

МЕТОД ОБНАРУЖЕНИЯ ПОДОЗРИТЕЛЬНЫХ ТРАНЗАКЦИЙ БАНКОВСКИХ КЛИЕНТОВ НА ОСНОВЕ СИСТЕМЫ РАСПОЗНАВАНИЯ ЭМОЦИЙ

Козьминых С. И.¹, Татаренков В. С.²

DOI: 10.21681/2311-3456-2024-3-129-140

Цель статьи: разработка метода выявления транзакций, совершенных клиентами, подвергнутыми воздействию мошенников с помощью методов социальной инженерии, на основе анализа видеоданных лица с использованием нейросетевых методов распознавания эмоций.

Метод исследования: анализ современных нейросетевых моделей и подходов, используемых для решения задачи распознавания эмоций; анализ архитектур нейронных сетей, обрабатывающих видеоизображение или последовательность кадров; разработка и программная реализация метода обнаружения подозрительных транзакций с использованием искусственных нейронных сетей по видеоданным лица человека; экспериментальное исследование и оценка разработанного метода.

Полученный результат: разработан метод выявления подозрительных транзакций, основанный на нейросетевых методах распознавания лицевых эмоций клиентов банка, подвергшихся воздействию злоумышленников. Реализована комбинированная структура нейронной сети с использованием архитектуры пригодной для обработки графической информации и информации, представленной во временной последовательности, для решения задачи распознавания эмоций. Создан программный прототип, позволяющий оценивать эмоциональное состояние наблюдаемого человека по видеоданным лица и способный определять нахождение человека в негативном эмоциональном состоянии. Были проанализированы результаты разработанного метода. Даны рекомендации по перспективам его применения и дальнейшим исследованиям данной темы.

Научная новизна: предложен новый метод выявления подозрительных транзакций, основанный на решении задачи распознавания эмоций по видео с применением комбинации CNN и LSTM архитектур нейронных сетей.

Ключевые слова: долгая краткосрочная память, машинное обучение, распознавание эмоций, рекуррентные нейронные сети, сверточные нейронные сети, CNN, LSTM.

METHOD FOR DETECTING SUSPICIOUS TRANSACTIONS OF BANKING CLIENTS BASED ON EMOTION RECOGNITION SYSTEM

Kozminykh S. I.³, Tataronkov V. S.⁴

The purpose of the article: to develop a method for detecting transactions made by customers exposed to fraud using social engineering methods based on the analysis of video data of a person using neural network methods of emotion recognition.

1 Козьминых Сергей Игоревич, доктор технических наук, доцент, профессор кафедры информационной безопасности Финансового университета при Правительстве РФ, профессор кафедры прикладной информатики и информационной безопасности РЭУ им. Г.В.Плеханова, г. Москва, Россия. E-mail: SlKozminykh@fa.ru, Kozminykh.SI@rea.ru

2 Татаренков Владислав Сергеевич, аспирант кафедры информационной безопасности Финансового университета при Правительстве РФ г. Москва, Россия. E-mail: vt96@mail.ru

3 Sergey I. Kozminykh, Doctor of Technical Sciences, Associate Professor, Professor of the Department of Information Security of the Financial University under the Government of the Russian Federation, Professor of the Department of Applied Informatics and Information Security of Plekhanov Russian University of Economics, Moscow, Russia. E-mail: SlKozminykh@fa.ru, Kozminykh.SI@rea.ru

4 Vladislav S. Tataronkov, postgraduate student of the Department of Information Security of the Financial University under the Government of the Russian Federation, Moscow, Russia. E-mail: vt96@mail.ru

Research method: analysis of modern neural network models and approaches used to solve the problem of emotion recognition; analysis of neural network architectures that process a video image or sequence of frames; development and software implementation of a method for detecting suspicious transactions using artificial neural networks based on video data of a person's face; experimental research and evaluation of the developed method.

The result obtained: a method for detecting suspicious transactions based on neural network methods for recognizing the facial emotions of bank customers exposed to intruders has been developed. A combined neural network structure is implemented using an architecture suitable for processing graphical information and information presented in a time sequence to solve the problem of emotion recognition. A software prototype has been created that allows you to assess the emotional state of an observed person from video data of a person and is able to determine whether a person is in a negative emotional state. The results of the developed method were analyzed. Recommendations are given on the prospects of its application and further research on this topic.

Scientific novelty: a new method for detecting suspicious transactions is proposed, based on solving the problem of recognizing emotions from video using a combination of CNN and LSTM architectures of neural networks.

Keywords: long-term short-term memory, machine learning, emotion recognition, recurrent neural networks, convolutional neural networks, CNN, LSTM.

Введение

На сегодняшний день, согласно опубликованным отчетам об исследованиях рынка систем обнаружения и распознавания эмоций, данные системы находят широкое применение в различных сферах деятельности человека⁵. Современные системы распознавания эмоций способны не только классифицировать стандартные типы эмоций, такие как радость, грусть, злость и так далее, но и определять специфические эмоции.

Вначале рассмотрим одну из таких сфер, которой является медицина и здравоохранение, где подобного рода системы призваны определять состояние людей по визуальному наблюдению с возможностью дополнительного анализа показателей, поступающих с медицинских измерительных приборов [1,2]. Также в опубликованных работах встречается описание применения систем распознавания эмоций для диагностики стрессового и психического расстройств [3,4]. Следующей рассматриваемой сферой является сфера автотранспорта и дальних грузоперевозок, для которой ведутся разработки и исследования систем по определению утомления или стрессового состояния человека [5,6]. Основной задачей, решаемой данными системами, является заблаговременное

определение критического состояния водителя по камерам из салона машины для предотвращения возникновения потенциальных дорожно-транспортных происшествий. Данные примеры исследований и разработок показывают, что современные методы для распознавания эмоций способны использоваться для определения психоэмоционального состояния человека и сложных глубинных эмоций.

Рассмотрим банковскую сферу. Поиск информации о внедрении систем обнаружения и распознавания эмоций в банках дает результаты, где эти системы применяются в основном в маркетинговых целях⁶. В частности, это применяется для определения удовлетворения клиента качеством обслуживания или определения эмоционального состояния клиента для последующей рекомендации ему того или иного банковского продукта. В свою очередь, по статистике, ежегодно публикуемой компаниями регуляторами⁷ в сфере информационной безопасности можно сделать вывод о стабильно высоком уровне телефонного и СМС-мошенничества, направленного

5 1. Emotion Detection and Recognition Market, By Technology Type (Facial Expression Recognition, Speech Emotion Recognition, and Biometric Recognition), By Component (Hardware, Software, and Services), By Application, and By Region Forecast to 2032 // Reports and Data : сайт. – 2023. – URL: <https://www.reportsanddata.com/report-detail/emotion-detection-and-recognition-market> (дата обращения: 01.12.2023).
2. Ankit Gupta. Emotion Detection and Recognition market Research Report Information By Technology (Bio Sensors Technology, Machine Learning, Pattern Recognition, Feature Extraction and 3D Modelling, Natural Language Processing (NLP), Others), By Service (Storage and Maintenance, Consulting and Integration), By Application (Law Enforcement, Surveillance and Monitoring, Marketing & Advertising, Media & Entertainment, Others), And By Region (North America, Europe, Asia-Pacific, And Rest Of The World) – Market Forecast Till 2030 // Market Research Future : сайт. – 2024. – URL: <https://www.marketresearchfuture.com/reports/emotion-detection-recognition-market-3193> (дата обращения: 19.01.2024).

6 1. Будущее за эмоциями: как Альфа-Банк улучшает клиентский опыт // Forbes : новостной портал. – 2023. – URL: <https://www.forbes.ru/brandvoice/501354-budusee-za-emociami-kak-al-fa-bank-ulucsaet-klientskij-opyt> (дата обращения: 04.12.2023).
2. Нейросети банка «Точка» научились распознавать эмоции клиентов // Adindex : новостной портал. – 2023. – URL: <https://adindex.ru/news/digital/2023/06/1/312986.html> (дата обращения: 05.12.2023).
7 1. Kaspersky Who Calls: в первом квартале 2023 года доля столкнувшихся с классическим телефонным мошенничеством выросла на три процентных пункта // Kaspersky : сайт. – 2023. – URL: https://www.kaspersky.ru/about/press-releases/2023_kaspersky-who-calls-v-pervom-kvartale-2023-goda-dolya-stolknuvshih-sya-s-klassicheskim-telefonnym-moshennichestvom-vyros-la-na-tri-procentnyh-punkta (дата обращения: 06.12.2023).
2. Инциденты информационной безопасности: итоги I квартала 2023 года // CBR : сайт. – 2023. – URL: <https://cbr.ru/press/event/?id=15814> (дата обращения: 07.12.2023).
3. Рост количества мошеннических операций за 2023 год — статистика от ЦБ РФ // Finadvice MTS : сайт. – 2023. – URL: <https://finadvice.mts.ru/blog/rost-kolichestva-moshennicheskikh-operatsii-za-2023-god-statistika-ot-tsb-rf> (дата обращения: 08.12.2023).

на запугивание жертвы и доведение его до стрессового состояния, в котором жертва произведет денежный перевод на счет мошенников. Переводы в большинстве таких случаев осуществляются через мобильное приложение или через банкомат, а именно, через устройства, в которых есть камера для получения и анализа видеосигнала. Таким образом имеется возможность отследить эмоциональное состояние клиента и пометить его транзакции, в случае, если они были совершены в стрессовом состоянии.

В данной статье рассмотрен метод определения подозрительных банковских транзакций с использованием нейросетевых технологий по распознаванию эмоций в период стрессового состояния клиента. Приведен анализ современных подходов и моделей нейронных сетей, используемых для решения задачи распознавания эмоций по видеоизображению. Дано описание используемого набора данных для обучения реализуемой модели, алгоритма предобработки данных и итоговой архитектуры модели на основе нейросетей с CNN и LSTM архитектурами.

1. Анализ архитектур нейронных сетей и методов машинного обучения, используемых в системах распознавания эмоций и обработки временных последовательностей данных

Для реализации предложенного в данной статье метода анализировались публикации, в которых применялись нейросетевые архитектуры и алгоритмы машинного обучения для обработки графической информации, представленной изображениями, последовательностями кадров или видео. А также статьи в области решения задачи по распознаванию эмоций и обработки последовательностей данных для выявления в них хронологических зависимостей.

В статье [7] приведен обзор современных методик и подходов к моно- или мультимодальному распознаванию эмоций, в том числе и по лицевой информации. В качестве традиционных методов извлечения лицевых признаков упоминаются следующие источники: Local Binary Pattern (LBP), Active Appearance Model (AAM), Active Shape Model (ASM), Histograms of Oriented Gradient (HOG), Gabor Wavelet Transform. К недостаткам традиционных методов относят необходимость ручного вмешательства в подготовку данных, а также неспособность извлечь и сохранить всю семантическую информацию лица, необходимую для определения эмоций. Для нивелирования данных особенностей традиционных методов предлагается использование сверточных нейронных сетей (CNN). Для работы с последовательностью изображений указывается применение архитектуры трехмерной сверточной нейронной сети 3D-CNN.

В публикации [8] приведен литературный обзор статей по теме распознавания эмоций по лицу с указанием используемых методов, наборов данных

и технологий. В большинстве работ выделяют 7 базовых эмоций, которые учитываются при обучении и тестировании построенных моделей: злость (anger), грусть (sadness), радость (happiness), отвращение (disgust), удивление (surprise), страх (fear) и нейтральное состояние (neutral). Для данных классов эмоций применяют группировку по двум категориям: позитивные эмоции (удивление, радость) и негативные эмоции (злость, грусть, отвращение, страх). Из приведенных в работе наборов данных, для дальнейшего поиска источников по получению к ним доступа были отобраны наборы CK+ и AFEW, как наиболее подходящие по описанию содержимого для решения поставленной задачи. В качестве широко используемого метода для извлечения лицевых признаков указывается нейросетевая архитектура сверточных нейронных сетей. Для определения лица на изображении используются также обученные нейронные сети или традиционный метод Виолы-Джонса.

В статьях [9–13] для выделения лицевых признаков при обработке изображений лица также используются вариации архитектуры сверточной нейронной сети.

В работе [14] используется комбинированная архитектура нейронной сети, состоящая из частей сверточной нейронной сети и блоков долгой краткосрочной памяти. Такая архитектура применяется для обработки последовательности данных, которая в работе представлена речевой записью. В публикации [15] также применяется комбинация сверточной и рекуррентной нейросетевых архитектур для обработки последовательности данных, представленных измерениями фотоплетизмографа. В статье [16] комбинация архитектур CNN и LSTM используется в качестве подхода для классификации действий человека по активностям на основе видеоданных. Как отмечают авторы, данный подход к распознаванию по видео показывает свою эффективность.

Статьи [17,18] демонстрируют возможность размещения и осуществления работы на мобильных устройствах и встраиваемых системах, построенных на базе глубокого машинного обучения.

Анализ публикаций показал, что самым популярным методом для извлечения признаков из графической информации является метод машинного обучения на основе сверточной нейронной сети (CNN). Данный метод решает недостатки традиционных подходов, а также обладает большим количеством изученных и исследованных вариаций моделей. Сверточная нейронная сеть может быть представлена адаптированной моделью из 3-х слоев, моделью по типу VGG-16 из 16-ти слоев, моделью по типу GoogLeNet из 22-х слоев и т.д. Рекуррентные нейронные сети (RNN) широко применяются для анализа

данных, представленных какими-либо временными последовательностями (последовательность измерений, звуковая запись, видео) и извлечения из них признаков, формирующих контекст и зависимость между элементами этой последовательности. Из архитектур рекуррентных нейронных сетей исследователи выделяют модель ячеек долгой краткосрочной памяти (LSTM). Ее преимуществами над традиционными моделями RNN являются: преодоление проблемы исчезающего градиента, способность запоминать информацию и учитывать ее на протяжении множества шагов, работа с последовательностями переменной длины. У LSTM модели также есть усовершенствованная модель Bi-LSTM, которая имеет возможность обрабатывать информацию от прошлых значений к будущим, а также в обратном порядке от будущих значений к прошлым.

Исследователи в задачах распознавания глубоких эмоций (усталость, стресс и т.д.) используют как классические наборы данных с базовыми эмоциями, вручную группируя их по признакам в рамках своей задачи, так и наборы данных, собранные с учетом специфики распознаваемых видов эмоций. В качестве популярных метрик оценки качества разрабатываемых систем для решения задач много классовой классификации применяется метрика ассигасы и матрица ошибок (confusion matrix).

2. Разработка системы распознавания стрессового эмоционального состояния клиента банка по видеоданным

Для реализации системы распознавания стрессового эмоционального состояния по лицевым данным, полученным из видео, использовался следующий инструментарий: язык программирования Python, библиотека алгоритмов компьютерного зрения OpenCV, библиотека машинного обучения PyTorch [19], среда разработки PyCharm и интерактивная облачная среда для разработки и выполнения кода Google Colab [20].

2.1. Используемые наборы данных (dataset-ы)

Для обучения и тестирования построенной нейронной сети необходимо использовать наборы коротких видео с людьми, испытывающими эмоции, распределенные по категориям: нормальное состояние (Normal), слабое стрессовое состояние (Weak stress) и сильное стрессовое состояние (Strong stress) [21].

В качестве основного обрабатываемого набора данных использовался DFEW [22], являющийся похожим на набор AFEW, но имеющий упрощенный способ получения доступа к его данным. DFEW составлен на основе 15 906 видеоклипов. Из каждого такого видеоклипа извлекались кадры, на которых определялось лицо, после чего кадр обрезался в соответствии с прямоугольной зоной найденного

лица. В свою очередь, видеоклипы прошли процесс классификации по 7 базовым эмоциям, которую осуществляли эксперты. После проведения такой обработки данных пользователям-исследователям набора данных предлагается использовать 11 697 объектов, размеченных по 7 классам (Happy, Sad, Neutral, Angry, Surprise, Disgust, Fear), где каждый объект представлен набором-последовательностью изображений извлеченного лица из видеоклипа.

Таблица 1

Параметры набора данных DFEW

№	Параметр	Значение
1	Количество объектов общее, шт.	11 697
2	Количество объектов класса Happy, шт.	2445
3	Количество объектов класса Sad, шт.	1894
4	Количество объектов класса Neutral, шт.	2669
5	Количество объектов класса Angry, шт.	2173
6	Количество объектов класса Surprise, шт.	1469
7	Количество объектов класса Disgust, шт.	145
8	Количество объектов класса Fear, шт.	902
9	Максимальная высота кадра, пикс.	256
10	Минимальная высота кадра, пикс.	256
11	Максимальная ширина кадра, пикс.	256
12	Минимальная ширина кадра, пикс.	256
13	Максимальное количество кадров у объекта, шт.	847
14	Среднее количество кадров у объекта, шт.	70
15	Минимальное количество кадров у объекта, шт.	20
16	Цветовая модель изображений	RGB

После адаптации данных для обучения и тестирования разрабатываемой нейронной сети, итоговый набор выглядит таким образом, как представлено на рис. 1.

Применительно к данному набору использовался способ с распределением базовых классов эмоций по целевым категориям. Исходя из круговой модели классификации эмоций Рассела [23], было решено распределить данные базовых эмоций следующим образом: Normal (Happy, Neutral), Weak stress (Sad, Surprise, Disgust) и Strong stress (Angry, Fear), представлено на рис. 2.

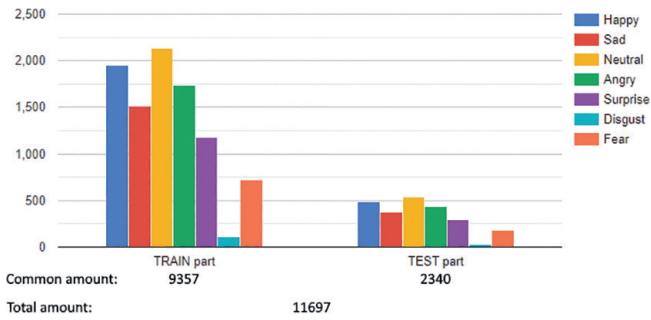


Рис. 1. Распределение данных по классам базовых эмоций по обучающей и тестирующей выборкам полученного набора из DFEW с классификацией 7 базовых эмоций

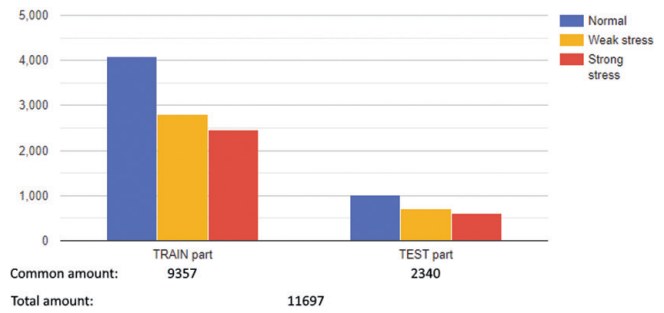


Рис. 2. Распределение данных по целевым категориям для распознавания после группировки базовых эмоций

2.2. Модель нейронной сети

Предлагаемая модель нейронной сети 3DCNN-LSTM построена с применением комбинации сверточных нейронных сетей (CNN) и долгой-краткосрочной памяти (LSTM). Данная архитектура позволяет извлекать признаки из последовательности графической информации, а потом выявлять в них скрытые связи и закономерности. Графическое представление архитектуры реализуемой модели представлено на рис. 3.

На вход нейронной сети подается последовательность из 20 изображений размером 128 x 128 пикселей с 3 цветовыми каналами. Параметр входного количества изображений был выбран исходя из минимального значения количества кадров для объекта во всем наборе данных. Размер подаваемого на вход изображения был подобран эвристическим путем. Во-первых, чтобы обеспечить обработку изображения в хорошем качестве, где лицо на изображении отображается без эффекта пиксельности. Во-вторых, чтобы обеспечить количество слоев сверточной нейронной сети, каждый из которых заканчивается подвыборкой, уменьшающей изображение вдвое. А также для сведения изображения до размеров, позволяющих преобразовать (flatten) его в вектор той размерности, которая требуется, чтобы у LSTM

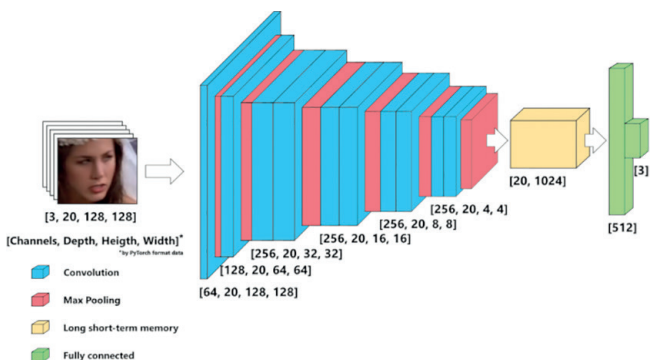


Рис. 3. Графическое представление архитектуры реализуемой модели

блока было количество обучаемых параметров, уместяющееся в оперативную память при обучении сети, и кроме того, не увеличивающее обучение этой сети до нецелесообразно долгих периодов времени.

Одновременно с предыдущим условием, необходимо, чтобы при выбранном количестве слоев не наблюдалось эффекта затухания градиента. Количество каналов было бы выбрано из учета особенности обработки кадров в наборе данных. При извлечении изображения лица из кадров видеоданных, после обнаружения его области, необходимо чтобы данное лицо могло быть выровнено путем применения аффинных преобразований, что приводит к появлению пустых областей в кадре, закрашенных черным цветом, как изображено на рис. 5. При выборе одного канала (оттенки серого), данная область могла внести помехи в процесс обучения, поэтому выбор был сделан в пользу 3 каналов (RGB).

```

Layer (type:depth-idx)      Output Shape      Param #
-----
CNN3D_LSTM_Net
-Conv3d: 1-1                [8, 64, 20, 128, 128] 1,792
-MaxPool3d: 1-2            [8, 64, 20, 64, 64] --
-Dropout: 1-3              [8, 64, 20, 64, 64] --
-Conv3d: 1-4                [8, 128, 20, 64, 64] 73,856
-MaxPool3d: 1-5            [8, 128, 20, 32, 32] --
-Dropout: 1-6              [8, 128, 20, 32, 32] --
-Conv3d: 1-7                [8, 256, 20, 32, 32] 295,168
-Conv3d: 1-8                [8, 256, 20, 32, 32] 590,080
-MaxPool3d: 1-9            [8, 256, 20, 16, 16] --
-Dropout: 1-10            [8, 256, 20, 16, 16] --
-Conv3d: 1-11              [8, 256, 20, 16, 16] 590,080
-Conv3d: 1-12              [8, 256, 20, 16, 16] 590,080
-MaxPool3d: 1-13          [8, 256, 20, 8, 8] --
-Dropout: 1-14            [8, 256, 20, 8, 8] --
-Conv3d: 1-15              [8, 256, 20, 8, 8] 590,080
-Conv3d: 1-16              [8, 256, 20, 8, 8] 590,080
-MaxPool3d: 1-17          [8, 256, 20, 4, 4] --
-Dropout: 1-18            [8, 256, 20, 4, 4] --
-Conv3d: 1-19              [8, 256, 20, 4, 4] 590,080
-Conv3d: 1-20              [8, 256, 20, 4, 4] 590,080
-MaxPool3d: 1-21          [8, 256, 20, 2, 2] --
-Flatten: 1-22             [8, 20, 1024] --
-LSTM: 1-23                [8, 20, 512] 3,149,824
-Linear: 1-24              [8, 512] 262,656
-Dropout: 1-25            [8, 512] --
-Linear: 1-26              [8, 3] 1,539
-----
Total params: 7,915,395
Trainable params: 7,915,395
    
```

Рис. 4. Подробное представление модели с помощью встроенной функции библиотеки PyTorch summary()



Рис. 5. Пример области пустых пикселей, закрашенных черным цветом

На первом этапе выделения признаков из объектов данных последовательность кадров проходит обработку через структуру 3DCNN сверточной нейронной сети. В ходе такой обработки извлекаются пространственные признаки каждого изображения в последовательности, которые затем подаются на вход структуре долгой-краткосрочной памяти LSTM. Данная структура анализирует и выявляет основные временные признаки последовательности. На последнем этапе данные из LSTM блока подаются на вход блока, состоящего из полносвязных слоев. Последний полносвязный слой содержит 3 нейрона, выход из которых показывает принадлежность классу.

2.2.1. Сверточный слой (Convolutional layer)

В сверточном слое к данным, представленным *n*-мерной структурой, применяются *n*-мерные ядра свертки с заданным шагом движения по исходной структуре данных. После прохождения всех ядер по данным формируется выходная карта признаков. В общем случае, математическая формула 3D сверточного слоя, выглядит следующим образом. Для входной структуры данных *X* формата (*C_{in}*, *D_{in}*, *H_{in}*, *W_{in}*), где *C_{in}* – количество каналов, *D_{in}* – глубина, *H_{in}* – высота, *W_{in}* – ширина. С ядром (фильтром) *W* размерности (*C_{in}*, *D_{kernel}*, *H_{kernel}*, *W_{kernel}*):

$$Y_{i,j,k} = \sum_{l=1}^{C_{in}} \sum_{m=1}^{D_{kernel}} \sum_{n=1}^{H_{kernel}} \sum_{p=1}^{W_{kernel}} X_{l,i+m-1,j+n-1,k+p-1} * W_{l,m,n,p} + bias \tag{1}$$

Эта формула выполняется независимо для каждого выходного канала *C_{out}*. Размеры выходной структуры данных (*C_{out}*, *H_{out}*, *W_{out}*) рассчитываются с учетом шага сдвига ядра (*D_{stride}*, *H_{stride}*, *W_{stride}*) следующим образом:

$$\begin{aligned} D_{out} &= \left\lfloor \frac{D_{in} - D_{kernel} + 1}{D_{stride}} \right\rfloor \\ H_{out} &= \left\lfloor \frac{H_{in} - H_{kernel} + 1}{H_{stride}} \right\rfloor \\ W_{out} &= \left\lfloor \frac{W_{in} - W_{kernel} + 1}{W_{stride}} \right\rfloor \end{aligned} \tag{2}$$

2.2.2. Слой подвыборки (Pooling layer)

Слой подвыборки или пулинга применяется для уменьшения карт признаков с сохранением наиболее важной информации. В данной модели используется слой 3D макспулинга (3D Max Pooling Layer), его математическая формула в общем виде выглядит следующим образом. Для входной структуры данных *X* размерности (*C*, *D_{in}*, *H_{in}*, *W_{in}*), где *C* – количество каналов, *D_{in}* – глубина, *H_{in}* – высота, *W_{in}* – ширина. С размером окна (*D_{pool}*, *H_{pool}*, *W_{pool}*) и параметрами шага (*D_{stride}*, *H_{stride}*, *W_{stride}*):

$$Y_{c,i,j,k} = \max_{m=1 \dots D_{pool}} \max_{n=1 \dots H_{pool}} \max_{p=1 \dots W_{pool}} X_{c,(i-1) \times D_{stride} + m,(j-1) \times H_{stride} + n,(k-1) \times W_{stride} + p} \tag{3}$$

Размеры выходной структуры данных (*C*, *D_{out}*, *H_{out}*, *W_{out}*) рассчитываются следующим образом:

$$\begin{aligned} D_{out} &= \left\lfloor \frac{D_{in} - D_{pool} + 1}{D_{stride}} \right\rfloor \\ H_{out} &= \left\lfloor \frac{H_{in} - H_{pool} + 1}{H_{stride}} \right\rfloor \\ W_{out} &= \left\lfloor \frac{W_{in} - W_{pool} + 1}{W_{stride}} \right\rfloor \end{aligned} \tag{4}$$

Модель долгой-краткосрочной памяти (LSTM)

Нейронная сеть LSTM состоит из цепочки блоков (ячеек), в которые подаются признаки из последовательности. Особенностью ячейки LSTM от традиционных рекуррентных нейронных сетей является наличие дополнительного канала, который позволяет сохранять важную информацию в долгосрочной перспективе. Внутреннее устройство ячейки LSTM приведено на рис. 6.

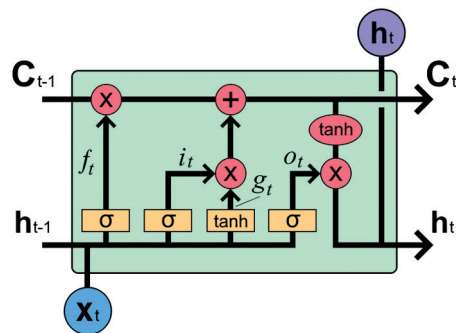


Рис. 6. Внутреннее устройство ячейки LSTM

Условные обозначения:

- C_{t-1}* – вектор долгосрочного контента с предыдущего шага;
- h_{t-1}* – вектор признаков скрытого состояния с предыдущего шага;
- x_t* – входной вектор признаков текущего шага;
- h_t* – вектор признаков скрытого состояния, полученный на текущем шаге;

- C_t – вектор долгосрочного контента, полученный на текущем шаге;
- σ – слой нейронной сети с сигмоидальной функцией активации;
- \tanh – слой нейронной сети с функцией активации гиперболический тангенс;
- \oplus – поэлементное сложение;
- \otimes – поэлементное умножение;
- \tanh – поэлементное вычисление гиперболического тангенса.

Вычисления, происходящие в ячейке LSTM, где W_j – весовые коэффициенты слоя j , $[a,b]$ – объединение векторов a и b и b_j – это смещение (bias) на слое j , выглядят следующим образом:

1. Для фазы утраты информации (забывание ненужного):

$$f_t = \sigma(W_f * [h_{t-1}, x_t] + b_f)$$

2. Для фазы сохранения информации (запоминание нового):

$$i_t = \sigma(W_i * [h_{t-1}, x_t] + b_i),$$

$$g_t = \tanh(W_g * [h_{t-1}, x_t] + b_g)$$

3. Для фазы нового состояния (формирование выходного значения в канале долгосрочного контента и выходного скрытого состояния):

$$o_t = \sigma(W_o * [h_{t-1}, x_t] + b_o),$$

$$C_t = f_t \otimes C_{t-1} \oplus i_t \otimes g_t, \quad (5)$$

$$h_t = o_t \otimes \tanh(C_t)$$

2.2.4. Полносвязный слой (Dense layer)

Стандартный полносвязный нейронный слой состоит из связанных между собой перцептронов. В общем случае, математическая формула данного слоя выглядит следующим образом:

$$Y = \sum_{i=1}^{Amount_{prev\ layer\ neurons}} (w_i * x_i) + bias \quad (6)$$

2.2.5. Слой исключения (Dropout)

Специализированный слой регуляризации нейронной сети, используемый для предотвращения или уменьшения влияния эффекта переобучения сети. Данный слой работает так, что во время процесса обучения, он случайным образом приравнивает элементы входной последовательности к нулю с вероятностью, заданной параметром, тем самым предотвращая взаимoadaptацию нейронов.

2.2.6. Функции активации

В разрабатываемой модели предполагается использование следующих функций активаций: выпрямитель (ReLU), сигмоидальная функция (Sigmoid), гиперболический тангенс tanh. Данные функции

применяются к каждому значению входной структуры данных, представленной изображением или тензором, при преобразовании входных данных в выходные с помощью заданной функции.

Функция активации ReLU применяется после каждого слоя свертки, а также после каждого полносвязного слоя, кроме выходного, в сверточной нейронной сети. График функции представлен на рис. 7. Популярность ее применения в глубоких сверточных нейронных сетях обусловлена ее нетребовательностью к вычислительным ресурсам за счет выполнения простых математических операций, разреженностью активации, а также возможностью роста значения на выходе нейрона, увеличивающего влияние активации этого нейрона.

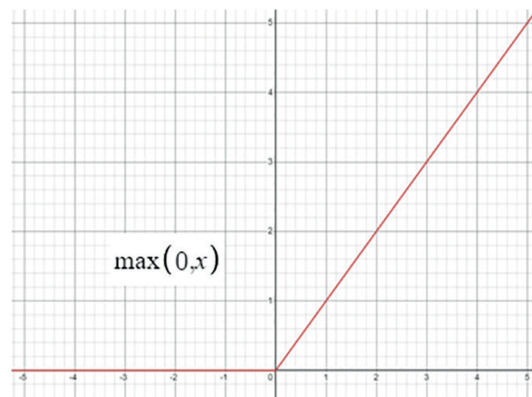


Рис. 7. График функции активации ReLU

Сигмоидальная функция активации преобразует входное значение, лежащее в интервале от отрицательной бесконечности до положительной бесконечности в значение от 0 до 1. В разрабатываемой модели данная функция используется в ячейке LSTM. Данная функция за счет своей области значений в блоке LSTM выступает в качестве «ключа», регулируя, какие данные будут интегрированы в канал хранения долгосрочного контента. График функции представлен на рис. 8.

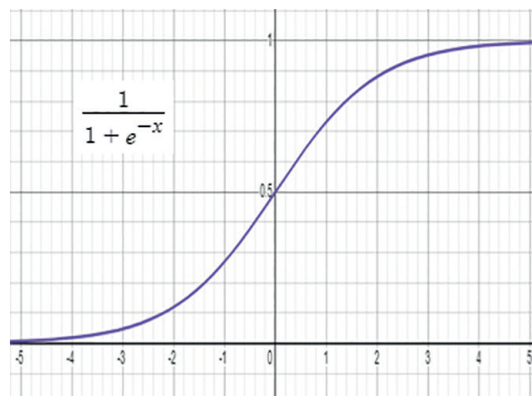


Рис. 8. График функции активации Sigmoid

Функция активации гиперболический тангенс также применяется в ячейке LSTM. Ее применение в паре с сигмоидальной функцией обусловлено тем, что на практике сходимость решения происходит быстрее и уменьшается влияние проблемы затухания градиента. Ее выходные значения лежат в диапазоне от -1 до 1, а график функции представлен на рис. 9.

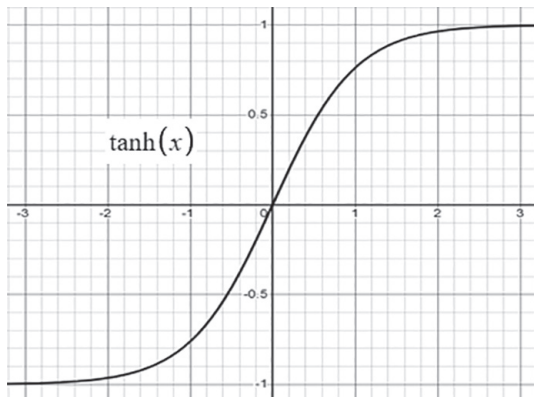


Рис. 9. График функции активации \tanh

В данной работе решается задача много классовой классификации, в контексте которой широко применяется функция активации Softmax. Данная функция часто используется в выходном нейронном слое для нормализации значений слоя в вероятностное распределение, где нейрон с наибольшим значением будет иметь высокое значение вероятности. Формула

расчета значений по входному вектору размерности n , для каждого значения x_i приведена ниже:

$$\text{softmax}(x_i) = \frac{\exp(x_i)}{\sum_{j=1}^n \exp(x_j)} \quad (7)$$

На этапе написания программного кода и реализации предложенной в статье модели, было принято решение об отказе использования данной функции активации из-за особенностей работы используемой функции потерь CrossEntropy из библиотеки машинного обучения PyTorch. В ходе тестирования выявилось, что при использовании данной функции скорость обучения снижается, а также значение потери на эпохах обучения не стремится к нулю, а имеет более высокий порог остановки. Также, в различных примерах, приведенных на официальных ресурсах⁸, в которых используется данная библиотека, видно, что исследователи не всегда применяют ее в своих моделях.

3. Программная реализация и экспериментальные результаты

Программная реализация описанной ранее модели состоит из следующих этапов:

- импорт необходимых библиотек;
- реализация необходимых классов и функций для работы с набором данных;

8 1. Training a Classifier // PyTorch.org : сайт. - URL: https://pytorch.org/tutorials/beginner/blitz/cifar10_tutorial.html (дата обращения: 20.12.2023).
2. What is torch.nn really? // PyTorch.org : сайт. - URL: https://pytorch.org/tutorials/beginner/nn_tutorial.html (дата обращения: 21.12.2023).

```
class DFew_DataSet(Dataset):
    def __init__(self, root_path_of_part_directory, path_of_excel_annotation, transform=None):
        self.root_path = root_path_of_part_directory
        self.path_of_excel_annotation = path_of_excel_annotation
        xls = pd.ExcelFile(path_of_excel_annotation)
        sheetX = xls.parse(0)
        self.length = len(sheetX["video_name, label"])
        self.transform = transform
        anno_list = []
        for i in range(len(sheetX["video_name, label"])):
            anno_list.append(str(sheetX["video_name, label"][i]).split("."))
        self.anno_list = anno_list.copy()
    def __len__(self):
        return self.length
    def __getitem__(self, idx):
        name, label = self.anno_list[idx]
        while (len(name) != 5):
            name = "0" + name
        path_to_item = self.root_path + "\\\" + name
        res = frames_from_directory_reading(path_to_item)
        for i in range(len(res)):
            res[i] = self.transform(res[i])
        res = torch.stack(res, 0)
        sample = (res, torch.as_tensor(dfew_classes_labels_union[int(label)-1]))
        return sample
```

Схема 1

- трансформация данных в требуемый вид для обучения и тестирования;
- построение модели нейронной сети, обучение, тестирование и вывод оценочных показателей.

При написании программной части работы были задействованы следующие библиотеки: PyTorch (библиотека, предоставляющая возможности осуществления машинного обучения), OpenCV (библиотека, предоставляющая возможности работы с фото, видео-файлами и другой графической информацией), numpy (библиотека, предоставляющая возможности для математических вычислений и преобразований, а также работы с многомерными массивами данных), matplotlib (библиотека, предоставляющая возможности для построения графиков), pandas (библиотека, предоставляющая возможности для работы с таблицами, в том числе и с excel-таблицами), tqdm (библиотека, предоставляющая возможности для визуализации прогресса работы при итерации по массивам данных).

Для осуществления возможности использования стороннего (не встроенного в саму библиотеку, как класс) набора данных, необходимо реализовать собственный класс, наследуемый от класса `torch.utils.data.Dataset`, в котором будут переопределены методы `__len__` и `__getitem__` (Схема 1).

Для перевода данных в вид, подходящий для загрузки данных на вход нейронной сети для процесса обучения и тестирования, необходимо определить данные, представленные объектом класса `Dataset`, как объект класса `torch.utils.data.DataLoader`. Данное преобразование позволяет формировать батчи данных, а также итерироваться по набору данных, работая только с определенным количеством объектов, загруженных в оперативную память. Далее приведем фрагмент примера кода, выполняющего

```
class CNN3D_LSTM(nn.Module):
    def __init__(self):
        super().__init__()
        # 1
        self.conv1_1 = nn.Conv3d(3,
            64, kernel_size=(1, 3, 3), padding="same")
        self.pool1 = nn.MaxPool3d((1, 2, 2), (1, 2, 2))
        self.dropout1 = nn.Dropout(p=0.25, inplace=False)
        # 2
        ...

    def forward(self, x):
        x = x.transpose(1, 2)
        x = F.relu(self.conv1_1(x))
        x = self.pool1(x)
        x = self.dropout1(x)
        ...
        return x
```

данные преобразования для тестовой части набора данных:

```
test_data = DFEW_Dataset(...params for class...)
batch_size_var=8
test_loader = DataLoader(test_data, batch_size=batch_size_var, shuffle = False)
```

После работы с данными, наступает этап построения нейросетевой модели. Библиотека PyTorch позволяет строить нейронные сети различными способами, предлагая разработчикам-исследователям разный уровень контроля над такими параметрами сети, как количество слоев, размерности данных на входе и выходе каждого слоя, функции активации слоев и т.д. В текущем варианте исполнения был выбран способ, в котором реализуется класс нейронной сети с двумя методами: `__init__` и `forward`. В методе `__init__` определяется каждый слой сети, со своими требуемыми параметрами, в методе `forward` расписывается порядок прохождения данных, а также применяемые функции активации к значениям, полученным с выходов слоев. Сокращенный пример фрагмента кода демонстрирует данный способ (Схема 2).

Далее идет реализация цикла обучения. Для тренировки нейронной сети использовался стандартный цикл, взятый из примеров документации к библиотеке. Перед циклом определяются: количество эпох, алгоритм оптимизации и функция потерь. Затем начинается цикл обучения, фрагмент которого приведен далее:

```
for epoch in tqdm(range(num_epochs)):
    net.train()
    running_loss = 0.0
    for i, batch in enumerate(tqdm(train_loader)):
        x_batch, y_batch = batch
        x_batch=x_batch.to(device)
        y_batch=y_batch.to(device)
```

Схема 2

```
optimizer.zero_grad()
y_pred = net(x_batch)
loss=loss_fn(y_pred,y_batch)
loss.backward()
optimizer.step()
```

Таблица 2

Параметры для обучения нейронной сети

№	Параметр	Значение
1	Размер батча	8
2	Количество обрабатываемых кадров, взятых из видео	20
3	Алгоритм оптимизации	Adam
4	Функция потерь	Cross Entropy
5	Количество эпох	50
6	Параметр p для слоя исключения Dropout внутри сверточных слоев	0.25
7	Параметр p для слоя исключения Dropout внутри полносвязных слоев	0.5
8	Параметр скорости обучения	0.0001

Отметим основные функции, участвующие в цикле. Функция `net.train()` – переводит нейронную сеть в режим обучения. В режиме обучения у сети вычисляются градиенты и работают вспомогательные слои, предотвращающие эффект переобучения, наподобие `nn.Dropout()`. Функция `optimizer.zero_grad()` – вызывается для обнуления рассчитанных на предыдущем шаге градиентов, чтобы не было эффекта «накопления». Функция `loss.backward()` – вызывается для вычисления градиентов. Функция `optimizer.step()` – вызывается для осуществления обновления весовых коэффициентов сети.

После цикла обучения идет цикл тестирования нейронной сети и вычисления различных оценочных метрик. Фрагмент основного цикла тестирования для вычисления точности работы сети приведен далее:

```
with torch.no_grad():
    net.eval()
    for data in tqdm(test_loader):
        x_batch, y_batch = data
        x_batch = x_batch.to(device)
        y_batch = y_batch.to(device)
        y_pred = net(x_batch)
        _, predicted = torch.max(y_pred.data, 1)
        total += y_batch.size(0)
        correct += (predicted == y_batch).sum().item()
```

Основным отличием от цикла обучения является применение здесь следующих функций: `torch.no_grad()` – вызывается в контекстном менеджере для отключения вычисления градиентов, так как в этом нет необходимости на этапе тестирования, что экономит вычислительные и временные ресурсы; `net.eval()` – переводит нейронную сеть в режим оценки, отключая работу вспомогательных слоев, предотвращающих эффект переобучения, наподобие `nn.Dropout()`.

Процесс обучения модели производился с применением параметров, указанных в таблице 2.

Обучение производилось с переносом вычислительной логики на GPU при помощи технологии CUDA на видеокарте NVIDIA GeForce RTX 3080. График потерь на каждой эпохе представлен на рис. 10.

Показатели точности распознавания вычислялись по всей тестовой выборке, а также отдельно по классам в ней. Итоговая общая точность алгоритма, подсчитанная с использованием метрики `accuracy`, составила: 87%. Показатели точности распознавания целевых классов представлены на рис. 11.

Для оценки эффективности решения поставленной задачи, предлагаемой нейросетевой модели,

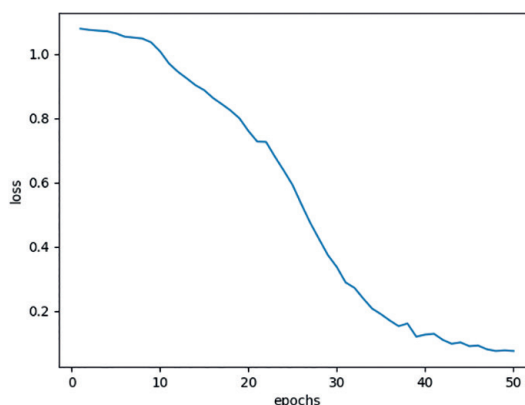


Рис. 10. График потерь на каждой эпохе

```
Accuracy for class: Normal is 90.3 %
Accuracy for class: Weak Stress is 83.1 %
Accuracy for class: Strong Stress is 87.8 %
```

Рис. 11. Показатели точности распознавания целевых классов

были также реализованы структуры с применением базовых сетей 3DCNN и LSTM. На каждой из структур был проведен процесс обучения, а затем тестирования, с вычислением общей точности работы алгоритма распознавания. В ходе адаптации базовых моделей к решаемой задаче были осуществлены следующие действия: для 3DCNN архитектуры после сверточных слоев и перед выпрямлением карт активации для подачи на вход полносвязного слоя применялся метод усреднения 3D тензора по оси количества кадров. Для LSTM архитектуры была уменьшена размерность кадров со 128 пикселей до 45, а также уменьшено количество каналов с 3 до 1, с тем чтобы количество параметров сети позволяло производить обучение за целесообразное время и объем памяти, занимаемый ими, был меньше

Решение поставленной задачи с помощью разных структур нейронных сетей

Метод	Кол-во эпох	Размер батча	Функция потерь	Алгоритм оптимизации	Количество обучаемых параметров	Точность
3DCNN	50	8	Cross Entropy	Adam	5,027,715	48%
LSTM	50	8	Cross Entropy	Adam	13,549,571	45%
3DCNN+LSTM	50	8	Cross Entropy	Adam	7,915,395	87%

оперативной памяти компьютера. Полученные результаты точности работы распознавания с применением разных архитектур нейронных сетей приведены в таблице 3.

Также, для визуализации точности работы предлагаемой модели была построена матрица путаницы (confusion matrix). Каждая строка матрицы отражает целевой класс эмоций, а каждый столбец – прогнозируемый моделью класс эмоций. Матрица путаницы представлена на рис. 12.

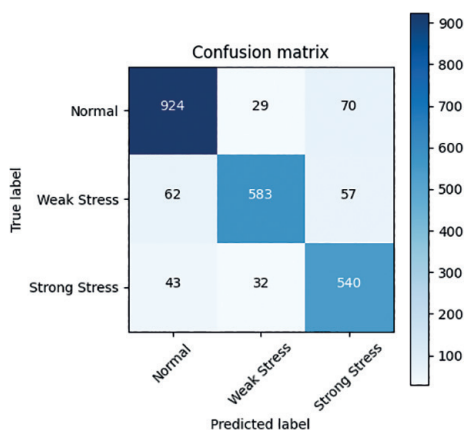


Рис. 12. Матрица путаницы

Экспериментальный результат работы построенной модели показал, что представленный метод позволяет решить задачу обнаружения негативного (сильного стрессового) состояния человека по видеоданным его лица. Точность работы алгоритма является удовлетворительной для эксперимента, так как для обучения использовался набор данных, составленный из «in the wild» видеоклипов, в которых присутствуют различные повороты головы, артефакты в виде затемнения или световых бликов на лицевой области и т.п.

Структура данной модели подходит для работы с графической информацией, представленной последовательностью изображений. Скорость обучения вне учета параметра learning rate прямо пропорционально зависит от размерности признаков на входном и скрытом слоях ячеек LSTM, так как при текущих параметрах сети данный блок содержит наибольшее

количество обучаемых параметров по сравнению с другими слоями. Использование в полносвязных слоях блоков исключения (dropout) позволило избежать переобучения сети.

Заключение

В данной статье представлен метод выявления транзакций, совершаемых клиентами, подвергнутыми психологическому воздействию мошенников, на основе использования нейросетевой модели распознавания эмоций. Архитектура модели составлена из комбинации сверточной нейронной сети (3DCNN) и рекуррентной нейронной сети (в качестве долгой краткосрочной памяти (LSTM)). Данная модель позволяет работать с последовательностью изображений лица, полученных из видеоданных. Построение нейронной сети, ее обучение и тестирование были реализованы на языке Python с использованием библиотеки PyTorch. Экспериментальные результаты подтвердили работоспособность данного подхода и показали перспективность его применения для определения негативного (сильного стрессового) эмоционального состояния человека с целью отнесения совершенных им транзакций к категории подозрительных, поскольку данное состояние может быть обусловлено психологическим воздействием мошенников. Данный метод наряду с методами распознавания мошеннического воздействия по телефону, а также антифрод-системами, позволит выстроить много эшелонированную защиту пользователей банковских услуг, способную выявлять подозрительные транзакции и рассматривать их в особом порядке.

Дальнейшими исследованиями данной тематики является уменьшение входной размерности признаков в блоке LSTM с сохранением точности работы алгоритма для увеличения скорости обучения и уменьшения занимаемого объема памяти весами нейронной сети за счет уменьшения обучаемых параметров. Планируется повышение точности за счет использования других наборов данных, таких же, как и DFEW, составленных «in the wild». Далее, планируется анализ и изучение возможности замены блока LSTM на bi-LSTM или GRU, а также расширение архитектуры сети для обработки входных данных, представленных координатами ключевых точек лица.

Литература

1. Ahmad F. M. Mansor, Ahmad A. Zainuddin, Zulkeflee Khalidin. Patient Monitoring System using Computer Vision for Emotional Recognition and Vital Signs Detection. // ResearchGate: портал. – 2020. – URL: https://www.researchgate.net/publication/344399775_Patient_Monitoring_System_using_Computer_Vision_for_Emotional_Recognition_and_Vital_Signs_Detection (дата обращения: 02.12.2023).
2. Mei Wang, Ziyang Huang, Yuancheng Li, Lihong Dong, Hongguang Pan. Maximum weight multi-modal information fusion algorithm of electroencephalographs and face images for emotion recognition // Computers & Electrical Engineering. – 2021. – Vol. 94. – DOI: 10.1016/j.compeleceng.2021.107319. – ISSN 0045-7906.
3. Cuiting Xu, Chunchuan Yan, Mingzhe Jiang, Fayadh Alenezi, Adi Alhudaif, Norah Alnaim, Kemal Polat, Wanqing Wu. A novel facial emotion recognition method for stress inference of facial nerve paralysis patients // Expert Systems with Applications. – 2022. – Vol. 197. – DOI: 10.1016/j.eswa.2022.116705. – ISSN 0957-4174.
4. Shichuan Du, Aleix M. Martinez. Compound facial expressions of emotion: from basic research to clinical applications // Dialogues in Clinical Neuroscience. – 2015. – № 17:4. – Pages 443–455. – DOI: 10.31887/DCNS.2015.17.4/sdu.
5. Zhongshan Chen, Xinning Feng, Shengwei Zhang. Emotion detection and face recognition of drivers in autonomous vehicles in IoT platform // Image and Vision Computing. – 2022. – Vol. 128. – DOI: 10.1016/j.imavis.2022.104569. – ISSN 0262-8856.
6. Zepf Sebastian, Hernandez Javier, Schmitt Alexander, Minker Wolfgang, Picard Rosalind. Driver Emotion Recognition for Intelligent Vehicles: A Survey // ACM Computing Surveys. – 2020. – DOI: 10.1145/3388790.
7. Lian H, Lu C, Li S, Zhao Y, Tang C, Zong Y. A Survey of Deep Learning-Based Multi-modal Emotion Recognition: Speech, Text, and Face // Entropy (Basel). – 2023. – № 25(10):1440. – DOI: 10.3390/e25101440.
8. Prameela Naga, Swamy Das Marri, Raiza Borreo. Facial emotion recognition methods, datasets and technologies: A literature survey // Materials Today: Proceedings. – 2023. – Vol. 80. – Pages 2824–2828. – DOI: 10.1016/j.matpr.2021.07.046.
9. Chahak Gautam, K. R Seeja. Facial emotion recognition using Handcrafted features and CNN // Procedia Computer Science. – 2023. – Vol. 218. – Pages 1295–1303. – DOI: 10.1016/j.procs.2023.01.108.
10. Zia Ullah, Lin Qi, Asif Hasan, Muhammad Asim. Improved Deep CNN-based Two Stream Super Resolution and Hybrid Deep Model-based Facial Emotion Recognition // Engineering Applications of Artificial Intelligence. – 2022. – Vol. 116. – DOI: 10.1016/j.engappai.2022.105486.
11. Elham S. Salama, Reda A. El-Khoribi, Mahmoud E. Shoman, Mohamed A. Wahby Shalaby. A 3D-convolutional neural network framework with ensemble learning techniques for multi-modal emotion recognition // Egyptian Informatics Journal. – 2021. – Vol. 22. – Issue 2. – Pages 167–176. – DOI: 10.1016/j.eij.2020.07.005.
12. Radha Priyadharsini G, Krishnaveni K. A novel framework using binary attention mechanism based deep convolution neural network for face emotion recognition // Measurement: Sensors. – 2023. – Vol. 30. – DOI: 10.1016/j.measen.2023.100881.
13. Anjali R, J. Babitha, Rithika W, Ms. Reeya S.L. Stress Detection Based on Emotion Recognition Using Deep Learning // National Conference on Smart Systems and Technologies. – 2021. – Vol. 8. – Issue 7. – Pages 109–114.
14. Orhan Atila, Abdulkadir Şengür. Attention guided 3D CNN-LSTM model for accurate speech based emotion recognition // Applied Acoustics. – 2021. – Vol. 182. – DOI: 10.1016/j.apacoust.2021.108260.
15. Wafa Mellouk, Wahida Handouzi. CNN-LSTM for automatic emotion recognition using contactless photoplethysmographic signals // Biomedical Signal Processing and Control. – 2023. – Vol. 85. – DOI: 10.1016/j.bspc.2023.104907.
16. El Mehdi Saoudi, Jaafar Jaafari, Said Jai Andaloussi. Advancing human action recognition: A hybrid approach using attention-based LSTM and 3D CNN // Scientific African. – 2023. – Vol. 21. – DOI: 10.1016/j.sciaf.2023.e01796.
17. Emanuel Di Nardo, Vincenzo Santopietro, Alfredo Petrosino. Emotion recognition at the edge with AI specific low power architectures // Microprocessors and Microsystems. – 2021. – Vol. 85. – DOI: 10.1016/j.micpro.2021.104299.
18. Yi Chen, Jun Bin, Chao Kang. Application of machine vision and convolutional neural networks in discriminating tobacco leaf maturity on mobile devices // Smart Agricultural Technology. – 2023. – Vol. 5. – DOI: 10.1016/j.atech.2023.100322.
19. Deyuan Qu, Sudip Dhakal, Dominic Carrillo. Facial Emotion Recognition using CNN in PyTorch. – URL: <https://arxiv.org/pdf/2312.10818.pdf> (дата обращения: 14.12.2023).
20. Glen Berman. Machine Learning practices and infrastructures. – URL: <https://arxiv.org/pdf/2307.06518.pdf> (дата обращения: 14.12.2023).
21. Ramesh Naidu P, Pruthvi Sagar S, Praveen K, Kiran K, Khalandar K. Stress Recognition Using Facial Landmarks and Cnn (Alexnet) // Journal of Physics: Conference Series. – 2021. – 2089(1):012039 – DOI: 10.1088/1742-6596/2089/1/012039.
22. Xingxun Jiang, Yuan Zong, Wenming Zheng, Chuangao Tang, Wanchuang Xia, Cheng Lu, Jiateng Liu. DFEW: A Large-Scale Database for Recognizing Dynamic Facial Expressions in the Wild // Proceedings of the 28th ACM International Conference on Multimedia. – 2020. – Pages 2881–2889. – DOI: 10.48550/arXiv.2008.05924.
23. Andrea Scarantino. Core Affect and Natural Affective Kinds // Philosophy of Science. – 2009. – Vol. 76. – Issue 5. – Pages 940–957. – DOI: 10.1086/605816

