

А. Адикова^{1*}, А. Адамова²

¹Казахский агротехнический университет им. С. Сейфуллина, г. Нур-Султан, Казахстан

²Astana IT University, г. Нур-Султан, Казахстан

*e-mail: aido4ka85@mail.ru

ИССЛЕДОВАНИЯ И АНАЛИЗ КЛАССИФИКАТОРОВ ДЛЯ ПРИМЕНЕНИЯ В РАСПОЗНАВАНИИ ЭМОЦИЙ

Аннотация

На сегодняшний день, наблюдается широкое использование метода коммуникаций через онлайн-платформы, такие как обеспечения безопасности при получении конфиденциальных документов, осуществления электронных платежей и вместе с тем в процессе получения образования. Массовое использование онлайн-платформ для общения порождает необходимость систем распознавания человеческих эмоций. В системах распознавания лиц и человеческих эмоций информация собирается, анализируется и обрабатывается с помощью специальных методов и алгоритмов обучения. В работе рассматриваются и анализируются классификаторы для распознавания лиц, такие как каскадный классификатор Хаара, метод извлечения признаков, локальная бинарная гистограмма (LBP), нейронные сверточные сети (CNN), которые способствуют эффективно реализовать методы распознавания эмоций. Данные алгоритмы извлекают и используют лицевые ориентиры, функции, паттерны двигательного поведения лица для обнаружения эмоций из различных форм данных, таких как изображения, видео и т. д.

Ключевые слова: распознавание, эмоции, классификатор, сверточные нейронные сети.

Аңдатпа

А. Адикова¹, А. Адамова²

¹ С. Сейфуллин атындағы Қазақ агротехникалық университеті, Нұр-Сұлтан қ., Қазақстан

² Astana IT University, Нұр-Сұлтан қ., Қазақстан

ЭМОЦИЯНЫ ТАНУДА ҚОЛДАНУ ҮШІН ЖІКТЕУШТЕРДІ ЗЕРТТЕУ ЖӘНЕ ТАЛДАУ

Бүгінгі таңда құпия құжаттарды қабылдау, электронды төлемдерді жүзеге асыр, сонымен қатар білім алу процесінде және басқа салаларда да, онлайн-платформалар арқылы коммуникация әдісін кеңінен қолдану байқалады. Интернеттегі байланыс платформаларын жаппай пайдалану адамның эмоцияларын тану жүйелерін қажет етеді. Адамның беті мен эмоцияларын тану жүйелерінде ақпарат арнайы әдістер мен оқыту алгоритмдері арқылы жиналады, талданады және өңделеді. Бұл жұмыста Хаар каскадты классификаторы, белгілерді алу әдісі, жергілікті екілік гистограмма (LBP), эмоцияны тану әдістерін тиімді қолдануға ықпал ететін нейрондық конвульсиялық желілер (CNN) сияқты бет-әлпетті тану классификаторлары қарастырылады және талданады. Бұл алгоритмдер кескіндер, бейнелер және т.б. сияқты деректердің әртүрлі пішіндерінен эмоцияларды анықтау үшін бет белгілерін, мүмкіндіктерді және бет-әлпет қозғалысы үлгілерін шығарады және пайдаланады.

Түйін сөздер: тану, эмоциялар, жіктеушітер, конвульсиялық нейрондық желілер.

Abstract

STUDY AND ANALYSIS OF CLASSIFIERS FOR USE IN EMOTION RECOGNITION

Adikova A.¹, Adamova A.²

¹ S. Seifullin Kazakh AgroTechnical University, Nur-Sultan, Kazakhstan

² Astana IT University, Nur-Sultan, Kazakhstan

Today, there is a widespread use of the method of communication through online platforms, such as ensuring security when receiving confidential documents, making electronic payments, and at the same time in the process of obtaining education. The massive use of online platforms for communication creates the need for human emotion recognition systems. In facial recognition systems and human emotions, information is collected, analyzed and processed using special methods and learning algorithms. The paper considers and analyzes classifiers for face recognition, such as the Haar cascade classifier, the feature extraction method, local binary histogram (LBP), neural convolutional networks (CNN), which contribute to the effective implementation of emotion recognition methods. These algorithms extract and use facial landmarks, features, and facial movement patterns to detect emotions from various forms of data such as images, videos, etc.

Keywords: recognition, emotions, classifier, convolutional neural networks.

Введение

На сегодняшний день распознавание лиц остается одной из самых актуальных направлений компьютерного зрения. Эта технология уже доступна и используется в разных сферах. От камер, которые фокусируются на лице еще до того, как сделан снимок, до социальных сетей, когда автоматически помечаются лица людей после загрузки изображения. Также их используют для идентификации преступников по видеозаписи с камер наблюдения и для разблокировки телефона всего лишь посмотрев на него. Но по лицу человека можно не только его идентифицировать, но и определить его намерения основываясь на его эмоции. Правильное прогнозирование настроений может стать источником роста для развития различных сфер, изучения алгоритмов для чтения эмоций является наиболее исследуемой сферой в современной науке [1].

Методы исследования

В мире баз данных и машинного обучения существуют множество методов распознавания, которые применимы для создания систем, идея которых состоит в том, чтобы воспроизвести мыслительный процесс человека на основе данных обучения (в виде изображений и видео людей) и попытаться сегментировать эмоции, присутствующие в этих данных. Применимые в этих системах алгоритмы распознавания являются самыми исследуемыми в современном машинном обучении:

- PCA (principal component analysis) - метод главных компонент;
- HoG (Histogram of Oriented Gradients) - гистограммы ориентированных градиентов;
- DNN (Deep Neural Network) – искусственная нейронная сеть;
- Haar Cascade – каскад Хаара;
- MTCNN (Multi-task Cascaded Convolutional Neural Networks) - многозадачная каскадная сверточная нейронная сеть;
- LBP (Local Binary Pattern Histogram) - локальные бинарные шаблоны.

Рассмотрим некоторые из них, в работе [2] рассматривается метод главных компонент (PCA), разработанный на самой ранней стадии и по-прежнему привлекающий исследователей своим свойством уменьшать размер данных без потери важной информации. В методе PCA было предложено несколько вариантов стандартного подхода для конкретных ситуаций. Вместе с тем создание системы распознавания лиц на основе PCA по-прежнему занимает много времени, потому что существуют различные проблемы, которые необходимо учитывать в практических применениях, такие как освещение, выражение лица или угол съемки [2]. OpenCV, популярный набор инструментов для обработки изображений, например, имеет встроенные библиотеки для стандартных алгоритмов PCA, но эти библиотеки имеют ограниченные возможности настройки [3]. Кроме того, необходимо учитывать несколько вариантов PCA, поскольку лучшие результаты получаются, когда подходящий вариант PCA используется для экстремальных ситуаций, таких как как неравномерное освещение или преувеличенное выражение лица. Кроме того, связанные шаги, как обнаружение лиц и предварительная обработка, также играют важную роль с точки зрения всего процесса распознавания лиц.

Также в работе [4] рассматривается алгоритм гистограммы ориентированных градиентов (HoG). Суть алгоритма HOG состоит в том, чтобы вместо использования каждого отдельного направления градиента каждого отдельного пикселя изображения, группировать пиксели в маленькие ячейки. Для каждой ячейки вычисляются все направления градиента и группируют их в несколько бинов ориентации. Суммируя получают величину градиента в каждом образце. Таким образом, более сильные градиенты вносят больший вес в свои бункеры, а эффекты малых случайных ориентаций из-за шума снижаются. Эта гистограмма дает картину доминирующей ориентации этой клетки. Выполнение перечисленных этапов для всех ячеек формирует представление о структуре изображения.

Следующий пример это DNN, который упоминается в работе [5] – это тип нейронной сети, смоделированной как многослойный перцептрон (MLP), который обучается с помощью алгоритмов для изучения представлений из наборов данных без какого-либо ручного проектирования экстракторов признаков. Как следует из названия, глубокое обучение состоит из большего количества уровней обработки, что контрастирует с моделью поверхностного обучения с меньшим количеством уровней единиц. Переход от поверхностного к глубокому обучению позволил сопоставить более сложные и нелинейные функции. Это улучшение было дополнено распространением более дешевых устройств обработки, таких как универсальный графический процессор (GPGPU), и большого объема

набора данных (больших данных) для обучения. Хотя GPGPU менее мощны, чем процессоры, количество ядер параллельной обработки в них на порядки превышает количество ядер процессора. Это делает GPGPU лучше для реализации DNN.

Классификаторы для распознавания лиц

Методы Хаара и LBP называют классификаторами изображений или компьютерной программой, которая определяет, является ли изображение позитивным – если имеет в себе лица людей или негативным – без лиц. Классификатор обучается на сотнях тысяч изображений лиц и других изображений, чтобы научиться правильно классифицировать новое изображение. Библиотека OpenCV, предоставляет два предварительно обученных и готовых к использованию классификаторов обнаружения лиц Классификатор Хаара и LBP [3].

Классификаторы для определения лиц обрабатывают изображения в серых оттенках. Классификатор Хаара является алгоритмом машинного обучения, созданный Полем Алта и Майклом Джонса, который обучается на основе множества положительных изображений лиц и отрицательных изображений без лиц [6].

Классификатор начинается с извлечения функций Хаара из каждого изображения. Классификатор состоит из пары черт, которые показаны как отдельные окна с особенностями. Каждое окно размещается на изображении для расчета отдельного признака (Рисунок 1). Эта функция представляет собой одно значение, полученное путем вычитания суммы пикселей под белой частью окна из суммы пикселей под черной частью окна. Теперь все возможные размеры каждого окна размещены во всех возможных местах каждого изображения, чтобы вычислить множество функций [6,7].



Рисунок 1. Различные этапы визуализации классификатора Хаара

Например, на изображении выше извлекаются две функции. Первый фокусируется на том, что область глаз часто темнее, чем область носа и щек. Вторая особенность основана на том, что глаза темнее переносицы. Но на практике большинство из этих характеристик не имеет значения [6]. Например, при использовании на щеке окна становятся неактуальными, потому что ни одна из этих областей не темнее или светлее, чем другие области на щеках, все сектора здесь такие же. Поэтому нерелевантные функции быстро отбрасываются и остаются только те, которые актуальны, с помощью алгоритма Adaboost [8].

AdaBoost - это процесс обучения распознаванию лиц, который выбирает только те функции, которые, как известно, улучшают точность классификации классификатора. В конце алгоритм учитывает тот факт, что, как правило, большая часть области изображения не является областью лица. Учитывая это, было бы лучше иметь простой метод, чтобы проверить, является ли окно областью без лица, и если это не так, немедленно отбросьте его и не обрабатывайте его снова. Таким образом, мы можем сосредоточиться в основном на той области, где находится лицо [8].

Вместе с тем каскадный классификатор LBP, как и любой другой классификатор также необходимо обучать на сотнях изображений. LBP – это визуальный текстурный дескриптор, из которых состоят наши лица, которые также состоят из микровизуальных паттернов. Таким образом, особенности LBP извлекаются, чтобы сформировать вектор признаков, который классифицирует лицо от лица, не являющегося лицом [9].

Каждый тренировочный образ разделен на несколько блоков, для каждого блока LBP просматривает 9 пикселей (окно 3×3) за раз, где особое внимание уделяется пикселю,

расположенному в центре окна. Затем сравнивается значение центрального пикселя со значением каждого соседнего пикселя в окне 3×3 . Для каждого соседнего пикселя, который больше или равен центральному пикселю, устанавливается значение 1, а для остальных - 0. После считываются обновленные значения пикселей (которые могут быть 0 или 1) по часовой стрелке и формирует двоичное число. Затем идет преобразование двоичных чисел в десятичное, и это десятичное число является новым значением центрального пикселя. Данные операции прodelьваются для каждого пикселя в блоке. Затем значение каждого блока преобразуется в гистограмму и в результате получается по одной гистограмме для каждого блока изображения [10]. В итоге, алгоритм объединяет гистограммы блоков, чтобы сформировать единый вектор признаков для одного изображения, который содержит все особенности. Таким образом извлекая особенности LBP из изображения. Например, код LBP для пикселя в (x_s, y_s) задается формулой (1). [10]:

$$LBP_{P,R}(x_c, y_c) = \sum_{(p=0..7)} S(g_p - g_c)2^p, S(x) = \{1, x \geq 0 \text{ and } 0, x < 0\} \quad (1)$$

где, g_c – значение серого центрального пикселя, g_p – значение серого соседнего пикселя g_c , $P = 8$ (максимум 8 соседей для центрального пикселя в матрице 3×3).

LBP просто реализовать, поскольку двоичный рисунок остается неизменным, несмотря на изменения освещения или яркости. Увеличение яркости или условие освещения приведет к увеличению интенсивности всех пикселей на одно и то же значение, сохраняя при этом относительную разницу.

Для сравнения работы двух классификаторов проведен анализ обученных систем в OpenCV.

Для тестирования распознавания было приведены тестовые изображений лиц и они были загружены в онлайн систему для тестирования. Ниже приведен пример работы классификатора Хаара (Рисунок 2)



Рисунок 2. Распознавание с классификатором Хаара в OpenCV

Также был исследован второй метод распознавания LBP в OpenCV. По результатам данного тестирования приведен следующий анализ характеристик методов (Таблица 1).

Таблица 1. Характеристики алгоритмов распознавания

Алгоритм	Преимущество	Недостаток
Haar	Высокая точность обнаружения Низкий уровень ложноположительных результатов	Вычислительно сложный и медленный Более длительное время для обучения Менее точный на темных лицах Ограничен в сложных условиях освещения Менее устойчив к окклюзиям
LBP	Вычислительно простой и быстрый Более короткое время для обучения Устойчив к местным изменениям освещенности Устойчив к окклюзиям	Менее точный Высокий уровень ложноположительных результатов

Каждый классификатор обнаружения лиц OpenCV имеет свои плюсы и минусы, но основные различия заключаются в точности и скорости. Если требуются более точные обнаружения, лучше всего подойдет классификатор Хаара. Этот алгоритм больше подходит для таких технологий, как системы безопасности или высокотехнологичное преследование. Но классификатор LBP работает быстрее, поэтому его следует использовать в мобильных приложениях или встроенных системах. Соответственно алгоритм распознавания следует выбирать в зависимости от реализуемой области [11].

Также в данном исследовании был рассмотрен следующий алгоритм применимый в распознавании не только лица, но и определении эмоций человека. Этот алгоритм MTCNN (многокаскадная сверточная сеть) используемая в Face Emotion Recognizer (обычно известный как FER) – библиотеке Python с открытым исходным кодом, созданная и поддерживаемая Джастином Шенком, и используемая для анализа тональности изображений или видео. Распознавание в FER как раз строится на самых развивающихся технологиях, как сверточная нейронная сеть с весами.

Например, Конструктор параметров MTCNN (многокаскадная сверточная сеть) - это метод обнаружения лиц, где для него установлено значение «Истина», модель MTCNN используется для обнаружения лиц, а когда для него установлено значение «Ложь», функция использует классификатор OpenCV HaarCascade по умолчанию [1].

Многозадачная каскадная сверточная нейронная сеть (MTCNN) - это подход, основанный на глубоком обучении, который использует три каскадных сверточных нейронных сети (CNN) для быстрого и точного обнаружения лиц. В MTCNN обнаружение лиц и выравнивание лиц выполняются вместе в режиме многозадачного обучения, что позволяет обнаруживать невыровненные лица лучше. В таблице 2 показаны основные плюсы и минусы данного алгоритма распознавания [12].

Таблица 2. MTCNN

Преимущества	Недостатки
<ol style="list-style-type: none"> 1. Работает в реальном времени на GPU. 2. Очень точное обнаружение лица. 3. Обнаруживает лица в разных масштабах. 4. Работает для разной ориентации лица. 5. Работает с изображениями с окклюзиями. 	<ol style="list-style-type: none"> 1. Медленная загрузка процессора.

Этапы программного обеспечения для распознавания эмоций

В данном классификаторе используются такие команды, как `detect_emotions ()` - которая является функцией, используемой для классификации обнаружения эмоций, и регистрирует вывод по шести категориям, а именно: «страх», «нейтральный», «счастливый», «грустный», «гнев» и «отвращение». Каждая эмоция рассчитывается, и результат оценивается по шкале от 0 до 1.

Программа начинается с ввода изображения или видео, которые необходимо проанализировать. Схематично показано на рисунке 3. Конструктор FER инициализируется присвоением ему классификатора обнаружения лиц (OpenCV HaarCascade или MTCNN). Затем мы вызываем функцию обнаружения эмоций этого конструктора, передавая ей входной объект (изображение или видео). Достигнутый результат – это набор эмоций, для каждой из которых указано значение. Наконец, функция `'top_emotion'` может отобразить самые ценные эмоции объекта и вывести их значение.

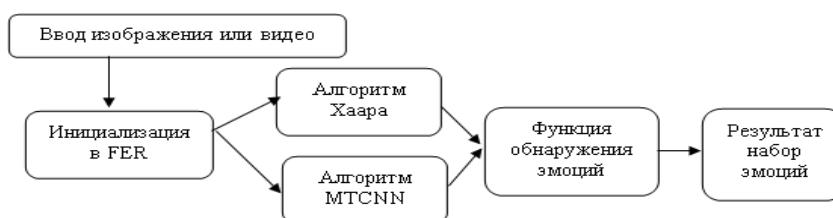


Рисунок 3. Процесс распознавания эмоций

Реализацию этого алгоритма для изображений можно коротко рассмотреть следующим образом. Первым делом устанавливаем и инициализируем необходимые библиотеки.

1. Установка алгоритма Face Emotion Recognizer и всех необходимых библиотек

```
[1] pip install fer
[2] pip install opencv-python
[5] pip install numpy
[6] pip install matplotlib
import cv2
[7] from fer import FER
import matplotlib.pyplot as plt
%matplotlib inline
```

2. Присвоением классификатора для обнаружения лица и определения базовых эмоции объекта.

```
test_image_two = plt.imread("/content/test3.JPG")
captured_emotions_two = emo_detector.detect_emotions(test_image_two)
print(captured_emotions_two)
plt.imshow(test_image_two)
dominant_emotion_two, emotion_score_two = emo_detector.top_emotion(test_image_two)
print(dominant_emotion_two, emotion_score_two)
```

```
[[{'box': (138, 42, 185, 185), 'emotions': {'angry': 0.0, 'disgust': 0.0, 'fear': 0.0, 'happy': 1.0, 'sad': 0.0, 'surprise': 0.0, 'neutral': 0.0}}]]
happy 1.0
```



3. Регистрация вывода по шести категориям и определения основную

```
dominant_emotion, emotion_score = emo_detector.top_emotion(test_image_two)
print(dominant_emotion, emotion_score)
```

```
happy 1.0
```

Рисунок 4. Определения основной эмоции

Код индивидуально принимает изображения в качестве входных данных и детализирует различные эмоции и их уровни интенсивности при выходе. Затем с помощью функции `top_emotion()` извлекает наиболее доминирующую эмоцию (рисунок 4).

Чтобы выбрать алгоритм обнаружения лиц, идеально подходящий для этой задачи, все метода были протестированы в общей сложности на 300 случайно выбранных изображениях из обучающей выборки (100 изображений, принадлежащих каждому классу эмоций). Количество лиц, обнаруженных каждым методом для 300 изображений, можно увидеть на рисунке 5 [12].

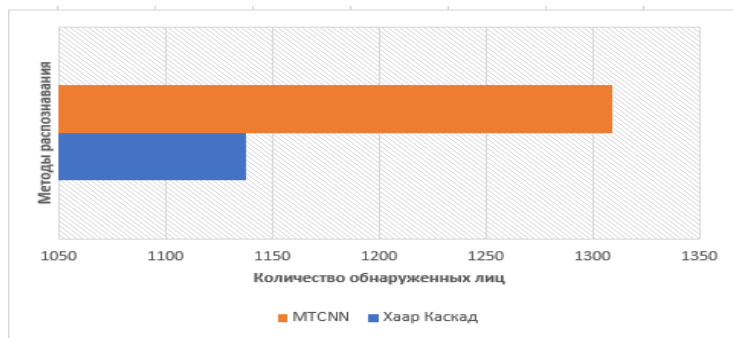


Рисунок 5. Диаграмма сравнения методов распознавания

Заключение

В результате эксперимента было обнаружено, что детектор лиц MTCNN является наиболее точным из набора данных. Однако детектор лиц MTCNN работал довольно медленно. Но вместе с

тем было обнаружено, что детектор лиц Хаара в OpenCV обнаружил также довольно большое количество лиц, но дал много ложных срабатываний.

В связи с вышеизложенным данное исследование приступило к дальнейшей реализации классификатора обнаружения эмоций именно с МТСNN, но в ходе исследования могут возникнуть и другие характеристики, которые могут повлиять на изменения выбранного метода распознавания.

Список использованной литературы:

- 1 Rahulraj Singh. *The Ultimate Guide to Emotion Recognition from Facial Expressions using Python* [Электронный ресурс]. – 2020 – URL: <https://towardsdatascience.com/the-ultimate-guide-to-emotion-recognition-from-facial-expressions-using-python-64e58d4324ff>
- 2 Peng Peng, Ivens Portugal, Paulo Alencar, Donald Cowan. *A face recognition software framework based on principal component analysis* // PLoS ONE16(7) – 2021.
- 3 Valentina Alt.o *Face recognition with OpenCV: Haar Cascade* // DataSeries *Imagine the future of data* - 2019.
- 4 В. М. Рябов, Ю. А. Иванова. *Детектирование объектов на изображении на основе комбинации hog+svm* // Томский политехнический университет: Матер. конф. – Томск, 2020.
- 5 Ajay Shrestha and Ausif Mahmood *Review of Deep Learning Algorithms and Architectures* // IEEE Access – 2019, PP (99): 1-1.
- 6 Адитья Муттал. *Haar Cascades, Explained* [Электронный ресурс]. – 2020 – URL: <https://medium.com/analytics-vidhya/haar-cascades-explained-38210e57970d>
- 7 Girija Shankar Behera. *Face Detection with Haar Cascade guide* [Электронный ресурс]. – 2020 – URL: <https://towardsdatascience.com/face-detection-with-haar-cascade-727f68dafd08>
- 8 Serign Modou Bah, Fang Ming. *An improved face recognition algorithm and its application in attendance management system* // Array Volume 5 - March 2020, 100014.
- 9 Ruth Mare. *Understanding Facial Recognition Using Local Binary Pattern Histogram (LBPH) Algorithm* [Электронный ресурс]. – 2021 – URL: <https://www.section.io/engineering-education/understanding-facial-recognition-using-local-binary-pattern-histogram-algorithm/>
- 10 Parth Singh. *Understanding Face Recognition using LBPH algorithm* // Data Science Blogathon – 2021.
- 11 SuperDataScience Team. *Face detection using OpenCV and Python: A beginner's guide* [Электронный ресурс]. – 2020 – URL: <https://www.superdatascience.com/blogs/opencv-face-recognition>
- 12 Chi-Feng Wang. *What's the Difference Between Haar-Feature Classifiers and Convolutional Neural Networks?* [Электронный ресурс]. – 2018 – URL: <https://towardsdatascience.com/whats-the-difference-between-haar-feature-classifiers-and-convolutional-neural-networks-ce6828343aeb>

References:

- 1 Rahulraj Singh. *The Ultimate Guide to Emotion Recognition from Facial Expressions using Python* [Электронный ресурс]. – 2020 – URL: <https://towardsdatascience.com/the-ultimate-guide-to-emotion-recognition-from-facial-expressions-using-python-64e58d4324ff>
- 2 Peng Peng, Ivens Portugal, Paulo Alencar, Donald Cowan. *A face recognition software framework based on principal component analysis* // PLoS ONE16(7) – 2021.
- 3 Valentina Alt.o *Face recognition with OpenCV: Haar Cascade* // DataSeries *Imagine the future of data* - 2019.
- 4 В. М. Рябов, Ю. А. Иванова. *Детектирование объектов на изображении на основе комбинации hog+svm* // Томский политехнический университет: Матер. конф. – Томск, 2020.
- 5 Ajay Shrestha and Ausif Mahmood *Review of Deep Learning Algorithms and Architectures* // IEEE Access – 2019, PP (99): 1-1.
- 6 Адитья Муттал. *Haar Cascades, Explained* [Электронный ресурс]. – 2020 – URL: <https://medium.com/analytics-vidhya/haar-cascades-explained-38210e57970d>
- 7 Girija Shankar Behera. *Face Detection with Haar Cascade guide* [Электронный ресурс]. – 2020 – URL: <https://towardsdatascience.com/face-detection-with-haar-cascade-727f68dafd08>
- 8 Serign Modou Bah, Fang Ming. *An improved face recognition algorithm and its application in attendance management system* // Array Volume 5 - March 2020, 100014.
- 9 Ruth Mare. *Understanding Facial Recognition Using Local Binary Pattern Histogram (LBPH) Algorithm* [Электронный ресурс]. – 2021 – URL: <https://www.section.io/engineering-education/understanding-facial-recognition-using-local-binary-pattern-histogram-algorithm/>
- 10 Parth Singh. *Understanding Face Recognition using LBPH algorithm* // Data Science Blogathon – 2021.
- 11 SuperDataScience Team. *Face detection using OpenCV and Python: A beginner's guide* [Электронный ресурс]. – 2020 – URL: <https://www.superdatascience.com/blogs/opencv-face-recognition>
- 12 Chi-Feng Wang. *What's the Difference Between Haar-Feature Classifiers and Convolutional Neural Networks?* [Электронный ресурс]. – 2018 – URL: <https://towardsdatascience.com/whats-the-difference-between-haar-feature-classifiers-and-convolutional-neural-networks-ce6828343aeb>