МРНТИ 28.23.37

10.51889/2959-5894.2025.92.4.016

М.С. Жасузак[1,2*], Ж.Н. Кудиретулла[1], Ж.А. Бурибаев[1,2], А.С. Еримбетова[2]

[1]Казахский национальный университет имени аль-Фараби, г.Алматы, Казахстан.
[2]Институт информационных и вычислительных технологий, г.Алматы, Казахстан
[*]e-mail: zhassuzak.mukhtar@gmail.com

РАЗРАБОТКА СИСТЕМЫ РАСПОЗНАВАНИЯ КАЗАХСКОГО ЖЕСТОВОГО ЯЗЫКА С ИСПОЛЬЗОВАНИЕМ МЕТОДОВ КОМПЬЮТЕРНОГО ЗРЕНИЯ И ГЛУБОКОГО ОБУЧЕНИЯ

Аннотация

В статье рассматривается разработка системы распознавания казахского жестового языка и повышение её эффективности. Актуальной задачей является создание системы, способной распознавать казахский жестовый язык в режиме реального времени с целью облегчения коммуникации среди людей с нарушениями слуха и речи. В предлагаемом методе используется сверточная нейронная сеть YOLOv5 и библиотека Mediapipe, обеспечивающие высокоточное распознавание жестов в реальном времени. Для анализа распознанных знаков и их семантической обработки применяется сеть долговременной и кратковременной памяти (LSTM). Разработанная система позволяет анализировать движения рук пользователя в реальном времени и формировать осмысленные предложения из распознанных жестов. Кроме того, для удобства представления результатов пользователю был разработан веб-сайт на основе фреймворка Django. Экспериментальные результаты показали, что предложенная система обеспечивает высокую точность и надежность при распознавании казахского жестового языка в реальном времени.

Ключевые слова: казахский жестовый язык, сверточная нейронная сеть, реальное время, построение предложений, веб-приложение, нейронная сеть, распознавание жестов.

М.С. Жасұзақ[1,2], Ж.Н. Құдіретулла[1], Ж.А. Бурибаев[1,2], А.С. Еримбетова[2]
[1]Әл-Фараби атындағы Қазақ Ұлттық университеті, Алматы қ., Қазақстан.
[2]Ақпараттық және есептеу технологиялары институты, Алматы қ., Қазақстан

КОМПЬЮТЕРЛІК КӨРУ ЖӘНЕ ТЕРЕҢ ОҚЫТУ ӘДІСТЕРІН ҚОЛДАНА ОТЫРЫП ҚАЗАҚ ЫМ-ИШАРА ТІЛІН ТАНУ ЖҮЙЕСІН ӘЗІРЛЕУ

Аңдатпа

Мақала қазақ ым-ишара тілін тану жүйесін әзірлеу және оның тиімділігін арттыру мәселелері қарастырылады. Бүгінгі таңда маңызды міндет – есту және сөйлеу қабілеті бұзылған адамдардың қарым-қатынасын жеңілдету мақсатында қазақ ым-ишара тілін нақты уақыт режимінде тани алатын жүйе құру. Ұсынылып отырған әдісте нақты уақыт режимінде ымдарды жоғары дәлдікпен тануды қамтамасыз ететін YOLOv5 свертпелі нейрондық желісі мен Mediapipe кітапханасы қолданылады. Танылған белгілерді талдау және олардың мағыналық өңдеу үшін ұзақ және қысқа мерзімді жады желісі (LSTM) пайдаланылады. Әзірленген жүйе пайдаланушының қол қимылдарын нақты уақыт режимінде талдап, танылған ымдардан мағыналы сөйлемдер құрауға мүмкіндік береді. Сонымен қатар, нәтижелерді пайдаланушыға ыңғайлы түрде ұсыну үшін Django фреймворкі негізінде веб-сайт жасалды. Эксперименттік нәтижелер ұсынылған жүйенің нақты уақыт режимінде қазақ ым-ишара тілін тануда жоғары дәлдік пен сенімділікті қамтамасыз ететінін көрсетті.

Түйін сөздер: қазақ ым-ишара тілі, свертпелі нейрондық желі, нақты уақыт, сөйлем құрау, веб-қосымша, нейрондық желі, ымдарды тану.

M.S. Zhassuzak[1,2], Zh.N. Kudiretulla[1], Zh.A. Buribaev[1,2], A.S. Yerimbetova[2]
[1]Al-Farabi Kazakh National University, Almaty, Kazakhstan
[2]Institute of Information and Computational Technologies, Almaty, Kazakhstan

DEVELOPMENT OF A KAZAKH SIGN LANGUAGE RECOGNITION SYSTEM USING COMPUTER VISION AND DEEP LEARNING METHODS

Abstract

This paper discusses the development of a Kazakh Sign Language recognition system and the enhancement of its effectiveness. An urgent task is the creation of a system capable of recognizing Kazakh Sign Language in real-time to facilitate communication among individuals with hearing and speech impairments. The proposed method employs a YOLOv5 convolutional neural network and the Mediapipe library, ensuring high-precision real-time gesture recognition. For the analysis and semantic processing of the recognized signs, a Long Short-Term Memory (LSTM) network is utilized. The developed system allows for the real-time analysis of users' hand movements and the generation of meaningful sentences from the recognized gestures. Furthermore, a web application based on the Django framework was developed to conveniently present the results to users. Experimental results demonstrated that the proposed system provides high accuracy and reliability in recognizing Kazakh Sign Language in real-time.

Keywords: Kazakh Sign Language, convolutional neural network, real-time processing, sentence construction, web application, neural network, gesture recognition.

Введение

Основные положения

В ходе исследования была разработана система распознавания казахского жестового языка, способная работать в реальном времени и обеспечивать высокую точность и надёжность распознавания. Архитектура системы основана на сочетании нейросети YOLOv5 для локализации жестов, библиотеки MediaPipe для отслеживания движений рук и рекуррентной модели LSTM для анализа временных последовательностей. Экспериментальные результаты подтвердили эффективность подхода: достигнуты показатели точности до 97,6 % при распознавании жестов и до 93,4 % при интерпретации жестов в слова. Система успешно визуализирует результаты через веб-интерфейс, что делает её практичным инструментом для коммуникации с людьми с нарушениями слуха и речи.

На сегодняшний день основным средством общения для людей с нарушением слуха является жестовый язык. Однако его широкому распространению препятствуют множество ограничений. Одним из главных является нехватка людей, владеющих жестовым языком, а также различия в его использовании в зависимости от культурных и региональных особенностей. Кроме того, для прямого перевода жестов требуются профессиональные переводчики, услуги которых не всегда доступны. Системы автоматического распознавания жестов представляют собой эффективное решение данной проблемы. Эти системы позволяют точно распознавать знаки и движения жестового языка с помощью методов компьютерного зрения и глубокого обучения, предоставляя возможность общения в текстовой или звуковой форме [1, 2].

В настоящее время сверточные нейронные сети архитектуры YOLO (You Only Look Once) демонстрируют высокую эффективность в области распознавания объектов на видеоданных [3]. Модель YOLOv5 отличается способностью к быстрому и точному распознаванию в реальном времени [4]. Несмотря на эффективность данной модели при распознавании жестов, при работе с казахским жестовым языком её точность может снижаться из-за таких факторов, как фоновый шум, скорость движения и разнообразие жестов рук.

Использование одной лишь модели YOLOv5 для локализации движений рук при распознавании жестов недостаточно. Для определения значений отдельных жестов и их семантической интерпретации необходимо применение дополнительных рекуррентных нейронных сетей. С этой целью используется сеть долговременной и кратковременной памяти (LSTM). LSTM – это разновидность рекуррентной нейронной сети с высокой способностью к сохранению и обработке временных зависимостей [5]. Она позволяет анализировать

временные последовательности признаков, полученных с помощью YOLOv5, и формировать из них осмысленные слова и предложения.

Для обеспечения удобства использования системы, распознанные знаки и предложения отображаются на веб-сайте, разработанном на основе фреймворка Django [6]. Django представляет собой функционально расширенный веб-фреймворк, обеспечивающий визуализацию данных в реальном времени.

Целью данного проекта является разработка высокоточной системы распознавания казахского жестового языка в реальном времени путём интеграции моделей YOLOv5 и Mediapipe. Система анализирует движения рук в режиме реального времени, определяет их значение и формирует осмысленные предложения на основе распознанных жестов.

Методы исследования

В данном разделе описаны основные методы и инструменты, использованные при создании системы для распознавания казахского жестового языка в режиме реального времени. В ходе исследования была реализована интеграция современных архитектур глубокого обучения, библиотек для обработки видео и технологий, обеспечивающих веб-интерфейс [7, 8, 9].

Основная цель системы – в реальном времени определять движения рук пользователя из видеопотока, распознавать их смысл и формировать осмысленные предложения из распознанных знаков. Для этого были использованы следующие компоненты. Интеллектуальная система для распознавания казахского жестового языка в реальном времени была разработана на основе интеграции передовых технологий: в первую очередь, для точного определения жестов на видеозаписи применялась высокоскоростная и надёжная нейронная сеть YOLOv5 [4,10]; для извлечения координат 21 ключевой точки каждого обнаруженного жеста использовалась платформа MediaPipe, разработанная Google [11]; семантическое распознавание, учитывающее последовательные изменения движений во времени, реализовано с помощью нейронной сети LSTM (Long Short-Term Memory) [5,12]; распознанные знаки и слова отображались в реальном времени в удобном для пользователя веб-интерфейсе, реализованном на фреймворке Django [6]; обучение всей модели осуществлялось на специально собранном наборе видео- и координатных данных, охватывающем 42 буквы казахского жестового алфавита и различные части речи [13]; разработка велась на языке Python 3.10 с использованием современных библиотек, таких как OpenCV, NumPy, Matplotlib, MediaPipe, PyTorch, TensorFlow и Django, что обеспечило доступность и эффективность решения [10,11,14,15,16].

Сеть LSTM способна распознавать только временные изменения жестов. Для точного определения соответствующего слова важны входной сигнал (x_t), предыдущее состояние (h_{t-1}) и память (C_t). При анализе временной последовательности движений руки пользователя ключевым является решение о необходимости сохранения новой информации в памяти. Это решение принимается с помощью так называемого «входного гейта» (input gate), работа которого основана на сигмоидной функции и описывается следующим выражением:

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

где:

i_t –значение входного гейта;

σ – сигмоидная функция активации;

W_i – матрица весов входного гейта;

$h_{t-1,} x_t$ – предыдущее скрытое состояние и входной вектор;

b_i – смещение.

Если пользователь выполняет жест «сен» («ты»), система сравнивает текущее изменение координат (x_t) с предыдущим состоянием (h_{t-1}) и определяет, представляет ли новая информация смысловую ценность.

Если $i_t \approx 1$, жест считается значимым и сохраняется в памяти. Если система решает сохранить жест, то вычисляется содержательная информация, подлежащая запоминанию. Она определяется с помощью гиперболического тангенса:

$$C_t=tanh(W_C \cdot [h_{t-1,}x_t]+b_C) \quad (2)$$

где:

C_t – кандидат на сохранение в памяти;

tanh – функция гиперболического тангенса;

W_C – матрица весов;

b_C – смещение;

$h_{t-1,}x_t$ – предыдущее состояние и текущий вход.

Пример:

При выполнении жеста «сен» система, анализируя координаты, распознаёт его как слово «ты» и подготавливает его к сохранению в памяти.

Окончательный результат – распознанное слово или смысл – определяется выходным состоянием модели, которое передаётся в визуальный или другой внешний интерфейс:

$$h_t = o_t \cdot tanh(C_t) \quad (3)$$

где:

h_t – текущее выходное значение скрытого слоя;

o_t – значение выходного гейта;

C_t – обновленное состояние памяти;

tanh – функция активации.

Пример:

Если пользователь выполняет последовательные жесты «сен – автобус – бар» («ты – автобус – иди»), система распознаёт каждое слово и формирует из них связное предложение, которое отображается на экране.

В задаче автоматического определения и локализации движений рук сверточные нейронные сети (CNN – Convolutional Neural Networks) играют ключевую роль. Архитектуры CNN обладают высокой способностью извлекать пространственные признаки из видеоданных, что позволяет эффективно выделять области жестов на видеокадрах [7].

В предложенной системе для точного обнаружения жестов на кадрах использовалась модель YOLOv5 (You Only Look Once, версия 5). Эта модель относится к одноэтапным (one-stage) детекторам и обеспечивает высокую скорость обработки в режиме реального времени [8]. YOLOv5 одновременно анализирует всё изображение, точно локализуя жесты рук и определяя их с помощью ограничивающих рамок (bounding boxes).

Архитектура YOLOv5 на Рисунке 1 состоит из нескольких ключевых компонентов:

- Backbone (CSPDarknet53): используется для извлечения начальных признаков изображения. Блоки Cross Stage Partial (CSP) уменьшают потерю информации и повышают вычислительную эффективность.
- Neck (FPN + PANet): обеспечивает многоуровневую обработку признаков изображения, что позволяет эффективно распознавать как мелкие, так и крупные объекты. Архитектуры Feature Pyramid Network (FPN) и Path Aggregation Network (PANet) работают совместно.
- Head: финальный компонент, отвечающий за выдачу результатов детекции. С помощью механизма anchor-based определяется местоположение и класс объекта.

В предлагаемой системе YOLOv5 распознаёт жесты рук на каждом кадре видеопотока, а затем передаёт выделенные области в модели Mediapipe и LSTM для дальнейшей обработки. Эффективность YOLOv5 оказывает прямое влияние на общую точность системы, обеспечивая высокую скорость и стабильность распознавания жестового языка [9-10].

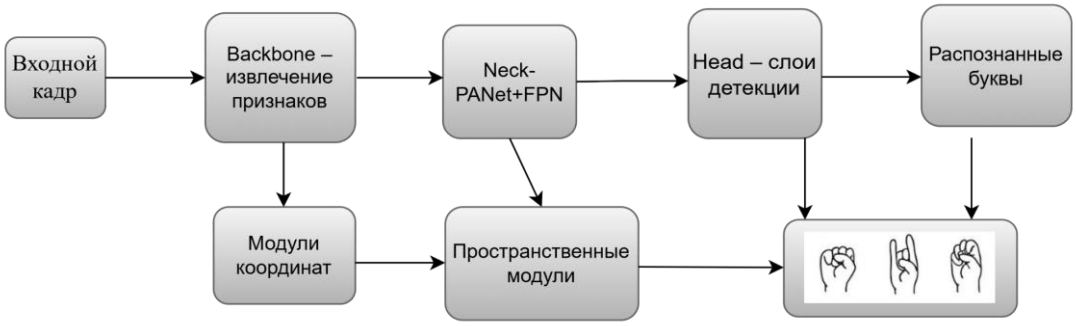


Рисунок 1. Архитектурная схема модели YOLOv5 для распознавания жестов рук

Использованный набор данных был сформирован на основе дактильного алфавита казахского жестового языка. Датасет сбалансирован и содержит более 10 000 изображений, охватывающих 40 символов казахского алфавита (40 классов). На рисунке 2 представлены примеры жестов, соответствующих каждой букве казахского алфавита.

Рисунок 2. Алфавит казахского жестового языка

Символы «Ъ» и «Ь» не входят в собственную фонетическую систему казахского языка, поэтому они используются исключительно в заимствованных словах, то есть при записи слов, пришедших из других языков (рисунок 3). Эти знаки служат для обозначения оттенков произношения согласных – мягкости или твёрдости. Однако в казахском языке такие фонологические контрасты отсутствуют, поэтому данные символы не были интегрированы в активную языковую практику.

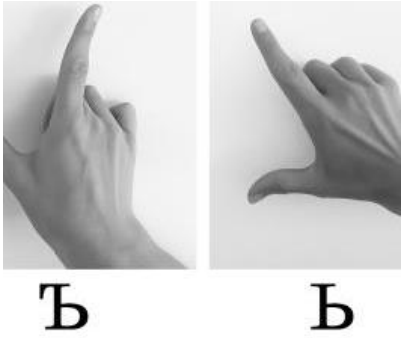


Рисунок 3. Особые знаки казахского алфавита

В задаче распознавания букв казахского жестового языка из изображений для выделения пространственных признаков и их точной классификации применяются глубокие архитектуры сверточных нейронных сетей (CNN). В рамках данного исследования были использованы три популярные модели CNN – GoogleNet, ResNet-101 и VGG-19, и проведено сравнение их эффективности.

Модель VGG (рисунок 4) отличается простой, но глубокой архитектурой. Она состоит из сверточных слоев размером 3×3 и направлена на извлечение пространственных признаков из видеоданных. Однако из-за большого количества параметров модель показала низкую точность (2,23 %) при распознавании букв казахского жестового языка. Это указывает на ограниченные возможности VGG-19 при различении сложных жестов.

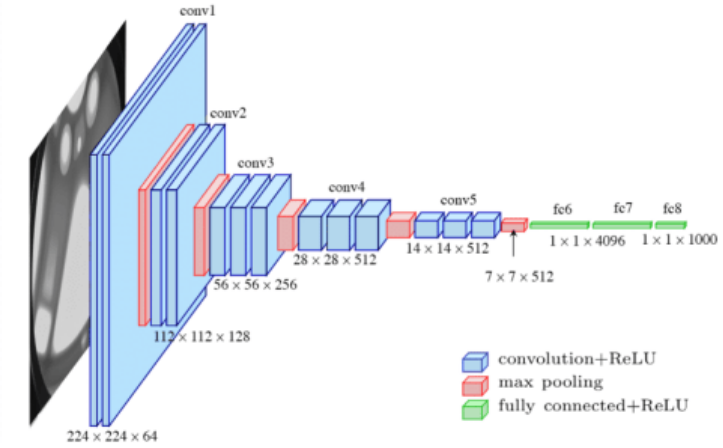


Рисунок 4. Архитектура VGG-19

Архитектура ResNet (рисунок 5) предотвращает потерю информации за счёт остаточных связей (skip connections), что позволяет эффективно обучать даже очень глубокие модели. В задаче распознавания букв казахского жестового языка модель ResNet достигла 100 % точности, успешно справившись с различением сложных жестов и особенностей движений. Эта архитектура была выбрана как наиболее эффективная.

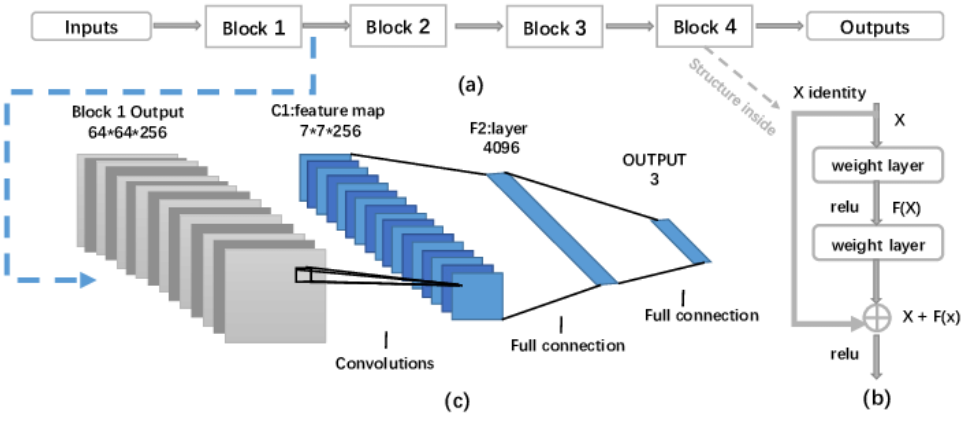


Рисунок 5. Архитектура ResNet-101

Модель GoogleNet (рисунок 6) использует Inception-блоки для распознавания признаков изображения на разных масштабах. В каждом блоке одновременно применяются фильтры различных размеров, что позволяет эффективно выявлять разнообразные особенности жестов. При распознавании букв казахского жестового языка данная модель достигла точности 88,01 %, продемонстрировав оптимальный результат.

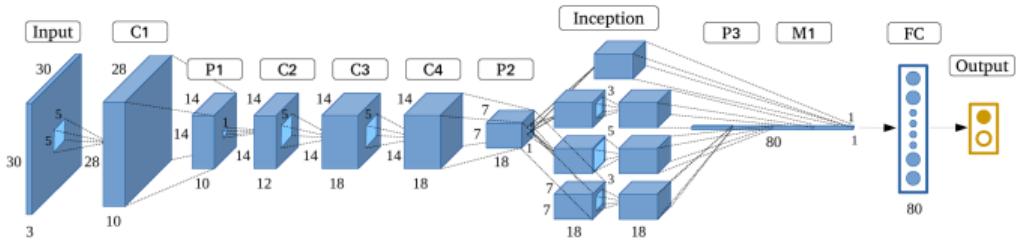


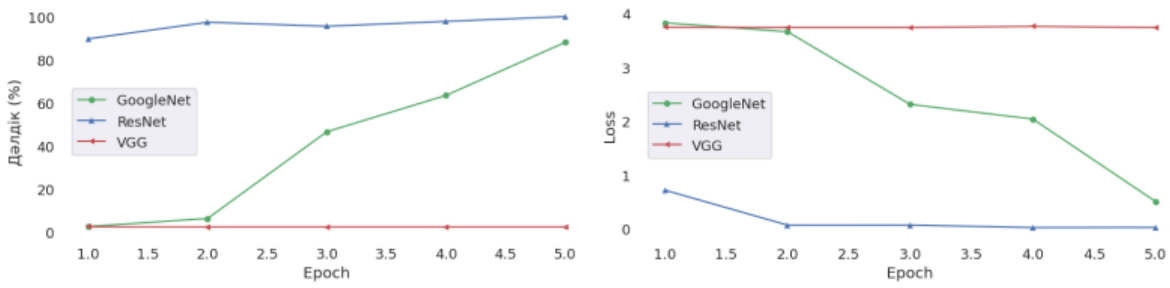
Рисунок 6. Архитектура GoogleNet

В целом, модель VGG отличается простотой и удобством для обучения, GoogleNet – эффективной обработкой и экономией вычислительных ресурсов, тогда как ResNet продемонстрировала наилучшие результаты благодаря высокой точности и глубокой архитектуре. Эти выводы подтверждаются и результатами предыдущих исследований [14, 15] (рисунок 7).

Epoch	GoogleNet Accuracy (%)	GoogleNet Loss	ResNet Accuracy (%)	ResNet Loss	VGG Accuracy (%)	VGG Loss
1.0	2.43	3.8241	89.68	0.7141	2.28	3.741
2.0	6.17	3.6574	97.37	0.0648	2.23	3.7381
3.0	46.46	2.3082	95.5	0.0654	2.23	3.7379
4.0	63.36	2.036	97.77	0.0192	2.23	3.7575
5.0	88.01	0.5053	100.0	0.0224	2.23	3.7374

Рисунок 7. Сравнительные показатели архитектур сверточных нейронных сетей

Эффективность системы распознавания казахского жестового языка была оценена на основе специально собранных видеозаписей и координатных данных. Общий набор данных, состоящий более чем из 10 000 изображений, был разделён на 80 % для обучения и 20 % для тестирования. Показатели точности (accuracy) системы рассчитывались по результатам тестового набора. Сравнительная точность трёх предложенных архитектур (GoogleNet, ResNet-101, VGG-19) представлена на рисунке 8. Согласно результатам, модель ResNet-101 показала высокую и стабильную точность на тестовом наборе, в то время как GoogleNet достигла высоких результатов только на последних эпохах. Модель VGG-19 продемонстрировала низкую точность на всех этапах обучения.

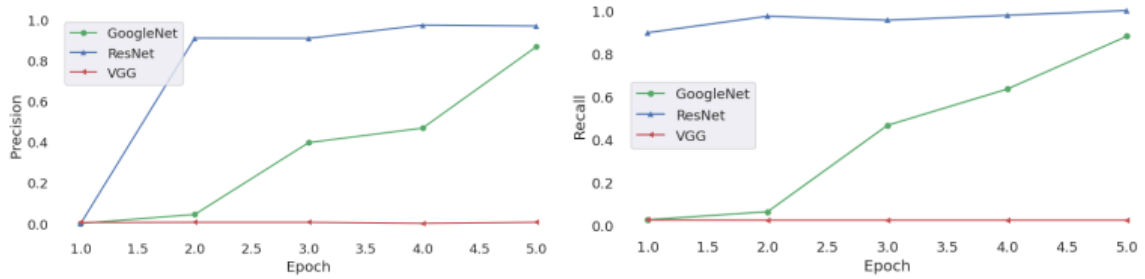


(a) (b)

Рисунок 8. Сравнительная оценка моделей CNN и ансамблевого метода: (a) Точность (Accuracy); (b) Функция потерь (Loss function).

Графическая интерпретация метрик precision (точность) и recall (полнота), представленных на рисунках 9.a и 9.b, демонстрирует, что модель VGG-19 не способна чётко различать заданный класс от других. Более того, при классификации с использованием VGG-19 значительная часть положительных примеров остаётся нераспознанной. В то же время модели

ResNet-101 и ансамблевый подход показывают высокие значения recall при определении конкретных классов, достигая значительно лучших результатов.



(a) (b)

Рисунок 9. Сравнительная оценка моделей CNN и ансамблевого метода: (a) Точность (Precision); (b) Полнота (Recall).

Результаты исследования

Для оценки эффективности предложенной системы было проведено несколько экспериментов на основе специально подготовленного набора данных. В датасет были включены 40 букв казахского жестового алфавита, а также распространённые местоимения, существительные и глаголы. Каждый жест записывался многократно с участием разных пользователей.

На рисунке 10 представлена визуализация жестов, соответствующих буквам казахского алфавита. Эти знаки использовались в обучающем наборе данных системы, при этом каждому жесту был присвоен конкретный класс. Подобные визуальные данные позволяют системе точно распознавать буквы и служат основой для извлечения пространственных признаков с помощью сверточных нейронных сетей (CNN).

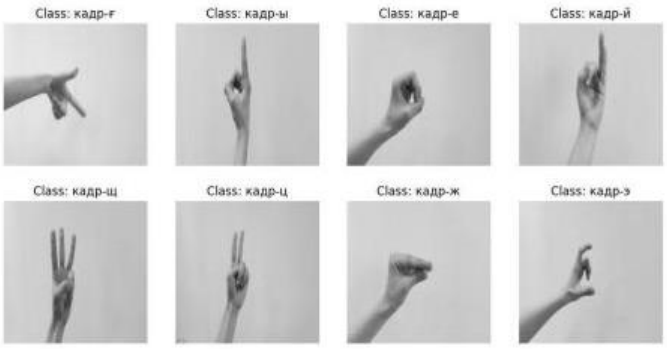


Рисунок 10. Буквенные жесты казахского жестового языка

Извлечение координат рук (MediaPipe): с помощью библиотеки MediaPipe из видеопотока были точно и стабильно извлечены координаты 21 ключевой точки руки. Эти координаты описывают пространственную конфигурацию жеста и обеспечивают надёжность работы системы. MediaPipe продемонстрировала устойчивую производительность даже при изменениях освещения и фона [3].

Скорость работы: система функционировала со средней скоростью 20 кадров в секунду, что соответствует требованиям реального времени. Каждый распознанный жест отображался на экране в течение 6 секунд. В случае формирования предложения ранее распознанные слова сохранялись и дополнялись новыми.

На следующем графике (рисунок 11) представлены показатели точности (precision) и полноты (recall) компонентов YOLOv5 и LSTM по эпохам обучения.

Показатель precision демонстрирует, насколько точно система определяет истинно положительные примеры, и повысился с 75 % до 97,6 % за 10 эпох.

Recall отражает способность модели находить все истинно положительные случаи, увеличившись с 70 % до 95,3 %. Точность на уровне слов (Word-level accuracy) на основе LSTM также улучшилась с 68 % до 93,4 %, показывая эффективность распознавания движений в зависимости от их временной последовательности. Эти результаты подтверждают, что компоненты системы постепенно обучаются и адаптируются к выполнению стабильных и достоверных предсказаний.

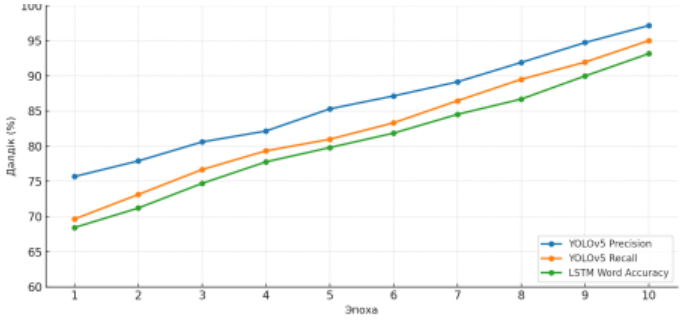


Рисунок 11. Результаты точности по компонентам системы

В целом, согласованность компонентов системы и корректность выбранных архитектур обеспечили высокую точность работы в режиме реального времени. Система стабильно преобразовывала каждый жест, показанный пользователем, в слова и предложения, демонстрируя надёжность распознавания.

Веб-интерфейс предложенной системы представляет собой интерактивную платформу для распознавания казахского жестового языка в реальном времени (см. рисунок 12). Интерфейс автоматически обрабатывает жесты руки, выполненные перед камерой, распознаёт соответствующую казахскую букву и визуально отображает результат на экране. Система реализована с использованием веб-фреймворка Django на языке Python и интегрирована с моделями глубокого обучения.

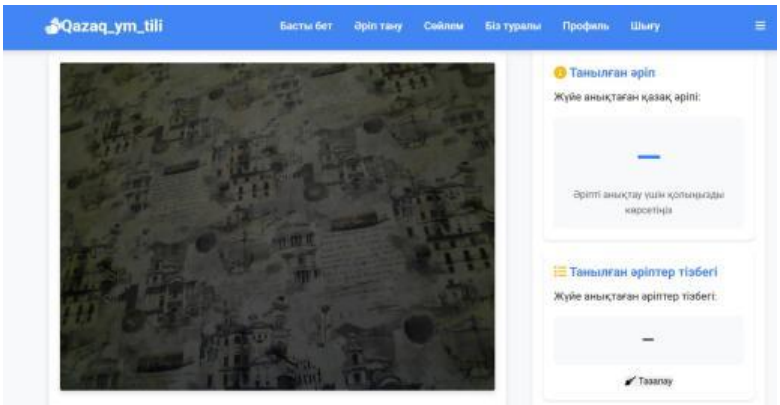


Рисунок 12. Визуальный интерфейс системы, работающей в режиме реального времени

Разработанная в рамках данного проекта система включает в себя следующие ключевые компоненты и обеспечивает удобный пользовательский интерфейс. Через главную страницу пользователь знакомится с общей целью системы и получает доступ к основным функциям. Модуль распознавания букв обрабатывает жесты рук из видеопотока в режиме реального времени и определяет соответствующие буквы казахского жестового языка. Распознанные буквы передаются в модуль построения предложений, где в соответствии с порядком они формируют полносмысленные фразы. Дополнительно в интерфейсе предусмотрен раздел «О нас» с информацией об авторах проекта, а также функции «Профиль» и «Выход» для управления действиями пользователя. Все компоненты системы работают согласованно,

обеспечивая высокоточную визуализацию и распознавание казахского жестового языка в реальном времени.

На следующем изображении представлена полная архитектурная структура веб-ориентированной системы для распознавания казахского жестового языка (рисунок 13).

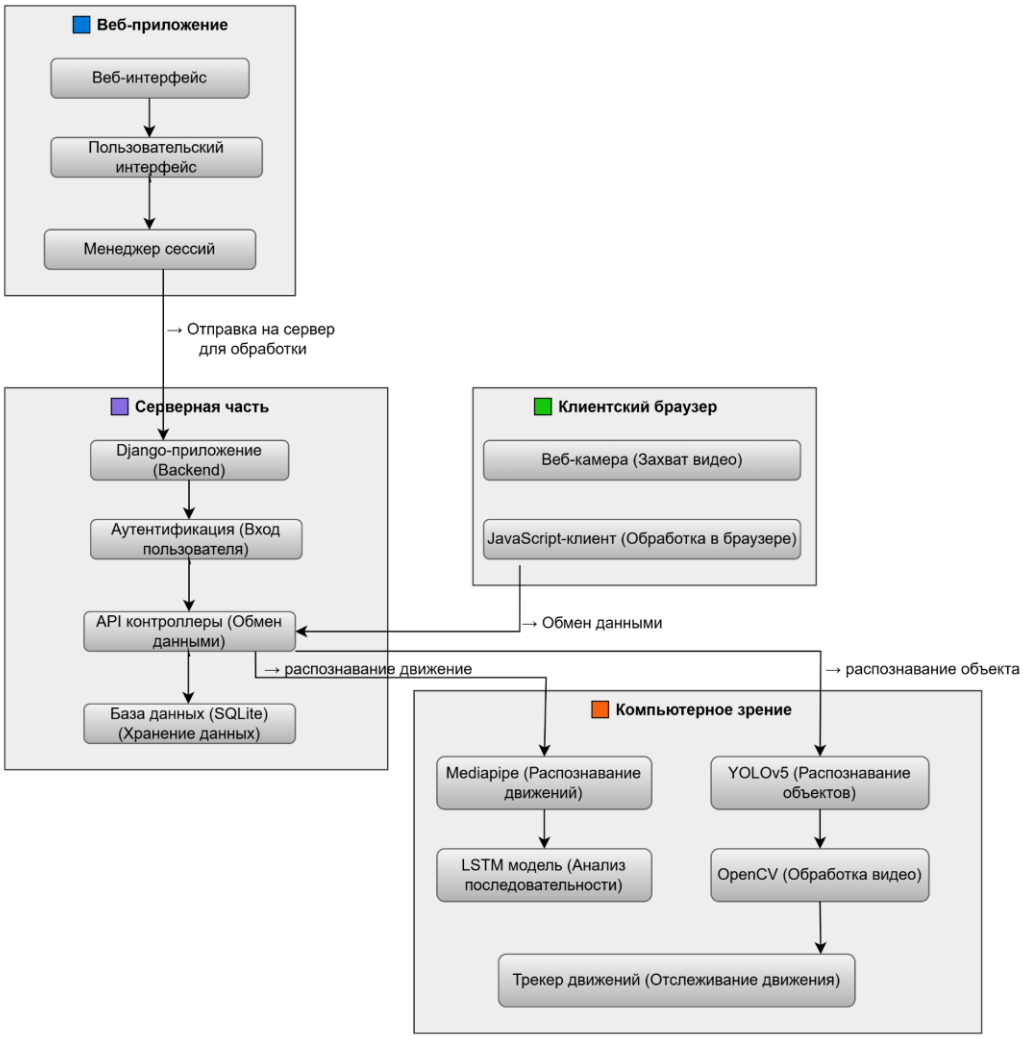


Рисунок 13. Полная архитектурная структура системы

Система состоит из трех основных компонентов: веб-приложения (пользовательский интерфейс), клиентской части браузера (обработка с помощью камеры и JavaScript) и серверной части (бэкенд на Django и база данных). Пользователь подключается к системе через веб-интерфейс и показывает жест с помощью камеры. Видеопоток обрабатывается на стороне клиента с использованием JavaScript и отправляется на сервер. На сервере фреймворк Django на языке Python принимает данные и передаёт их через аутентификацию и API-контроллер в блок обработки. Внутри блока компьютерного зрения:

– Mediapipe извлекает координаты 21 ключевой точки руки,

– YOLOv5 выполняет локализацию руки на изображении,

– LSTM анализирует временные последовательности движений и распознаёт осмысленные буквы или слова. Все полученные данные сохраняются в базе данных SQLite, а результат отображается пользователю в режиме реального времени.

На рисунке 15 показаны примеры жестов, распознанных системой в реальном времени. Для каждого кадра были визуализированы 21 опорная точка жеста с помощью MediaPipe, а полученные координаты обработаны через LSTM-сеть.

Согласно результатам, жесты, соответствующие буквам «L», «G», «R» и «Z», были распознаны со 100 % точностью. Жест «A» был определён с точностью 86,59 %, а «O» – с 80,60 %. Эти показатели демонстрируют высокую надёжность системы в различении жестов, принадлежащих к разным классам.

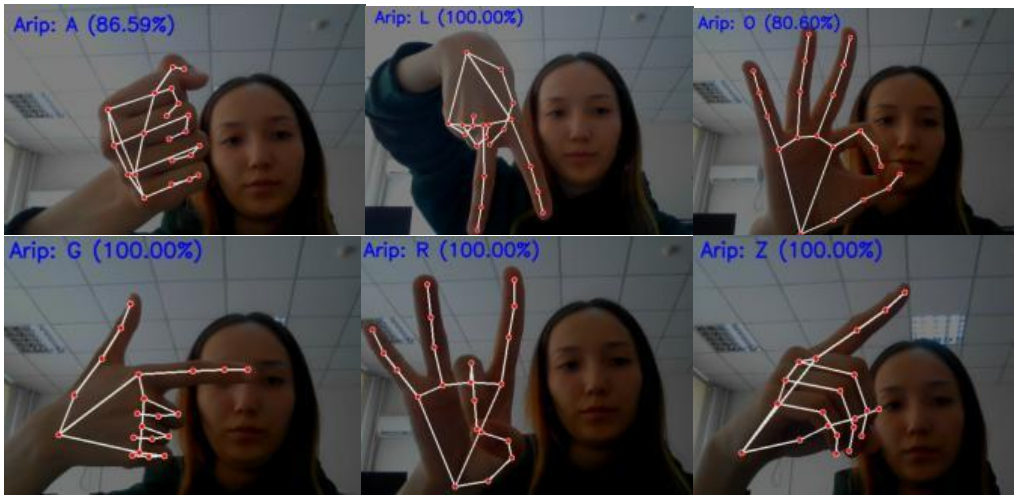


Рисунок 14. Результаты распознавания букв казахского жестового языка

Система точно определяет 21 ключевую точку с помощью MediaPipe, а благодаря локализации YOLOv5 и временной аналитике LSTM достигается высокая точность распознавания. Показатели точности варьируются от 80 % до 100 %, что свидетельствует об эффективном обучении модели и её устойчивости при использовании. Несмотря на разнообразие жестов, система сохраняет когнитивную согласованность и визуальную стабильность.

На приведённой иллюстрации (рисунок 15) продемонстрирована способность системы в реальном времени распознавать жесты пользователя по трем основным частям речи: местоимение, существительное и глагол. Распознанные слова объединяются в логической последовательности и формируют полноценное предложение. Финальный результат – предложение «sen jumysty jasaysyn» – отображается в текстовом виде и озвучивается с помощью TTS-технологии. Таким образом, система позволяет формировать осмысленные предложения в соответствии с синтаксическими нормами казахского языка на основе жестов.

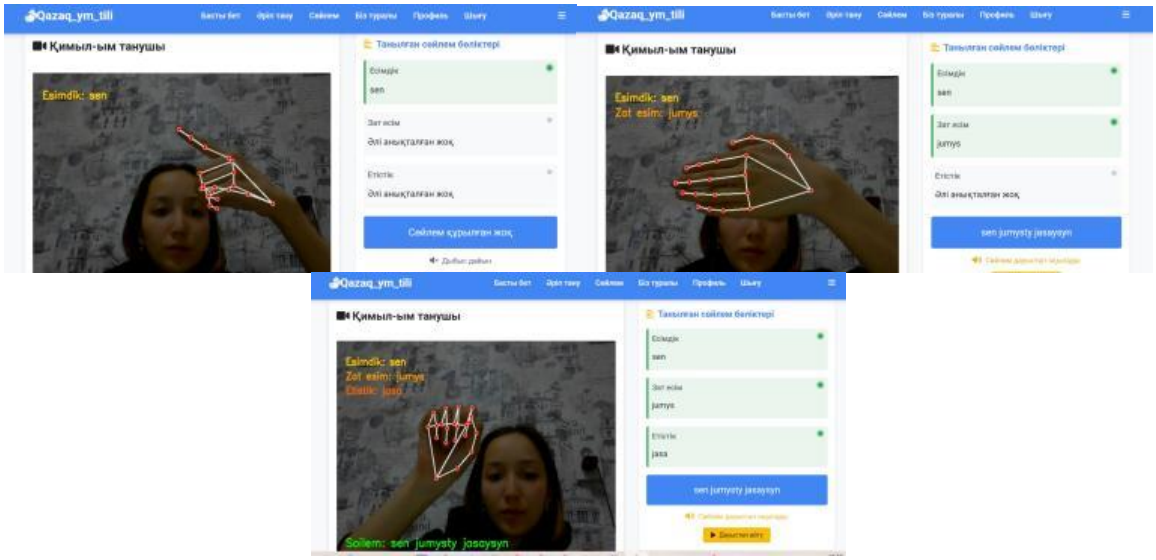


Рисунок 15. Результаты построения предложений в системе распознавания жестового языка

Основываясь на полученных результатах, была оценена функциональная эффективность системы и её синтаксическая согласованность. Жесты, выполняемые пользователем, распознаются с высокой точностью в режиме реального времени и объединяются в логически связанное предложение с сохранением смысловой структуры. Кроме того, система обеспечивает корректность морфологического построения и соблюдение синтаксических норм казахского языка.

Дискуссия

Разработка системы распознавания казахского жестового языка в режиме реального времени представляет собой значительный шаг в направлении инклюзивных технологий, направленных на преодоление барьеров коммуникации для лиц с нарушениями слуха и речи. Предложенная архитектура, сочетающая возможности сверточной нейросети YOLOv5, координатного трекинга MediaPipe и рекуррентной модели LSTM, демонстрирует сбалансированную эффективность на этапах локализации, извлечения ключевых признаков и анализа временных зависимостей.

Высокие значения метрик Precision (97,6 %) и Recall (95,3 %) на этапе обнаружения жестов свидетельствуют о достаточной чувствительности и специфичности модели YOLOv5. Однако важным фактором устойчивой производительности остаётся правильная организация обучающего набора: высокая вариативность поз, освещённости и фона может влиять на точность локализации. Кроме того, использование MediaPipe позволило упростить процесс извлечения координатных признаков, однако система может быть чувствительна к перекрытию рук и частичной окклюзии.

Применение LSTM-архитектуры обеспечило возможность анализа последовательных кадров, что особенно критично для распознавания динамических жестов, где временной контекст определяет смысловую нагрузку. Сравнительный анализ моделей классификации показал превосходство ResNet над другими архитектурами, что коррелирует с современными исследованиями, подтверждающими эффективность остаточных связей в глубоких сетях.

Однако наряду с положительными результатами, система имеет ряд ограничений. Прежде всего, она работает с ограниченным набором слов и фраз, не охватывая синтаксическую структуру языка. Отсутствие модуля генерации аудиоречи также ограничивает возможности взаимодействия между глухими пользователями и слышащими. Кроме того, устойчивость системы при условиях внешних шумов (например, изменении освещения, появлении посторонних объектов в кадре) требует дополнительного тестирования и адаптации.

Заключение

В ходе настоящего исследования была разработана и экспериментально подтверждена работоспособность многокомпонентной системы для распознавания казахского жестового языка в реальном времени. В её основу легли передовые алгоритмы компьютерного зрения и обработки временных последовательностей, включая YOLOv5, MediaPipe и LSTM. Система показала высокую точность на всех этапах: от локализации и извлечения координат до семантической интерпретации жестов.

Полученные результаты демонстрируют потенциал применения подобных систем в социальных и образовательных сферах, а также в сервисах цифровой коммуникации. Работа системы в режиме 20 кадров в секунду с возможностью формирования осмысленных фраз подтверждает её прикладную ценность и технологическую зрелость.

В перспективе планируется расширение функциональности системы за счёт включения синтаксического анализа казахского жестового языка, реализации голосовой озвучки распознанных фраз и интеграции поддержки мультиязычия. Кроме того, рассматриваются возможности адаптации системы для мобильных и веб-платформ, что расширит доступность технологии для широкой аудитории.

Благодарность

Статья подготовлена при финансовой поддержке Министерства Науки и высшего образования Республики Казахстан в рамках грантового исследования № BR24992875.

Список использованных источников

1. *Akhmetkali, A. 2025. "Kazakh Scientist Develops AI to Translate Sign Language into Kazakh." The Astana Times. https://astanatimes.com/2025/02/kazakh-scientist-develops-ai-to-translate-sign-language-into-kazakh/.*
2. *"Gesture Recognition of the Kazakh Alphabet Based on Machine and Deep Learning Models." 2024. Procedia Computer Science, Elsevier. https://www.sciencedirect.com/science/article/pii/S1877050924017757.*
3. *Bochkovskiy, Alexey, Chien-Yao Wang, and Hong-Yuan M. Liao. 2020. "YOLOv4: Optimal Speed and Accuracy of Object Detection." arXiv preprint arXiv:2004.10934. https://arxiv.org/abs/2004.10934*
4. *Ultralytics. 2023. YOLOv5: Real-Time Object Detection. GitHub Repository. https://github.com/ultralytics/yolov5.*
5. *Hochreiter, Sepp, and Jürgen Schmidhuber. 1997. "Long Short-Term Memory." Neural Computation 9 (8): 1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735.*
6. *Pinkham, Andrew. 2015. Django Unleashed. Boston: Pearson Education. https://books.google.com/books/about/Django_Unleashed.html?id=gC0BCwAAQBA.*
7. *Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E. Hinton. 2012. "ImageNet Classification with Deep Convolutional Neural Networks." Advances in Neural Information Processing Systems 25: 1097–1105. https://papers.nips.cc/paper_files/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf.*
8. *Simonyan, Karen, and Andrew Zisserman. 2015. "Very Deep Convolutional Networks for Large-Scale Image Recognition." arXiv preprint arXiv:1409.1556. https://arxiv.org/abs/1409.1556.*
9. *He, Kaiming, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. "Deep Residual Learning for Image Recognition." Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR): 770–778. https://doi.org/10.1109/CVPR.2016.90.*
10. *Szegedy, Christian, Wei Liu, Yangqing Jia, et al. 2015. "Going Deeper with Convolutions." Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). https://doi.org/10.1109/CVPR.2015.7298594.*
11. *Lugaresi, Camillo, Jason Tang, Hartwig Nash, et al. 2019. "MediaPipe: A Framework for Building Perception Pipelines." arXiv preprint arXiv:1906.08172. https://arxiv.org/abs/1906.08172.*
12. *Kingma, Diederik P., and Jimmy Ba. 2015. "Adam: A Method for Stochastic Optimization." International Conference on Learning Representations (ICLR). https://arxiv.org/abs/1412.6980.*
13. *Abdrakhmanova, G., et al. 2023. "Development of Kazakh Sign Language Dataset and Preliminary Evaluation." Proceedings of the International Conference on Language Resources and Evaluation (LREC). https://aclanthology.org/2023.lrec-main.1017.*
14. *Bradski, Gary. 2000. "The OpenCV Library." Dr. Dobb's Journal of Software Tools. https://opencv.org/.*
15. *Van Rossum, Guido, and Fred L. Drake Jr., eds. 2002. Python 3 Reference Manual. CreateSpace. https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=09b1d693676a152621799e2cfff562a369d10204.*
16. *Paszke, Adam, Sam Gross, Francisco Massa, et al. 2019. "PyTorch: An Imperative Style, High-Performance Deep Learning Library." Advances in Neural Information Processing Systems 32: 8024–8035. https://papers.nips.cc/paper_files/paper/2019/file/bdbca288fee7f92f2bfa9f7012727740-Paper.pdf.*