Муниципальное бюджетное общеобразовательное учреждение «Лицей № 161» городского округа город Уфа Республики Башкортостан

Проектная работа

**«Уникализация видеороликов с помощью python'а»**

по информатике

Выполнили: Чугайнов Глеб, Фатхиев Рузаль.

Класс: 10 ТИ , 10 ТФ.

Научный руководитель: Ардуанова Екатерина Леонидовна.

Уфа, 2024 год

ОГЛАВЛЕНИЕ стр.

ВВЕДЕНИЕ 3

ГЛАВА 1. Применение Python’a для уникализации видеороликов

1.1 Понятие Python’a как языка программирования 5

1.2 Приемы уникализации видеороликов 6

1.3 Собственные идеи по уникализации видео 7

ГЛАВА 2. Процесс создания кода для уникализации

2.1 Шаги к уникальному контенту 8

2.2 Детальный обзор процесса работы 9

ВЫВОДЫ 12

ЗАКЛЮЧЕНИЕ 13

СПИСОК ЛИТЕРАТУРЫ 14

ПРИЛОЖЕНИЕ 15

**Введение**

В наше современное время наблюдается тенденция к постоянному упрощению в различных аспектах жизни. Современный образ жизни также характеризуется тем, что люди предпочитают удобство и эффективность. Например, вместо того чтобы выходить для получения пищи, они предпочитают воспользоваться услугами курьерской доставки, что еще один пример упрощения повседневных процессов.  
 В контексте современных тенденций, стандартные процессы упрощаются, включая создание видеоконтента. Вместо долгих видеороликов на YouTube, предпочтение отдается коротким форматам, таким как TikTok или Shorts. Люди все чаще используют курьерскую доставку вместо посещения магазинов для получения пищи. Наша цель - упростить процесс создания авторского видеоконтента, делая его более доступным и привлекательным для широкой аудитории.

Эффективное создание видео обычно требует наличия высококачественного микрофона, камеры, наушников, гипнотизирующей музыки и привлекательного фона. В свете вышеупомянутого, наш продукт представляет собой комплексное решение, объединяющее указанные компоненты. Результатом использования данного продукта является абсолютно уникальное видеоконтент, создаваемое искусственно обученным роботом. Этот робот обладает способностью обходить алгоритмы, направленные на обеспечение уникальности контента на платформах YouTube и TikTok. Таким образом, пользователи могут безопасно загружать множество роликов, избегая блокировок со стороны платформы или теневых блокировок, которые могут препятствовать распространению и популяризации их видеоматериалов.

Данный проект будет **актуален**, ведь подходит под тренд индустриального обществ: упрощение повседневных дел. Наш проект помогает упростить создание видеороликов для разных платформ. Исследование и разработка методов уникализации видеороликов с использованием Python позволит расширить творческие возможности и сделать контент более привлекательным для аудитории.

**Цель:**

Индивидуальный проект заключается в исследовании и разработке методов уникализации видеороликов с использованием языка программирования Python.

**Задачи:**

1)Рассмотреть различные техники, включая алгоритмы обработки изображений и звука, а также методы машинного обучения, которые позволят создавать уникальные вариации видеороликов.

2)Описать практические примеры применения этих методов и разработаем программный инструмент на Python для упрощения процесса уникализации.  
 3)Разработать программу для уникализации видео.

**Гипотеза:**

Применение методов уникализации с использованием Python может эффективно создавать вариации видеороликов, обеспечивая высокий уровень уникальности, сохранение качества, визуальной привлекательности, увеличение скорости создания множества уникального контента , что делает этот подход перспективным инструментом для индустриального общества.

**Методы исследования:** использование библиотек python’a, изучение алгоритмов площадок видеохостингов, работа с формулами математики, асинхронной многопоточности для быстрой работы с файлами, работа с фотографиями для наложения текста в каждый кадр видео.

**Практическая значимость:** автоматизация создания большого количества видео роликов за счет программного обеспечения, без участия человека.

**Глава 1. Применение Python’a для уникализации видеороликов  
  
1.1 Понятие Python’a как языка программирования**

Python представляет собой интерпретируемый, высокоуровневый язык программирования, разработанный Гвидо ван Россумом в конце 1980-х годов. Язык изначально создавался с упором на читаемость кода и простоту синтаксиса, что содействует быстрой разработке программ. Python обладает широким спектром применений, включая веб-разработку, научные вычисления, автоматизацию, искусственный интеллект и многие другие области.

Python ставит в приоритет **читаемость кода**, что делает его подходящим для начинающих и улучшает совместную разработку. Синтаксис языка обладает простотой и выразительностью. Python интерпретируемый, что означает отсутствие этапа компиляции. Код выполняется построчно интерпретатором Python, что упрощает тестирование и отладку. Python располагает обширной библиотекой стандартных модулей, а также богатым экосистемой сторонних библиотек. Это способствует удобной реализации различных задач без необходимости написания кода с нуля. Язык построен на принципах **объектно-ориентированного** программирования (ООП), что позволяет структурировать код в виде объектов, обладающих свойствами и методами.

Этот язык программирования является самым востребованным и популярным на данный момент. Он используется во многих сферах: веб-разработка, научные вычисления, машинное обучение и искусственный интеллект, сценарии и автоматизация.

Python представляет собой мощный и универсальный язык программирования, который завоевал широкое распространение благодаря своей простоте, читаемости кода и обширной экосистеме. Независимо от области применения, Python остается эффективным инструментом для решения разнообразных задач в области программирования и разработки.

**1.2 Приемы уникализации видеороликов**

Уникализация видеороликов - это процесс изменения или модификации видеоматериала с целью сделать его уникальным или отличающимся от оригинала. Это может включать в себя различные методы редактирования и обработки видеоматериала, чтобы создать новый контент на основе существующего.

Процесс уникализации видео может быть использован в различных областях, таких как создание контента для социальных медиа, рекламы, обучающих материалов и других видов видеопродукции. В некоторых случаях, уникализация может включать в себя технологии и методы искусственного интеллекта для изменения содержания или структуры видеоролика.  
Вот несколько приемов уникализации видеороликов:

1. Изменение скорости воспроизведения;

2. Обрезка и монтаж;

3. Фильтры и эффекты;

4. Добавление текста и графики;

5. Использование эффектов наложения;

6. Создание гиперреального контента;

7. Использование инструментов для смены фона;

8. Ускорение или замедление звука.

**1.3 Собственные идеи по уникализации видео.**

В рамках проведенного эксперимента внедрялась невидимая геометрическая структура на изображение в специфической области кадра, что сопровождалось ухудшением качества видеоматериала. Такое воздействие влекло за собой изменения в процессах обнаружения объектов, влияя тем самым на функциональность алгоритмов восприятия изображения. В результате происходило легкое, но заметное уникализирование видео-контента. Однако, следует отметить, что в современные периоды времени, когда алгоритмы, такие как алгоритм YouTube, претерпевают существенные изменения и становятся более продвинутыми в применении искусственного интеллекта, представленный метод внесения невидимых структур оказывается недостаточно эффективным. Наш алгоритм, разработанный с использованием передовых методов, успешно преодолевает блокировки.  
  
Внесение дополнительного текстового контента, который динамически изменяет свое местоположение в течение определенного времени на видео, представляет собой эффективный механизм для повышения степени уникальности видеоматериала. Данный метод обеспечивает создание визуальных изменений, которые могут служить средством обхода алгоритмов обнаружения контента, применяемых платформой YouTube. Несмотря на то, что внедряемый текст может оставаться статическим, его изменение в расположении и цвете обеспечивает эффект, при котором алгоритмы YouTube классифицируют данный контент как уникальное видео. Таким образом, несмотря на повторяемость текста, изменения в его визуальных атрибутах обеспечивают восприятие как нового и оригинального контента, что является эффективным методом обхода современных алгоритмических механизмов платформы.

**Глава 2. Процесс создания кода для уникализации**

**2.1 Шаги к Уникальному Контенту**

Ход работы включал следующие этапы:

1) Проведено исследование существующих мультимедийных приложений, предназначенных для обработки видео. Проанализированы их функциональные возможности и методы воздействия на видеоматериал.

2) Осуществлено изучение основ языка программирования Python в версии 3.12.0. Параллельно было проведено основательное исследование библиотек, предназначенных для обработки видеоряда (cv2, moviepy), обработки звука (pydub), а также взаимодействия с файловой системой (os) и управления процессами (shutil).

3) Следующим шагом стало приступление к реализации робота с использованием заранее разработанного плана. Робот был спроектирован для автоматизации процесса создания уникальных видеороликов. Этот процесс включал в себя применение предварительно разработанных алгоритмов, направленных на генерацию контента с высокой степенью индивидуализации.

Для разработки программного обеспечения был проведен анализ видеоредакторов и методов их взаимодействия с видео. В ходе анализа выявлено, что эффекты в видеоредакторах представляют собой математические формулы, включающие графику и движение, цветовые эффекты, фильтры и анимации. На основе полученных данных и после анализа функционала видеоредакторов, было приступлено к разработке программного обеспечения.   
  
 Наш Python-скрипт разработан с целью уникализации коротких видеоматериалов, таких как YouTube Shorts и TikTok. Данный скрипт предоставляется в виде телеграм-бота и работает на платформе Python версии 3.12.0, используя библиотеки для обработки видео (cv2, moviepy), звука (pydub), файловой системы (os) и управления процессами (shutil).  
 Уникальность нашего бота обусловлена новым методом уникализации контента, включающим добавление анимированного текста, уникального аудиофайла и особого алгоритма. Алгоритм предполагает, что один из каждых трех кадров создается низкого качества, в то время как два кадра имеют высокое качество. Эта стратегия позволяет избежать возможных ограничений в виде теневых банов на платформах, таких как YouTube и TikTok.

**2.2 Детальный Обзор Процесса Работы.**

Сам процесс создания робота начинается с написания кода в IDE для Python - Pycharm версии 2023.3.1. Для оптимизации процессора и избежания выходов за рамки приложения была создана виртуальная среда через Docker. С учетом знаний, полученных из книги «Пол Бэрри. Изучаем программирование на Python», был разработан скрипт, который получил название «audio.py».   
 Бот начинает с загрузки видео в формате mp4 через телеграм-бота. После получения видеоролика бот декомпозирует его на 24 кадра в секунду, учитывая психофизиологические особенности человека, способного воспринимать примерно 24 кадра в секунду. Важно отметить, что конкретное число кадров может изменяться в зависимости от динамических характеристик видео. Этот универсальный подход к обработке видео обеспечивает создание уникального контента с учетом различий в длине, частоте кадров, звуке, качестве и размере видеоролика. Текст, появляющийся в случайной точке экрана по осям X и Y, добавляет дополнительную уникальность контенту. Этот текст начинает гармоничные колебания по синусоидной траектории, обеспечивая равномерное движение по обеим осям.

Скрипт предназначен для уникализации аудиодорожек коротких видеоматериалов. Сначала была создана структура папок: «musics\_input» для исходных аудиофайлов и «musics\_output» для выходных файлов. В папку «musics\_input» помещены 20 аудиофайлов музыки длиной 10 секунд каждый. В коде скрипта указано, что выбирается случайный аудиофайл, затем он обрабатывается с использованием библиотеки pydub, сохраняя уникальные характеристики. Таким образом, создается уникальный звук, который способствует избежанию обнаружения системами YouTube и TikTok.  
 Затем был разработан скрипт с наименованием «img\_to\_vid.py», в котором были задействованы библиотеки, такие как OpenCV (cv2) и операционная система (os) для эффективной обработки файлов. Эти библиотеки были выбраны из-за их высокой скорости и удобства при прямой работе с файловой системой. Далее была создана функция, которая создает массив, содержащий пути ко всем кадрам в указанной папке. Затем определен путь для сохранения готового видеоролика, и установлено количество кадров в секунду (30 fps). Кодек для видео был установлен как mp4. Процесс кодирования видео был осуществлен с учетом установленных параметров, таких как выходной путь, кодек, частота кадров, а также высота и ширина, которые были оставлены без изменений. Важно отметить, что видеоролик на данном этапе представляет собой слайд-шоу с частотой кадров 30 в секунду.

Для оптимизации системы был разработан отдельный скрипт для очистки временных файлов в папках. Скрипт использует модули os, random и shutil. Три функции были созданы для удаления лишних файлов, случайного пропуска нескольких кадров и создания ложных кадров с учетом особенностей формата имен файлов. Эти меры помогают обеспечить качественную и уникальную обработку видеоматериалов.   
 Далее в коде осуществляется использование публичного репозитория для взаимодействия с ffmpeg с целью уникализации предварительно созданного видеоматериала с внесением текста. Этот подход направлен на обеспечение дополнительной защиты от обнаружения видеоматериала системой.

На видео применяются невидимые для человеческого глаза фигуры, при этом происходит наклон видео от 0 до 10 градусов и изменение цветовой гаммы на незначительные смещения, не воспринимаемые человеческим глазом, но выявляемые алгоритмами.

Импорт библиотек представляет собой важную часть кода, включающую в себя asyncio, cv2 (OpenCV), numpy, os, shutil, time, moviepy, pydub и другие, предназначенные для обработки видео, работы с изображениями, аудиогенерации и прочих операций.

Определены следующие функции:

1. `format\_timedelta()`: Функция для форматирования длительности времени в строку;

2. `get\_saving\_frames\_durations()`: Получение списка длительностей сохранения кадров из видео;

3. `main()`: Основная функция, выполняющая обработку видео;

Основная часть кода включает следующие шаги:

1. Открытие видеофайла с использованием OpenCV;

2. Получение частоты кадров (FPS) видео;

3. Определение длительностей сохранения кадров в соответствии с установленным FPS;

4. Цикл обработки каждого кадра видео: извлечение кадра, сохранение его в виде отдельного изображения (jpg);

5. Запуск других функций для генерации аудио, очистки рабочей директории и прочих операций;

6. Конкатенация видеофайла и аудиодорожки;

7. Создание нового видеофайла с аудиодорожкой;

Код также включает операции с файловой системой, генерацию аудиофайлов и прочие операции. В конце происходит конкатенация видео и аудио, а также создание нового файла видео при помощи библиотеки moviepy.

**Вывод**

1) В ходе исследования были разработаны и реализованы эффективные алгоритмы уникализации роликов с использованием языка программирования Python. Эти алгоритмы способны эффективно обрабатывать видеоматериалы, обеспечивая высокий уровень уникальности контента.   
2) Был проведен обширный анализ существующих методов обработки видео и аудио в контексте уникализации. Это включало в себя исследование методов распознавания объектов, аудиоанализа и других техник, которые могут быть использованы для создания более качественных и уникальных результатов.   
3) Исследование также затронуло возможности применения методов машинного обучения для улучшения процесса уникализации. Это включало в себя использование нейронных сетей и других технологий для более точного анализа и модификации видео- и аудиоданных.

**Заключение**

В ходе нашего научного исследования по уникализации роликов с использованием Python мы продемонстрировали, что данный инструмент является мощным средством для создания уникального медиаконтента. Разработанные алгоритмы, базирующиеся на языке программирования Python, позволяют эффективно модифицировать видео- и аудиоматериалы, создавая уникальные версии с сохранением качества и содержания.

Важным результатом нашей работы является выявление потенциала Python в области обработки медиаданных. Простота и гибкость этого языка программирования совместно с различными библиотеками для обработки видео (например, OpenCV) и звука (например, librosa) делают его мощным инструментом для решения сложных задач уникализации контента.

Этим исследованием мы подтвердили нашу гипотезу о том, что применение методов уникализации способствует созданию множества видеороликов с высоким уровнем уникальности, сохраняя при этом качество и визуальную привлекательность. Кроме того, мы обнаружили, что использование этих методов также ускоряет процесс создания уникального контента.

Таким образом, результаты нашей работы не только подтверждают эффективность уникализации роликов с применением Python, но и ставят новые задачи и вызовы для дальнейших исследований в этой области, призывая к совершенствованию методов и расширению возможностей для создания уникального и качественного медиаконтента.

**Список литературы**

1. Работа с видеофайлами с помощью Python | Форум информационной безопасности — Codeby.net [Электронный ресурс] - Режим доступа:

https://codeby.net/threads/rabota-s-videofajlami-s-pomoschju-python.80169.

2. Use FFMpeg in Python – Bannerbear [Электронный ресурс] - Режим доступа:

https://www.bannerbear.com/blog/how-to-use-ffmpeg-in-python-with-examples.

3. YouTube: Записки Арбитражника: УНИКАЛИЗАЦИЯ видео для TikTok / Reels / Shorts с помощью FFMPEG! [Электронный ресурс] - Режим доступа:

https://youtu.be/Gx1me7nt7\_Y?si=lNLmsFFS9GI\_2eKV

4. Используйте обработку видео Python - Русские Блоги [Электронный ресурс] - Режим доступа:

https://russianblogs.com/article/18163856547/

**Приложение**

 

  
