ПРАВИТЕЛЬСТВО РОССИЙСКОЙ ФЕДЕРАЦИИ
ФГАОУ ВО НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ
«ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»

Факультет компьютерных наук
Образовательная программа «Прикладная математика и информатика»

**Отчет о программном проекте на тему:**

# Анализ видео-фрагмента интервью с целью получения лучшего изображения лица

**Выполнила:**

студентка группы БПМИ208
Мэн Сыфэй _____ 18.05.2023
(подпись) (дата)

**Принял руководитель проекта:**

Передерин Д.А.
Приглашенный преподаватель, доцент
Школы востоковедения ФМЭиМП НИУ ВШЭ

_____ 15/05/2023
(подпись) (дата)

Москва 2023

# Содержание

# Аннотация

Автоматическое извлечение лучшего изображения лица из видеозаписи является важной и сложной задачей, которая имеет большое практическое значение в жизни. В данном проекте рассматривается алгоритм, который позволяет решить поставленную задачу с помощью нейронной сети и встроенных методов библиотек Dlib и OpenCV.

Алгоритм включает три этапа: детектирование лица при помощи get_frontal_face_detector() из библиотеки Dlib, проверку кадра по таким критериям, как открытость глаз и рта, уровень размытия изображения, направление взгляда и привлекательность лица, определяемая нейронной сетью. Изображения, не проходящие предыдущие этапы, не рассматриваются в системе дальше. По каждому критерию была установлена метрика оценивания, которая вносит положительный или отрицательный вклад в итоговую оценку. Наилучшим изображением считается тот кадр, который набирает наибольший балл. Для оценивания красоты лиц была использована нейронная сеть EfficientNet V2, обученная на датасете SCUT-FBP5500 с функцией потери `SmoothL1Loss`. Качество работы алгоритма было протестировано с помощью 100 видео-фрагментов интервью длительностью в районе пяти минут от студентов Школы Востоковедения НИУ ВШЭ. Результаты показали, что выбранные критерии были эффективны в определении наилучшего изображения лица. Итоговая версия программного обеспечения размещена на сайте.



# Ключевые слова

Детектирование лиц, Поиск ключевых точек лица, Исследование движений глаз, Оценка привлекательности лица, Обработка веб-приложения

# 1 Введение

За последние годы видеоклипы набирают большую популярность среди всех групп населения благодаря их способности быстро, наглядно и эффективно передавать информацию по сравнению с другими источниками информации. С развитием социальных сетей и видео платформ, таких как Youtube, Tiktok и VK, видеоролики становятся доступными для каждого человека и являются неотъемлемой частью жизни.

В связи с взрывающим количеством новых клипов, одной из самых важных задач является создание обложки видеозаписи, потому что качественное превью может существенно влиять на количество просмотров и подписчиков. Большинство миниатюр видеороликов показывают лица, потому что они способны создать эмоциональную связь со зрителями, естественным образом привлекая внимания людей и устанавливая более тесный контакт между автором и аудиторией. Несмотря на доминирующую роль видеороликов в жизни людей, автоматическое извлечение лучшего изображения лица из клипа все еще является сложной задачей.

На данный момент существуют в основном два способа получения обложки видео: люди либо сами загружают заранее созданное превью, либо выбирают лучший кадр из автоматически сгенерированных изображений. Разные видео платформы применяют различные подходы к генерации обложек для видеоклипов. Некоторые из них используют нейронные сети для автоматической фильтрации некачественных кадров, но в связи с разнообразным содержанием видеоконтента, качество изображения лица не является самым важным фактором для выбора миниатюры. Другие серверы предоставляют возможность выбрать превью из набора случайных кадров, однако получение подходящего изображения лица полностью полагается на удачу.

В связи с этими причинами данная работа имеет большую практическую значимость, так как в данный момент нет оптимального веб-приложения, которое позволяет автоматически получить лучшее изображение лица из клипа. Для оценивания подходящего кадра была установлена формула (1):

$$O_{final} = \alpha O_{beauty} + \beta O_{ear} + \gamma O_{gaze} + \theta O_{blur} - \delta O_{mar}, \tag{1}$$

где $O_{beauty}$ - оценка привлекательности лица, $O_{ear}$ - уровень открытости глаз, $O_{gaze}$ - оценка направления взгляда, $O_{blur}$ - степень размытия кадра, $O_{mar}$ - уровень открытости рта, и $\alpha$, $\beta$, $\gamma$, $\theta$, $\delta$ - соответствующие нормировочные коэффициенты.

Цель в задаче определения лучшего изображения лица заключается в максимизации итоговой оценки кадра (2).

$$O_{final} \to max \tag{2}$$

В задаче предсказания привлекательности лица будем минимизировать разницу между истинной оценкой красоты лица и предсказанным баллом (3).

$$\sum_{i=1}^{N} \|y_i - \hat{y}_i\| \| \to min, \tag{3}$$

где $\hat{y}$ - ответ выбранной модели, $y$ - истинное значение привлекательности лица.

Продуктом данного проекта является разработанное веб-приложение, которое принимает на вход видеоролик от пользователя и возвращает лучшее изображение лица из видео. Порядок работы алгоритма изображен в виде блок-схемы на рисунке 1.1.

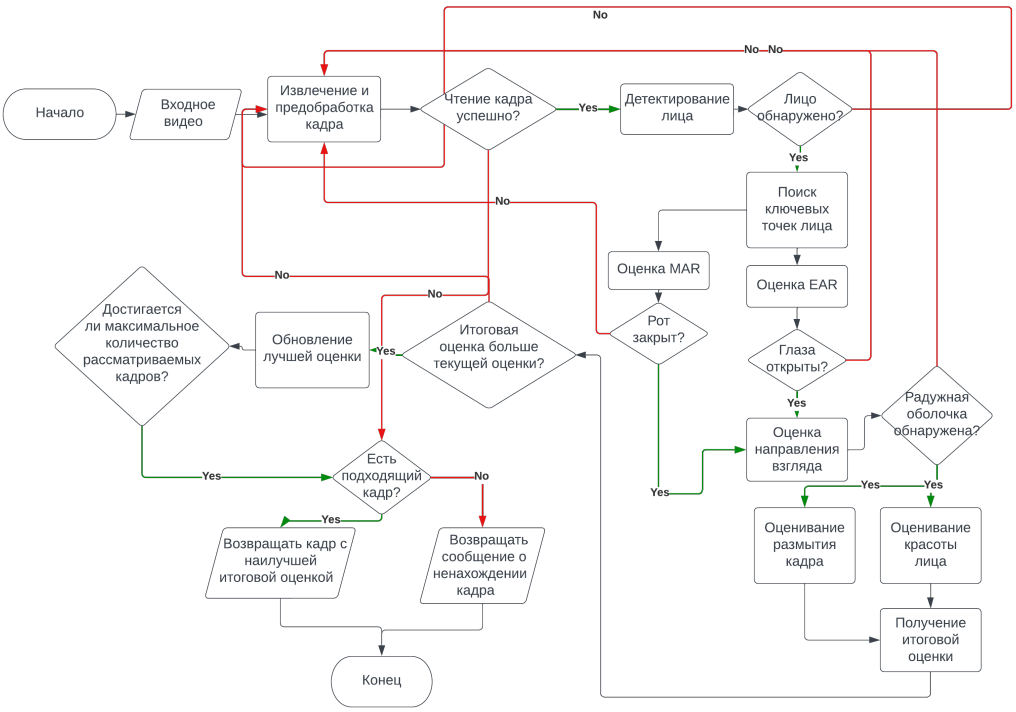



Рис. 1.1: Алгоритм получения лучшего изображения лица из видео

# 2 Обзор похожих программных решений

## 2.1 Подходы выбора обложки

### Получение обложки с помощью нейронной сети

Исследование от команды *YouTube Creator team* показывает [5], что использование

такой нейронной сети, как DNN модель, является эффективным подходом для выбора обложек видео на YouTube. Модель решает задачу бинарной классификации, которая определяет, является ли предложенный кадр качественным изображением. Примеры положительного класса были получены из миниатюр видео с большим количеством просмотров, а обложки плохого качества были взяты случайно из клипа. При загрузке видео на YouTube, система составляет выборку кадров с частотой одно изображение в секунду. Каждый кандидат-кадр оценивается моделью, и система выбирает изображение с наивысшей оценкой.

На практике выяснилось, что обложки, созданные моделью DNN, на 65% предпочтительнее, чем миниатюры, созданные предыдущим алгоритмом для автоматической генерации превью на платформе Youtube. Однако при проведении экспериментов[1] на видеофрагментах интервью было выявлено, что данный алгоритм не уделяет достаточного внимания выражению лица, что приводит к выбору изображения лица с неподходящей эмоцией. Примеры таких кадров изображены на рисунке 2.1. Таким образом, данный подход не решает поставленную задачу.

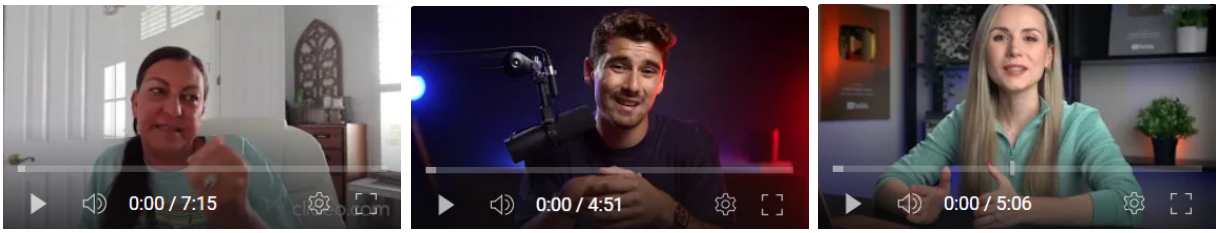



Рис. 2.1: Кадры, сгенерированные с помощью Youtube

**Получение обложки из рандомных кадров**

Другим популярным подходом является генерация случайных кадров. Например, видео платформа Vimeo предоставляет возможность выбрать обложку из набора случайно созданных кадров клипа. Другое программное решение[2] имеет похожую идею: веб-приложение выбирает случайный момент из загруженного видео и создает из него высококачественное JPG-изображение. В отличие от других видео платформ, данное приложение позволяет загрузить сразу несколько видео и скачать сгенерированные изображения, что является большим плюсом среди конкурентов.

Одним из достоинств данного метода является его высокая скорость работы. Однако, в процессе экспериментов на видео-фрагментах интервью было обнаружено, что полученные

[1]Видеоклипы доступны по ссылке: https://www.youtube.com/watch?v=KMWx15H0SFs&t=12s, https://www.youtube.com/watch?v=E1258eprZFM&t=143s, https://www.youtube.com/watch?v=WLQ6HyFbfKU&t=3s

[2]Приложение доступно по ссылке: https://olfu79.github.io/thumbnail-generator/, дата обр. 10.05.2023

кадры могут быть размытыми. Кроме того, алгоритм не делает акцент на выражении лица, что может привести к выбору неподходящего кадра. Примеры таких моментов изображены на рисунке 2.2. Следовательно, данный метод не является эффективным для решения поставленной проблемы.

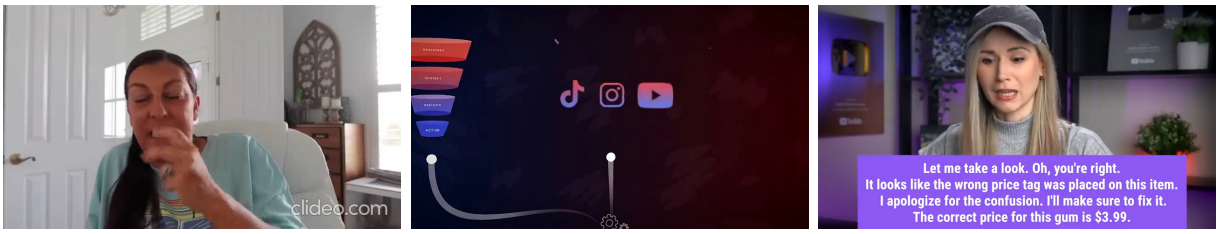



Рис. 2.2: Кадры, сгенерированные с помощью веб-приложения olfu79

## 2.2 Сравнение качества работы двух подходов

Для исследования качества работы алгоритмов, были проведены три эксперимента с видеороликами длительностью 7 минут 15 секунд, 4 минуты 51 секунда и 5 минут 6 секунд соответственно. По результатам выяснилось, что подход с применением нейронной сети создают качественные кадры, а генерация случайных кадров гораздо быстрее. Примеры полученных кадров изображены на рисунках 2.1 и 2.2, сравнение времени выполнения алгоритмов представлено в таблице 2.1.

| | Youtube | Olfu79 |
|---|---|---|
| Video 1 | 89.83 | 1.02 |
| Video 2 | 125.15 | 0.97 |
| Video 3 | 88.17 | 1.13 |

Таблица 2.1: Время выполнения алгоритмов в секундах

# 3 Описание предлагаемого метода

## 3.1 Первый этап: Детектирование лиц

Цель этого шага заключается в фильтрации кадров, не содержащих лиц. Для этого решено использовать детектор лица `get_frontal_face_detector()` из библиотеки Dlib, кототрый основан на HOG и SVM.

HOG, или гистограмма направленных градиентов, представляет собой популярный дескриптор, позволяющий упрощать изображение и извлекать из него полезную информацию.

Идея дескриптора HOG заключается в том, что распределение градиентов пикселей определяет внешний вид объекта, так как на границе наблюдается большое изменение величины градиентов. Чтобы создать дескриптор HOG, изображение сначала разбивается на ячейки, для каждой из которых вычисляются горизонтальные и вертикальные градиенты $g_x, g_y$, их величина $g = \sqrt{g_x^2 + g_y^2}$ и направление $\theta = arctan(\frac{g_y}{g_x})$. После этого ячейки объединяются в блоки, и вычисляется гистограмма градиентов на них. Из нормированных блоков строится финальный вектор признаков для изображения, который используется для обнаружения и распознавания объектов с помощью метода опорных векторов (SVM). Классификатор SVM основан на построении оптимальной гиперплоскости, разделяющей объекты разных классов.

Таким образом, для детектирования лица в кадре был использован метод `get_frontal_face_` из библиотеки Dlib, потому что он быстрее детектирует лица по сравнению с другими способами [3]. После завершения первого этапа все кадры, не содержащие лица, не допускаются для дальнейшей проверки.

## 3.2 Второй этап: Проверка кадра по жестким критериям

### 3.2.1 Обнаружение ключевых точек лица

Нахождение ключевых точек лица осуществляется при помощи метода `shape_predictor` из библиотеки Dlib. Алгоритм на вход принимает область лица, полученную на первом этапе, и возвращает 68 координат ключевых точек, в том числе входят контур лица, верхняя и нижняя губы, левый и правый углы рта, челюсть, левый и правый глаза и т.д. Местоположения всех ключевых точек лица изображены на рисунке 3.1.

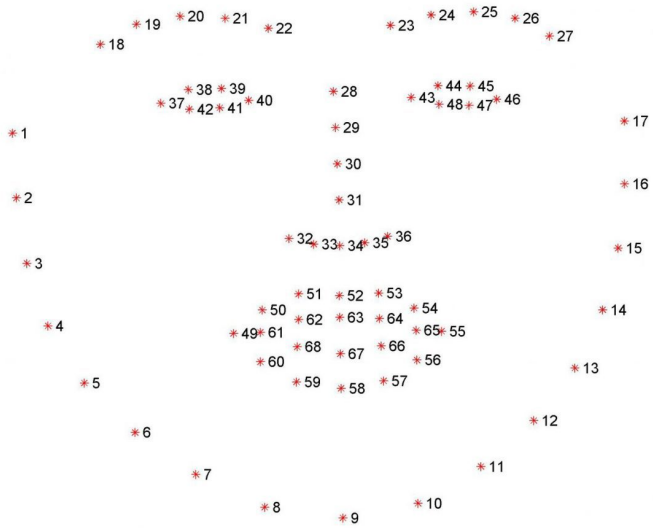



Рис. 3.1: 68 ключевых точек лица

По завершении данного этапа были вычислены координаты ключевых точек лица для анализа степени открытости глаз и рта, что является важным критерием для определения наилучшего изображения лица.

### 3.2.2 Определение открытости глаза

Для оценивания открытости глаза вычисляется соотношение сторон глаз (EAR), которое представляет собой отношение суммы длин вертикальных осей глаза к длине горизонтальной оси и вычисляется по формуле (4):

$$EAR = \frac{\|p_2 - p_6\| + \|p_3 - p_5\|}{2\|p_1 - p_4\|}, \tag{4}$$

где значения $p_1, p_2, p_3, p_4, p_5, p_6$ соответствуют координатам ключевых точек глаз, полученным из `shape_predictor` библиотеки Dlib. Согласно статье [2], EAR для открытых глаз представляет собой устойчивое значение, однако при моргании данное число резко падает ниже 0,25 и приближается к нулю. Примеры значений EAR для открытого и закрытого глаза представлены на рисунке 3.2. Таким образом, вычисление EAR не только фильтрует неподходящие кадры, но и оценивает уровень открытости глаза.

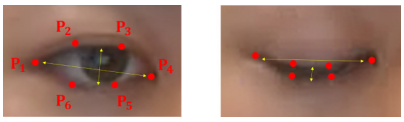



Рис. 3.2: EAR открытого и закрытого глаза

### 3.2.3 Определение открытости рта

Для определения степени открытости рта используется соотношение сторон рта (MAR), которое представляет собой отношение суммы длин вертикальных осей рта к длине горизонтальной оси (5):

$$MAR = \frac{\|p_2 - p_8\| + \|p_3 - p_7\| \|p_4 - p_6\|}{2\|p_1 - p_5\|}, \tag{5}$$

где значения $p_1, p_2, ..., p_8$ взяты из набора ключевых точек лица, полученных на предыдущих этапах. Для исключения влияния размера губ на соотношение, используются координаты внутренней области рта. Иллюстрация соотношения сторон рта показана на рисунке 3.3.

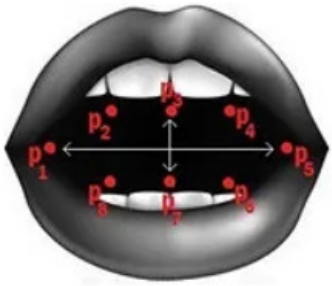



Рис. 3.3: MAR открытого рта

Высокое значение MAR указывает на более широкое открытие рта, что может свидетельствовать о том, что человек говорит в данный момент. Низкое значение MAR соответствует закрытому или сжатому рту, что считается приемленным для получения лучшего изображения лица.

### 3.2.4 Итог второго этапа

После вычисления EAR и MAR для рассматриваемого кадра, проводится их сравнение с пороговыми значениями, определенными на основе экспериментов. Кадры, в которых EAR или MAR ниже пороговых значений, считаются неподходящими и отбрасываются. В результате данного этапа получаются координаты ключевых точек лица и производится фильтрация неподходящих кадров по таким жестким критериям, как открытость глаз и рта.

## 3.3 Третий этап: Проверка кадра по мягким критериям

### 3.3.1 Определение направления взгляда

В связи с двумя причинами было принято решение проверить направление взгляда. Одним из оснований является желание отобрать изображения лица, в которых человек смотрит прямо в камеру. Другая причина заключается в обнаружении некорректного определения координат глаз, полученных при помощи детектора ключевых точек лица, что может привести к завышенному значению EAR. Для решения данной проблемы было решено проверить наличие радужной оболочки глаза, что является первым шагом в определении направления взгляда.

Проверка направления взгляда основана на отношении площадей склеры левой и правой части глаза. Если отношение приближается к единице, то есть площади склеры обеих частей примерно одинаковы, то взгляд направлен прямо вперед. В противном случае можно сделать вывод, что человек смотрит в сторону. Иллюстрация таких примеров показана на рисунке 3.4.

Рис. 3.4: Площадь склеры левой и правой части глаза при разных сторонах взгляда

Таким образом, с помощью определения направления взгляда, кадры, в котором человек смотрит прямо в камеру, имеют более высокую оценку.

### 3.3.2 Оценка степени размытия изображения

В связи с возможным движением головы при рассказе, изображение лица может быть размытым. Такие моменты клипа не соответствуют категории "лучшего кадра поэтому было принято ввести оценку, определяющую степени размытия изображения. Метод основан на вычислении дисперсии лапласиана кадра. Оператор Лапласа используется для выделения границы объекта, так как в таких областях интенсивность пикселей резко меняется. Размытое изображение не имеет четко определенных границ, поэтому лапласиан такого кадра будет примерно одинаковым везде. Следовательно, дисперсия данного значения будет меньше.

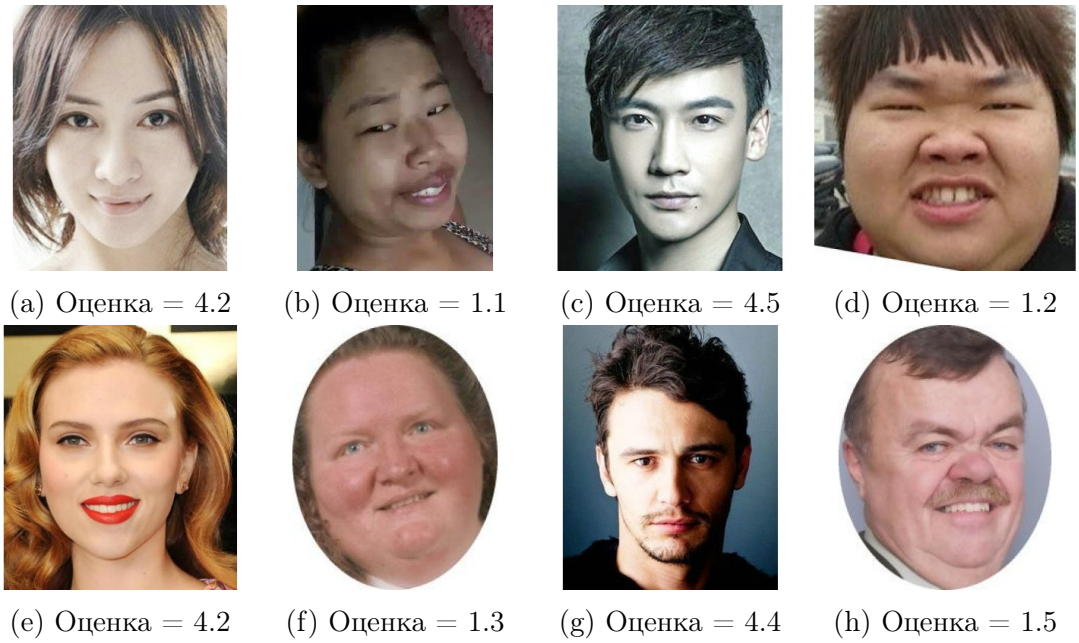
Основными преимуществами данного метода являются высокая скорость проверки и удобство реализации с использованием встроенного метода `cv2.Laplacian()` из библиотеки OpenCV. Возможным недостатком может быть чувствительность метода к шуму изображения, однако это не является критической проблемой, поскольку качество кадров остается стабильным в течение одного и того же видеоклипа.

Одним словом, путем расчета дисперсии лапласиана изображения можно определить степень его размытия.

### 3.3.3 Оценка привлекательности лица

**Описание использованного датасета** Датасет SCUT-FBP5500 [1], предоставленный Южно-Китайским технологическим университетом, является широко известным набором данных для определения привлекательности лиц человека. Набор состоит из 5500 фронтальных изображений лиц в возрасте от 15 до 60 лет и разделен на четыре подмножества с различными расами и полом, включая 2000 азиатских женщин, 2000 азиатских мужчин, 750 европейских женщин и 750 европейских мужчин. Каждое изображение помечено оценкой красоты в диапазоне от 1 до 5, где 5 соответствует наиболее привлекательному лицу, а 1 - наименее красивому. Каждое изображение было оценено 60 добровольцами в возрасте от

18 до 27 лет, и среднее значение всех оценок считалось истинным значением. Оценки, отклоняющиеся от целевого значения более чем на 2, рассматривались как выбросы. Примеры изображений лиц и их оценок представлены на рис. 3.5.



(a) Оценка = 4.2 (b) Оценка = 1.1 (c) Оценка = 4.5 (d) Оценка = 1.2

(e) Оценка = 4.2 (f) Оценка = 1.3 (g) Оценка = 4.4 (h) Оценка = 1.5

Рис. 3.5: Примеры лиц разных рас и пол и их оценки привлекательности

Согласно авторам статьи [1], доля выбросов мала: в подмножестве европейских женщин процент выбросов составляет 0.32% от всех оценок, в подмножестве европейских мужчин - 0.40%, для азиатских женщин - 0.29%, для азиатских мужчин - 0.42%.

**Аугментация изображений**

В связи с маленьким размером датасета было решено применить ряд методов аугментации данных.

Первый метод, названный **RandomRotation**, заключается в случайном повороте изображений на определенный угол для создания новых вариаций данных. В данном исследовании изображения поворачивались на угол от -30 до 30 градусов.

Второй метод, названный **GaussianBlur**, представляет собой операцию размытия изображения с использованием фильтра Гаусса, который уменьшает резкость границ и шумов на изображении. Этот метод принимает два параметра - `kernel_size` и `sigma`, которые определяют размер ядра фильтра и степень размытия соответственно. Чем больше эти параметры, тем более размытым будет полученное изображение. В данном исследовании значения параметров были установлены равными $(5, 5)$ и $(0.1, 5)$. GaussianBlur был выбран для аугментации изображений с целью повышения точности модели на более размытых изображениях.

Третий метод, названный **RandomHorizontalFlip**, основан на случайном отражении изображения по горизонтальной оси. Это позволяет модели обучаться на горизонтально перевернутых изображениях, что улучшает качество нейронной сети.

**Эксперименты с моделями**

Для выбора лучшей модели были проведены сравнения между ResNet18, RegNetY800MF и EfficientNetV2. Датасет был разделен на тренировочную и тестовую выборки в соотношении 60% к 40%. Обучающая выборка содержит 3300 изображений, в то время как тестовая выборка включает в себя 2200 изображений. В качестве метрики качества модели была выбрана средняя абсолютная ошибка (MAE), которая вычисляется по следующей формуле (6):

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|y_i - \hat{y}_i|, \tag{6}$$

где $y_i$ - истинное значение и $\hat{y}_i$ - ответ модели. Поскольку в данных существуют выбросы, в процессе обучения необходимо выбрать такую функцию потерь, которая не слишком чувствительна к аномальным значениям. Именно поэтому была выбрана функция потерь `SmoothL1Loss` (7):

$$l(x, y) = L = (l_1, ..., l_N)^T;$$

$$l_n = \begin{cases} 0.5(x_n - y_n)^2/\beta, & when|x_n - y_n| < \beta \\ |x_n - y_n| - 0.5\beta, & else \end{cases} \tag{7}$$

Для оценки качества модели были использованы не только MAE, но и RMSE и коэффициент Пирсона. Выбор данных метрик обусловлен необходимостью оценки модели с различных точек зрения, а также с целью сопоставления результатов с исследованием, проведенным авторами статьи[1]. В результате проведенного обучения было установлено, что архитектура RegNetY800MF начинает проявлять признаки переобучения после 19 эпохи, что сопровождается низким качеством модели. Аналогичные результаты были получены при обучении архитектуры ResNet18. В итоге, наилучшие показатели продемонстрировала архитектура EfficientNetV2. Сравнение качества моделей доступно в файле `Train.ipynb`, расположенном в репозитории на сайте Github.

**Обучение выбранной модели.**

После сравнения с разными моделями была выбрана EfficientNetV2 в качестве основной архитектуры.

EfficientNetV2 [4] представляет собой новую модель нейронной сети от компании Google для решения задачи в области компьютерного зрения. По сравнению с предыдущими моделями EfficientNetV2 имеет меньше параметров, и поэтому обучается быстрее, но при этом

достигает высокой точности. Графики изменения функции потерь `SmoothL1Loss` и средней абсолютной ошибки изображены на рисунке 3.6.

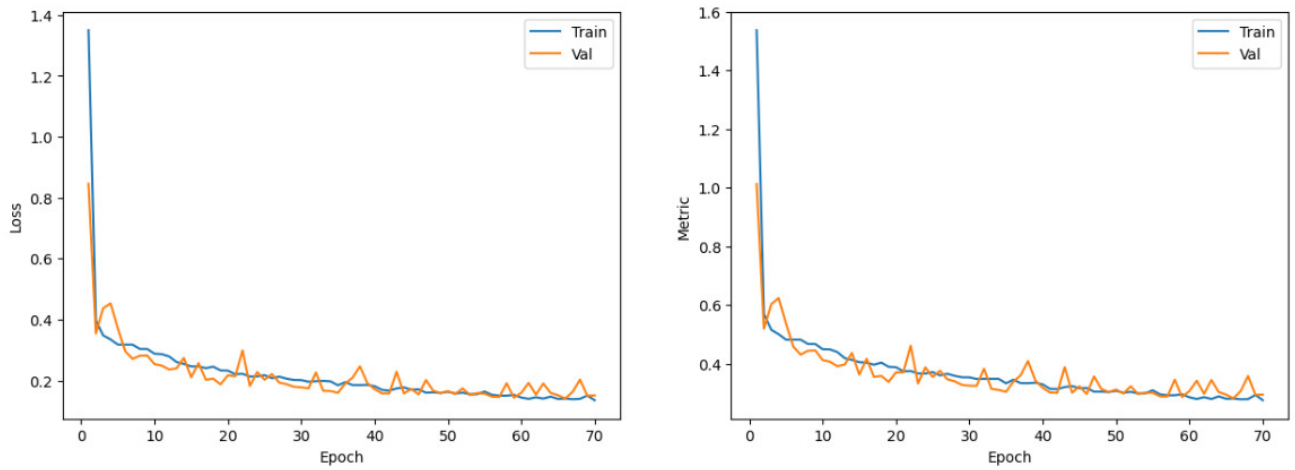



Рис. 3.6: EfficientNetV2

В процессе обучения входные изображения масштабируются таким образом, чтобы малая сторона равнялась 256 пикселей. Затем изображения разрезаются на квадрат размера 224 на 224 с помощью метода `CenterCrop`. Для оптимизации функции потерь используется Адам оптимизатор с длиной шага $10^{-4}$ и коэффициентом L2 регуляризации, равным $10^{-3}$. Модель содержит 20178769 параметров и обучается в течение 70 эпох с размером батча 32. В качестве функции потерь используется `SmoothL1Loss` с параметром `beta = 0.4`, а в качестве метрик качества модели используются `MAE, PC, RMSE` для сравнения результатов с качеством модели, предложенной автором в статье [1].

| | EfficientNetV2 | AlexNet |
|---|---|---|
| PC | 0.8469 | 0.8298 |
| MAE | 0.2927 | 0.2938 |
| RMSE | 0.3782 | 0.3819 |

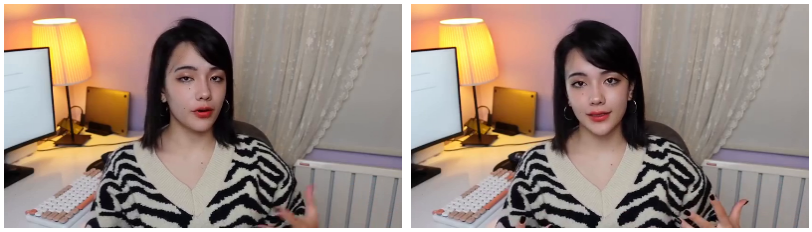
Таблица 3.1: Сравнение качества моделей на тестовой выборке

Результаты сравнения проведенного анализа представлены в таблице 3.1. Выяснилось, что EfficientNetV2 имеет более высокое качество по сравнению AlexNet[3], однако на практике вторая модель работает быстрее. Учитывая, что анализируемые кадры поступают с частотой 3 изображения в секунду, производительность EfficientNetV2 может считаться приемлемой.

**Примеры работы модели предсказывания красоты лица человека**

[3]Обученная модель доступна по ссылке: https://drive.google.com/file/d/1un5CjTz_49Lg6MTNQn99WD7FjFqEJGoY/view, пароль получения файла: 12345, дата обр. 10.05.2023

На рисунке 3.7 представлены оценки привлекательности лиц одного и того же человека с разным лицевым выражением.



(a) Оценка = 3.21 (b) Оценка = 3.67

Рис. 3.7: Сравнение оценки красоты человека с разным выражением лица

Из полученных результатов следует, что оценка привлекательности лица человека зависит не только от структуры лица, но и от его выражения. В связи с этим, оценивание красоты лица человека в данной работе является целесообразным для достижения наилучшего качества изображений.

### 3.3.4 Калибровка оценок

В связи с возможными расхождениями оценок, обусловленными индивидуальными особенностями людей или низким разрешением видео, была реализована процедура калибровки оценок относительно первому соответствующему кадру. Далее, относительные оценки были нормированы в интервале от -1 до 1. Таким образом, баллы по различным условиям оказывают влияние на итоговую оценку в положительном или отрицательном направлении, но при этом аномальные оценки какого-либо критерия не могут исказить баланс итоговой оценки.

### 3.3.5 Итог третьего этапа

Кадры, прошедшие на третий этап, были подвергнуты жестким критериям отбора наилучшего изображения лица. Для ранжирования таких кадров было принято решение использовать оценки по нескольким критериям, включая направление взгляда, уровень привлекательности лица и степень размытия кадра. После проведения оценки по всем критериям, кадр с наивысшим общим баллом считается наилучшим изображением лица в видеоклипе.

# 4 Тестирование и анализ полученных результатов

В рамках оценивания качества работы алгоритма были проверены 100 видеозаписей от студентов Школы Востоковедения НИУ ВШЭ. В каждом клипе присутствует только

один человек, и некоторые видеоклипы записаны в темной комнате с плохим качеством. Все лучшие изображения лиц, которые алгоритм генерировал, можно увидеть по этой ссылке. Можно увидеть, что все кадры соответствуют поставленным критериям: глаза открытые, рот закрытый или слегка приоткрытый, выражение нейтральное.

На рисунке 4.1 представлены некоторые примеры полученных изображений. Кроме того, в таблице 4.1 отображены временные характеристики работы алгоритма на 10 видеозаписях, а также их среднее значение.

Рис. 4.1: Лучшие изображения лиц, полученные алгоритмом

| | Длительность видео | Размер видео | Время выполнения |
|---|---|---|---|
| 1 | 4 мин. 12 сек. | 15.96 MB | 37.38 сек. |
| 2 | 3 мин. 11 сек. | 11.18 MB | 29.86 сек. |
| 3 | 3 мин. 27 сек. | 6.98 MB | 28.49 сек. |
| 4 | 3 мин. 19 сек. | 7.14 MB | 27.45 сек. |
| 5 | 3 мин. 14 сек. | 12.61 MB | 29.55 сек. |
| 6 | 2 мин. 19 сек. | 6.69 MB | 20.45 сек. |
| 7 | 7 мин. 23 сек. | 6.40 MB | 57.13 сек. |
| 8 | 5 мин. 13 сек. | 12.01 MB | 43.12 сек. |
| 9 | 4 мин. 03 сек. | 10.67 MB | 35.37 сек. |
| 10 | 4 мин. 23 сек. | 11.58 MB | 37.73 сек. |
| Среднее | 4 мин. 07 сек | 10.12 MB | 34.65 сек |

Таблица 4.1: Время выполнения алгоритма

# 5 Сравнение с известными аналогами

По сравнению с другими аналогами, веб-приложение данного проекта обращает больше внимания на качество лица. Более того, с помощью веб-приложения пользователи имеют возможность скачать лучшее изображение лица с произвольным размером. Такая функция будет большим преимуществом для тех, кто не желает загрузить клип в видео платформу.

На рисунке 5.1 представлено сравнение полученных обложек видеоконтента длительностью 11 минут 52 секунды, сгенерированных тремя подходами:

1 Получение превью с помощью DNN модели. Представитель: Youtube;

2 Генерация рандомного кадра. Представитель: веб-приложение Olfu;

3 Извлечение лучшего изображения лица по разным критериям, реализованное в данном проекте.


(a) Olfu (b) Project (c) Youtube

Рис. 5.1: Сравнение качества моделей

В таблице 5.1 представлены временные характеристики различных методов обработки видео. Для измерения времени обработки видеоматериала на платформе Youtube было принято решение начинать измерение после завершения загрузки видеоконтента.

| | Youtube | Project | Olfu |
|---|---|---|---|
| Время обработки (сек) | 108.26 | 61.45 | $< 1$ |

Таблица 5.1: Сравнение времени выполнения разных алгоритмов

Исходя из полученных результатов, можно сделать вывод, что лицевое выражение, представленное в кадре, сгенерированном с использованием веб-приложения Olfu и Youtube, является неприемлемым. Наиболее качественное изображение лица было получено при использовании программного решения данного проекта, и самым быстрым алгоритмом оказалась генерация случайного кадра, весь процесс которого занимался меньше секунды. Стоит отметить, что скорость выполнения алгоритма данного проекта оказалась на 46.81 секунд быстрее, чем алгоритма Youtube. В целом, можно утверждать, что программное решение данной работы демонстрирует хорошие результаты как с точки зрения качества кадра, так и с точки зрения эффективности выполнения работы.

# 6 Описание системы с точки зрения пользователя

Разработанное программное решение было протестировано на операционной системе Ubuntu 20.04 LTS с использованием процессора AMD Ryzen 5 3500U и объемом оперативной памяти 8 Гб. Представленная система является веб-приложением, реализованным с помощью Flask. Для использования данной утилиты пользователь может перейти по ссылке и загрузить видео в форматах `'mp4'`, `'mov'`, `'wmv'`, `'avi'`, `'mkv'`. Максимальный размер видеоконтента был установлен равным 60 МБ.

Пользователь имеет возможность вручную настроить две переменные: `num_frames` и `fps`. Значение `num_frames` определяет количество первых кадров, которые будут рассматриваться программой. В случае, если среди первых `num_frames` не обнаружено подходящего кадра, система автоматически анализирует следующие `num_frames` кадры до тех пор, пока не получено изображение, соответствующее критериям. Переменная `fps` определяет количество кадров, которые программа будет анализировать в единицу времени. Например, если значение переменной `fps` = 5, то система равномерно выбирает 5 кадров в секунду. По умолчанию программа рассматривает все видео и составляет выборку кадров с частотой 3 кадра в секунду.

Для запуска веб-приложения на локальном компьютере необходимо загрузить все файлы с сайта Github[4] и проверить соответствие версий пакетов, указанных в файле requirements.txt. Для загрузки необходимых пакетов следует выполнить команду `"./boostrap.sh"`, а для запуска веб-приложения - `"./start.sh"`.

Дополнительная информация о приложении может быть найдена в документации, размещенной на веб-приложении в разделе *About Project*[5].

# 7 Заключение

Выводы, которые можно сделать на основе данного исследования, свидетельствуют о том, что автоматическое извлечение лучшего изображения лица из клипа является важной и актуальной задачей в области видео анализа. Разработанное веб-приложение может быть полезным инструментом для всех, кто работает с видеоконтентом и хочет получить наиболее привлекательную обложку для своих видеороликов.

Однако, необходимо отметить, что существует ряд ограничений в использовании данного алгоритма. В частности, он был протестирован только на видео с одним человеком, по-

[4]Доступно по ссылке: https://github.com/mengsifei/CourseProject/tree/main
[5]Доступно по ссылке: http://1462839-cj10416.tw1.ru/about

этому необходимо дополнительное исследование для расширения его возможностей на более сложных видеоконтентах. В ходе проведения тестирования еще были обнаружены проблемы, связанные с низким качеством видеозаписи и наличием макияжа, которые могут привести к неправильным результатам. В перспективе возможно обучение отдельных моделей для распознавания ключевых точек лица и повышения разрешения изображения выбранного кадра.

Несмотря на вышеупомянутые ограничения и проблемы, разработанный алгоритм является важным шагом в автомотической генерации превью видео и может быть использован для создания более продвинутых систем обработки видеоконтента.

# Список литературы


[1] Lingyu Liang, Luojun Lin, Lianwen Jin, Duorui Xie и Mengru Li. “SCUT-FBP5500: A Diverse Benchmark Dataset for Multi-Paradigm Facial Beauty Prediction”. В: *arXiv:1801.06345* (2018).

[2] Sathasivam S., Mahamad A. K., Saon S., Sidek A, Som M. M. и Ameen H. A. “Drowsiness Detection System using Eye Aspect Ratio Technique”. В: *IEEE Student Conference on Research and Development (SCOReD)* (2020).

[3] Suwarno и Kevin. “Analysis of Face Recognition Algorithm: Dlib and OpenCV”. В: *JITE (Journal Of Informatics And Telecommunication Engineering)* 4.1 (2020), с. 173—184.

[4] Mingxing Tan и Quoc V. Le. “EfficientNetV2: Smaller Models and Faster Training”. В: *International Conference on Machine Learning* (2021).

[5] Weilong Yang и Min-hsuan Tsai. *Improving YouTube video thumbnails with deep neural nets*. URL: https://ai.googleblog.com/2015/10/improving-youtube-video-thumbnails-with.html (дата обр. 10.05.2023).