свою творчість та створювати складні 3D-об'єкти. Інтерфейс Blender під час анімації зображено на рисунку 2.6.

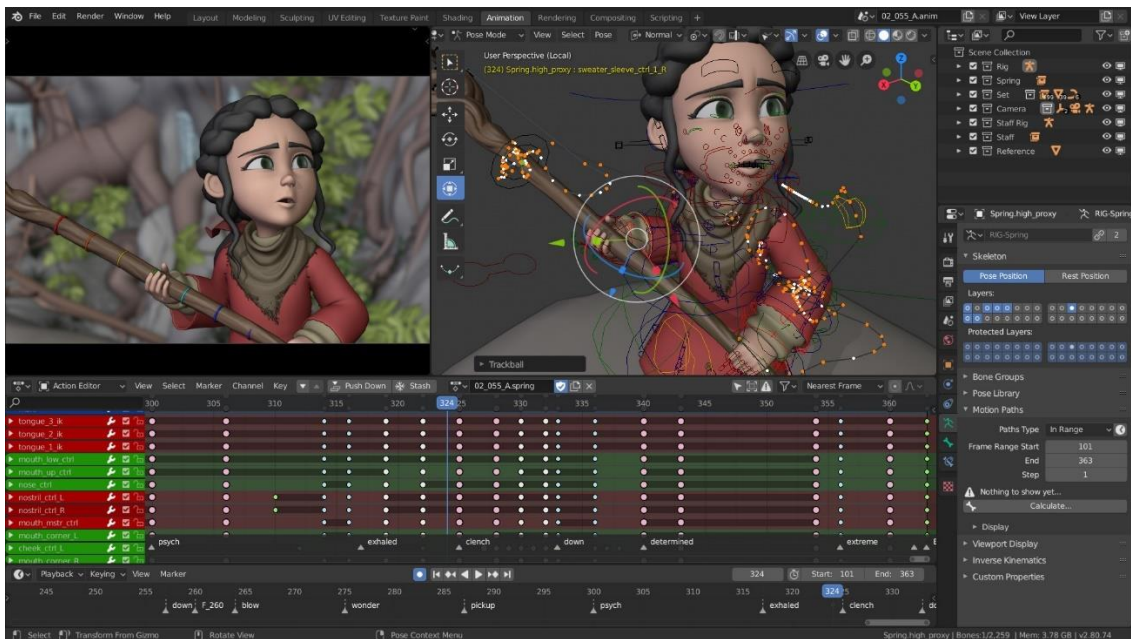


Рисунок 2.6 – Інтерфейс Blender під час анімації

- Розширені Інструменти для Ключових Кадрів: Blender має розширені засоби для створення ключових кадрів, що дозволяє детально контролювати анімаційні переходи та ефекти.

- Співпраця з Python: Blender підтримує мову програмування Python, що дозволяє користувачам створювати власні сценарії та автоматизувати процеси роботи.

- Широкі Можливості Рендерингу: Blender має різноманітні опції рендерингу. Особливо слід відзначити вбудовані дві системи рендерингу – Cycles та Eevee. Cycles забезпечує фотореалістичний рендеринг, дозволяючи створювати вражаючі візуальні ефекти з високою якістю. З іншого боку, Eevee дозволяє швидку попередню візуалізацію в реальному часі, полегшуючи процес роботи та прискорюючи вирішення завдань. Ця різноманітність рендерингу важлива для тих, хто шукає не лише високоякісні візуальні ефекти, але й ефективний та зручний процес роботи з анімацією та візуалізацією. Результати рендерингу Blender зображені на рисунку 2.7.



Рисунок 2.7 – Результат рендерінгу Blender

Звернемо увагу на деякі недоліки Blender порівняно з Maya для анімації обличчя:

- Менший професійний спектр використання: В області VFX та ігрової індустрії Maya частіше вважається стандартом через свою широкую популярність серед професіоналів.
- Система навігації: Для тих, хто привчений працювати в Maya, перехід на Blender може вимагати періоду адаптації, особливо стосовно системи навігації та інтерфейсу.
- Відсутність деяких пропрієтарних інструментів: Blender може виявитися менш обладнаним в ряді пропрієтарних інструментів та плагінів, які можуть бути доступні в Maya.
- Менший спектр підтримки та навчання: У відповідь на популярність Maya у навчальних закладах та онлайн-курсах, участь у спільноті та доступ до ресурсів для навчання можуть бути меншими для Blender.

Autodesk 3ds Max [18] - це інший потужний 3D редактор, який широко використовується у візуальному мистецтві, геймдевідженменті та архітектурному візуалізації. Розглянемо його основні переваги та

особливості для створення анімації обличчя:

- Моделювання та Анімація: 3ds Max пропонує широкий набір інструментів для створення 3D-моделей та їх анімації. Він підтримує скелетну анімацію, рігінг та текстурну роботу. Інтерфейс 3Ds Max під час анімації зображено на рисунку 2.8.

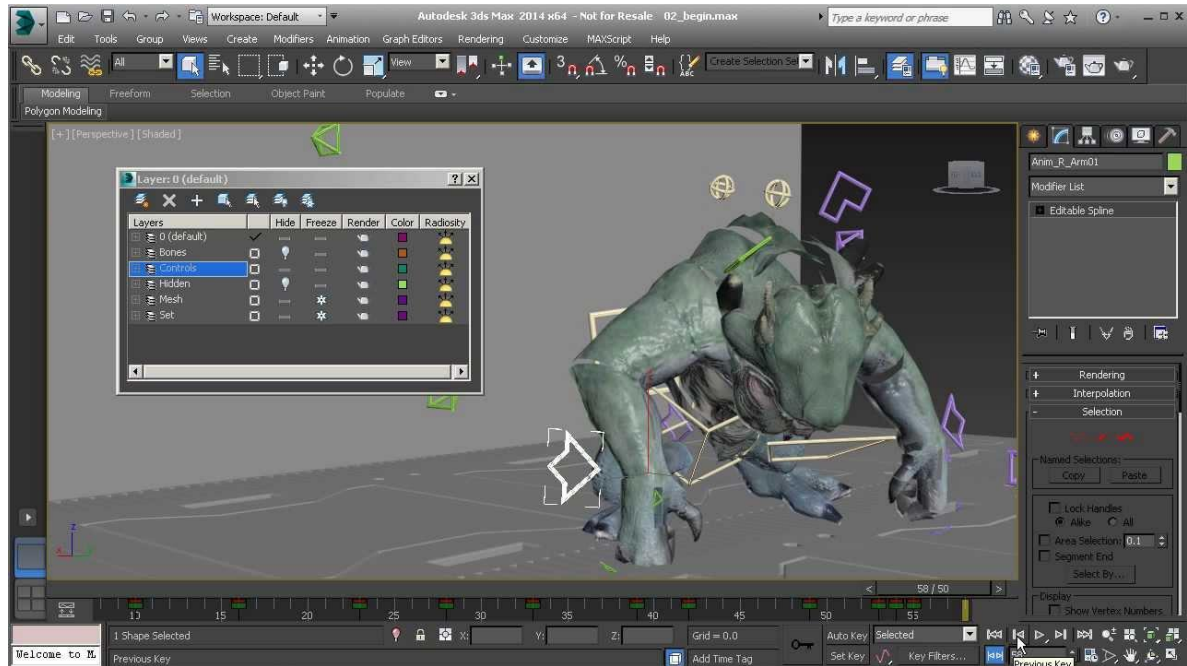


Рисунок 2.8 – Інтерфейс 3Ds Max під час анімації

- Можливості Рендерингу: 3ds Max має вбудовані засоби рендерингу, такі як Arnold. Arnold - це потужний рендерер, використовуваний для створення вражаючих візуальних ефектів та реалістичних зображень у світі 3D-моделювання та анімації. Його фізичний рендеринг враховує фізичні властивості світла, матеріалів і середовища, надаючи зображенням максимальний ступінь реалізму. Arnold володіє потужними засобами глобального освітлення, відтворює розповсюдження світла та вплив на об'єкти. Засоби для роботи з матеріалами та текстурами дозволяють досягати різноманітних ефектів. Його інтеграція з популярними програмами для 3D-моделювання, такими як Maya та 3ds Max, робить його універсальним інструментом у світі CGI. Arnold оптимізований для великих сцен, забезпечуючи високу якість рендерингу у фільмах. (рис. 2.9).



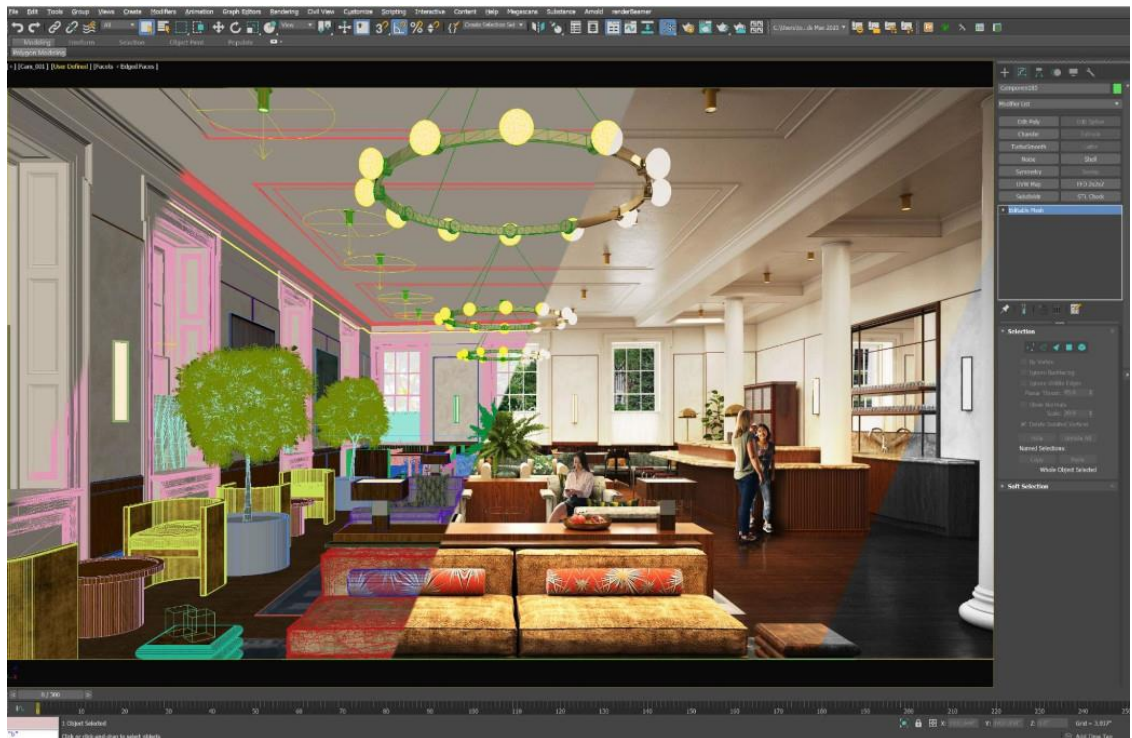


Рисунок 2.9 – Результат рендерінгу 3Ds Max

- Інтеграція з Іншими Продуктами Autodesk: Як частина сімейства продуктів Autodesk, 3ds Max має зручну інтеграцію з іншими програмами, такими як AutoCAD та Revit.

- Скриптовання та Розширення: 3ds Max підтримує мову програмування MAXScript, що дозволяє створювати власні інструменти та автоматизувати завдання. Редактор коду зображено на рисунку 2.10

```

1  Demonstrates simple animation.
2
3  from pymxs import runtime as rt # pylint: disable=import-error
4  import pymxs as mx # pylint: disable=import-error
5
6  print("Hello World Animation")
7
8  def print_interval(interval):
9      """Prints an animation interval"""
10     print("Current Animation Range: {}".format(interval.start, interval.end))
11
12 def set_animation_ranges():
13     """Changes the animation range from the default of 100 frames to 200 frames"""
14     print_interval(rt.animationRange)
15     rt.animationRange = rt.Interval(0, frames)
16     # The animation slider now shows 200 frames
17     print_interval(rt.animationRange)
18
19 def animate_transform(thing):
20     """Records an animation on the provided object"""
21     # select the object to animate so we will see the keyframes in the timeline
22     rt.select(thing)
23
24     # animate
25     with mx.animate(True):
26         with mx.redraw(True):
27             with mx.attime(0):
28                 thing.Pos = rt.Point3(50, 0, 0)
29             with mx.attime(50):
30                 thing.Pos = rt.Point3(100, 50, 0)
31             with mx.attime(100):
32                 thing.Pos = rt.Point3(50, 100, 0)
33             with mx.attime(150):
34                 thing.Pos = rt.Point3(-50, 50, 0)
35             with mx.attime(200):
36                 thing.Pos = rt.Point3(0, 0, 0)
37
38 def playback_animation():
39     """Play back the animation 3 times"""
40     rt.playbackloop = False
41     # play animation
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000

```

Рисунок 2.10 – Редактор коду 3Ds MAX

- Багатофункціональний Інтерфейс: Інтерфейс 3ds Max призначений для зручності користувача, надаючи доступ до інструментів та панелей у зручному форматі.

Тепер перейдемо до деяких недоліків 3ds Max у порівнянні з Maya для створення анімації обличчя:

- Обмежена Універсальність Анімації: У порівнянні з Maya, 3ds Max може мати обмежені можливості для роботи з анімацією обличчя, зокрема у сфері складної лицьової анімації.

- Вартість Ліцензій: Ліцензії на Autodesk 3ds Max можуть бути варті грошей, зокрема для незалежних артистів та початківців.

- Підтримка Тривіальна в Онлайн-Спільноті: Онлайн-спільнота для 3ds Max може бути менш активною порівняно з Maya, що може призвести до складнощів у вирішенні проблем та отриманні порад.

- Менше Розвинених Інструментів для Анімації Обличчя: Може бути менше розвинених інструментів для роботи з анімацією обличчя порівняно зі знаменитою системою лицьової анімації у Maya.

- Із цими відомостями можна краще зрозуміти, наскільки 3ds Max може відповісти на конкретні потреби у створенні анімації обличчя порівняно з іншими редакторами.

Після ретельного порівняння різних 3D редакторів для створення анімації обличчя, найкращим вибором стала Autodesk Maya. Вона вирізняється деталізованими та розвинутими інструментами для роботи з лицьовою анімацією, широким спектром можливостей у сфері ріггінгу, скелетної анімації та візуалізації. Інтеграція із спеціалізованими інструментами, такими як Advanced Skeleton, та активна спільнота користувачів створюють оптимальне середовище для високоякісної та ефективної роботи. Вибір Maya дозволяє впевнено взятися за завдання створення анімації обличчя та досягти бажаних результатів у світі тривимірної графіки та анімації.



## 2.2 Розробка методу розбиття слова на фонетичні звуки

Фонема [19] – це найменша одиниця звуку в мові, яка відрізняє одне слово від іншого. Це абстрактна мовна одиниця, яка несе значення та може впливати на семантику слова чи виразу. Фонемати визначаються звуковими характеристиками, такими як тон, довжина та інтенсивність звуку.

У фонології фонемати вважаються ключовим елементом для вивчення звукової структури мови. Вони грають важливу роль у визначенні фонетичних та фонематичних відмінностей між словами та вимовою звуків у мовленні.

Знання фонемати допомагає зрозуміти мовленнєві моделі мови і впливає на правильну вимову та сприйняття мови: при спробі вимовити звук /k/ у словах *cat* /kat/ або *fact* /fakt/, звук /k/ насправді не збігається для багатьох носіїв англійської мови, але вони не розпізнають їх як різні звуки. Загалом, звук /k/ у слові "кішка" передбачає вдих повітря і може передаватися як [kh], але не як [k]. Як саме вимовляти, залежить від фонологічного оточення звуку /k/. Наприклад, [khat] не протиставляється [kat], оскільки [khat] не сприймається як два окремих звуки в англійській мові.

Щоб розпізнати, які фонемати використовуються у словах, лінгвісти створили словники. Одним з найбільших словників, що відображає фонемати у словах, є словник Карнегі-Меллона.

Словник Карнегі-Меллон для вимови (CMU Pronouncing Dictionary) [19] - це відомий англійський словник, який містить фонетичні транскрипції для багатьох слів англійської мови. Цей словник розроблений університетом Карнегі-Меллон і часто використовується в області оброблення природної мови та комп'ютерної лінгвістики.

Кожне слово в цьому словнику супроводжується транскрипцією фонетичного вимовлення, що дозволяє програмам та системам розпізнавання мови правильно вимовляти слова. Це важливий інструмент для розробників голосових технологій, текстово-голосових систем та інших додатків,

пов'язаних з обробкою мовлення.

CMUdict (CMU Pronouncing Dictionary) надає фонетичні транскрипції для багатьох слів англійської мови. Основна мета словника полягає в тому, щоб забезпечити комп'ютерам та програмам інформацію про те, як правильно вимовляти слова.

Основні особливості CMUdict:

1) Фонетичні транскрипції: Для кожного слова у словнику надається транскрипція, яка вказує, як слід вимовляти це слово.

2) Використання в технологіях голосового розпізнавання: CMUdict часто використовується в галузі голосового розпізнавання, де точність вимовлення слів є критично важливою.

3) Дослідження в області оброблення природної мови: Словник використовується в різних дослідницьких проектах та додатках, пов'язаних з обробкою природної мови.

4) Підтримка мовних технологій: Словник допомагає у покращенні якості мовних технологій, таких як синтез та розпізнавання мови. Забезпечення точної фонетичної інформації дозволяє створювати більш ефективні та точні системи.

5) Аналіз мовлення: CMUdict використовується для аналізу та дослідження особливостей мовлення. Вивчення вимови слів та взаємозв'язку між фонетичною структурою та значенням слів.

CMUdict став важливим ресурсом для розвитку технологій, пов'язаних з обробкою мовлення, і його використання сприяє розвитку голосових та текстово-голосових.

Останній випуск – 0,7b; він містить понад 134 000 записів.

Ця база даних відіграє значущу роль у різних галузях, сприяючи розвитку технологій та полегшуючи вивчення та використання англійської мови. Розширення його застосування в сучасних технологіях робить CMUdict невід'ємною частиною інструментарію у галузі обробки мовлення та комп'ютерної лінгвістики.

Метод розбиття слова на фонетичні звуки виконує процес аналізу слова для визначення його фонетичної транскрипції. Основні кроки цього методу включають:

1. Нормалізація слова: Слово має бути переведене до нижнього регістру або виконана інша попередня обробка для уніфікації даних.

2. Перевірка наявності фонетичного запису: Перевіряється, чи вже існує відомий фонетичний запис слова в словнику. Дерево перевірки фонетичного запису словнику Карнегі-Меллон зображено на 2.11.

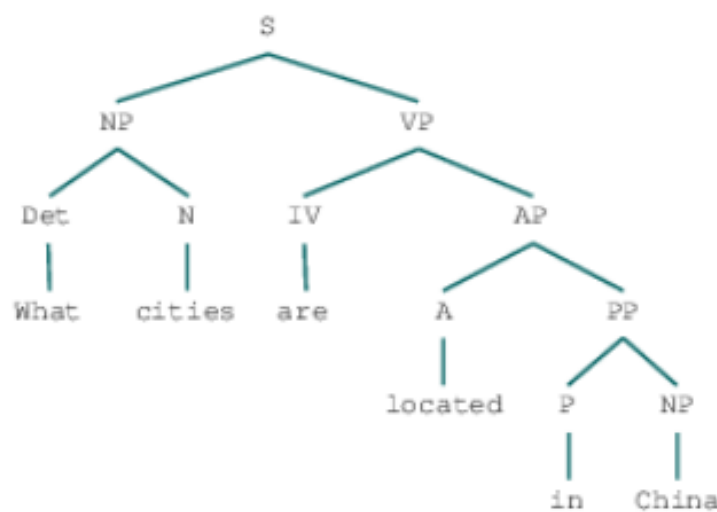


Рисунок 2.11 – Дерево перевірки фонетичного запису словнику Карнегі-Меллон

3. Розбиття на частини: Слово розбивається на частини, такі як префікс і суфікс, для подальшого аналізу.

4. Рекурсивний аналіз: Виконується аналіз кожної частини слова з використанням рекурсії для подальшого розбиття.

5. Формування фонетичних комбінацій: Об'єднання фонетичних записів частин слова для утворення всіх можливих комбінацій.

6. Повернення результату: Повертається список фонетичних транскрипцій або спеціальний маркер, якщо розбиття неможливе.

Цей метод грає важливу роль у визначенні того, як обличчя реагує на вимовлені слова, сприяючи створенню реалістичної анімації мовлення.

### 2.3 Розробка математичної моделі анімаційного зображення

Тривимірний вектор - це математичний об'єкт, який характеризується трьома числовими значеннями і представляє собою точку або напрямок у тривимірному просторі. Кожна з цих числових компонент вектора відповідає одній з трьох координат (X, Y, Z) в просторі.

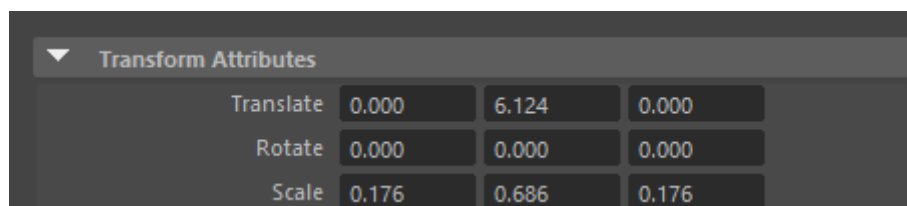
Такі вектори широко використовуються в геометрії, фізиці, комп'ютерній графіці та інших областях для опису положення об'єктів, напрямку руху, сили та інших параметрів, пов'язаних з тривимірним простором.

Наприклад, тривимірний вектор може бути визначений як  $v = \langle x, y, z \rangle$ , де  $x, y, z$  - це числові компоненти вектора:

$$\begin{pmatrix} x_u \\ y_u \\ z_u \end{pmatrix} + \begin{pmatrix} x_v \\ y_v \\ z_v \end{pmatrix} = \begin{pmatrix} x_u + x_v \\ y_u + y_v \\ z_u + z_v \end{pmatrix} \quad (2.1)$$

$$\alpha \begin{pmatrix} x_u \\ y_u \\ z_u \end{pmatrix} = \begin{pmatrix} \alpha x_u \\ \alpha y_u \\ \alpha z_u \end{pmatrix}.$$

Кістки анімації - це 3D-вектори. Окрім атрибутів координат, існують атрибути масштабування та обертання, які впливають на обробку анімованого зображення (рис. 2.12).



Transform Attributes			
Translate	0.000	6.124	0.000
Rotate	0.000	0.000	0.000
Scale	0.176	0.686	0.176

Рисунок 2.12 – Атрибути анімаційної кістки

Різні форми вуст можна створити, змінюючи координати кісток. Існує 15 різних форм рота. (рис. 2.13).



Рисунок 2.13 – П'ятнадцять видів форм вуст анімаційного персонажа

Ключовий кадр (або кадр анімації) - це спеціальний кадр в послідовності анімації, який визначає ключові параметри об'єкта чи персонажа в конкретний момент часу. В анімаційній роботі ключові кадри використовуються для визначення позиції, орієнтації, масштабу та інших важливих характеристик об'єкта.

Основні риси ключового кадру:

1) Визначення параметрів: Кожен ключовий кадр містить інформацію про значення певних параметрів (наприклад, координати, обертання, розмір) об'єкта в конкретний момент часу.

2) Опорний пункт: Це базовий кадр, відносно якого визначаються зміни у подальших кадрах, формуючи плавний перехід між ними.

3) Використання в програмах для анімації: У програмах для створення анімації ключові кадри використовуються для встановлення важливих моментів у русі об'єктів чи персонажів.

4) Послідовність: Ключові кадри утворюють послідовність, яка задає рух чи зміну стану об'єкта з часом.

Використання ключових кадрів дозволяє аніматорам та розробникам керувати та налаштовувати анімаційні об'єкти, роблячи процес створення анімації більш ефективним та гнучким.

5) Плавні переходи між кадрами: Однією з ключових переваг використання ключових кадрів є можливість створення плавних переходів між ними. За допомогою інтерполяції між значеннями параметрів на ключових кадрах аніматор може забезпечити природний та плавний рух об'єкта.

Використання ключових кадрів в анімаційному процесі спрощує роботу та робить його більш інтуїтивно зрозумілою, надаючи аніматорам зручний інструментарій для створення вражаючих та динамічних анімаційних сцен.

Time Slider - це інтерфейсний елемент у програмах для роботи з анімацією, який дозволяє користувачеві візуально контролювати та маніпулювати часовою шкалою в анімаційному проєкті. Цей інструмент дозволяє переміщатися по часовій шкалі, переглядати та редагувати ключові кадри, а також визначати тривалість та темп анімації.

Основні характеристики Time Slider:

1) Переміщення в часі: Користувач може використовувати Time Slider для переміщення по анімаційній послідовності та швидкого перегляду різних частин проєкту.

2) Маркер часу: Вказує поточний момент часу в анімації, де користувач може розміщати та редагувати ключові кадри.

3) Редагування ключових кадрів: Ключові кадри можна додавати, видаляти або редагувати безпосередньо за допомогою Time Slider.

Time Slider є важливим елементом для аніматорів та розробників, які працюють з анімаційними проєктами, оскільки він спрощує керування часовими параметрами та полегшує вирішення завдань, пов'язаних з часовою шкалою в анімації (рис. 2.14).

Аніматори використовують Time Slider для плавного переміщення по анімаційному проєкту, знаходження ключових моментів та оптимізації тимчасових параметрів. Цей інструмент робить процес створення анімації більш інтуїтивним та легким для керування, дозволяючи аніматорам точно

налаштовувати рухи та динаміку об'єктів.

Загалом, Time Slider виступає як невід'ємна частина інтерфейсу для роботи з анімацією в Autodesk Maya, надаючи користувачам засіб для точного керування часом та ключовими параметрами їхніх анімаційних проєктів.

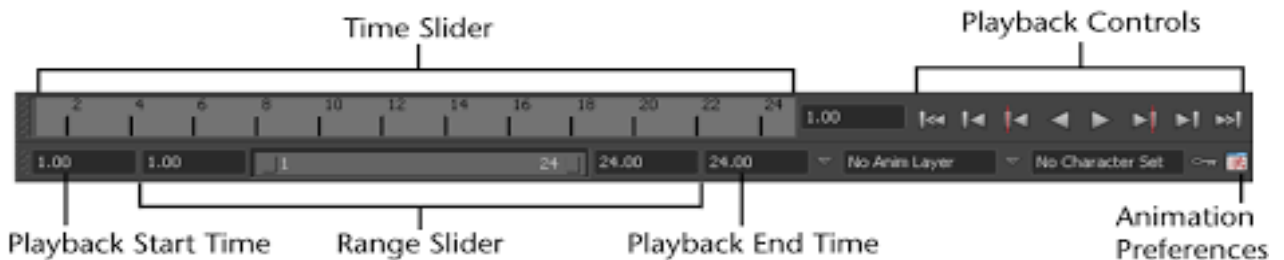


Рисунок 2.14 – Загальний вигляд інтерфейсу Time Slider

Формули використовуються, щоб визначити, коли замінити анімацію форми вуст. Ці формули відрізняються для приголосних і голосних, оскільки голосні зникають повільніше, між приголосні:

$$C_x = C_{x-1} + P \cdot W + N, \quad (2.2)$$

де  $C_x$  – кадр анімації в якій буде ведено форму вуст;  $C_{x-1}$  – останній кадр анімації в який було введено форму вуст;  $P$  – знак пунктуації. Якщо перед словом був знак «. ;» то значення  $P=3$ . Якщо знаку не було то  $P=1$ ;  $W$  – вказує чи є буква початком нового слова. Якщо так то  $W=2$ , якщо ні то  $0$ ;  $N$  – чи фонетичний звук приголосною, якщо так то  $N=3$ , а якщо ні то  $N=5$ .

Для визначення сили відкриття щелепи анімації форми вуст використовуються формула. Вона встановлює силу відкриття ротової порожнини залежно від голосного чи приголосного фонетичного звуку. Дана формула приставлена для приголосних фонетичних звуків:

$$S_x = \frac{S_{x-1}}{E}, \quad (2.3)$$

де  $S_x$  – сила з якою буде відкриватися щелепа;  $S_{x-1}$  – сила з якою відкривалася минала щелепа;  $E$  – індекс зниження сили відкриття щелепи

Для голосних літер  $S_x$  завжди дорівнює максимальному значінню відкриття щелепи у розмові.

## 2.4 Розробка структурної схеми функціонування програмного засобу

Структурна схема — це графічне зображення системи чи процесу, що використовується для візуального представлення його структури та взаємозв'язків між складовими елементами.

Структурна схема функцій програмного засобу відображає призначення і взаємодію основних структурних елементів програми та їх взаємодію з користувачем.

Програмні модулі Visual Ternary Ranking включають модуль для розбиття слів на частини мови, модуль для перетворення мови у форми рота, модуль для призначення форм рота певним кадрам анімації та модуль вибору файлів. Входом до програми є головне меню, з якого можна перейти до інших модулів програми.

Розроблена загальна структурна схема функціонування програмного модулю оброблення полігональних моделей анімаційних зображень зображено на рисунку 2.15.

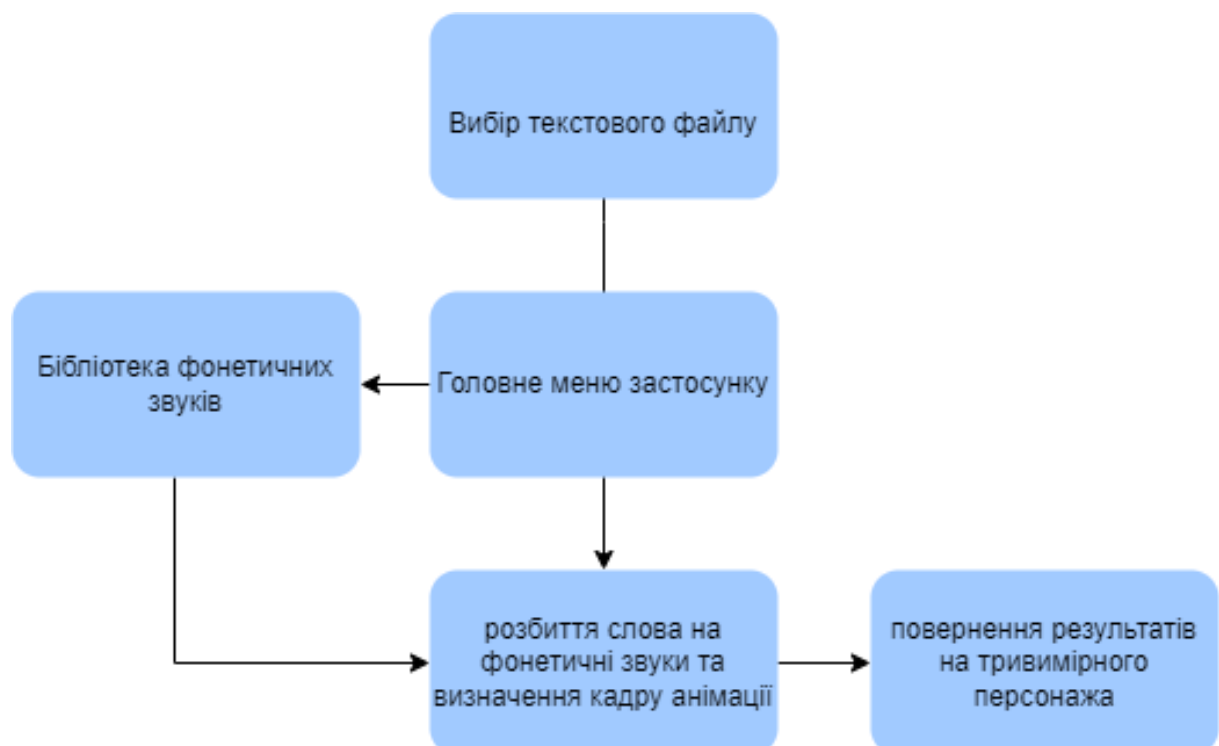


Рисунок 2.15 – Структурна схема функціонування програмного модулю



Ця діаграма відтворює функціональність програмного модулю для роботи з анімацією обличчя в середовищі Maya на найвищому рівні абстракції. Одночасно важливо приділити увагу аспектам функціонування операційної системи Windows у відношенні до життєвого циклу застосунків.

Діаграма прецедентів є інструментом визначення та візуалізації можливостей системи та взаємодії з її користувачами. У цій діаграмі прецеденти, які представляють собою конкретні функціональні можливості системи, відображаються разом із взаємодією акторів — учасників цих можливостей.

Діаграма прецедентів, що зображена на рисунку 2.16, показує взаємодію користувача з за стосунком, його можливі дії та реакції системи на них.

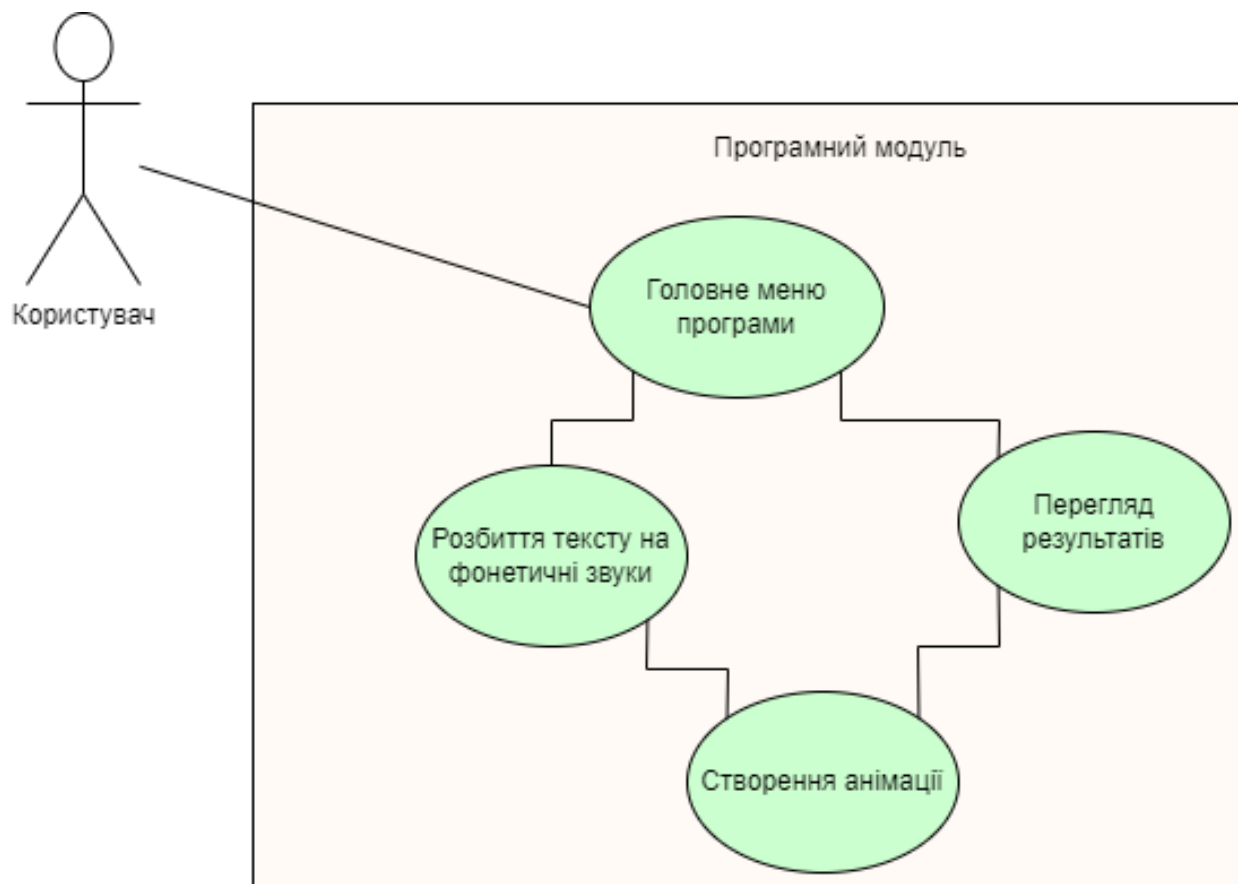


Рисунок 2.16 – Діаграма прецедентів програмного модулю

Для розбиття слів на фонемі було використано тридцять дев'ять фонем. У таблиці 2.1 наведено всі фонемі та їхні значення.

Таблиця 2.1 – Перелік фонем

№ з/п	Фонема	Приклад	Переклад	№ з/п	Фонема	Приклад	Переклад
1	AA	odd	AA D	2	AE	at	AE T
3	AH	hut	HH AH T	4	AO	ought	AO T
5	AW	cow	K AW	6	AY	hide	HH AY D
7	B	be	B IY	8	CH	cheese	CH IY Z
9	D	dee	D IY	10	DH	thee	DH IY
11	EH	Ed	EH D	12	ER	hurt	HH ER T
13	EY	ate	EY T	14	F	fee	F IY
15	G	green	G R IY N	16	HH	he	HH IY
17	IH	it	IH T	18	IY	eat	IY T
19	JH	gee	JH IY	20	K	key	K IY
21	L	lee	L IY	22	M	me	M IY
23	N	knee	N IY	24	NG	ping	P IH NG
25	OW	oat	OW T	26	OY	toy	T OY
27	P	pee	P IY	28	R	read	R IY D
29	S	sea	S IY	30	SH	she	SH IY
31	T	tea	T IY	32	TH	theta	THEY T AH
33	UH	hood	HH UH D	34	UW	two	T UW
35	V	vee	V IY	36	W	we	W IY
37	Y	yield	Y IY L D	38	Z	zee	Z IY
29	ZH	seizure	S IY ZH ER				

## **2.5 Розробка інтерактивних анімацій обличчя для програмного засобу оброблення та синхронізації контенту з елементами візуалізації**

Для анімацій обличчя потрібно використовувати не лише ліпсінг губами. Також для природньої анімації потрібно щоб обличчя кліпало та видавало емоції. Для створення максимально природної та реалістичної анімації, досліджується широкий спектр рухів та емоцій, які виникають в реальному обличчі. Це включає в себе рух бров, очей, кліпання, міміку, рухи відділень обличчя та інші аспекти.

Крім того, для досягнення ще більшої реалістичності може використовуватися технологія захоплення руху, яка дозволяє точно відтворити рухи обличчя реальної людини у віртуальному середовищі. Це може бути корисно в контексті відеоігор, віртуальної реальності та інших областей, де реалістична анімація є ключовим елементом.

За допомогою спеціальних конфігурацій, користувачі матимуть можливість точно налаштовувати параметри анімації обличчя для досягнення конкретних емоцій. Ці конфігурації будуть включати параметри, такі як інтенсивність кліпання губ, рухи брів, і вираз очей.

Наприклад, для вираження щастя можна налаштовувати більш інтенсивне та швидке кліпання губ, підняття брів і радісний вираз очей. У той час, для вираження суму можна регулювати повільніше та менш інтенсивне кліпання губ, опущення брів і вираз сумних очей.

Це дозволяє користувачам створювати різноманітні обличчя, відповідні різним емоційним станам, що робить програмний модуль більш гнучким та придатним для різноманітних сценаріїв використання в індустрії тривимірної анімації.

Програмний модуль буде надавати користувачам 3 ключові налаштування, які впливають на реалістичність та емоційність анімації обличчя.

Частота кліпання очей: Користувачі матимуть можливість регулювати частоту та швидкість кліпання очей, що дозволить створювати виразні ефекти в залежності від ситуації. Наприклад, швидке та інтенсивне кліпання може вказувати на захоплення чи враження.

Кліпання очей відбувається коли кістки біля віка переходять таку форму щоб віко закривало око як це продемонстровано на рисунку 2.17.

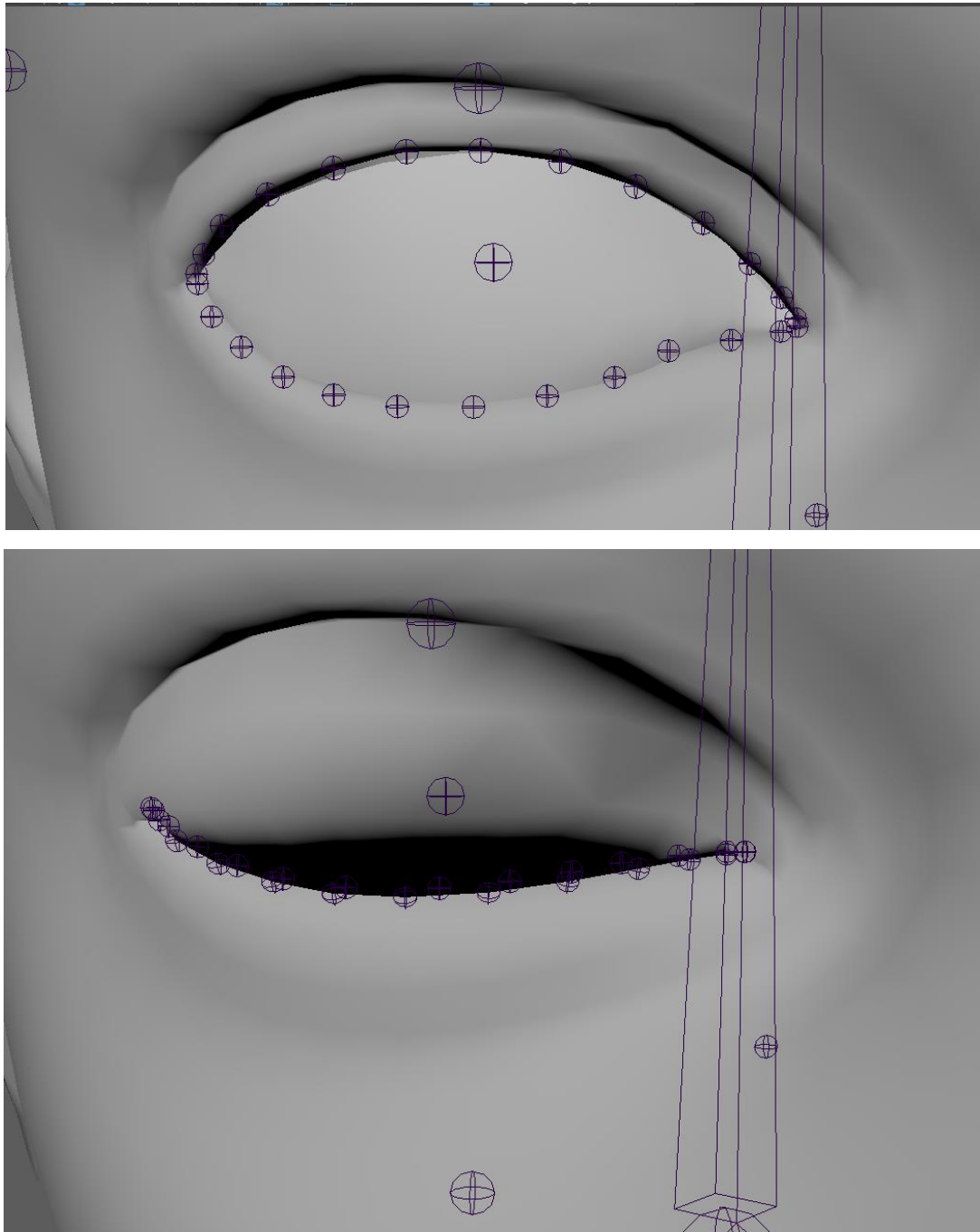


Рисунок 2.17 – Момент кліпання очей персонажа

Емоція: Конфігурації для різних емоцій будуть включати параметри, такі як рухи губ, підняття чи опущення брів, вирази очей та інші аспекти. Користувачі зможуть обирати з попередньо визначених емоцій чи створювати власні комбінації.

Разом із тим, програмний засіб передбачає використання спеціального ползунка для емоцій, що дозволить плавно переміщатись від сумного настрою до веселого.

Користувачі зможуть інтуїтивно налаштовувати цей параметр, визначаючи ступінь емоційності анімації від виразно сумного до яскраво веселого. Такий підхід забезпечить більш гнучкий та індивідуальний підхід до створення різноманітних сценаріїв використання обличчя в тривимірній анімації.

Для створення емоції потрібно перемістити кістки вуст так щоб сформувати форму потрібну для видворення реалістичного виразу обличчя, як це зображено на рисунку 2.18.

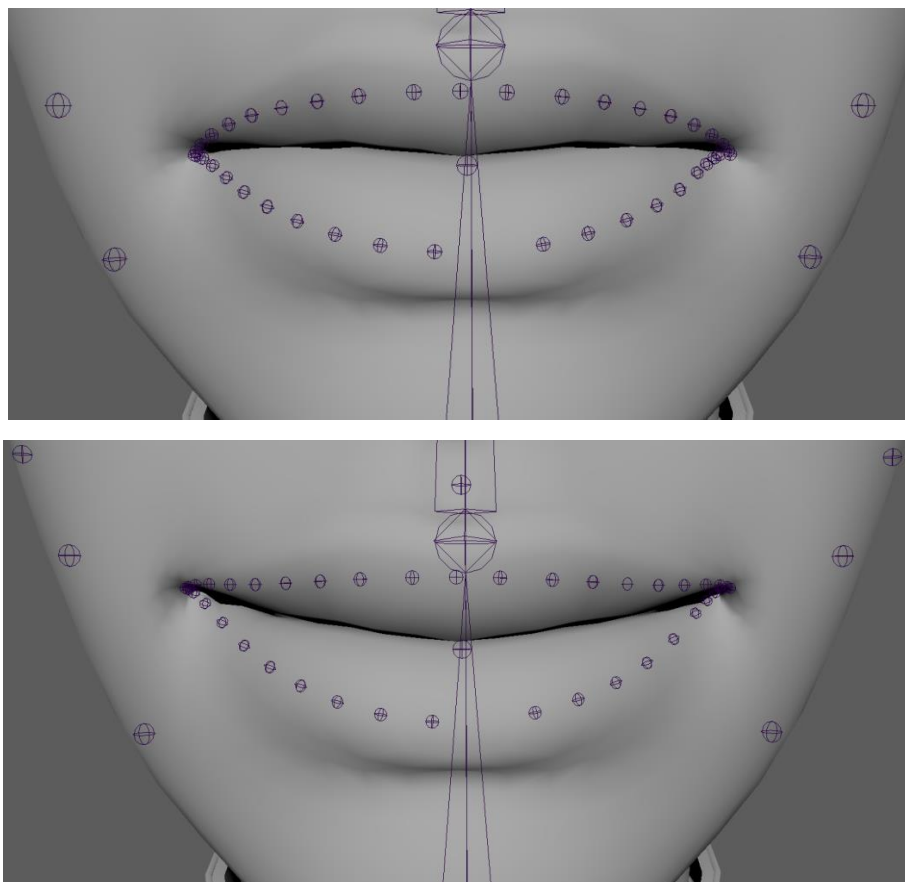


Рисунок 2.18 – Сумна емоція на обличчі персонажа

Рух голови: Це налаштування дозволить контролювати рухи голови тривимірного персонажа. Відхилення та нахил голови можуть бути використані для підсилення емоцій та виразності анімації.

Використовуйте ключові кадри для визначення потрібно рухати голову від одного положення до іншого. Рухайтеся голова за допомогою кісти Head\_M так, щоб передати реалістичні рухи голови під час говоріння, як це зображено на рисунку 2.19.

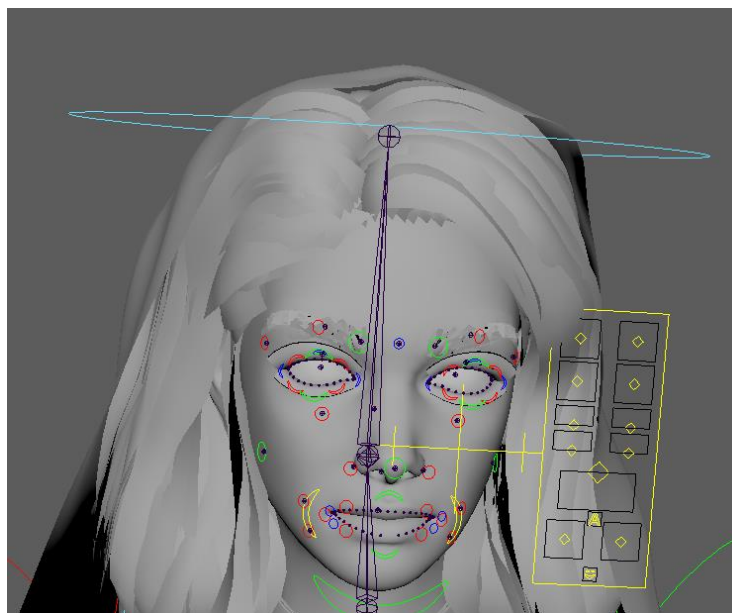
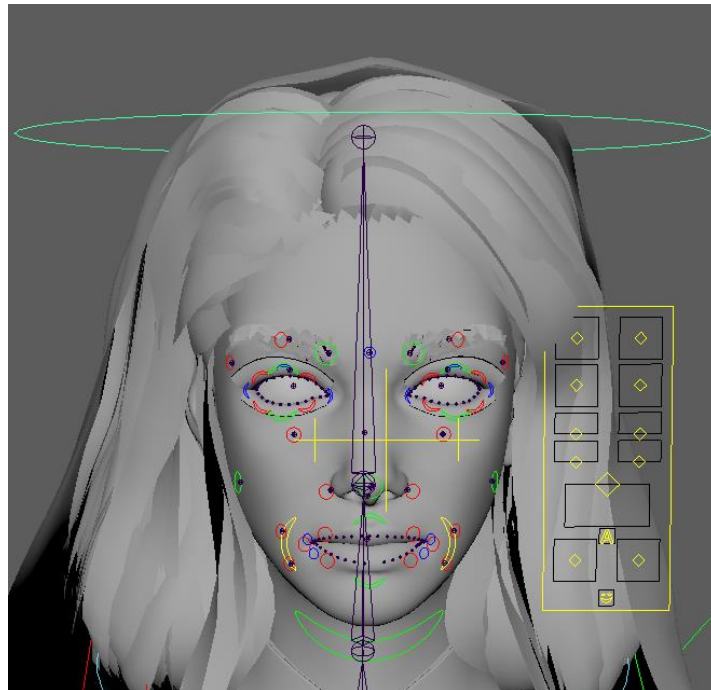


Рисунок 2.19 – Анімація нахилу голови персонажа

## 2.6 Розробка алгоритму функціонування програмного засобу оброблення та синхронізації контенту з елементами візуалізації

Алгоритм реалізації підходу до обробки полігональних моделей анімованих зображень (рис. 2.9):

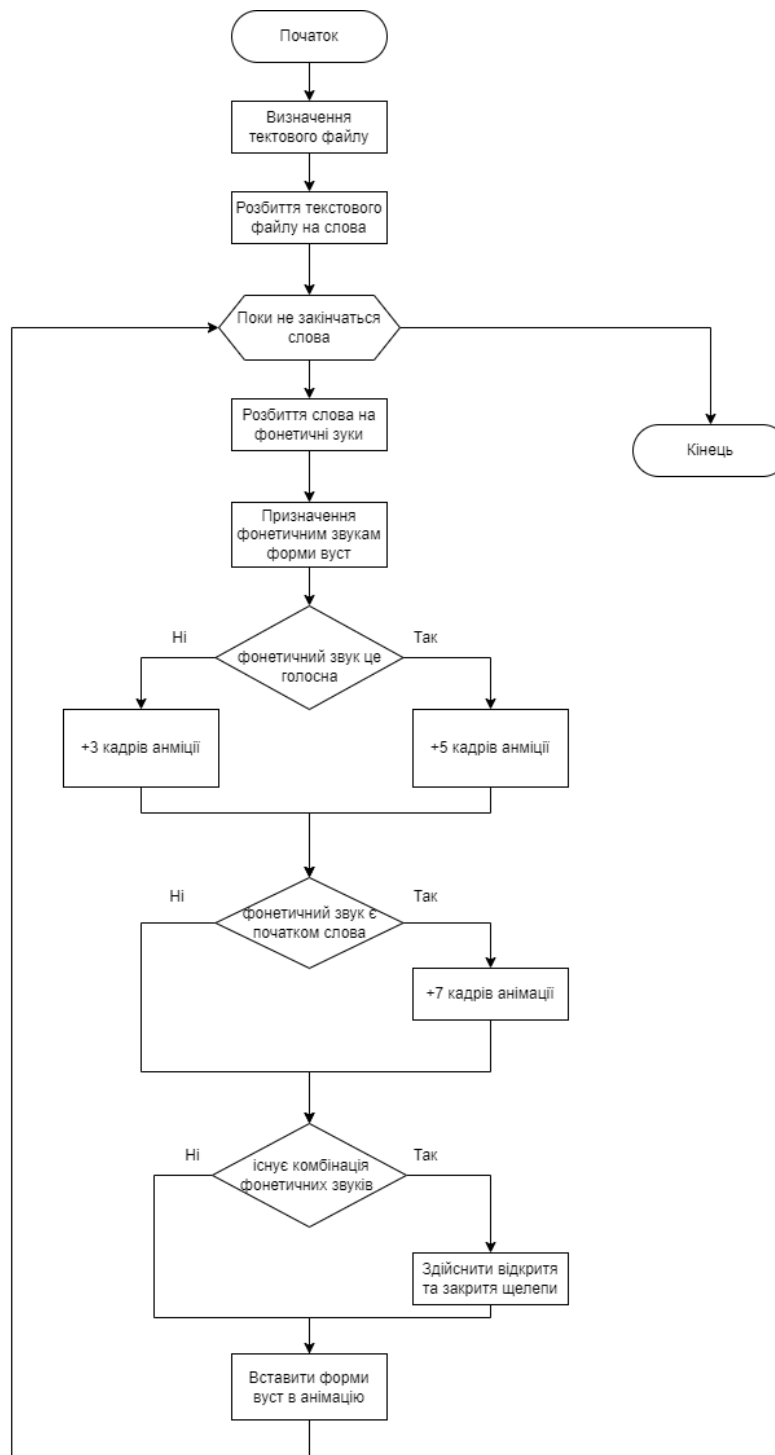


Рисунок 2.20 – Алгоритм функціонування програмного засобу оброблення та синхронізації контенту з елементами візуалізації



1. Початок алгоритму функціонування програмного засобу оброблення та синхронізації контенту з елементами візуалізації;
2. Визначення текстового файлу користувача;
3. Розбиття тексту на слова;
4. Розбиття слова на фонетичні звуки;
5. Призначення фонетичним звукам форми вуст;
6. Якщо фонетичний звук є голосною то додаєм 3 кадри анімації, якщо голосна то 5 кадрів;
7. Якщо це початок слова то додаєм до останнього кадру анімації 7 кадрів;
8. Якщо була комбінація приголосної, голосної, приголосної фонетичного звуку то викриваємо і закриваємо щелепу;
9. Підставляєм форми вуст у визначені фонетичні звуки;
10. Якщо в текстовому файлі існують ще слова повторюємо починаючи з пункту 4;
11. Відображати лицьову анімацію;
12. Кінець алгоритму функціонування програмного модулю оброблення полігональних моделей анімаційних зображень.

## **2.7 Висновки**

1. Аналіз тривимірних редакторів показав, що найкращим для реалізації програмного засобу оброблення та синхронізації контенту з елементами візуалізації є тривимірний редактор Maya, оскільки Maya володіє широким спектром інструментів для роботи з анімацією, включаючи роботу з ключовими кадрами, скелетною анімацією, а також мозаїкою фонем та відтворенням емоцій через рухи обличчя.

2. Розроблено метод розбиття слова на фонетичні звуки та метод конвертації цих фонетичних звуків у реалістичні анімаційні рухи обличчя. Цей метод спрощує створення високоякісної лицьової анімації для

автоматизації генерації медіаконтенту.

3. Розроблено загальну математичну модель анімаційного зображення, в основі якої лежать принципи фізичної моделювання рухів обличчя та фонетичних звуків. Ця модель сприяє створенню більш точної та емоційно насиченої анімації обличчя на основі текстових файлів.

4. Розроблено загальну структурну схему функціонування програмного засобу, що визначає ключові етапи та компоненти процесу анімації обличчя на основі текстових файлів. Структурна схема чітко визначає послідовність операцій, від введення текстового файлу до генерації реалістичної анімації обличчя.

5. Розроблено інтерактивні анімації обличчя для програмного засобу оброблення та синхронізації контенту з елементами візуалізації та впроваджено інтерактивні можливості для створення живого та реалістичного образу обличчя. Інтеграція анімації очей, виразів обличчя та руху голови додає додатковий рівень емоційної виразності до синтезованого образу.

6. Розроблено алгоритм функціонування програмного засобу оброблення та синхронізації контенту з елементами візуалізації. Алгоритм функціонування програмного засобу забезпечує ефективну обробку та синхронізацію контенту з візуалізацією на основі текстових файлів. Процес розбиття слів на фонетичні звуки і їх подальша конвертація в анімаційні рухи реалізовані із застосуванням математичних моделей та структур.

## **3 ПРОГРАМНА РЕАЛІЗАЦІЯ МОДУЛЯ ОБРОБЛЕННЯ ПОЛІГОНАЛЬНИХ МОДЕЛЕЙ АНІМАЦІЙНИХ ЗОБРАЖЕНЬ**

### **3.1 Обґрунтування вибору мови програмування та особливості платформи реалізації**

Python [21]– це високорівнева мова програмування, яка широко використовується в індустрії візуальних ефектів та 3D-анімації, зокрема в Autodesk Maya. Давайте розглянемо ключові аспекти використання Python у Maya:

#### **Простота та Читабельність:**

Python славиться своєю простотою та читабельністю коду. Це важливо, оскільки у Maya велика кількість коду використовується для створення та управління об'єктами та анімацією.

#### **Можливості API:**

Python API для Maya надає доступ до широкого спектру функціоналу програми. Це означає, що ви можете автоматизувати практично будь-яку задачу, яку ви зазвичай виконуєте вручну.

#### **Скрипти та Плагіни:**

Python дозволяє створювати короткі скрипти для автоматизації рутинних завдань, а також більш складні плагіни для розширення функціоналу Maya.

#### **Інтеграція з Іншими Інструментами:**

Багато інших інструментів візуальних ефектів та рендерингу також підтримують Python, що дозволяє легко інтегрувати різні етапи вашого процесу роботи.

#### **Спільнота та Ресурси:**

Python має велику та активну спільноту. Це означає, що ви можете знайти безліч ресурсів, підручників та форумів, які можуть допомогти вам розвиватися як програміст у візуальних ефектах.

#### Мультиплатформеність:

Python є мультиплатформеною мовою, що означає, що ваш код буде працювати на різних операційних системах без змін.

#### Розширюваність:

Python дозволяє розширювати функціонал Maya, створюючи нові інструменти та підгонюючи середовище під ваші потреби.

Python у Maya — це потужний інструмент для створення скриптів, автоматизації рутинних завдань та створення розширень, що робить його популярним серед візуальних ефектів та анімацій.

MEL є вбудованою мовою програмування для Autodesk Maya, спеціально розробленою для використання в цьому програмному забезпеченні. Давайте розглянемо ключові аспекти використання MEL:

#### Інтеграція з Maya:

MEL [22] розроблено для тісної інтеграції з інтерфейсом та функціоналом самої Maya. Це дозволяє легко управляти об'єктами, анімацією та іншими аспектами програми.

#### Команди та Сценарії:

MEL базується на командах, які можна викликати з командного рядка, а також використовувати для написання скриптів та сценаріїв.

#### Шаблони та Процедури:

Мова підтримує використання шаблонів та процедур, що полегшує структурування коду та його повторне використання.

#### Широке Використання:

MEL використовується для автоматизації завдань, створення плагінів та розширень, а також для реалізації складних анімаційних сценаріїв.

#### Інтерактивність:

MEL підтримує інтерактивний режим, де користувач може виконувати команди безпосередньо з командного рядка.

Недоліки MEL порівняно з Python в контексті візуальних ефектів та анімації в Maya:

#### Обмежена Функціональність:

MEL може бути обмеженим в порівнянні з Python у відношенні функціональності та можливостей, оскільки воно було спеціально створено для роботи в середовищі Maya.

#### Читабельність та Структура Коду:

Python зазвичай вважається більш читабельною мовою з хорошою структурою коду, що полегшує розробку та обслуговування великих проектів, що може бути важливо в області візуальних ефектів.

#### Ширший Спектр Застосування:

Python є більш універсальною мовою програмування, що означає, що вона може використовуватися не тільки для роботи з Maya, а й для інших завдань у сфері візуальних ефектів та програмування.

#### Активна Спільнота:

Python має значно більшу та активну спільноту, що означає більше ресурсів та підтримки для програмістів.

Хоча MEL є потужним інструментом для роботи в Maya, Python, завдяки своїй універсальності та зручності в розробці, стає все більш популярним в середовищі візуальних ефектів та анімації.

C++ [23] є потужною та високопродуктивною мовою програмування, широко використовуваною в галузі комп'ютерної графіки та візуальних ефектів, включаючи анімацію. Основні аспекти використання C++ для цих завдань включають:

#### Висока Швидкість Виконання:

C++ компілюється безпосередньо в машинний код, що дозволяє досягати високої швидкодії виконання, що важливо для обробки великого обсягу графічних даних та обчислень.

#### Доступ до Ресурсів Пам'яті:

Можливість прямого управління пам'яттю дозволяє ефективно

працювати з великими обсягами даних, що є ключовим для графічних застосувань.

Розширені Бібліотеки та Фреймворки:

C++ має багато бібліотек та фреймворків, спрямованих на роботу з графікою, що полегшує розробку візуальних ефектів та анімацій.

Широка Підтримка Від Галузі Грифіки:

Спільнота розробників у галузі графіки широко використовує C++, що забезпечує обмін знань та ресурсів.

Об'єктно-Орієнтована Парадигма:

C++ підтримує об'єктно-орієнтоване програмування, що полегшує організацію та управління складними проектами.

Недоліки C++ порівняно з Python в контексті візуальних ефектів та анімації:

Складність та Об'єм Коду:

C++ може вимагати більше коду для досягнення тих самих результатів порівняно з Python, що може призвести до більшого обсягу та складності кодової бази.

Системне Програмування:

Враховуючи, що C++ більше використовується для системного програмування, він може бути менш зручним для швидкого розроблення та експериментів, які зазвичай потрібні в області візуальних ефектів.

Брак Гнучкості:

Python є більш гнучкою мовою та дозволяє швидше змінювати та тестувати код, що важливо в області візуальних ефектів, де частіше потрібно експериментувати з різними підходами.

У висновку, Python став оптимальним вибором для реалізації програмного засобу, оскільки він поєднує в собі легкість використання, гнучкість та широкі можливості, необхідні для успішної розробки візуальних ефектів та анімацій у контексті тривимірного редактора.

### 3.2 Аналіз середовища розробки

Script Editor в Maya є важливим інструментом для написання та виконання сценаріїв (скриптів) для автоматизації завдань та взаємодії з тривимірними об'єктами. Основна мова сценаріїв в Maya - це Maya Embedded Language (MEL).

Script Editor включає інтерактивне середовище, де можна вводити та виконувати команди MEL. Він також надає зручний інтерфейс для написання, редагування та збереження скриптів. Користувач може використовувати його для експериментів з командами, тестування нових ідей і вивчення MEL.

Майстри тривимірної графіки часто використовують Python для програмування в Maya через його могутню API та багатofункціональні бібліотеки. Python дозволяє розробникам створювати скрипти для автоматизації, обробки даних, створення інтерфейсів користувача та навіть для розширення функціональності Maya.

Однією з ключових областей програмування в Maya є створення власного інтерфейсу користувача. За допомогою MEL або Python можна створювати вікна, панелі та елементи керування, що дозволяє розробникам створювати спеціалізовані інструменти, оптимізовані для їхніх потреб.

Крім того, програмування в Maya включає розробку плагінів, які додають нові функції та можливості до програми. Розробники можуть використовувати C++, Python або MEL для створення розширень, які розширюють функціональність Maya згідно з їхніми потребами.

Node Editor — це інструмент, який використовується в графічних програмах та редакторах для візуального програмування та налаштування графічних або обчислювальних процесів. У контексті тривимірного редактора, такого як Maya чи Blender, Node Editor дозволяє користувачам створювати та змінювати графи взаємодії між об'єктами, матеріалами,





### 3.3 Програмна реалізація засобу оброблення та синхронізації контенту з елементами візуалізації

Для реалізації алгоритму розбиття слова на фонетичні звуки було використано бібліотеку NLTK [24].

Natural Language Toolkit (NLTK) [24] - це бібліотека для мови програмування Python, яка надає інструменти для роботи з природною мовою (Natural Language Processing, NLP). NLTK містить різноманітні модулі та ресурси, які допомагають в обробці та аналізі текстів.

Деякі основні можливості NLTK включають:

- Токенізація: Розділення тексту на токени (слова або фрази).
- Стемінг і лематизація: Визначення основи слова (стема) або його нормальної форми.
- Частотний аналіз: Підрахунок частоти вживання слів у тексті.
- Морфологічний аналіз: Аналіз морфем та будови слова.
- Синтаксичний аналіз: Визначення структури речень та граматичних зв'язків.

NLTK також містить різноманітні корпуси текстів і ресурси для навчання та розробки моделей для обробки природної мови.

Ця бібліотека широко використовується у сферах обробки природної мови, машинного навчання та комп'ютерної лінгвістики. Вона надає зручний інтерфейс для вивчення та аналізу текстових даних, допомагаючи дослідникам та розробникам реалізовувати різноманітні завдання, пов'язані з мовним аналізом. NLTK становить цінний інструментарій для обробки природної мови, використовується для розробки та вдосконалення алгоритмів, а також для навчання моделей, спрямованих на вирішення завдань у сфері мовленнєвого аналізу.

Багато великих проектів та досліджень у галузі NLP базуються на NLTK через його потужність та гнучкість. Зокрема, він використовується для створення та навчання моделей машинного навчання, розпізнавання іменованих сутностей, сентимент-аналізу та багатьох інших аспектів обробки

природної мови.

NLTK дозволяє дослідникам та розробникам виконувати різноманітні завдання, від базового аналізу тексту до складніших лінгвістичних вивчень, роблячи його ключовим інструментом для розвитку сучасних рішень у галузі обробки мовленнєвих даних.

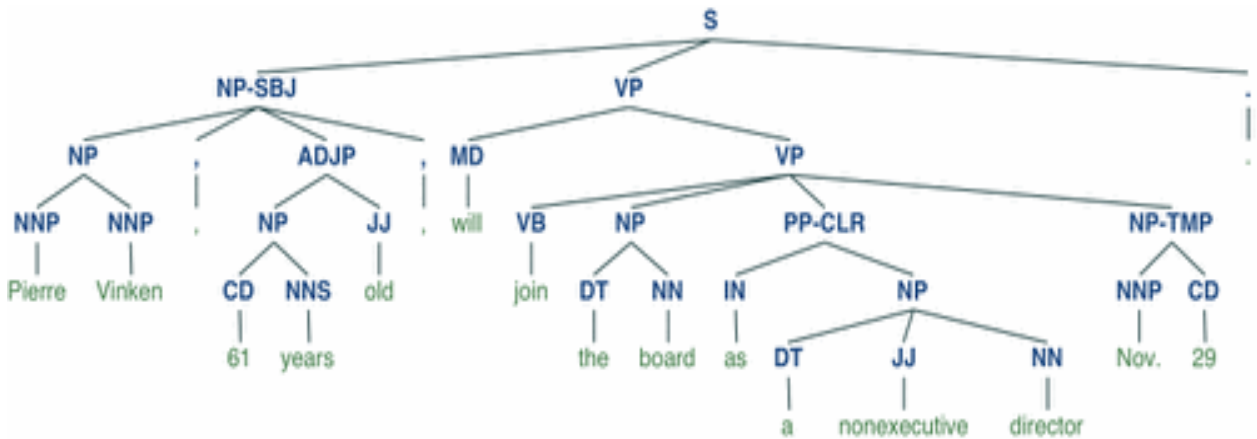


Рисунок 3.2 – Схема функціонування бібліотеки фонетичного синтезу NLTK

NLTK підходить для лінгвістів, інженерів, студентів, викладачів, дослідників та промислових користувачів. NLTK доступний для Windows, MacOS X та Linux. Перш за все, NLTK - це безкоштовний проект з відкритим вихідним кодом, керований спільнотою.

NLTK (Natural Language Toolkit) використовується для розбиття слова на фонетичні звуки за допомогою різноманітних функцій, що надаються цією бібліотекою. Однією з ключових можливостей є токенізація, яка дозволяє розділити текст на окремі токени, що можуть представляти слова чи фрази. Також, NLTK має функції для стемінгу та лематизації, що дозволяють визначати основу слова або його нормальну форму.

Окрім цього, бібліотека надає можливості частотного та морфологічного аналізу, що допомагає визначати структуру текстів та аналізувати будову слів. NLTK дозволяє використовувати різноманітні корпуси текстів та ресурси для розробки моделей для обробки природної мови.

Дерево пошуку - це структура даних, яка використовується для зберігання та організації даних з можливістю ефективного пошуку, вставки та вилучення. В основі дерева пошуку лежить принцип впорядкування, що дозволяє швидше локалізувати конкретний елемент.

Основні елементи дерева пошуку:

**Корінь (Root):** Початковий вузол дерева, від якого починається структура. Він не має батьківського вузла.

**Вузол (Node):** Елемент дерева, який містить значення та може мати дочірні вузли.

**Гілка (Edge):** З'єднання між вузлами, вказує напрямок від батьківського вузла до дочірнього.

**Піддерево (Subtree):** Частина дерева, яка сама є деревом. Складається з батьківського вузла та всіх його нащадків.

**Лист (Leaf):** Вузол, який не має дочірніх вузлів. Листя зазвичай містять значення даних.

**Впорядкування (Ordering):** Властивість дерева пошуку, що визначає, як значення розміщуються в дереві. У лівому піддереві значення менше за значення в поточному вузлі, а в правому - більше.

Операції над деревом пошуку включають:

**Пошук (Search):** Знаходження вузла з конкретним значенням.

**Вставка (Insert):** Додавання нового вузла з певним значенням.

**Вилучення (Delete):** Видалення вузла з певним значенням.

Дерева пошуку забезпечують швидкий доступ та ефективне управління даними, зокрема в контексті операцій пошуку. Вони застосовуються в різних областях, включаючи бази даних, операційні системи та алгоритми штучного інтелекту.

Завдяки цим функціям, NLTK стає потужним інструментом для роботи з природною мовою, а використання його у проекті дозволяє розширити можливості обробки мовленнєвих даних. Використання бібліотеки NLTK спрощує аналіз тексту та допомагає вирішувати завдання, пов'язані з

фонетичним синтезом та іншими аспектами обробки природної мови.

Для створення інтерфейсу було застосовано PyQt.

PyQt [26] — це відкрита бібліотека для розробки графічних інтерфейсів користувача (GUI) в мові програмування Python. PyQt базується на бібліотеці Qt, яка є однією з найпопулярніших бібліотек для розробки GUI у світі програмування. Вона надає програмістам інструменти для створення ефективних та естетично зовнішніх інтерфейсів для своїх програм.

PyQt включає в себе різноманітні класи та модулі для роботи з графікою, подіями, мережами, базами даних та іншими аспектами програмного забезпечення. Вона дозволяє програмістам легко створювати вікна, кнопки, меню та інші елементи інтерфейсу, а також обробляти події, пов'язані з користувачем.

Використання PyQt дозволяє розробникам створювати крос-платформенні програми, які можуть працювати на різних операційних системах, таких як Windows, Linux та macOS. PyQt добре інтегрується з іншими популярними бібліотеками та інструментами для розробки програмного забезпечення, що робить його зручним і потужним вибором для створення GUI в програмах на Python.

PyQt є важливим інструментом для розробників Python, оскільки він дозволяє легко створювати графічні інтерфейси користувача для своїх програм. Заснований на Qt, цей фреймворк надає широкий спектр функцій для роботи з графікою, подіями, мережами і базами даних. Його використання дозволяє розробникам створювати ефективні та привабливі інтерфейси, а також легко обробляти події, пов'язані з користувачем.

Однією з ключових переваг PyQt є його крос-платформенність, що дозволяє створювати програми, які працюють на різних операційних системах, що робить його універсальним і зручним для розробників, які хочуть, щоб їх програми були доступні на різних платформах.

Бібліотека добре інтегрується з іншими інструментами та бібліотеками Python, що робить її потужним і гнучким вибором для розробки програмного

забезпечення. PyQt відкриває широкі можливості для створення професійних та стильних користувацьких інтерфейсів в різних галузях, від наукових досліджень до створення додатків з графічним інтерфейсом користувача.

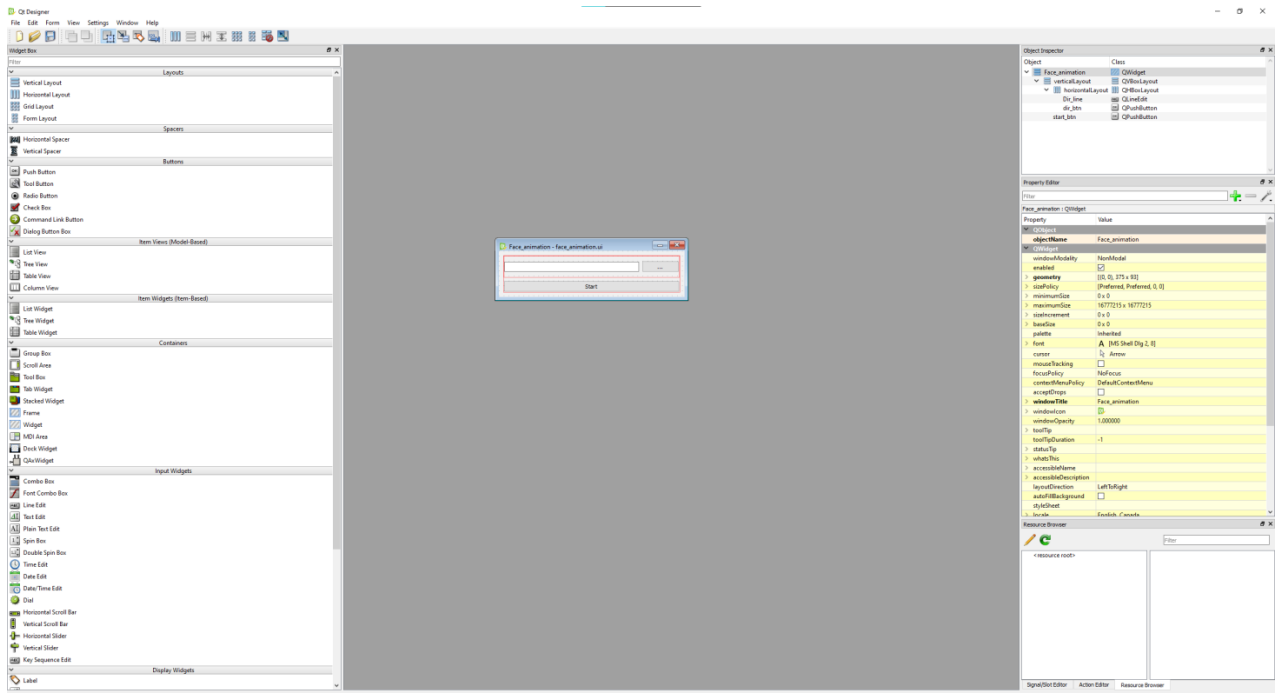


Рисунок 3.2 – Загальний вигляд інтерфейсу Qt Designer

Функція `def __init__(self)` відповідає за роботу з інтерфейсом та вивом інших функцій:

- Завантаження інтерфейсу проводиться через `loadUI` з використанням файлу `face_animation.ui`, який ймовірно містить елементи керування.
- Визначення `controls` за допомогою `lsUI` дозволяє отримати всі елементи управління вікном.
- Кожний елемент управління присвоюється як атрибут об'єкта, зробивши його доступним для подальшого використання.
- Ініціалізація кнопок, зокрема кнопки вибору директорії та кнопки запуску обробки файлу.

Функція `openDir(self,tmp)` відповідає за вибор файлу в `dir selector` за допомогою модулів анімаційного редактору MAYA:

- Функція викликається при натисканні кнопки вибору директорії.
- Використання `fileDialog2` дає можливість користувачеві вибрати директорію, а шлях до обраної директорії виводиться у консоль та встановлюється в текстове поле.

Функція `fileProcessing(self, tmp)` зчитує інформацію з файлу поки вона в ньому існує, та передає цю інформацію іншим функціям:

```
dir=cmds.textField( self.Dir_line, text = True, q = True ):
```

Отримує шлях до директорії, введений користувачем через текстове поле інтерфейсу.

```
with open(self.dir[0], 'r') as f:
```

Виводить кожен рядок файлу у консоль для відладки.

```
for world in line.split():
```

Запускає цикл, що перебирає слова в рядку, розділені пробілами.

```
cmu_pronunciation = self.wordbreak(world)[0]:
```

Визначає фонетичну транскрипцію для кожного слова, використовуючи функцію `wordbreak` та обирає перший елемент (оскільки `wordbreak` повертає список).

```
phomens = [self.convert_cmu_to_phonem(cmu) for cmu in cmu_pronunciation]:
```

Перетворює фонетичні записи з CMU в фонемі, використовуючи функцію `convert_cmu_to_phonem`.

Функція `wordbreak(tmp, s)` розбиває інформацію в файлі на окремі слова, щоб потім передавати ці слова в функцію `convert_cmu_to_phonem(tmp, cmu)`, яка свою чергу розбиває ці слова на фонетичні звуки по циклу:

- Розбиває слова на фонетичні склади, використовуючи арпабет та рекурсивний підхід.
- Використовується сортування по відстані від середини слова для ефективного розділення.

Функція `face_animation(self, phomens)` відповідає за створення анімації тривимірного персонажа на основі отриманих фонетичних звуків. В даній

функції існують команди для роботи з тривимірною графікою такі як:

1) `setAttr` [15] – Встановлює значення атрибута вузла залежності. Команда ``setAttr`` в Autodesk Maya використовується для встановлення значень атрибутів об'єкта в 3D-сцені. Ця команда дозволяє змінювати параметри та властивості об'єктів, такі як положення, обертання, розміри, матеріали тощо. Наприклад, якщо у вас є 3D-об'єкт (наприклад, модель персонажа), ви можете використати команду ``setAttr``, щоб змінити його атрибути. Наприклад: `cmds.setAttr('myCharacter.translateX', 10.0)` Ця команда змінить положення об'єкта ``myCharacter`` по осі X на значення 10.0. Ви можете використовувати ``setAttr`` для редагування різних атрибутів залежно від потреб вашого проекту.

2) `Select` [15] – Ця команда використовується для внесення об'єктів до або виключення з активного списку. Якщо жоден із п'яти прапорців `[-add, -af, -r, -d, -tgl]` не вказано, за замовчуванням об'єкти в активному списку замінюються заданим списком об'єктів; Команда ``select`` в Autodesk Maya використовується для вибору об'єктів в сцені. За допомогою цієї команди ви можете визначати, які об'єкти будуть активними або виділеними у вашій сцені. Наприклад, якщо у вас є об'єкти з іменами "object1", "object2" і "object3", ви можете використати команду ``select``, щоб вибрати їх: `cmds.select('object1', 'object2', 'object3')`. Ця команда вибере об'єкти з іменами 'object1', 'object2' і 'object3' у вашій сцені. Ви також можете використовувати різні параметри та умови для вибору об'єктів, таких як за типом об'єкта, по імені, за матеріалом тощо.

3) `setKeyframe` [15] – Ця команда створює ключові кадри для вказаних об'єктів або активних об'єктів, якщо жодного не вказано в командному рядку. Часом за замовчуванням для нових ключових кадрів є поточний час. Перевизначте цю поведінку за допомогою прапорця «-t» у командному рядку. Значенням за замовчуванням для ключового кадру є поточне значення атрибута, для якого встановлено ключовий кадр. Перевизначте цю поведінку за допомогою прапорця «-v» у командному рядку. Команда ``setKeyframe`` в



Autodesk Maya використовується для встановлення ключових кадрів для анімаційних параметрів об'єкта на конкретний час. Наприклад, якщо ви хочете встановити ключовий кадр для параметра "translateX" об'єкта на значенні 5 на кадрі 10, ви можете використати такий код: `cmds.setKeyframe('yourObject', attribute='translateX', value=5, time=10)` Ця команда встановить ключовий кадр для об'єкта з іменем 'yourObject' для параметра "translateX" на значенні 5 на кадрі 10. За допомогою команди `'setKeyframe'` ви можете контролювати анімаційні зміни параметрів об'єктів у вашій сцені.

4) Joint [15] – Команда `'joint'` в контексті тривимірної графіки та анімації використовується для створення **\*\*суглобів\*\*** або **\*\*коштовних точок\*\*** (які також іноді називають "координатними точками") в цифровому об'ємному просторі, такому як сцена 3D-моделювання.

Створення суглоба (joint): Команда `'joint'` використовується для створення нового суглоба. Суглоби можуть використовуватися для створення скелетної структури для тривимірних персонажів або об'єктів. Приклад використання:

```
python joint -p 0 0 0 -n "joint1".
```

У цьому прикладі:

- `'-p 0 0 0'` встановлює позицію суглоба в тривимірному просторі (в даному випадку, початкова позиція у координатах  $x=0, y=0, z=0$ ).

- `'-n "joint1"'` встановлює ім'я суглоба як "joint1".

Контроль суглобів (joint control): Суглоби часто використовуються для створення кісток та анімаційних скелетів. Контрольні точки або іконки суглобів визначають, як суглоб повинен поводитися під час руху об'єкта чи персонажа.

Дана функція визначає кадр анімації який потрібно ставити форму вуст, відкриття щелепи, останій кадр анімації, кадр виключення форми вуст із анімації:

### 3.4 Висновки

1. Обґрунтовано вибір мови програмування для реалізації програмного засобу. Вибір мови програмування є критичним аспектом розробки програмного засобу для анімації обличчя на основі текстових файлів. В якості мови програмування обрано мову Python, оскільки ця мова містить велику кількість бібліотек, які орієнтовані на реалізацію функцій анімації та функцій розпізнавання фонетичних звуків.

2. Проаналізовано середовища розробки Maya та Script Editor. Середовища розробки Maya та Script Editor є ключовими компонентами в процесі створення анімації обличчя на основі текстових файлів. Maya, як професійний 3D-редактор, надає розширені можливості для створення та редагування тривимірних сцен, а також інтеграції скриптів та плагінів. Script Editor, з іншого боку, є потужним інструментом для написання, відлагодження та тестування скриптів.

3. Розроблено програмний засіб оброблення та синхронізації контенту з елементами візуалізації, який дозволяє ефективно обробляти тексти з фонетичними звуками, анімувати обличчя та взаємодіяти з тривимірною моделлю обличчя в середовищі Maya.

## **4. ТЕСТУВАННЯ ПРОГРАМНОГО ЗАСОБУ ОБРОБЛЕННЯ ТА СИНХРОНІЗАЦІЇ КОНТЕНТУ З ЕЛЕМЕНТАМИ ВІЗУАЛІЗАЦІЇ**

### **4.1 Аналіз підходів до тестування програмного засобу**

При виконанні тестування програмного засобу виникла необхідність перевірити не лише коректність створених анімацій, але й ефективність розбиття слів на фонетичні звуки. Також було важливо переконатися в правильному функціонуванні модулю при обробці різних типів текстових файлів. Це дозволяло забезпечити не тільки відповідність задуманому функціоналу, але і забезпечити універсальність застосунку.

Під час тестування програмного засобу, що відтворює анімацію обличчя на основі текстового файлу в середовищі Maya, важливо зосередитися на ретельному аналізі різних аспектів функціоналу. Ось розгорнута стратегія тестування:

#### **1. Тестування інтерфейсу:**

- Оцініть зручність та зрозумілість інтерфейсу програми для користувачів.

- Перевірте можливості введення тексту для генерації анімації.

#### **2. Тестування завантаження файлів:**

- Переконайтеся, що програма належним чином завантажує текстові файли для створення анімації.

#### **3. Тестування анімації обличчя:**

- Перевірте, чи анімація відтворюється коректно, відповідно до введеного тексту.

- Здійсніть тестування на різних тривимірних обличчях, переконайтеся в універсальності програмного засобу.

#### **4. Тестування точності відтворення фонем:**

- Впевніться, що програма правильно виділяє фонем з тексту та адекватно їх відтворює у вигляді анімації.

#### 5. Тестування реалістичності анімації:

- Оцініть реалістичність анімації обличчя, звертаючи увагу на відтворення емоцій та виразів.

#### 6. Тестування на швидкість і продуктивність:

- Визначте швидкість генерації анімації та виявіть можливі затримки при обробці обсягу великих текстових файлів.

#### 7. Тестування сумісності:

- Перевірте, чи програма коректно працює на різних версіях Maya та під різними операційними системами.

#### 8. Тестування обробки помилок:

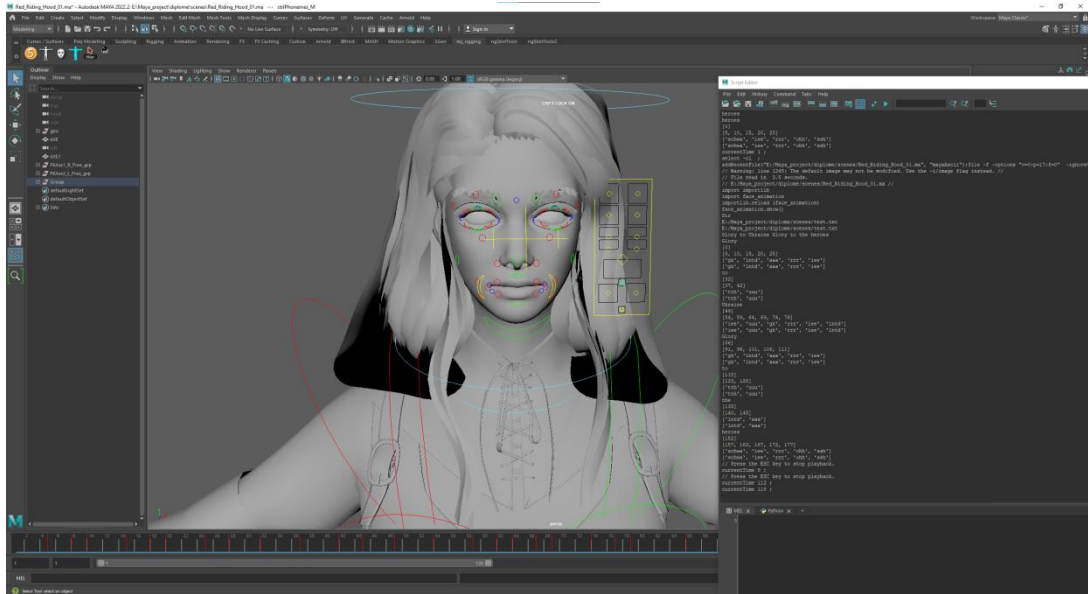
- Створіть тестові сценарії з помилковим вводом та переконайтеся, що програма адекватно обробляє помилки та надає користувачу інформацію про них.

### **4.2 Тестування програмного забезпечення**

Під час тестування приділялося уваги не лише позитивним сценаріям використання, але і виявленню можливих варіантів помилок. Це включало в себе тестування реакції системи на некоректні вхідні дані, а також перевірку стійкості програмного модулю до різних умов роботи. Такий підхід дозволяє забезпечити надійність та стабільність програми в реальних умовах використання.

Для тестування було використано два текстових файли з коротким та довгим реченням. У першому файлі написано “Glory to Ukraine! Glory to the heroes”. В другому текстовому файлі записаний гімн України на англійській мові. Результатом має бути різна лицьова анімація.

Перший файл правильно розбив всі слова на фонетичні звуки. Довжина анімації вийшла 180 кадрів. Анімації правильно виставлений час форм вуст та роботи щелепи. Результати тестування зображені на рисунку 3.3.



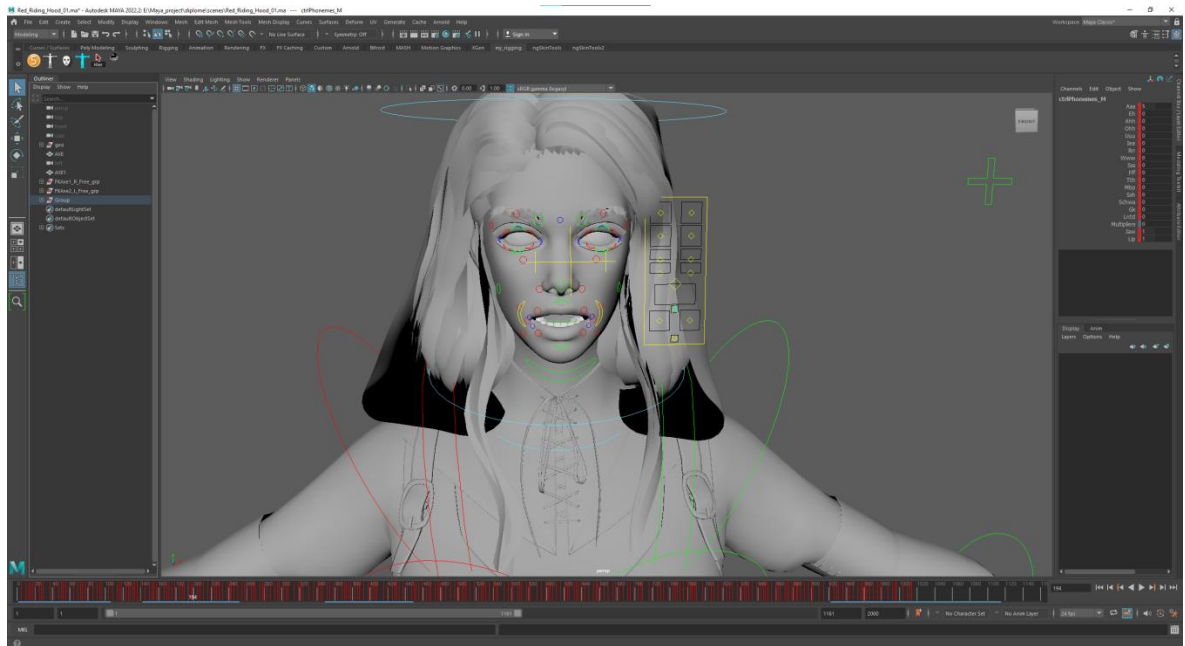
```

Glory to Ukraine Glory to the heroes
Glory
[0]
[5, 10, 15, 20, 25]
['gk', 'lntd', 'aaa', 'rrr', 'iee']
['gk', 'lntd', 'aaa', 'rrr', 'iee']
to
[32]
[37, 42]
['tth', 'uuu']
['tth', 'uuu']
Ukraine
[49]
[54, 59, 64, 69, 74, 79]
['iee', 'uuu', 'gk', 'rrr', 'iee', 'lntd']
['iee', 'uuu', 'gk', 'rrr', 'iee', 'lntd']
Glory
[86]
[91, 96, 101, 106, 111]
['gk', 'lntd', 'aaa', 'rrr', 'iee']
['gk', 'lntd', 'aaa', 'rrr', 'iee']
to
[118]
[123, 128]
['tth', 'uuu']
['tth', 'uuu']
the
[135]
[140, 145]
['lntd', 'aaa']
['lntd', 'aaa']
heroes
[152]
[157, 162, 167, 172, 177]
['schwa', 'iee', 'rrr', 'ohh', 'ssh']
['schwa', 'iee', 'rrr', 'ohh', 'ssh']

```

Рисунок 4.1 – Результати роботи програмного модулю синтезу текстового файлу

Другий файл правильно розбив всі слова на фонетичні звуки. Довжина анімації вийшла 1041 кадрів. Анімації правильно виставлений час форм вуст та роботи щелепи. Результати тестування зображені на рисунку 3.4.



```

Ukraine has not yet perished nor her glory nor her freedom

Ukraine
[0]
[5, 10, 15, 20, 25, 30]
['iee', 'uuu', 'gk', 'rrr', 'iee', 'lntd']
['iee', 'uuu', 'gk', 'rrr', 'iee', 'lntd']
has
[37]
[42, 47, 52]
['schwa', 'aaa', 'ssh']
['schwa', 'aaa', 'ssh']
not
[59]
[64, 69, 74]
['lntd', 'aaa', 'tth']
['lntd', 'aaa', 'tth']
yet
[81]
[86, 91, 96]
['iee', 'eh', 'tth']
['iee', 'eh', 'tth']
perished
[103]
[108, 113, 118, 123, 128, 133]
['mbp', 'eh', 'rrr', 'iee', 'sss', 'tth']
['mbp', 'eh', 'rrr', 'iee', 'sss', 'tth']
nor
[140]
[145, 150, 155]
['lntd', 'aaa', 'rrr']
['lntd', 'aaa', 'rrr']
her
[162]
[167, 172]
['schwa', 'rrr']
['schwa', 'rrr']
glory
[179]
[184, 189, 194, 199, 204]
['gk', 'lntd', 'aaa', 'rrr', 'iee']
['gk', 'lntd', 'aaa', 'rrr', 'iee']
nor
[211]
[216, 221, 226]
['lntd', 'aaa', 'rrr']
['lntd', 'aaa', 'rrr']
her
[233]
[238, 243]
['schwa', 'rrr']
['schwa', 'rrr']
freedom
[250]
[255, 260, 265, 270, 275, 280]
['fff', 'rrr', 'iee', 'lntd', 'aaa', 'mbp']
['fff', 'rrr', 'iee', 'lntd', 'aaa', 'mbp']
Upon us fellow Ukrainians fate shall smile once more
  
```

Рисунок 4.2 – Результаты работы программного модуля синтеза текстового файла

Проведемо тестування анімації з емоціям та кліпанням очей. В рамках тестування було проведено детальний аналіз емоційної виразності та реалістичності кліпання очей в анімації обличчя. Оцінювалися якість та точність відтворення різних емоцій, а також натуральність рухів очей під час кліпання.

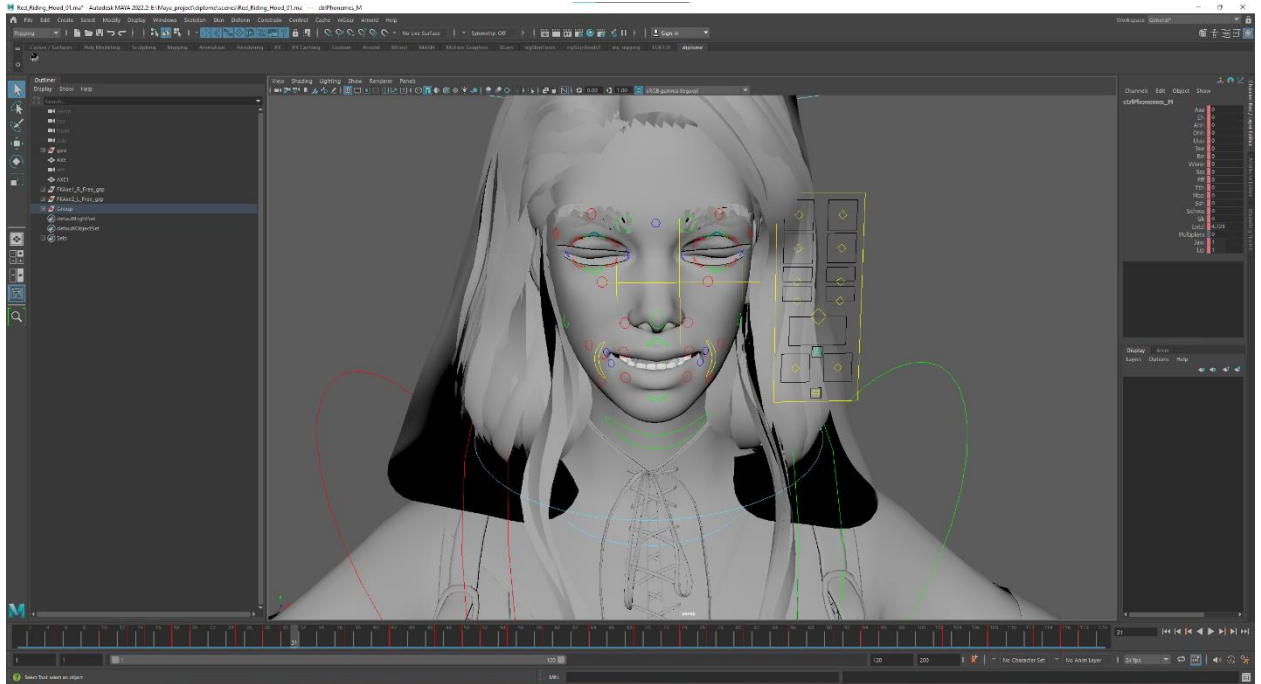


Рисунок 4.3 – Результати роботи програмного засобу разом з кліпанням

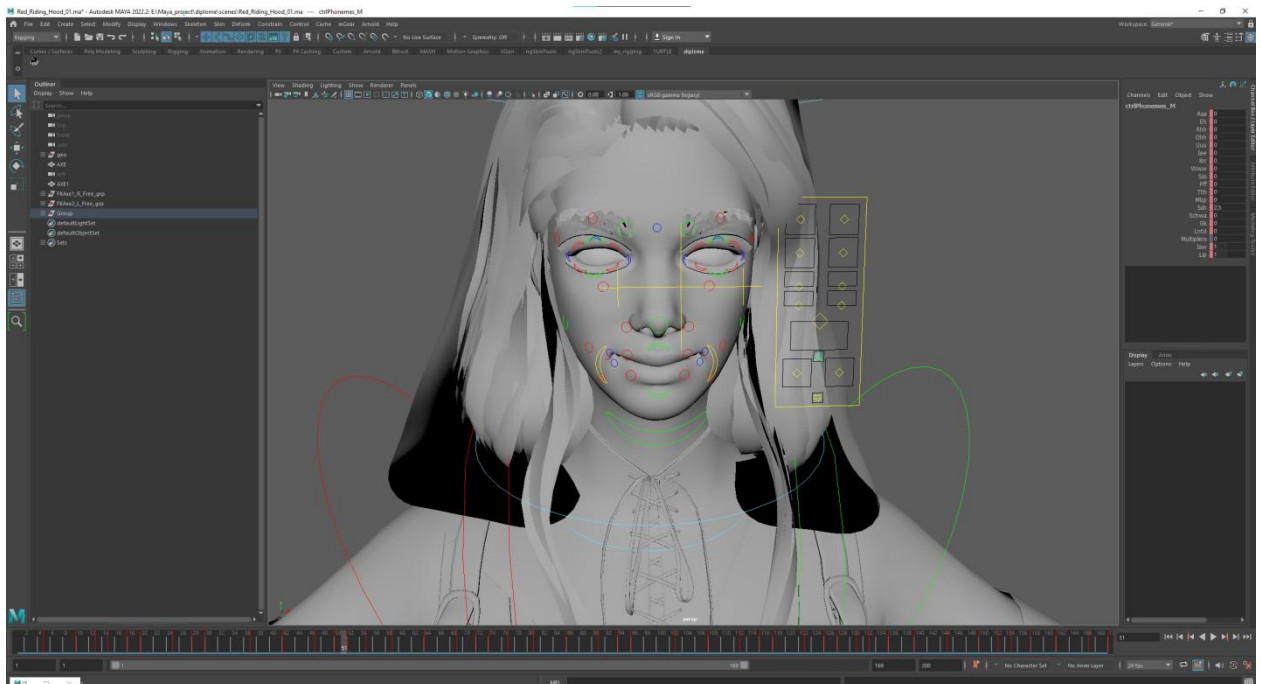


Рисунок 4.4 – Результати роботи програмного засобу разом з емоціями

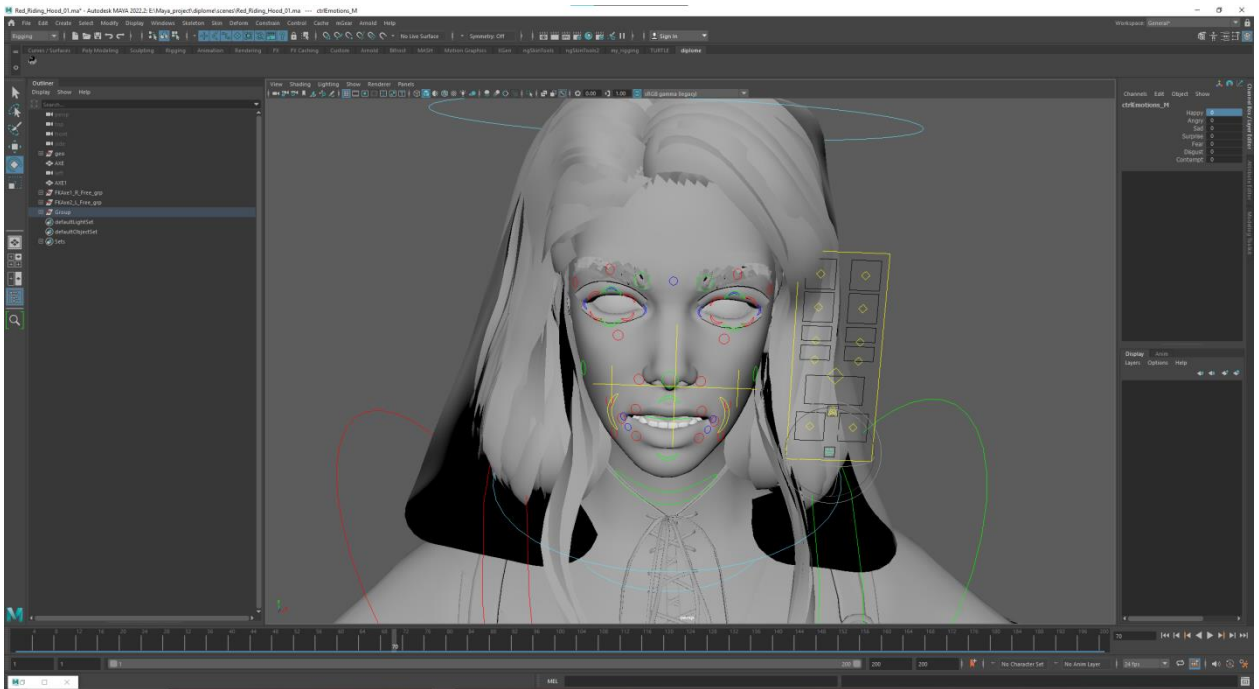


Рисунок 4.5 – Результати роботи програмного засобу разом з нехилом голови

### 4.3 Розробка інструкції користувача

#### 1. Встановлення:

1.1 Помістіть файл модулю в папку сценаріїв Maya ( usually `c:\Users\User\Documents\maya\2022\scripts\`;

1.2 Створити кнопку на полиці за допомогою команди:  
`face_animation.show()`;

1.3 Натисніть на кнопку і використовуйте.

#### 2. Використання:

Після відкриття головного меню (рис. 3.5) потрібно вибрати текстовий файл (рис. 3.6) та натиснути на кнопку start.



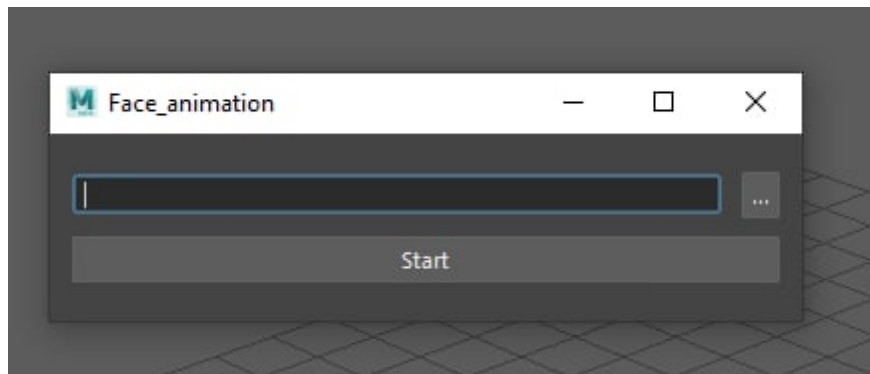


Рисунок 3.8 – Загальний вигляд інтерфейсного вікна типу «Головне меню»

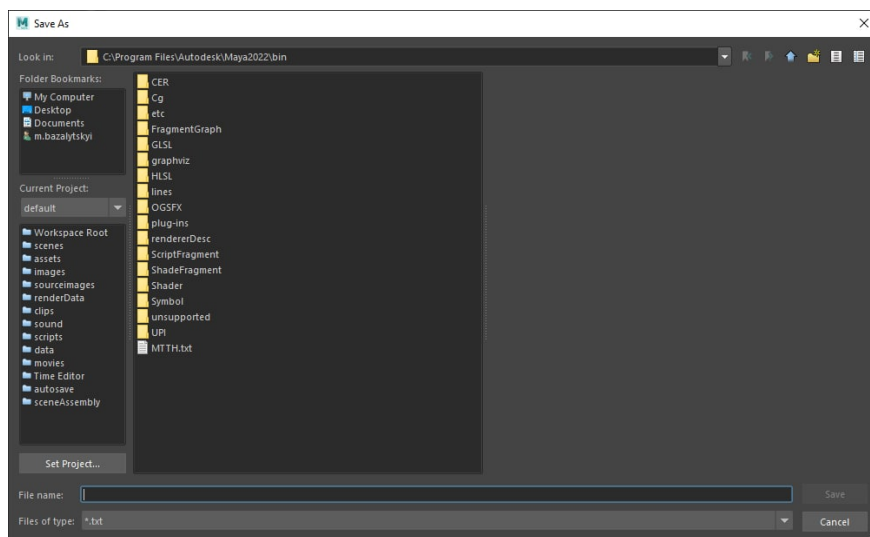


Рисунок 3.9 – Меню вибору файлу

#### 4.4 Висновки

1. Під час тестування програмного засобу для відтворення анімації обличчя на основі текстового файлу в середовищі Мауа було виявлено його високу ефективність, точність та зручність використання. Програма демонструє стабільну роботу, забезпечуючи реалістичне відображення виразів обличчя. Вона також відповідає стандартам сумісності та стабільності, що робить її ефективним інструментом для творчих завдань у галузі 3D-графіки.

2. Розроблено інструкцію користувача по встановленню та використанню програмного засобу.

## 5 ЕКОНОМІЧНА ЧАСТИНА

Науково-технічна розробка має право на існування та впровадження, якщо вона відповідає вимогам часу, як в напрямку науково-технічного прогресу та і в плані економіки. Тому для науково-дослідної роботи необхідно оцінювати економічну ефективність результатів виконаної роботи.

Магістерська кваліфікаційна робота за темою «Методи та програмні засоби оброблення та синхронізації контенту з елементами візуалізації» відноситься до науково-технічних робіт, які орієнтовані на виведення на ринок (або рішення про виведення науково-технічної розробки на ринок може бути прийнято у процесі проведення самої роботи), тобто коли відбувається так звана комерціалізація науково-технічної розробки. Цей напрямок є пріоритетним, оскільки результатами розробки можуть користуватися інші споживачі, отримуючи при цьому певний економічний ефект. Але для цього потрібно знайти потенційного інвестора, який би взявся за реалізацію цього проекту і переконати його в економічній доцільності такого кроку.

Для наведеного випадку нами мають бути виконані такі етапи робіт:

- 1) проведено комерційний аудит науково-технічної розробки, тобто встановлення її науково-технічного рівня та комерційного потенціалу;
- 2) розраховано витрати на здійснення науково-технічної розробки;
- 3) розрахована економічна ефективність науково-технічної розробки у випадку її впровадження і комерціалізації потенційним інвестором і проведено обґрунтування економічної доцільності комерціалізації потенційним інвестором.

## 5.1 Проведення комерційного та технологічного аудиту науково-технічної розробки

Метою проведення комерційного і технологічного аудиту дослідження за темою «Методи та програмні засоби оброблення та синхронізації контенту з елементами візуалізації» є оцінювання науково-технічного рівня та рівня комерційного потенціалу розробки, створеної в результаті науково-технічної діяльності.

Оцінювання науково-технічного рівня розробки та її комерційного потенціалу рекомендується здійснювати із застосуванням 5-ти бальної системи оцінювання за 12-ма критеріями, наведеними в табл. 5.1 [26].

Таблиця 5.1 – Рекомендовані критерії оцінювання науково-технічного рівня і комерційного потенціалу розробки та бальна оцінка

Бали (за 5-ти бальною шкалою)					
	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 працездатність продукту в реальних умовах
Ринкові переваги (недоліки)					
2	Багато аналогів на малому ринку	Мало аналогів на малому ринку	Кілька 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	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	Технічні та споживчі властивості продукту значно кращі, ніж в
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

Таблиця 5.1 – Продовження таблиці

Бали (за 5-ти бальною шкалою)					
	0	1	2	3	4
<b>Ринкові перспективи</b>					
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
7	Активна конкуренція великих компаній на ринку	Активна конкуренція	Помірна конкуренція	Незначна конкуренція	Конкуренція немає
<b>Практична здійсненність</b>					
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. Джерела фінансування ідеї 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	Необхідна розробка регламентних документів та отримання великої кількості дозвільних документів на виробництво та реалізацію продукту	Необхідно отримання великої кількості дозвільних документів на виробництво та реалізацію продукту, що вимагає значних коштів та часу	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

Результати оцінювання науково-технічного рівня та комерційного потенціалу науково-технічної розробки потрібно звести до таблиці.

Таблиця 5.2 – Результати оцінювання науково-технічного рівня і комерційного потенціалу розробки експертами

Критерії	Експерт (ПІБ, посада)		
	1	2	3
	Бали:		
1. Технічна здійсненність концепції	4	4	5
2. Ринкові переваги (наявність аналогів)	4	4	5
3. Ринкові переваги (ціна продукту)	3	3	3
4. Ринкові переваги (технічні властивості)	4	4	3
5. Ринкові переваги (експлуатаційні витрати)	3	3	3
6. Ринкові перспективи (розмір ринку)	3	4	3
7. Ринкові перспективи (конкуренція)	3	3	4
8. Практична здійсненність (наявність фахівців)	3	3	3
9. Практична здійсненність (наявність фінансів)	3	3	3
10. Практична здійсненність (необхідність нових матеріалів)	3	3	3
11. Практична здійсненність (термін реалізації)	3	4	4
12. Практична здійсненність (розробка документів)	4	3	3
Сума балів	40	41	42
Середньоарифметична сума балів $СБ_c$	41		

За результатами розрахунків, наведених в таблиці 5.2, зробимо висновок щодо науково-технічного рівня і рівня комерційного потенціалу розробки. При цьому використаємо рекомендації, наведені в табл. 5.3 [26].

Таблиця 5.3 – Науково-технічні рівні та комерційні потенціали розробки

Середньоарифметична сума балів $СБ_c$ , розрахована на основі висновків експертів	Науково-технічний рівень та комерційний потенціал розробки
41...48	Високий
31...40	Вище середнього
21...30	Середній
11...20	Нижче середнього
0...10	Низький

Згідно проведених досліджень рівень комерційного потенціалу розробки за темою «Методи та програмні засоби оброблення та синхронізації контенту з елементами візуалізації» становить 41 бал, що, відповідно до таблиці 5.3, свідчить про комерційну важливість проведення даних досліджень (рівень комерційного потенціалу розробки високий).

## 5.2 Розрахунок витрат на проведення науково-дослідної роботи

Витрати, пов'язані з проведенням науково-дослідної роботи на тему «Методи та програмні засоби оброблення та синхронізації контенту з елементами візуалізації», під час планування, обліку і калькулювання собівартості науково-дослідної роботи групуємо за відповідними статтями.

### 5.2.1 Витрати на оплату праці

До статті «Витрати на оплату праці» належать витрати на виплату основної та додаткової заробітної плати керівникам відділів, лабораторій, секторів і груп, науковим, інженерно-технічним працівникам, конструкторам, технологам, креслярам, копіювальникам, лаборантам, робітникам, студентам, аспірантам та іншим працівникам, безпосередньо зайнятим виконанням конкретної теми, обчисленої за посадовими окладами, відрядними розцінками, тарифними ставками згідно з чинними в організаціях системами оплати праці.

#### Основна заробітна плата дослідників

Витрати на основну заробітну плату дослідників ( $Z_o$ ) розраховуємо у відповідності до посадових окладів працівників, за формулою [26]:

$$Z_o = \sum_{i=1}^k \frac{M_{ni} \cdot t_i}{T_p}, \quad (5.1)$$

де  $k$  – кількість посад дослідників залучених до процесу досліджень;

$M_{ni}$  – місячний посадовий оклад конкретного дослідника, грн;

$t_i$  – число днів роботи конкретного дослідника, дні;

$T_p$  – середнє число робочих днїв в мїсяцї,  $T_p=21$  день.

$$Z_o = 30000,00 \cdot 20 / 21 = 28571,43 \text{ грн.}$$

Проведенї розрахунки зведемо до таблицї.

Таблиця 5.4 – Витрати на заробїтну плату дослїдникїв

Найменування посади	Мїсячний посадовий оклад, грн	Оплата за робочий день, грн	Число днїв роботи	Витрати на заробїтну плату, грн
Керївник проекту	45000	2142,86	66	141428,57
Інженер-розробник програмного забезпечення	40000	1904,76	66	125714,29
Middle 3D-artist	40000	1904,76	66	125714,29
Консультант економічних питань	3 20000	952,3809524	3	2857,14
Всього				395714,29

#### Основна заробїтна плата робїтникїв

Витрати на основну заробїтну плату робїтникїв ( $Z_p$ ) за вїдповїдними найменуваннями робїт НДР розраховуємо за формулою:

$$Z_p = \sum_{i=1}^n C_i \cdot t_i, \quad (5.2)$$

де  $C_i$  – погодинна тарифна ставка робїтника вїдповїдного розряду, за виконану вїдповїдну роботу, грн/год;

$t_i$  – час роботи робїтника при виконаннї визначеної роботи, год.

Погодинну тарифну ставку робїтника вїдповїдного розряду  $C_i$  можна визначити за формулою:

$$C_i = \frac{M_M \cdot K_i \cdot K_c}{T_p \cdot t_{zm}}, \quad (5.3)$$

де  $M_M$  – розмір прожиткового мїнімуму працездатної особи, або мїнімальної мїсячної заробїтної плати (в залежностї вїд дїючого законодавства), приймемо  $M_M=6700,00$  грн;

$K_i$  – коефїцієнт мїжквалїфікаційного спїввїдношення для встановлення тарифної ставки робїтнику вїдповїдного розряду (табл. Б.2, додаток Б) [26];

$K_c$  – мінімальний коефіцієнт співвідношень місячних тарифних ставок робітників першого розряду з нормальними умовами праці виробничих об'єднань і підприємств до законодавчо встановленого розміру мінімальної заробітної плати.

$T_p$  – середнє число робочих днів в місяці, приблизно  $T_p = 21$  дні;

$t_{зм}$  – тривалість зміни, год.

$$C_1 = 6700,00 \cdot 1,10 \cdot 1,65 / (21 \cdot 8) = 72,38 \text{ грн.}$$

$$З_{р1} = 72,38 \cdot 6,00 = 434,3 \text{ грн.}$$

Таблиця 5.5 – Величина витрат на основну заробітну плату робітників

Найменування робіт	Тривалість роботи, год	Розряд роботи	Тарифний коефіцієнт	Погодинна тарифна ставка, грн	Величина оплати на робітника грн
Установка електронно-обчислювального обладнання	6	2	1,1	72,38	434,30
Підготовка робочого місця дослідника	4	2	1,1	72,38	289,54
Інсталяція програмного забезпечення	2,2	5	1,7	111,87	246,11
Тестування системи	1	2	1,1	72,38	72,38
Всього					1042,33

Додаткова заробітна плата дослідників та робітників

Додаткову заробітну плату розраховуємо як 10 ... 12% від суми основної заробітної плати дослідників та робітників за формулою:

$$З_{дод} = (З_o + З_p) \cdot \frac{H_{дод}}{100\%}, \quad (5.4)$$

де  $H_{дод}$  – норма нарахування додаткової заробітної плати. Прийmemo 12%.

$$З_{дод} = (395714,29 + 1042,33) \cdot 12 / 100\% = 47610,79 \text{ грн.}$$

### 5.2.2 Відрахування на соціальні заходи



Нарахування на заробітну плату дослідників та робітників розраховуємо як 22% від суми основної та додаткової заробітної плати дослідників і робітників за формулою:

$$Z_n = (Z_o + Z_p + Z_{доо}) \cdot \frac{H_{zn}}{100\%} \quad (5.5)$$

де  $H_{zn}$  – норма нарахування на заробітну плату. Приймаємо 22%.

$$Z_n = (395714,29 + 1042,33 + 47610,79) \cdot 22 / 100\% = 97760,83 \text{ грн.}$$

### 5.2.3 Сировина та матеріали

До статті «Сировина та матеріали» належать витрати на сировину, основні та допоміжні матеріали, інструменти, пристрої та інші засоби і предмети праці, які придбані у сторонніх підприємств, установ і організацій та витрачені на проведення досліджень.

Витрати на матеріали ( $M$ ), у вартісному вираженні розраховуються окремо по кожному виду матеріалів за формулою:

$$M = \sum_{j=1}^n H_j \cdot C_j \cdot K_j - \sum_{j=1}^n B_j \cdot C_{ej}, \quad (5.6)$$

де  $H_j$  – норма витрат матеріалу  $j$ -го найменування, кг;

$n$  – кількість видів матеріалів;

$C_j$  – вартість матеріалу  $j$ -го найменування, грн/кг;

$K_j$  – коефіцієнт транспортних витрат, ( $K_j = 1,1 \dots 1,15$ );

$B_j$  – маса відходів  $j$ -го найменування, кг;

$C_{ej}$  – вартість відходів  $j$ -го найменування, грн/кг.

$$M_1 = 3 \cdot 200,00 \cdot 1,1 - 0,000 \cdot 0,00 = 660,0 \text{ грн.}$$

Проведені розрахунки зведемо до таблиці.

Таблиця 5.6 – Витрати на матеріали

Найменування матеріалу, марка, тип, сорт	Ціна за од, грн	Норма витрат, од	Величина відходів, кг	Ціна відходів, грн/кг	Вартість витраченого матеріалу, грн
Офісний папір	200	3	0	0	660
Папір для записів	110	1	0	0	121

Таблиця 5.6 – Продовження таблиці

Найменування матеріалу, марка, тип, сорт	Ціна за од, грн	Норма витрат, од	Величина відходів, кг	Ціна відходів, грн/кг	Вартість витраченого матеріалу, грн
Канцелярське приладдя (набір офісного працівника)	175	2	0	0	385
Картридж для принтера Canon LBP6500	1100	1	0	0	1210
Flesh-пам'ять	130	1	0	0	143
Всього					2519

#### 5.2.4 Розрахунок витрат на комплектуючі

Витрати на комплектуючі ( $K_6$ ), які використовують при проведенні НДР на тему «Методи та програмні засоби оброблення та синхронізації контенту з елементами візуалізації» відсутні.

#### 5.2.5 Спецустаткування для наукових (експериментальних) робіт

До статті «Спецустаткування для наукових (експериментальних) робіт» належать витрати на виготовлення та придбання спецустаткування необхідного для проведення досліджень, також витрати на їх проектування, виготовлення, транспортування, монтаж та встановлення.

Балансову вартість спецустаткування розраховуємо за формулою:

$$B_{\text{спец}} = \sum_{i=1}^k C_i \cdot C_{\text{пр.}i} \cdot K_i, \quad (5.7)$$

де  $C_i$  – ціна придбання одиниці спецустаткування даного виду, марки, грн;

$C_{\text{пр.}i}$  – кількість одиниць устаткування відповідного найменування, які придбані для проведення досліджень, шт.;

$K_i$  – коефіцієнт, що враховує доставку, монтаж, налагодження устаткування тощо, ( $K_i = 1,10 \dots 1,12$ );

$k$  – кількість найменувань устаткування.

$$B_{\text{спец}} = 550000 \cdot 1 \cdot 1,1 = 60500 \text{ грн.}$$

Отримані результати зведемо до таблиці:

Таблиця 5.7 – Витрати на придбання спецустаткування по кожному виду

Найменування устаткування	Кількість, шт	Ціна за одиницю, грн	Вартість, грн
ASUS S500MC-5114000460 Intel i5-11400/16/512F/NVD3060-12/No (90PF02H1-M00LP0)	2	52000,00	114400
HP Elite 600-G9 TWR, Intel i7-12700, 16GB, F512GB, UMA, кл+м, 3р, W (6U3Y6EA)	1	55000,00	60500
Всього			174900

### 5.2.6 Програмне забезпечення для наукових (експериментальних) робіт

До статті «Програмне забезпечення для наукових (експериментальних) робіт» належать витрати на розробку та придбання спеціальних програмних засобів і програмного забезпечення, (програм, алгоритмів, баз даних) необхідних для проведення досліджень, також витрати на їх проектування, формування та встановлення.

Балансову вартість програмного забезпечення розраховуємо за формулою:

$$B_{\text{прог}} = \sum_{i=1}^k C_{\text{прог}} \cdot C_{\text{прог.і}} \cdot K_i, \quad (5.8)$$

де  $C_{\text{прог}}$  – ціна придбання одиниці програмного засобу даного виду, грн;

$C_{\text{прог.і}}$  – кількість одиниць програмного забезпечення відповідного найменування, які придбані для проведення досліджень, шт.;

$K_i$  – коефіцієнт, що враховує інсталяцію, налагодження програмного засобу тощо, ( $K_i = 1,10 \dots 1,12$ );

$k$  – кількість найменувань програмних засобів.

$$V_{прз} = 42000,00 \cdot 1 \cdot 1,1 = 47040 \text{ грн.}$$

Отримані результати зведемо до таблиці:

Таблиця 5.8 – Витрати на придбання програмних засобів по кожному виду

Найменування програмного засобу	Кількість , шт	Ціна за одиницю , грн	Вартість , грн
Autodesk Maya	1	42000	47040
Всього			47040

### 5.2.7 Амортизація обладнання, програмних засобів та приміщень

В спрощеному вигляді амортизаційні відрахування по кожному виду обладнання, приміщень та програмному забезпеченню тощо, розраховуємо з використанням прямолінійного методу амортизації за формулою:

$$A_{обл} = \frac{Ц_{б}}{T_{в}} \cdot \frac{t_{вик}}{12}, \quad (5.9)$$

де  $Ц_{б}$  – балансова вартість обладнання, програмних засобів, приміщень тощо, які використовувались для проведення досліджень, грн;

$t_{вик}$  – термін використання обладнання, програмних засобів, приміщень під час досліджень, місяців;

$T_{в}$  – строк корисного використання обладнання, програмних засобів, приміщень тощо, років.

$$A_{обл} = (55000,00 \cdot 3) / (3 \cdot 12) = 4583,33 \text{ грн.}$$

Проведені розрахунки зведемо до таблиці.

Таблиця 5.9 – Амортизаційні відрахування по кожному виду обладнання

Найменування обладнання	Балансова вартість, грн	Строк корисного використання, років	Термін використання обладнання, місяців	Амортизаційні відрахування, грн
ASUS S500MC-5114000460 Intel i5-11400/16/512F/NVD3060-12/No (90PF02H1-M00LP0)	104000 (2*52000)	3	3	8666,67
HP Elite 600-G9 TWR, Intel i7-12700, 16GB, F512GB, UMA, кл+м, 3р, W (6U3Y6EA)	55000	3	3	4583,33
Робоче місце дослідника	10000	5	3	500,00
Оргтехніка	6000	4	3	429,69
Приміщення лабораторії	212000	20	3	375,00
ОС Windows 11	36000 (12000*3)	2	3	2650,00
Autodesk Maya	42000	2	3	4500,00
Прикладний пакет Microsoft Office 2019	15690 (3*5230)	2	3	5250,00
<b>Всього</b>				<b>28915,94</b>

### 5.2.8 Паливо та енергія для науково-виробничих цілей

Витрати на силову електроенергію ( $B_e$ ) розраховуємо за формулою:

$$B_e = \sum_{i=1}^n \frac{W_{yi} \cdot t_i \cdot C_e \cdot K_{\epsilon ni}}{\eta_i}, \quad (5.10)$$

де  $W_{yi}$  – встановлена потужність обладнання на визначеному етапі розробки, кВт;

$t_i$  – тривалість роботи обладнання на етапі дослідження, год;

$C_e$  – вартість 1 кВт-години електроенергії, грн; (вартість електроенергії визначається за даними енергопостачальної компанії), прийmemo  $C_e = 7,50$  грн;

$K_{eni}$  – коефіцієнт, що враховує використання потужності,  $K_{eni} < 1$ ;

$\eta_i$  – коефіцієнт корисної дії обладнання,  $\eta_i < 1$ .

$$B_e = 0,6 \cdot 528,0 \cdot 7,50 \cdot 0,95 / 0,97 = 2327,0 \text{ грн.}$$

Проведені розрахунки зведемо до таблиці.

Таблиця 5.10 – Витрати на електроенергію

Найменування обладнання	Встановлена потужність, кВт	Тривалість роботи, год	Сума, грн
ASUS S500MC-5114000460 Intel i5-11400/16/512F/NVD3060-12/No (90PF02H1-M00LP0),	0,6	528	2327,0
HP Elite 600-G9 TWR, Intel i7-12700, 16GB, F512GB, UMA, кл+м, 3р, W (6U3Y6EA)	0,6	528	2327,0
ASUS S500MC-5114000460 Intel i5-11400/16/512F/NVD3060-12/No (90PF02H1-M00LP0),	0,6	528	2327,0
Робоче місце дослідника	0,15	528	581,75
Оргтехніка	0,45	10	33,05
Всього			7595,84

### 5.2.9 Службові відрядження

До статті «Службові відрядження» дослідної роботи належать витрати на відрядження штатних працівників, працівників організацій, які працюють за договорами цивільно-правового характеру, аспірантів, зайнятих розробленням досліджень, відрядження, пов'язані з проведенням випробувань машин та приладів, а також витрати на відрядження на наукові

з'їзди, конференції, наради, пов'язані з виконанням конкретних досліджень.

Витрати за статтею «Службові відрядження» розраховуємо як 20...25% від суми основної заробітної плати дослідників та робітників за формулою:

$$B_{cv} = (Z_o + Z_p) \cdot \frac{H_{cv}}{100\%}, \quad (5.11)$$

де  $H_{cv}$  – норма нарахування за статтею «Службові відрядження», приймемо  $H_{cv} = 20\%$ .

$$B_{cv} = (395714,29 + 1042,33) \cdot 20 / 100\% = 79351,32 \text{ грн.}$$

5.2.10 Витрати на роботи, які виконують сторонні підприємства, установи і організації

Витрати за статтею «Витрати на роботи, які виконують сторонні підприємства, установи і організації» розраховуємо як 30...45% від суми основної заробітної плати дослідників та робітників за формулою:

$$B_{cn} = (Z_o + Z_p) \cdot \frac{H_{cn}}{100\%}, \quad (5.12)$$

де  $H_{cn}$  – норма нарахування за статтею «Витрати на роботи, які виконують сторонні підприємства, установи і організації», приймемо  $H_{cn} = 30\%$ .

$$B_{cn} = (395714,29 + 1042,33) \cdot 30 / 100\% = 119026,98 \text{ грн.}$$

5.2.11 Інші витрати

До статті «Інші витрати» належать витрати, які не знайшли відображення у зазначених статтях витрат і можуть бути віднесені безпосередньо на собівартість досліджень за прямими ознаками.

Витрати за статтею «Інші витрати» розраховуємо як 50...100% від суми основної заробітної плати дослідників та робітників за формулою:

$$I_e = (Z_o + Z_p) \cdot \frac{H_{ie}}{100\%}, \quad (5.13)$$

де  $H_{ie}$  – норма нарахування за статтею «Інші витрати», приймемо  $H_{ie} = 50\%$ .

$$I_6 = (395714,29 + 1042,33) \cdot 50 / 100\% = 198378,31 \text{ грн.}$$

### 5.2.12 Накладні (загально виробничі) витрати

До статті «Накладні (загально виробничі) витрати» належать: витрати, пов'язані з управлінням організацією; витрати на винахідництво та раціоналізацію; витрати на підготовку (перепідготовку) та навчання кадрів; витрати, пов'язані з набором робочої сили; витрати на оплату послуг банків; витрати, пов'язані з освоєнням виробництва продукції; витрати на науково-технічну інформацію та рекламу та ін.

Витрати за статтею «Накладні (загально виробничі) витрати» розраховуємо як 100...150% від суми основної заробітної плати дослідників та робітників за формулою:

$$B_{нзв} = (Z_o + Z_p) \cdot \frac{H_{нзв}}{100\%}, \quad (5.14)$$

де  $H_{нзв}$  – норма нарахування за статтею «Накладні (загально виробничі) витрати», прийmemo  $H_{нзв} = 100\%$ .

$$B_{нзв} = (395714,29 + 1042,33) \cdot 100 / 100\% = 396\,756,61 \text{ грн.}$$

Витрати на проведення науково-дослідної роботи розраховуємо як суму всіх попередніх статей витрат за формулою:

$$B_{заг} = Z_o + Z_p + Z_{доп} + Z_n + M + K_e + B_{спец} + B_{прз} + A_{обл} + B_e + B_{св} + B_{сп} + I_e + B_{нзв}. \quad (5.15)$$

$$B_{заг} = 395714,29 + 1042,33 + 47610,79 + 97760,82976 + 2519 + 0,00 + 47040 + 174900 + 28915,94 + 7595,84 + 79351,32 + 119026,98 + 198378,31 + 396756,61 = 1596612,24 \text{ грн.}$$

Загальні витрати  $ЗВ$  на завершення науково-дослідної (науково-технічної) роботи та оформлення її результатів розраховується за формулою:

$$ЗВ = \frac{B_{заг}}{\eta}, \quad (5.16)$$

де  $\eta$  - коефіцієнт, який характеризує етап (стадію) виконання науково-дослідної роботи, прийmemo  $\eta = 0,5$ .

$$ЗВ = 1596612,24 / 0,5 = 3193224,48 \text{ грн.}$$



### 5.3 Розрахунок економічної ефективності науково-технічної розробки при її можливій комерціалізації потенційним інвестором

В ринкових умовах узагальнюючим позитивним результатом, що його може отримати потенційний інвестор від можливого впровадження результатів тієї чи іншої науково-технічної розробки, є збільшення у потенційного інвестора величини чистого прибутку.

Результати дослідження проведені за темою «Методи та програмні засоби оброблення та синхронізації контенту з елементами візуалізації» передбачають комерціалізацію протягом 3-х років реалізації на ринку.

В цьому випадку майбутній економічний ефект буде формуватися на основі таких даних:

$\Delta N$  – збільшення кількості споживачів продукту, у періоди часу, що аналізуються, від покращення його певних характеристик;

1-й рік – 1000 користувачів;

2-й рік – 2000 користувачів;

3-й рік – 1500 користувачів.

$N$  – кількість споживачів які використовували аналогічний продукт у році до впровадження результатів нової науково-технічної розробки, прийmemo 35000 користувачів;

$C_o$  – вартість програмного продукту у році до впровадження результатів розробки, прийmemo 550000,00 грн;

$\pm \Delta C_o$  – зміна вартості програмного продукту від впровадження результатів науково-технічної розробки, прийmemo 25000,00 грн.

Можливе збільшення чистого прибутку у потенційного інвестора  $\Delta \Pi_i$  для кожного із 3-х років, протягом яких очікується отримання позитивних результатів від можливого впровадження та комерціалізації науково-технічної розробки, розраховуємо за формулою [26]:

$$\Delta\Pi_i = (\pm\Delta C_o \cdot N + C_o \cdot \Delta N)_i \cdot \lambda \cdot \rho \cdot \left(1 - \frac{\vartheta}{100}\right), \quad (5.17)$$

де  $\lambda$  – коефіцієнт, який враховує сплату потенційним інвестором податку на додану вартість. У 2023 році ставка податку на додану вартість складає 20%, а коефіцієнт  $\lambda = 0,8333$ ;

$\rho$  – коефіцієнт, який враховує рентабельність інноваційного продукту. Прийmemo  $\rho = 30\%$ ;

$\vartheta$  – ставка податку на прибуток, який має сплачувати потенційний інвестор, у 2023 році  $\vartheta = 18\%$ ;

Збільшення чистого прибутку 1-го року:

$$\Delta\Pi_1 = (25000,00 \cdot 35000,00 + 575000,00 \cdot 1000) \cdot 0,83 \cdot 0,3 \cdot (1 - 0,18/100\%) = 297131100 \text{ грн.}$$

Збільшення чистого прибутку 2-го року:

$$\Delta\Pi_2 = (25000,00 \cdot 35000,00 + 575000,00 \cdot (1000 + 2000)) \cdot 0,83 \cdot 0,3 \cdot (1 - 0,18/100\%) = 532786800 \text{ грн.}$$

Збільшення чистого прибутку 3-го року:

$$\Delta\Pi_3 = (25000,00 \cdot 35000,00 + 5750000,00 \cdot (1000 + 2000 + 1500)) \cdot 0,83 \cdot 0,3 \cdot (1 - 0,18/100\%) = 709528575 \text{ грн.}$$

Приведена вартість збільшення всіх чистих прибутків  $ПП$ , що їх може отримати потенційний інвестор від можливого впровадження та комерціалізації науково-технічної розробки:

$$ПП = \sum_{i=1}^T \frac{\Delta\Pi_i}{(1 + \tau)^i}, \quad (5.18)$$

де  $\Delta\Pi_i$  – збільшення чистого прибутку у кожному з років, протягом яких виявляються результати впровадження науково-технічної розробки, грн;

$T$  – період часу, протягом якого очікується отримання позитивних результатів від впровадження та комерціалізації науково-технічної розробки, роки;

$\tau$  – ставка дисконтування, за яку можна взяти щорічний прогнозований рівень інфляції в країні,  $\tau = 0,25$ ;

$t$  – період часу (в роках) від моменту початку впровадження науково-технічної розробки до моменту отримання потенційним інвестором додаткових чистих прибутків у цьому році.

$$ПП = 297131100/(1+0,25)^1 + 532786800/(1+0,25)^2 + 709528575/(1+0,25)^3 = 941967062,40 \text{ грн.}$$

Величина початкових інвестицій  $PV$ , які потенційний інвестор має вкласти для впровадження і комерціалізації науково-технічної розробки:

$$PV = k_{инв} \cdot 3B, \quad (5.19)$$

де  $k_{инв}$  – коефіцієнт, що враховує витрати інвестора на впровадження науково-технічної розробки та її комерціалізацію, приймаємо  $k_{инв} = 5$ ;

$3B$  – загальні витрати на проведення науково-технічної розробки та оформлення її результатів, приймаємо 3193224,48 грн.

$$PV = k_{инв} \cdot 3B = 5 \cdot 3193224,48 = 15966122,41 \text{ грн.}$$

Абсолютний економічний ефект  $E_{абс}$  для потенційного інвестора від можливого впровадження та комерціалізації науково-технічної розробки становитиме:

$$E_{абс} = ПП - PV \quad (5.20)$$

де  $ПП$  – приведена вартість зростання всіх чистих прибутків від можливого впровадження та комерціалізації науково-технічної розробки, 941967062,40 грн;

$PV$  – теперішня вартість початкових інвестицій, 15966122,41грн.

$$E_{абс} = III - PV = 941967062,40 - 15966122,41 = 926000939,99 \text{ грн.}$$

Внутрішня економічна дохідність інвестицій  $E_g$ , які можуть бути вкладені потенційним інвестором у впровадження та комерціалізацію науково-технічної розробки:

$$E_g = \sqrt[T_{жс}]{1 + \frac{E_{абс}}{PV}} - 1, \quad (5.21)$$

де  $E_{абс}$  – абсолютний економічний ефект вкладених інвестицій, 926000939,99грн;

$PV$  – теперішня вартість початкових інвестицій, 15966122,41грн;

$T_{жс}$  – життєвий цикл науково-технічної розробки, тобто час від початку її розробки до закінчення отримання позитивних результатів від її впровадження, 3 роки.

$$E_g = \sqrt[3]{1 + \frac{E_{абс}}{PV}} - 1 = (1 + 926000939,99 / 15966122,41)^{1/3} = 2,89.$$

Мінімальна внутрішня економічна дохідність вкладених інвестицій  $\tau_{мін}$

:

$$\tau_{мін} = d + f, \quad (5.22)$$

де  $d$  – середньозважена ставка за депозитними операціями в комерційних банках; в 2023 році в Україні  $d = 0,11$ ;

$f$  – показник, що характеризує ризикованість вкладення інвестицій, приймемо 0,18.

$\tau_{мін} = 0,11 + 0,18 = 0,29 < 2,89$  свідчить про те, що внутрішня економічна дохідність інвестицій  $E_g$ , які можуть бути вкладені потенційним інвестором у впровадження та комерціалізацію науково-технічної розробки вища

мінімальної внутрішньої дохідності. Тобто інвестувати в науково-дослідну роботу за темою «Методи та програмні засоби оброблення та синхронізації контенту з елементами візуалізації» доцільно.

Період окупності інвестицій  $T_{ок}$  які можуть бути вкладені потенційним інвестором у впровадження та комерціалізацію науково-технічної розробки:

$$T_{ок} = \frac{1}{E_g}, \quad (5.23)$$

де  $E_g$  – внутрішня економічна дохідність вкладених інвестицій.

$$T_{ок} = 1 / 2,89 = 0,35 \text{ року.}$$

$T_{ок} < 3$ -х років, що свідчить про комерційну привабливість науково-технічної розробки і може спонукати потенційного інвестора профінансувати впровадження даної розробки та виведення її на ринок.

## 5.5 Висновки

Згідно проведених досліджень рівень комерційного потенціалу розробки за темою «Методи та програмні засоби оброблення та синхронізації контенту з елементами візуалізації» становить 41 бал, що, свідчить про комерційну важливість проведення даних досліджень (рівень комерційного потенціалу розробки високий).

Також термін окупності становить 0,35 р., що менше 3-х років, що свідчить про комерційну привабливість науково-технічної розробки і може спонукати потенційного інвестора профінансувати впровадження даної розробки та виведення її на ринок.

Отже можна зробити висновок про доцільність проведення науково-дослідної роботи за темою «Методи та програмні засоби оброблення та синхронізації контенту з елементами візуалізації».

## ВИСНОВКИ

У магістерській кваліфікаційній роботі розроблено програмне забезпечення для автоматизації процесу проектування типових анімаційних зображень на платформі анімаційного редактора MAYA.

Основні результати роботи такі:

1. Аналіз методів обробки полігональних моделей анімованих зображень показав, що найбільш прийнятним для подальшого аналізу та використання є метод скелетної анімації, оскільки цей метод характеризується меншими обчислювальними затратами у порівнянні з аналогами, що дає можливість його реалізації чисто програмним способом.

2. Розглянуто особливості реалізації лінгвістичних засобів оброблення та синхронізації контенту з елементами візуалізації, виділено основні особливості міміки обличчя, які потребують реалізації в програмному модулі.

3. Показано, що відомі програмні реалізації фонетичного морфінгу для анімації обличчя вимагають участі оператора на різних етапах створення ліпсингової анімації, тому актуальною є задача автоматизації процесу анімації обличчя.

4. Аналіз тривимірних редакторів показав, що найкращим для реалізації програмного засобу оброблення та синхронізації контенту з елементами візуалізації є тривимірний редактор Maya, оскільки Maya володіє широким спектром інструментів для роботи з анімацією, включаючи роботу з ключовими кадрами, скелетною анімацією, а також мозаїкою фонем та відтворенням емоцій через рухи обличчя.

5. Розроблено метод розбиття слова на фонетичні звуки та метод конвертації цих фонетичних звуків у реалістичні анімаційні рухи обличчя. Цей метод спрощує створення високоякісної лицьової анімації для автоматизації генерації медіаконтенту.

6. Розроблено загальну математичну модель анімаційного зображення, в основі якої лежать принципи фізичної моделювання рухів обличчя та фонетичних звуків. Ця модель сприяє створенню більш точної та емоційно насиченої анімації обличчя на основі текстових файлів.

7. Розроблено загальну структурну схему функціонування програмного засобу,

що визначає ключові етапи та компоненти процесу анімації обличчя на основі текстових файлів.

8. Розроблено інтерактивні анімації обличчя для програмного засобу оброблення та синхронізації контенту з елементами візуалізації та впроваджено інтерактивні можливості для створення живого та реалістичного образу обличчя. Інтеграція анімації очей, виразів обличчя та руху голови додає додатковий рівень емоційної виразності до синтезованого образу.

9. Розроблено алгоритм функціонування програмного засобу оброблення та синхронізації контенту з елементами візуалізації. Алгоритм функціонування програмного засобу забезпечує ефективну обробку та синхронізацію контенту з візуалізацією на основі текстових файлів. Процес розбиття слів на фонетичні звуки і їх подальша конвертація в анімаційні рухи реалізовані із застосуванням математичних моделей та структур.

10. Обґрунтовано вибір мови програмування для реалізації програмного засобу. Вибір мови програмування є критичним аспектом розробки програмного засобу для анімації обличчя на основі текстових файлів. В якості мови програмування обрано мову Python, оскільки ця мова містить велику кількість бібліотек, які орієнтовані на реалізацію функцій анімації та функцій розпізнавання фонетичних звуків.

11. Проаналізовано середовища розробки Maya та Script Editor. Середовища розробки Maya та Script Editor є ключовими компонентами в процесі створення анімації обличчя на основі текстових файлів. Maya, як професійний 3D-редактор, надає розширені можливості для створення та редагування тривимірних сцен, а також інтеграції скриптів та плагінів. Script Editor, з іншого боку, є потужним інструментом для написання, відлагодження та тестування скриптів.

12. Розроблено програмний засіб оброблення та синхронізації контенту з елементами візуалізації, який дозволяє ефективно обробляти тексти з фонетичними звуками, анімувати обличчя та взаємодіяти з тривимірною моделлю обличчя в середовищі Maya.

13. Тестування програмного засобу для відтворення анімації обличчя на основі текстового файлу в середовищі Maya показало його високу ефективність, точність та зручність використання. Програма демонструє стабільну роботу, забезпечуючи

реалістичне відображення виразів обличчя. Вона також відповідає стандартам сумісності та стабільності, що робить її ефективним інструментом для творчих завдань у галузі 3D-графіки.

14. Розроблено інструкцію користувача по встановленню та використанню програмного засобу, яка містить детальний покроковий опис з експлуатації програмного засобу.

15. Розрахунок економічних показників показав високу окупність програмного засобу. Термін окупності складає лише 1 рік.



## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Базалицький М. Р. Програмний модуль оброблення полігональних моделей анімаційних зображень / Тези доповідей XVI міжнародній науково-практичній конференції «Інформаційні технології і автоматизація» – Одеса 2023
2. Базалицький М. Р. Методи та програмні засоби оброблення і синхронізації контенту та відповідного йому зображення / Тези доповідей Міжнародній науково-практичній Інтернет-конференції «Електронні інформаційні ресурси: створення, використання, доступ» – Вінниця 2023.
3. Software [Електронний ресурс] // wikipedia. – 2022. – Режим доступу до ресурсу: <https://en.wikipedia.org/wiki/Software>.
4. Програмний модуль. іт словник укр. [Електронний ресурс]: Режим доступу: [http://xn--r1a3b.xn-b1amgblet.xn-j1amh/index.php/Програмний\\_модуль](http://xn--r1a3b.xn-b1amgblet.xn-j1amh/index.php/Програмний_модуль).
5. 3D graphics [Електронний ресурс] // pcmag. – 2022. – Режим доступу до ресурсу: <https://www.pcmag.com/encyclopedia/term/3d-graphics>.
6. Parent R. Facial Animation [Електронний ресурс] / Rick Parent // Sciencedirect. – 2019. – Режим доступу до ресурсу: <https://www.sciencedirect.com/topics/computer-science/facial-animation>.
7. What is 3D Animation? Pointintimestudios. [Електронний ресурс]: Режим доступу: <https://pointintimestudios.com/what-is-3d-animation/>.
8. Скелетна Анімація. Adobe. [Електронний ресурс]: Режим доступу: <https://helpx.adobe.com/ua/animate/using/character-rigging-in-animate.html>.
9. Метод Understanding Skinning - The Vital Step for Any Rigging Project. pluralsight. [Електронний ресурс]: Режим доступу: <https://www.pluralsight.com/blog/film-games/understanding-skinning-vital-step-rigging-project>.
10. What is Motion Capture? Audiomotion. [Електронний ресурс]: Режим доступу: <https://www.audiomotion.com/blog/what-is-motion-capture.html>.
11. MetaHuman. Unrealengine. [Електронний ресурс]: Режим доступу: <https://www.unrealengine.com/en-US/metahuman-creator>.
12. OMNIVERSE AUDIO2FACE. Nvidia. [Електронний ресурс]: Режим доступу:

<https://www.nvidia.com/ru-ru/omniverse/apps/audio2face/>.

13. Animaze. Animaze. [Электронный ресурс]: URL: <https://www.animaze.us/>.

14. Лінгвістика [Электронный ресурс] // jnsm. – 2022. – Режим доступа до ресурсу: <https://www.jnsm.com.ua/cgibin/u/book/sis.pl?Qry=%CB%B3%ED%E3%E2%B3%F1%F2%E8%EA%E0>.

15. Maya. Autodesk. [Электронный ресурс]: Режим доступа: <https://www.autodesk.com/products/maya/>.

16. advanced skeleton [Электронный ресурс] // Animation Studios. – 2022. – Режим доступа до ресурсу: <https://www.animationstudios.com.au/advanced-skeleton>.

17. Blender [Электронный ресурс] // Blender. – 2023. – Режим доступа до ресурсу: <https://www.blender.org/>.

18. Autodesk 3ds Max [Электронный ресурс] // Autodesk. – 2023. – Режим доступа до ресурсу: <https://www.autodesk.com/products/3ds-max/overview?term=1-YEAR&tab=subscription>.

19. Splits and Mergers. Courses.nus.edu. [Электронный ресурс]: Режим доступа: <https://courses.nus.edu.sg/course/elltankw/history/phon/b.htm>.

20. Precedence Diagram Method. Acqnotes. [Электронный ресурс]: Режим доступа: <https://acqnotes.com/acqnote/tasks/precedence-diagram-method-pdm>.

21. Python. Wikipedia. [Электронный ресурс]: Режим доступа: <https://uk.wikipedia.org/wiki/Python>.

22. MEL [Электронный ресурс] // Autodesk. – 2022. – Режим доступа до ресурсу: <https://help.autodesk.com/view/MAYAUL/2023/ENU/?guid=GUID-60178D44-9990-45B4-8B43-9429D54DF70E>.

23. C Sharp [Электронный ресурс]: Режим доступа: [https://uk.wikipedia.org/wiki/C\\_Sharp](https://uk.wikipedia.org/wiki/C_Sharp).

24. Python-operators. Education-wiki. [Электронный ресурс]: Режим доступа: <https://uk.education-wiki.com/7002017-python-operators>.

25. Nltk. Nltk. [Электронный ресурс]: Режим доступа: <https://www.nltk.org/>.

26. Beginning PyQt: A Hands-on Approach to GUI Programming with PyQt6. – New-York: Apress, 2022. – 543 с. – (Apress). – (PyQt; кн. 1).

27. Методичні вказівки до виконання економічної частини магістерських кваліфікаційних робіт / Уклад. : В. О. Козловський, О. Й. Лесько, В. В. Кавецький. Вінниця : ВНТУ, 2021. 42 с.

28. Кавецький В. В. Економічне обґрунтування інноваційних рішень: практикум / В. В. Кавецький, В. О. Козловський, І. В. Причепя. Вінниця : ВНТУ, 2016. 113 с.


**ДОДАТОК А**

(обов'язковий)

Міністерство освіти і науки України  
Вінницький національний технічний університет  
Факультет інформаційних технологій та комп'ютерної інженерії

ЗАТВЕРДЖУЮ


д.т.н., проф. О. Н. Романюк

 "19" "09" 2023 р.**Технічне завдання**


на магістерську кваліфікаційну роботу «Методи та програмні засоби  
оброблення та синхронізації контенту з елементами візуалізації» за  
спеціальністю

**121 – Інженерія програмного забезпечення**

Керівник магістерської кваліфікаційної роботи:

 д. т. н. Романюк О. Н.  
"19" "09" 2023 р.

Виконав:

 студент гр.ЗПІ-22м Базалицький М. Р.  
"19" "09" 2023 р.

## **1. 1. Найменування та галузь застосування**

Магістерська кваліфікаційна робота: «Методи та програмні засоби оброблення та синхронізації контенту з елементами візуалізації».

Галузь застосування – комп’ютерна графіка.

## **2. Підстава для розробки.**

Підставою для виконання магістерської кваліфікаційної роботи (МКР) є індивідуальне завдання на МКР та наказ № 247 від “18” вересня 2023 року ректора ВНТУ про закріплення тем МКР.

## **3. Мета та призначення розробки.**

Метою роботи є зменшення часу проектування анімацій обличчя тривимірного персонажа за рахунок розробки програмних засобів оброблення та синхронізації контенту з елементами візуалізації.

Призначення роботи – розробка методів та програмних засобів оброблення та синхронізації контенту з елементами візуалізації.

## **4. Вихідні дані для проведення НДР**

Перелік основних літературних джерел, на основі яких буде виконуватись МКР:

1. Nltk. Nltk. [Електронний ресурс]: Режим доступу: <https://www.nltk.org/>.
2. Python-operators. Education-wiki. [Електронний ресурс]: Режим доступу: <https://uk.education-wiki.com/7002017-python-operators>.
3. Maya. Autodesk. [Електронний ресурс]: Режим доступу: <https://www.autodesk.com/products/maya/>.

## **5. Технічні вимоги**

середовище розробки – Maya;

мова програмування –Python;

метод анімації – скелетна анімація;

лінгвістичний словник – словник Карнегі-Меллона.

## 6. Конструктивні вимоги.

Графічна та текстова документація повинна відповідати діючим стандартам України.

## 7. Перелік технічної документації, що пред'являється по закінченню робіт:

- пояснювальна записка до МКР;
- технічне завдання;
- лістинги програми.

## 8. Вимоги до рівня уніфікації та стандартизації

При розробці програмних засобів слід дотримуватися уніфікації і ДСТУ.

## 9. Стадії та етапи розробки:

з/п	Назва етапів магістерської кваліфікаційної роботи	Строк виконання етапів роботи
1	Аналіз стану питання	19.09.23 – 25.09.23
2	Аналіз предметної області	26.09.23 – 28.09.23
3	розробка методу оброблення та синхронізації контенту з елементами візуалізації	29.09.23 – 7.10.23
4	розробка програмних засобів оброблення та синхронізації контенту з елементами візуалізації	8.10.23 – 24.11.23
5	Тестування роботи програмного модулю	25.11.23 – 26.11.23
6	Економічна частина	27.11.23 – 29.11.23
7	Оформлення матеріалів до захисту МКР	24.11.23 – 05.12.23

## 10. Порядок контролю та прийняття.

Виконання етапів магістерської кваліфікаційної роботи контролюється керівником згідно з графіком виконання роботи. Прийняття магістерської кваліфікаційної роботи здійснюється ДЕК, затвердженою зав. кафедрою згідно з графіком.

## ДОДАТОК Б

(Обов'язковий)

## ПРОТОКОЛ ПЕРЕВІРКИ НАВЧАЛЬНОЇ (КВАЛІФІКАЦІЙНОЇ) РОБОТИ

Тема роботи: **Методи та програмні засоби оброблення та синхронізації контенту з елементами візуалізації.**

Тип роботи: кваліфікаційна робота

Відділ : кафедра програмного забезпечення, ФІТКІ, ЗПІ – 22м

Обов'язковий керівник: д.т.н., професор Романюк О. Н.

Unicheck	
Оригінальність	90%
Схожість	10%

## Аналіз звіту подібності

запозичення, виявлені у роботі, оформлені коректно і не містять ознак плагіату.

Виявлені у роботі запозичення не мають ознак плагіату, але їх надмірна кількість викликає сумніви щодо цінності роботи і відсутності самостійності її автора. Роботу направити на редагування.

Виявлені у роботі запозичення є недобросовісними і мають ознаки плагіату та/або в ній зустрічаються навмисні спотворення тексту, що вказують на спроби приховування недобросовісних запозичень.

Підпис, відповідальна за перевірку



Черноволик Г. О.

Висновок прийнятого рішення: допустити до захисту

Знакомлені з повним звітом подібності, який був згенерований системою Unicheck

Автор роботи



Базалицький М. Р.

Керівник роботи



Романюк О. Н.



## ДОДАТОК

(довідниковий)

### Лістинг програми

```
import maya.cmds as cmds
import maya.mel as mel
import os
from PySide2 import QtCore, QtGui, QtWidgets
import sys
from itertools import product as iterprod
import pickle

f = open('D:/dictionary.pkl', 'rb')
arpabet = pickle.load(f)
f.close()
cmu_to_phonem = {"AA": "aaa",
                 "AH": "aaa",
                 "AW": "aaa",
                 "AE": "aaa",
                 "AO": "aaa",
                 "AY": "aaa",
                 "B": "mbp",
                 "CH": "schwa",
                 "D": "lntd",
                 "DH": "lntd",
                 "EH": "eh",
                 "ER": "rrr",
                 "EY": "iee",
                 "F": "fff",
                 "G": "gk",
                 "HH": "schwa",
                 "IH": "iee",
                 "IY": "iee",
                 "JH": "Schwa",
                 "K": "gk",
                 "L": "lntd",
                 "M": "mbp",
                 "N": "lntd",
                 "NG": "lntd",
                 "OY": "ohh",
                 "OW": "ohh",
                 "P": "mbp",
                 "R": "rrr",
                 "S": 'sss',
                 "SH": 'sss',
                 "T": "tth",
                 "TH": "tth",
                 "UH": "uuu",
                 "UW": "uuu",
                 "V": "www",
                 "W": "www",
                 "Y": "iee",
                 "Z": "ssh",
                 "ZH": "ssh"}
```



```

class FaceAnimationClass(object):
    def __init__(self):
        ui=os.path.join(os.path.dirname(__file__),'face_animation.ui')
        self.win= cmds.loadUI(f=ui)
        controls=[x for x in cmds.lsUI(dumpWidgets=True, long=True) if self.win in x]
        for c in controls:
            name= c.split(' ')[-1]
            if name:
                setattr(self,name,c)
        cmds.button(self.dir_btn, e=True,c=self.openDir)
        cmds.button(self.start_btn, e=True, c=self.fileProcessing)
        # cmds.button(self.start_btn, e=True,c=self.face_read)

    def openDir(self,tmp):
        print ('Dir')
        basicFilter = "*.txt"
        self.dir=cmds.fileDialog2(fileFilter=basicFilter, dialogStyle=2)
        print (self.dir[0])
        cmds.textField(self.Dir_line, edit=True, tx=self.dir[0])

    def face_animation(self, phomens):
        face_pose2=phomens
        if face_pose2[0]==",":
            self.last_cadr+=10

        else:
            self.last_cadr+=7

        face_time=[self.last_cadr-5]
        print(face_time)
        cmds.select('ctrlPhonemes_M')
        cmds.setAttr('ctrlPhonemes_M.aaa', 0)
        cmds.setAttr('ctrlPhonemes_M.eh', 0)
        cmds.setAttr('ctrlPhonemes_M.ahh', 0)
        cmds.setAttr('ctrlPhonemes_M.ohh', 0)
        cmds.setAttr('ctrlPhonemes_M.uuu', 0)
        cmds.setAttr('ctrlPhonemes_M.iee', 0)
        cmds.setAttr('ctrlPhonemes_M.rrr', 0)
        cmds.setAttr('ctrlPhonemes_M.www', 0)
        cmds.setAttr('ctrlPhonemes_M.sss', 0)
        cmds.setAttr('ctrlPhonemes_M.fff', 0)
        cmds.setAttr('ctrlPhonemes_M.tth', 0)
        cmds.setAttr('ctrlPhonemes_M.mbp', 0)
        cmds.setAttr('ctrlPhonemes_M.ssh', 0)
        cmds.setAttr('ctrlPhonemes_M.schwa', 0)
        cmds.setAttr('ctrlPhonemes_M.gk', 0)
        cmds.setAttr('ctrlPhonemes_M.lntd', 0)
        cmds.setKeyframe(t = face_time[0])

        #face_time=[0, 12, 24, 36, 48]
        face_time.pop(0)
        for x in face_pose2:
            face_time.append(self.last_cadr)
            self.last_cadr+=5

        print (face_time)
        print (face_pose2)
        size_animation=len(face_time)
        #face_ctrl_pose=[Aaa, Eh, Ahh, Ohh, Uuu, Iee, Rrr, Www, Sss, Fff, Tth, Mbp, Ssh, Schwa, Gk ,Lntd]

```

```

for x in range (size_animation):
    cmds.select('ctrlPhonemes_M')
    if face_pose2[x]=='aaa' or face_pose2[x]=='eh' or face_pose2[x]=='ahh' or face_pose2[x]=='ohh' or
face_pose2[x]=='uuu' or face_pose2[x]=='iee':
        if x>0:
            cmds.setAttr('ctrlPhonemes_M'+ "." +face_pose2[x-1], 0)
            cmds.setKeyframe(t= face_time[x-1]+5)
        else:
            cmds.setAttr('ctrlPhonemes_M'+ "." +face_pose2[x-1], 0)
            cmds.setKeyframe(t= face_time[x-1]+3)
    cmds.setAttr('ctrlPhonemes_M'+ "." +face_pose2[x], 5)
    cmds.setKeyframe(t= face_time[x])

```

```

def wordbreak(tmp, s):
    print (s)
    s = s.lower()
    if s in arpabet:
        return arpabet[s]
    middle = len(s)/2
    partition = sorted(list(range(len(s))), key=lambda x: (x-middle)**2-x)
    for i in partition:
        pre, suf = (s[:i], s[i:])
        if pre in arpabet and self.wordbreak(suf) is not None:
            return [x+y for x,y in iterprod(arpabet[pre], self.wordbreak(suf))]
    return None

```

```

def convert_cmu_to_phonem(tmp, cmu):
    filtered_cmu = ".join(filter(str.isalpha, cmu))
    return cmu_to_phonem[filtered_cmu]

```

```

def fileProcessing(self,tmp):
    dir=cmds.textField( self.Dir_line, text = True, q = True )
    self.last_cadr=-2
    print (dir)
    with open(self.dir[0], 'r') as f:
        for line in f:
            print(line)
            for world in line.split():
                cmu_pronumciation = self.wordbreak(world)[0]
                phomens = [self.convert_cmu_to_phonem(cmu) for cmu in cmu_pronumciation]
                self.face_animation(phomens)
            print(phomens)

```

```

def show(self):
    cmds.showWindow(self.win)

```

```

def show():
    w = FaceAnimationClass()
    w.show()

```

The Natural Language Toolkit (NLTK) is an open source Python library for Natural Language Processing. A free online book is available. (If you use the library for academic research, please cite the book.)

Steven Bird, Ewan Klein, and Edward Loper (2009).

Natural Language Processing with Python. O'Reilly Media Inc.

<https://www.nltk.org/book/>

```

isort:skip_file
"""

import os

# ////////////////////////////////////////////////////////////////////
# Metadata
# ////////////////////////////////////////////////////////////////////

# Version. For each new release, the version number should be updated
# in the file VERSION.

try:
    # If a VERSION file exists, use it!
    version_file = os.path.join(os.path.dirname(__file__), "VERSION")
    with open(version_file) as infile:
        __version__ = infile.read().strip()
except NameError:
    __version__ = "unknown (running code interactively?)"
except OSError as ex:
    __version__ = "unknown (%s)" % ex

if __doc__ is not None: # fix for the ``python -OO``
    __doc__ += "\n@version: " + __version__

# Copyright notice
__copyright__ = """\
Copyright (C) 2001-2023 NLTK Project.

Distributed and Licensed under the Apache License, Version 2.0,
which is included by reference.
"""

```

```

__license__ = "Apache License, Version 2.0"
# Description of the toolkit, keywords, and the project's primary URL.
__longdescr__ = """\
The Natural Language Toolkit (NLTK) is a Python package for
natural language processing. NLTK requires Python 3.7, 3.8, 3.9, 3.10 or 3.11."""
__keywords__ = [
    "NLP",
    "CL",
    "natural language processing",
    "computational linguistics",
    "parsing",
    "tagging",
    "tokenizing",
    "syntax",
    "linguistics",
    "language",
    "natural language",
    "text analytics",
]
__url__ = "https://www.nltk.org/"

# Maintainer, contributors, etc.
__maintainer__ = "NLTK Team"
__maintainer_email__ = "nltk.team@gmail.com"
__author__ = __maintainer__
__author_email__ = __maintainer_email__

# "Trove" classifiers for Python Package Index.
__classifiers__ = [
    "Development Status :: 5 - Production/Stable",
    "Intended Audience :: Developers",
    "Intended Audience :: Education",
    "Intended Audience :: Information Technology",
    "Intended Audience :: Science/Research",
    "License :: OSI Approved :: Apache Software License",
    "Operating System :: OS Independent",
    "Programming Language :: Python :: 3.7",
    "Programming Language :: Python :: 3.8",
    "Programming Language :: Python :: 3.9",

```

```

"Programming Language :: Python :: 3.10",
"Programming Language :: Python :: 3.11",
"Topic :: Scientific/Engineering",
"Topic :: Scientific/Engineering :: Artificial Intelligence",
"Topic :: Scientific/Engineering :: Human Machine Interfaces",
"Topic :: Scientific/Engineering :: Information Analysis",
"Topic :: Text Processing",
"Topic :: Text Processing :: Filters",
"Topic :: Text Processing :: General",
"Topic :: Text Processing :: Indexing",
"Topic :: Text Processing :: Linguistic",
]

```

```

from nltk.internals import config_java

```

```

# support numpy from pypy

```

```

try:

```

```

    import numpypy

```

```

except ImportError:

```

```

    pass

```

```

# Override missing methods on environments where it cannot be used like GAE.

```

```

import subprocess

```

```

if not hasattr(subprocess, "PIPE"):

```

```

    def _fake_PIPE(*args, **kwargs):

```

```

        raise NotImplementedError("subprocess.PIPE is not supported.")

```

```

    subprocess.PIPE = _fake_PIPE

```

```

if not hasattr(subprocess, "Popen"):

```

```

    def _fake_Popen(*args, **kwargs):

```

```

        raise NotImplementedError("subprocess.Popen is not supported.")

```

```

    subprocess.Popen = _fake_Popen

```

```

# Import top-level functionality into top-level namespace

```

```
from nltk.collocations import *
from nltk.decorators import decorator, memoize
from nltk.featsstruct import *
from nltk.grammar import *
from nltk.probability import *
from nltk.text import *
from nltk.util import *
from nltk.jsontags import *
from nltk.chunk import *
from nltk.classify import *
from nltk.inference import *
from nltk.metrics import *
from nltk.parse import *
from nltk.tag import *
from nltk.tokenize import *
from nltk.translate import *
from nltk.tree import *
from nltk.sem import *
from nltk.stem import *

# Packages which can be lazily imported
# (a) we don't import *
# (b) they're slow to import or have run-time dependencies
# that can safely fail at run time

from nltk import lazyimport

app = lazyimport.LazyModule("app", locals(), globals())
chat = lazyimport.LazyModule("chat", locals(), globals())
corpus = lazyimport.LazyModule("corpus", locals(), globals())
draw = lazyimport.LazyModule("draw", locals(), globals())
toolbox = lazyimport.LazyModule("toolbox", locals(), globals())

# Optional loading

try:
    import numpy
except ImportError:
    pass
else:
```

```

from nltk import cluster

from nltk.downloader import download, download_shell

try:
    import tkinter
except ImportError:
    pass
else:
    try:
        from nltk.downloader import download_gui
    except RuntimeError as e:
        import warnings

        warnings.warn(
            "Corpus downloader GUI not loaded "
            "(RuntimeError during import: %s)" % str(e)
        )

# explicitly import all top-level modules (ensuring
# they override the same names inadvertently imported
# from a subpackage)

from nltk import ccg, chunk, classify, collocations
from nltk import data, featstruct, grammar, help, inference, metrics
from nltk import misc, parse, probability, sem, stem, wsd
from nltk import tag, tbl, text, tokenize, translate, tree, util

# FIXME: override any accidentally imported demo, see https://github.com/nltk/nltk/issues/2116
[docs]def demo():
print("To run the demo code for a module, type nltk.module.demo()")
import codecs

try:
    codecs.lookup("mbscs")
except LookupError:
    ascii = codecs.lookup("ascii")
    func = lambda name, enc=ascii: {True: enc}.get(name == "mbscs")
    codecs.register(func)

```

```

import os

# Use the VERSION file to get NLTK version
version_file = os.path.join(os.path.dirname(__file__), "nltk", "VERSION")
with open(version_file) as fh:
    nltk_version = fh.read().strip()

# setuptools
from setuptools import find_packages, setup

# Specify groups of optional dependencies
extras_require = {
    "machine_learning": [
        "numpy",
        "python-crfsuite",
        "scikit-learn",
        "scipy",
    ],
    "plot": ["matplotlib"],
    "tgrep": ["pyparsing"],
    "twitter": ["twython"],
    "corenlp": ["requests"],
}

# Add a group made up of all optional dependencies
extras_require["all"] = {
    package for group in extras_require.values() for package in group
}

# Adds CLI commands
console_scripts = """
[console_scripts]
nltk=nltk.cli:cli
"""

_project_homepage = "https://www.nltk.org/"

setup(
    name="nltk",

```



```

description="Natural Language Toolkit",
version=nlk_version,
url=_project_homepage,
project_urls={
    "Documentation": _project_homepage,
    "Source Code": "https://github.com/nltk/nltk",
    "Issue Tracker": "https://github.com/nltk/nltk/issues",
},
long_description=""

```

The Natural Language Toolkit (NLTK) is a Python package for natural language processing. NLTK requires Python 3.7, 3.8, 3.9, 3.10 or 3.11."

```

license="Apache License, Version 2.0",
keywords=[
    "NLP",
    "CL",
    "natural language processing",
    "computational linguistics",
    "parsing",
    "tagging",
    "tokenizing",
    "syntax",
    "linguistics",
    "language",
    "natural language",
    "text analytics",
],
maintainer="NLTK Team",
maintainer_email="nltk.team@gmail.com",
author="NLTK Team",
author_email="nltk.team@gmail.com",
classifiers=[
    "Development Status :: 5 - Production/Stable",
    "Intended Audience :: Developers",
    "Intended Audience :: Education",
    "Intended Audience :: Information Technology",
    "Intended Audience :: Science/Research",
    "License :: OSI Approved :: Apache Software License",
    "Operating System :: OS Independent",
    "Programming Language :: Python :: 3.7",
    "Programming Language :: Python :: 3.8",

```

```
"Programming Language :: Python :: 3.9",
"Programming Language :: Python :: 3.10",
"Programming Language :: Python :: 3.11",
"Topic :: Scientific/Engineering",
"Topic :: Scientific/Engineering :: Artificial Intelligence",
"Topic :: Scientific/Engineering :: Human Machine Interfaces",
"Topic :: Scientific/Engineering :: Information Analysis",
"Topic :: Text Processing",
"Topic :: Text Processing :: Filters",
"Topic :: Text Processing :: General",
"Topic :: Text Processing :: Indexing",
"Topic :: Text Processing :: Linguistic",
],
package_data={"nltk": ["test/*.doctest", "VERSION"]},
python_requires=">=3.7",
install_requires=[
    "click",
    "joblib",
    "regex>=2021.8.3",
    "tqdm",
],
```

**ДОДАТОК Г**  
**(обов'язковий)**

**ІЛЮСТРАТИВНА ЧАСТИНА**

**МЕТОДИ ТА ПРОГРАМНІ ЗАСОБИ ОБРОБЛЕННЯ ТА СИНХРОНІЗАЦІЇ  
КОНТЕНТУ З ЕЛЕМЕНТАМИ ВІЗУАЛІЗАЦІЇ**

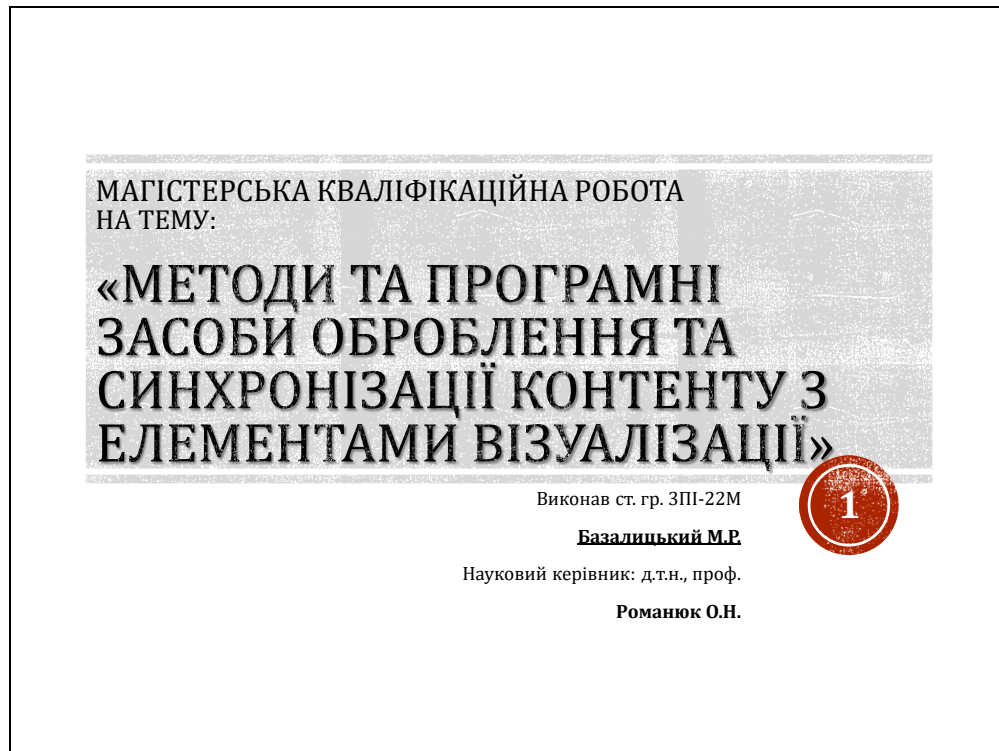


Рисунок Г.1 – Слайд презентації №1

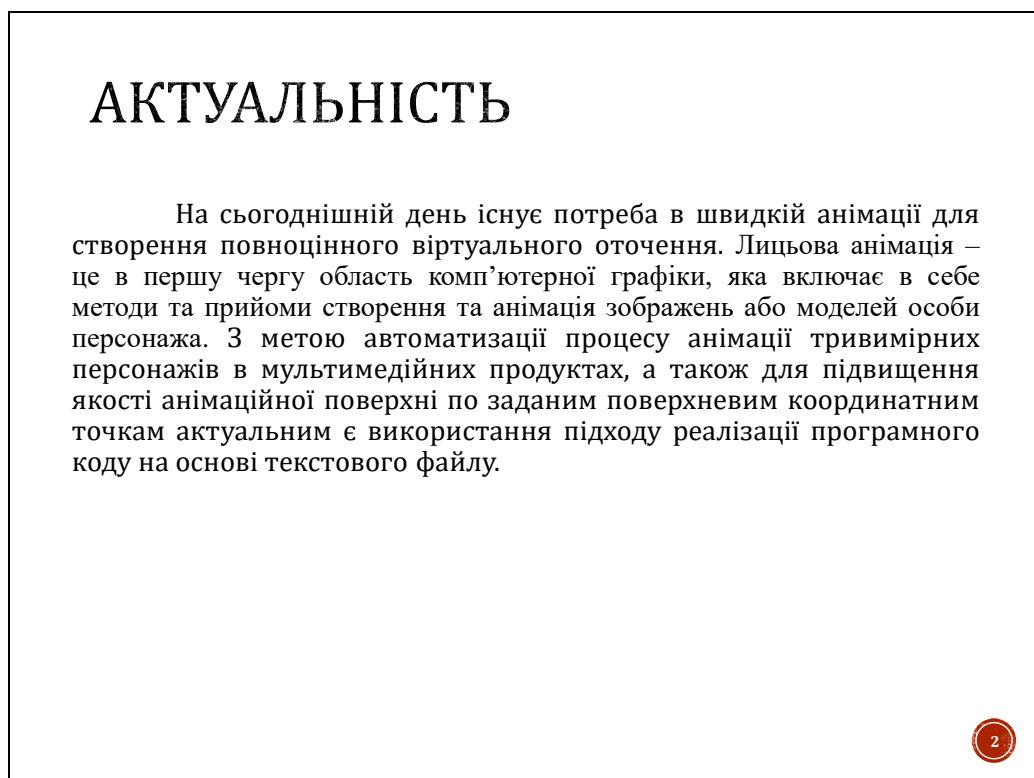


Рисунок Г.2 – Слайд презентації №2

## МЕТА РОЗРОБКИ, ОБ'ЄКТ І ПРЕДМЕТ ДОСЛІДЖЕННЯ

- **Метою** роботи є зменшення часу проектування анімацій обличчя тривимірного персонажа за рахунок розробки програмних засобів оброблення та синхронізації контенту з елементами візуалізації.
- **Об'єкт дослідження** – процес розробки методів і програмних засобів автоматизації створення анімацій обличчя шляхом перенесення інформації з тестового файлу в анімаційний редактор та трансформації її в фонему для відображення динамічних форм вуст.
- **Предмет дослідження** – методи та програмні засоби оброблення та синхронізації контенту з елементами візуалізації.

3

Рисунок Г.3 – Слайд презентації №3

### ЗАДАЧІ НА МАГІСТЕРСЬКОЇ КВАЛІФІКАЦІЙНА РОБОТИ:

- Проаналізувати предметну область розробки анімаційних зображень;
- Обґрунтувати вибір тривимірного редактору для розробки програмного модуля оброблення полігональних моделей анімаційних зображень.
- Розробити методи оброблення та синхронізації контенту і відповідного йому зображення.
- Розробити алгоритм функціонування програмних засобів оброблення синхронізації контенту та відповідного йому зображення.
- Розробити програмний засіб оброблення та синхронізації контенту з елементами візуалізації.
- Провести тестування розробленого програмного засобу оброблення і синхронізації контенту ти відповідного йому зображення.

### Апробація та публікація:

Основні положення магістерської кваліфікаційної роботи доповідалися та обговорювалися на XVI міжнародній науково-практичній конференції «Інформаційні технології і автоматизація» – Одеса 2023, на Міжнародній науково-практичній Інтернет-конференції «Електронні інформаційні ресурси: створення, використання, доступ» – Вінниця 2023.

4

Рисунок Г.4 – Слайд презентації №4

**Наукова новизна одержаних результатів:**

1. Вперше запропоновано математичну модель лицьової анімації, особливістю якої є використання текстових файлів для генерації анімації обличчя тривимірних персонажів, що дозволило скоротити час створення контенту на 75%.

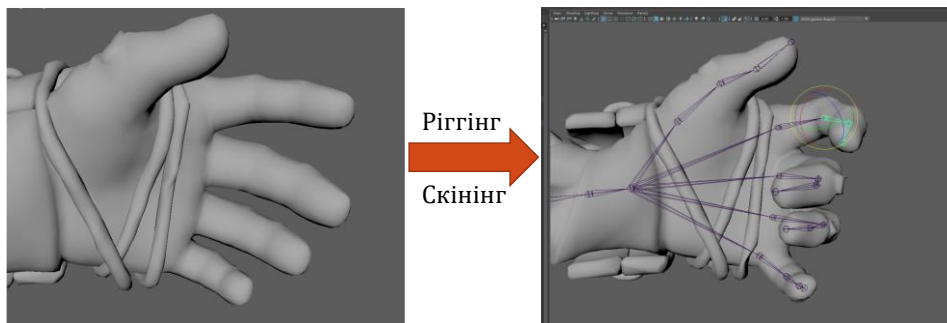
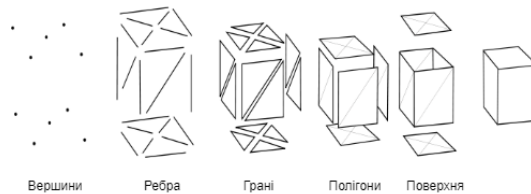
2. Подальшого розвитку отримав метод фонетичного морфінгу для анімації обличчя, у якому на відміну від існуючих використано обробка природної мови NLTK, що дозволило підвищити реалістичність анімації з урахуванням вимови та інтонації слів.

**Практична цінність отриманих результатів.** Практична цінність одержаних результатів полягає в тому, що на основі отриманих в магістерській кваліфікаційній роботі теоретичних положень запропоновано алгоритми та розроблено програмний засіб оброблення та синхронізації контенту з елементами візуалізації, який автоматично створює анімацію обличчя тривимірного персонажу на основі текстового файлу, а також вбудовано цей програмний засіб у тривимірний редактор MAYA.

5

Рисунок Г.5 – Слайд презентації №5

**АНАЛІЗ МЕТОДІВ ТА ЗАСОБІВ ОБРОБЛЕННЯ  
ПОЛІГОНАЛЬНИХ МОДЕЛЕЙ  
АНІМАЦІЙНИХ ЗОБРАЖЕНЬ**

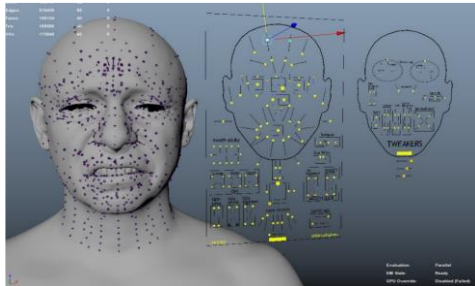


6

Рисунок Г.6 – Слайд презентації №6

## АНАЛОГИ ОБРОБЛЕННЯ ПОЛІГОНАЛЬНИХ МОДЕЛЕЙ АНІМАЦІЙНИХ ЗОБРАЖЕНЬ

MetaHuman



AUDIO2FACE



7

Рисунок Г.7 – Слайд презентації №7

## ЗАГАЛЬНІ МАТЕМАТИЧНІ МОДЕЛІ ОБРОБКИ АНІМАЦІЙНОГО ЗОБРАЖЕННЯ

$$C_x = C_{x-1} + P \cdot W + N,$$

де  $C_x$  – кадр анімації в якій буде ведено форму вуст;  $C_{x-1}$  – останній кадр анімації в який було ведено форму вуст;  $P$  – знак пунктуації. Якщо перед словом був знак «. , » то значення  $P=3$ . Якщо знаку не було то  $P=1$ ;  $W$  – вказує чи є буква початком нового слова. Якщо так то  $W=2$ , якщо ні то  $0$ ;  $N$  – чи фонетичний звук приголосною, якщо так то  $N=3$ , а якщо ні то  $N=5$ .

$$S_x = \frac{S_{x-1}}{E},$$

де  $S_x$  – сила з якою буде відкриватися щелепа;  $S_{x-1}$  – сила з якою відкривалася минула щелепа;  $E$  – індекс зниження сили відкриття щелепи.

Для голосних літер  $S_x$  завжди дорівнює максимальному значінню відкриття щелепи у розмові.

8

Рисунок Г.8 – Слайд презентації №8

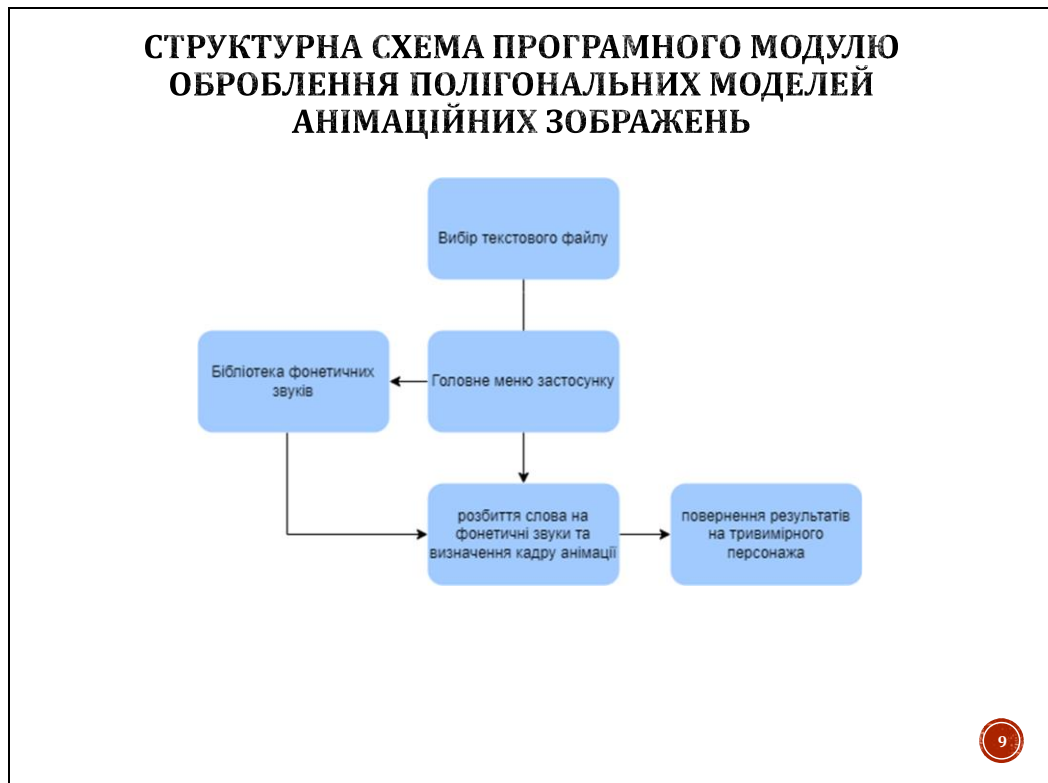


Рисунок Г.9 – Слайд презентації №9

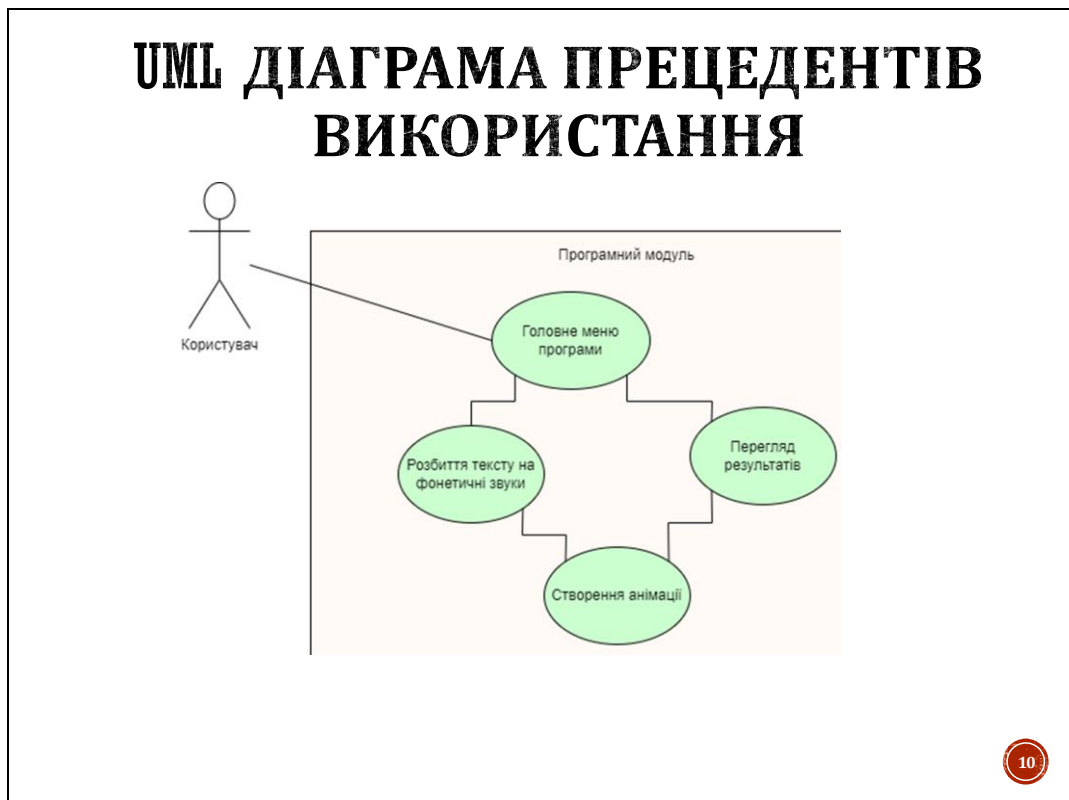


Рисунок Г.10 – Слайд презентації №10



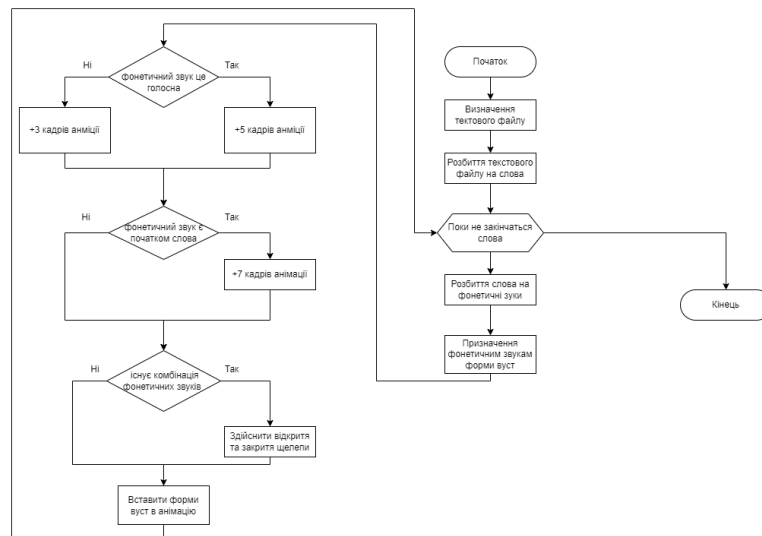
## ПЕРЕЛІК ФОНЕМ

№ з/п	Фонема	Приклад	Переклад	№ з/п	Фонема	Приклад	Переклад	№ з/п	Фонема	Приклад	Переклад	№ з/п	Фонема	Приклад	Переклад
1	AA	odd	AA D	2	AE	at	AE T	23	N	knee	N IY	24	NG	ping	P IH NG
3	AH	hut	HH AH T	4	AO	ought	AO T	25	OW	oat	OW T	26	OY	toy	T OY
5	AW	cow	K AW	6	AY	hide	HH AY D	27	P	pee	P IY	28	R	read	R IY D
7	B	be	B IY	8	CH	cheese	CH IY Z	29	S	sea	S IY	30	SH	she	SH IY
9	D	dee	D IY	10	DH	thee	DH IY	31	T	tea	T IY	32	TH	theta	TH EY T AH
11	EH	Ed	EH D	12	ER	hurt	HH ER T	33	UH	hood	HH UH D	34	UW	two	T UW
13	EY	ate	EY T	14	F	fee	F IY	35	V	vee	V IY	36	W	we	W IY
15	G	green	G R IY N	16	HH	he	HH IY	37	Y	yield	Y IY L D	38	Z	zee	Z IY
17	IH	it	IH T	18	IY	eat	IY T	29	ZH	seizure	S IY ZH ER				
19	JH	gee	JH IY	20	K	key	K IY								
21	L	lee	L IY	22	M	me	M IY								

11

Рисунок Г.11 – Слайд презентації №11

## СХЕМА АЛГОРИТМУ ОБРОБЛЕННЯ ПОЛІГОНАЛЬНИХ МОДЕЛЕЙ АНІМАЦІЙНИХ ЗОБРАЖЕНЬ



12

Рисунок Г.12 – Слайд презентації №12

## У РОЗРОБЦІ БУЛО ЗАСТОСОВАНО



Анімаційний редактор MAYA



середовище для розробки графічних інтерфейсів Qt Designer

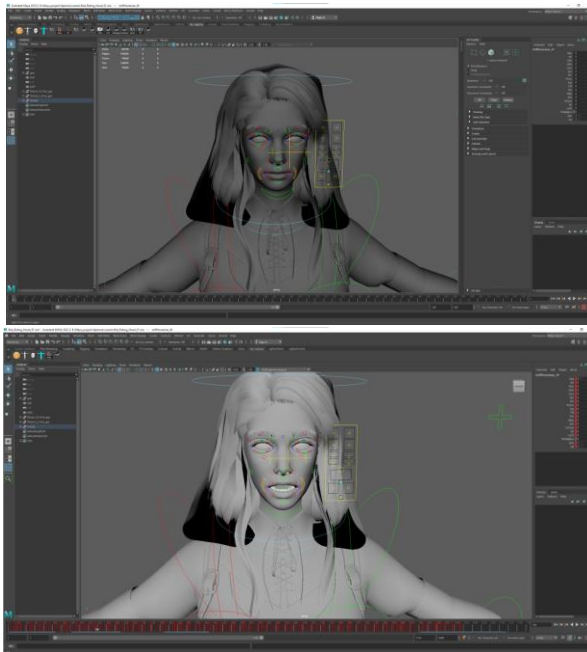


Набір інструментів природної мови NLTK

13

Рисунок Г.13 – Слайд презентації №13

## РЕЗУЛЬТАТ ТЕСТУВАННЯ ПРОГРАМНОГО МОДУЛЮ



```

Glory to Ukraine Glory to the heroes
Glory
[0]
[5, 10, 15, 20, 25]
['gk', 'lntd', 'aaa', 'rrr', 'iee']
['gk', 'lntd', 'aaa', 'rrr', 'iee']
to
[32]
[37, 42]
['tth', 'uuu']
['tth', 'uuu']
Ukraine
[49]
[54, 59, 64, 69, 74, 79]
['iee', 'uuu', 'gk', 'rrr', 'iee', 'lntd']
['iee', 'uuu', 'gk', 'rrr', 'iee', 'lntd']
Glory
[86]
[91, 96, 101, 106, 111]
['gk', 'lntd', 'aaa', 'rrr', 'iee']
['gk', 'lntd', 'aaa', 'rrr', 'iee']
to
[118]
[123, 128]
['tth', 'uuu']
['tth', 'uuu']
the
[135]
[140, 145]
['lntd', 'aaa']
['lntd', 'aaa']
heroes
[152]
[157, 162, 167, 172, 177]
['schwa', 'iee', 'rrr', 'ohh', 'ssh']
['schwa', 'iee', 'rrr', 'ohh', 'ssh']

```

14

Рисунок Г.14 – Слайд презентації №14

## ВИСНОВКИ

1. Аналіз методів обробки полігональних моделей анімованих зображень показав, що найбільш прийнятним для подальшого аналізу та використання є метод скелетної анімації, оскільки цей метод характеризується меншими обчислювальними затратами у порівнянні з аналогами, що дає можливість його реалізації чисто програмним способом.
2. Розглянуто особливості реалізації лінгвістичних засобів оброблення та синхронізації контенту з елементами візуалізації, виділено основні особливості міміки обличчя, які потребують реалізації в програмному модулі.
3. Показано, що відомі програмні реалізації фонетичного морфінгу для анімації обличчя вимагають участі оператора на різних етапах створення ліпсингової анімації, тому актуальною є задача автоматизації процесу анімації обличчя.
4. Аналіз тривимірних редакторів показав, що найкращим для реалізації програмного засобу оброблення та синхронізації контенту з елементами візуалізації є тривимірний редактор Maya, оскільки Maya володіє широким спектром інструментів для роботи з анімацією, включаючи роботу з ключовими кадрами, скелетною анімацією, а також мозаїкою фоном та відтворенням емоцій через рухи обличчя.
5. Розроблено метод розбиття слова на фонетичні звуки та метод конвертації цих фонетичних звуків у реалістичні анімаційні рухи обличчя. Цей метод спрощує створення високоякісної лицьової анімації для автоматизації генерації медіаконтенту.
6. Розроблено загальну математичну модель анімаційного зображення, в основі якої лежать принципи фізичної моделювання рухів обличчя та фонетичних звуків. Ця модель сприяє створенню більш точної та емоційно насиченої анімації обличчя на основі текстових файлів.
7. Розроблено загальну структурну схему функціонування програмного засобу, що визначає ключові етапи та компоненти процесу анімації обличчя на основі текстових файлів.
8. Розроблено інтерактивні анімації обличчя для програмного засобу оброблення та синхронізації контенту з елементами візуалізації та впроваджено інтерактивні можливості для створення живого та реалістичного образу обличчя. Інтеграція анімації очей, виразів обличчя та руху голови додає додатковий рівень емоційної виразності до синтезованого образу.

15

Рисунок Г.15 – Слайд презентації №15

## ВИСНОВКИ

9. Розроблено алгоритм функціонування програмного засобу оброблення та синхронізації контенту з елементами візуалізації. Алгоритм функціонування програмного засобу забезпечує ефективну обробку та синхронізацію контенту з візуалізацією на основі текстових файлів. Процес розбиття слів на фонетичні звуки і їх подальша конвертація в анімаційні рухи реалізовані із застосуванням математичних моделей та структур.
10. Обґрунтовано вибір мови програмування для реалізації програмного засобу. Вибір мови програмування є критичним аспектом розробки програмного засобу для анімації обличчя на основі текстових файлів. В якості мови програмування обрано мову Python, оскільки ця мова містить велику кількість бібліотек, які орієнтовані на реалізацію функцій анімації та функцій розпізнавання фонетичних звуків.
11. Проаналізовано середовища розробки Maya та Script Editor. Середовища розробки Maya та Script Editor є ключовими компонентами в процесі створення анімації обличчя на основі текстових файлів. Maya, як професійний 3D-редактор, надає розширені можливості для створення та редагування тривимірних сцен, а також інтеграції скриптів та плагінів. Script Editor, з іншого боку, є потужним інструментом для написання, відлагодження та тестування скриптів.
12. Розроблено програмний засіб оброблення та синхронізації контенту з елементами візуалізації, який дозволяє ефективно обробляти тексти з фонетичними звуками, анімувати обличчя та взаємодіяти з тривимірною моделлю обличчя в середовищі Maya.
13. Тестування програмного засобу для відтворення анімації обличчя на основі текстового файлу в середовищі Maya показало його високу ефективність, точність та зручність використання. Програма демонструє стабільну роботу, забезпечуючи реалістичне відображення виразів обличчя. Вона також відповідає стандартам сумісності та стабільності, що робить її ефективним інструментом для творчих завдань у галузі 3D-графіки.
14. Розроблено інструкцію користувача по встановленню та використанню програмного засобу, яка містить детальний покроковий опис з експлуатації програмного засобу.
15. Розрахунок економічних показників показав високу окупність програмного засобу. Термін окупності складає лише 1 рік.

16

Рисунок Г.16 – Слайд презентації №16