



ОТКРЫТАЯ НАУКА  
издательство

Международный журнал информационных технологий и  
энергоэффективности

Сайт журнала:

<http://www.openaccessscience.ru/index.php/ijcse/>



УДК 004.058:004.93

## ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ РАСПОЗНАВАНИЯ АЗБУКИ ГЛУХОНЕМЫХ

<sup>1</sup> Нагорных М. Э., <sup>2</sup> Быков А.Н., <sup>3,4</sup> Чернышёв С. А.

<sup>1,2</sup> Санкт-Петербургский государственный университет аэрокосмического приборостроения, Россия (190000, г. Санкт-Петербург, ул. Большая Морская, 67, лит. А),  
e-mail: <sup>1</sup> nagornykh\_max@mail.ru, <sup>2</sup> alexey\_bykovoff@mail.ru

<sup>3</sup> Санкт-Петербургский государственный университет промышленных технологий и дизайна, Россия (191186, г. Санкт-Петербург, ул. Большая Морская, 18)

<sup>4</sup> Санкт-Петербургский государственный экономический университет, Россия (191023, г. Санкт-Петербург, ул. Садовая, 21), e-mail: chernyshev.s.a@bk.ru

Последнее время наблюдается социальная направленность во внутренней политике различных государств. Особенно это заметно по требованиям, предъявляемым к образовательным организациям. Не должно существовать разделения между обычными студентами и студентами с ограничениями по здоровью. В связи с этим на первый план выходит инклюзивный подход к организации образовательного процесса. К сожалению, не все образовательные организации могут позволить себе специальное оборудование, необходимое для проведения занятий с участием студентов с ограничениями по здоровью. В статье подробно описывается процесс разработки программного обеспечения для распознавания азбуки глухонемых. Приводится краткий обзор и обоснование выбора модели обучения нейронной сети, а также продемонстрирован полученный результат.

Ключевые слова: глухонемые, нейросеть, Faster-RCNN, Python, Tensorflow.

## DEAF-MUTE ALPHABET RECOGNITION SOFTWARE

<sup>1</sup> Nagornykh M.E., <sup>2</sup> Bykov A.N., <sup>3,4</sup> Chernyshev S.A.

<sup>1,2</sup> Saint Petersburg State University of Aerospace Instrumentation, Russia (190000, Saint Petersburg, Bolshaya Morskaya st., 67, letter A),

e-mail: <sup>1</sup> nagornykh\_max@mail.ru, <sup>2</sup> alexey\_bykovoff@mail.ru

<sup>3</sup> Saint Petersburg State University of Industrial Technology and Design, Russia (191186, Saint-Petersburg, Bolshaya Morskaya st., 18)

<sup>4</sup> Saint Petersburg State University of Economics, Russia (191023, Saint Petersburg, Sadovaya st., 21), e-mail: chernyshev.s.a@bk.ru

Recently, there has been a social orientation in the internal politics of various states. This is especially noticeable in the requirements for educational organizations. There should be no division between ordinary students and students with disabilities. In this regard, an inclusive approach to the organization of the educational process comes to the fore. Unfortunately, not all educational organizations can afford the special equipment required to conduct classes with students with disabilities. The article describes in detail the process of developing software for recognizing the alphabet of the deaf-mute. A brief overview and justification of the choice of a neural network training model are given, and the result obtained is demonstrated.

Keywords: deaf-mute, neural network, Faster-RCNN, Python, Tensorflow.

## **Введение**

Большинство людей не понимают язык глухонемых, из-за чего при общении с глухонемыми может возникать проблема взаимопонимания в простых жизненных ситуациях. Примером таких ситуаций могут выступать: общение между преподавателем и учеником, обращение в медицинское учреждение или даже при покупках в магазинах и т.д. [1-3].

С развитием технологий появляется все больше возможностей для решения этих проблем. В статье представлен процесс разработки приложения, которое используя веб-камеру компьютера, производит конвертацию языка жестов в текст. Это достигается посредством использования нейронных сетей и машинного обучения. Данная технология активно развивается последние десятилетия и набирает популярность не только при решении промышленных, но и социально-значимых задач [4-9].

Создание такого программного обеспечения позволяет приблизиться к решению проблемы коммуникации, появляющейся при общении с глухонемыми людьми. Данная тема имеет остросоциальную необходимость, поскольку общество стремится к тому, чтобы люди не чувствовали себя ограниченными в своих возможностях.

### **1. Принцип работы.**

Алгоритм работы разработанного приложения состоит из следующих стадий:

- Выбор используемой веб-камеры из списка доступных на компьютере;
- Получение изображения из видео потока;
- Анализ его с помощью обученной нейросети, получение координат расположения жеста в видеопотоке и его классификация;
- Отображение распознанного символа в конце списка для составления слова.

### **2. Выбор нейронной сети.**

На сегодняшний день существует множество программных решений для создания и обучения нейронных сетей. В данном проекте использовалась Tensorflow. Эта библиотека привлекает пользователей своим широким спектром возможностей для решения задач классификации и активно поддерживается как её разработчиками, так и сообществом программистов [10, 11].

TensorFlow предоставляет несколько моделей обнаружения объектов (предварительно обученные классификаторы с конкретными архитектурами нейронных сетей). Некоторые модели (например, модель SSD-MobileNet) имеют архитектуру, которая обеспечивает более быстрое обнаружение, но с меньшей точностью, в то время как некоторые модели (например, модель Faster-RCNN) обеспечивают более медленное обнаружение, но большую точность.

Система Faster-RCNN состоит из двух модулей: первый модуль — это сверточная нейронная сеть, которая находит потенциальные объекты на изображении и разбивает их на области, а второй модуль — это Fast-RCNN, который использует предложенные области и является детектором. Вся система представляет собой единую унифицированную сеть для обнаружения объектов.

Faster-RCNN быстрее и лучше выполняет задачи поиска и классификации, чем ее предшественники. На сегодняшний день она широко используется и остается одним из лучших решений доступных пользователю. Именно поэтому в качестве модели обучения использовалась Faster-RCNN [12].

### 3. Процесс разработки программного обеспечения

На самой первой стадии был сформирован набор обучающих данных, где каждый экземпляр представлял из себя фотографию с указанным расположением жеста и его значением. Для этих целей использовалось приложение LabelImg [13], работа в котором представлена на рисунке 1.



Рисунок 1 - Пример обработки обучающего фото в LabelImg

Фотографии были сделаны на различном фоне и при различных условиях освещения. Это позволило избежать нахождения нейросетью ненужных взаимосвязей и сконцентрировать ее внимание на анализе жестов кистей рук.

В результате было сделано 1650 фотографий по 50 на каждый жест, из них 30% было выделено для тестовых данных.

Из набора маркированных изображений были сгенерированы TFRecords файлы, которые служат входными данными для модели обучения TensorFlow. На следующем шаге создавалась карта меток, на которой указывалось соответствие id объекта с его наименованием [14]. Пример такой карты представлен в Таблице 1.

Таблица 1 – Карта меток.

<pre>item{   id: 1   name: 'A' }</pre>	<pre>item{   id: 7   name: 'G' }</pre>	<pre>item{   id: 13   name: 'M' }</pre>
<pre>item{   id: 2   name: 'B' }</pre>	<pre>item{   id: 8   name: 'H' }</pre>	<pre>item{   id: 14   name: 'N' }</pre>

<pre> item{   id: 3   name: 'C' } </pre>	<pre> item{   id: 9   name: 'I' } </pre>	<pre> item{   id: 15   name: 'O' } </pre>
<pre> item{   id: 4   name: 'D' } </pre>	<pre> item{   id: 10   name: 'G' } </pre>	<pre> item{   id: 16   name: 'P' } </pre>
<pre> item{   id: 5   name: 'E' } </pre>	<pre> item{   id: 11   name: 'K' } </pre>	<pre> item{   id: 17   name: 'R' } </pre>
<pre> item{   id: 6   name: 'F' } </pre>	<pre> item{   id: 12   name: 'L' } </pre>	<pre> item{   id: 18   name: 'S' } </pre>

После выбора модели обучения и подготовки обучающей выборки следовало создание конфигурационного файла, в котором указывались параметры обучения: количество итераций, количество классов распознавания и пути к файлам обучения. В результате обучения нейросети, которое заняло 38 часов, был сформирован файл, содержащий весовые коэффициенты.

С помощью использования библиотеки OpenCV, в приложении реализовано получение видеопотока с веб-камеры компьютера [15]. Каждый кадр отправляется на обработку нейронной сетью, в которой вычисляются координаты области нахождения жеста и происходит его классификация. Если жест в течение 1.5 секунды встречается на кадрах с частотой выше 90%, он заносится в поле для индикации показанных жестов.

Для реализации пользовательского интерфейса, который представлен на рисунке 2, использовался кроссплатформенный фреймворк Qt, а точнее его реализация для языка программирования Python - PyQt.

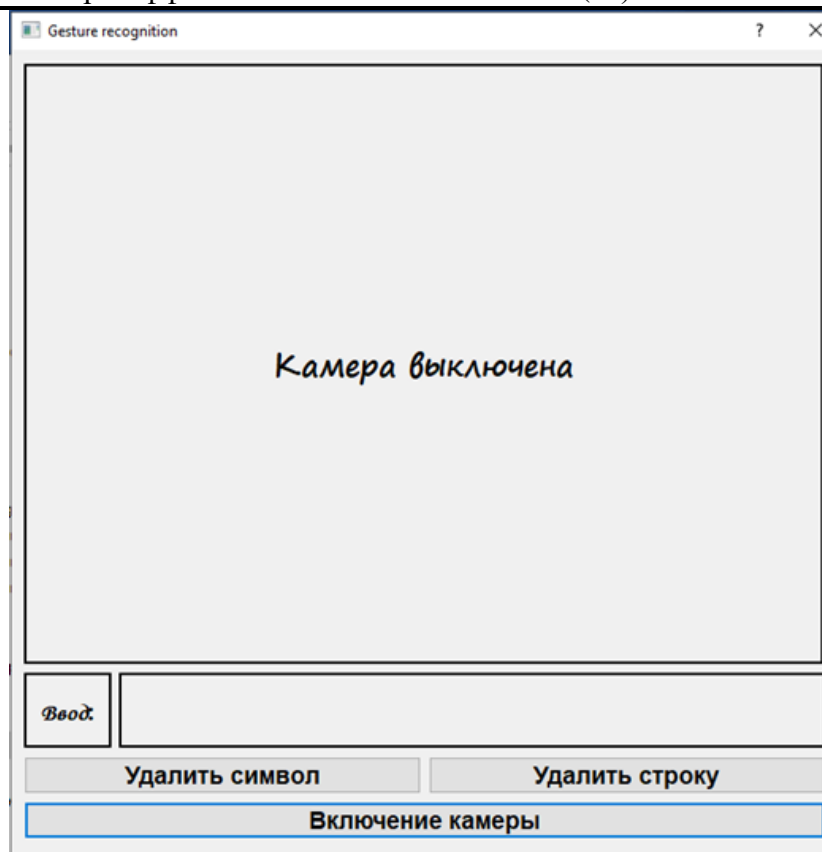


Рисунок 2 - Интерфейс приложения

Кнопка «Включить камеру» запускает видеопоток в приложении отдельно от потока обработки кадров и выводит изображение камеры на интерфейс. В случае возникновения ошибки со стороны пользователя или программы, нажатие на кнопку «Удалить символ» удаляет последний символ из слова. Аналогичную функцию выполняет кнопка «Удалить слово», после нажатия которой, происходит полное очищение строки поля вывода.

#### 4. Демонстрация работы

Пример ввода слов продемонстрирован на рисунках 3 и 4.

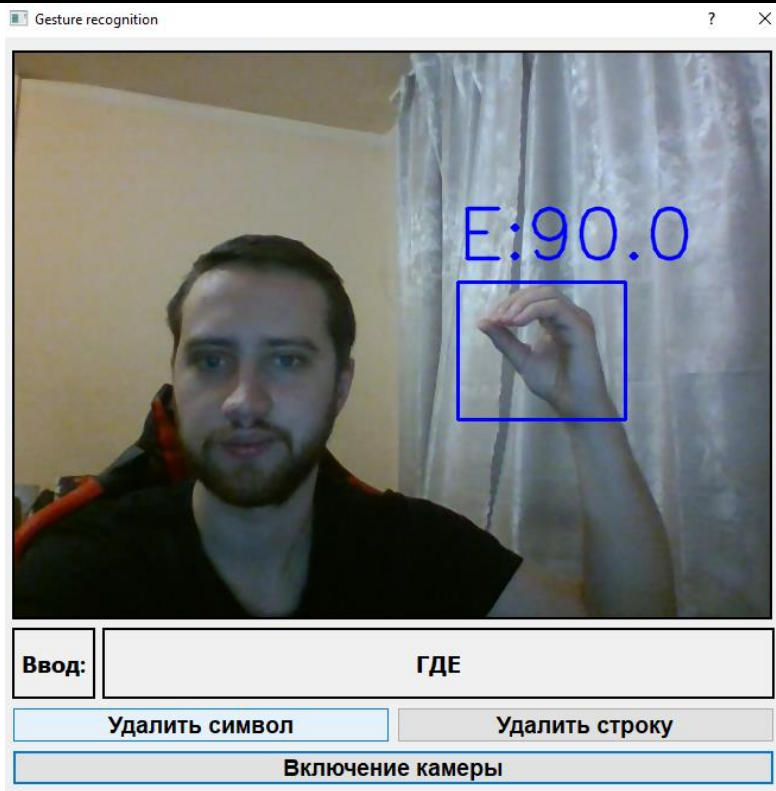


Рисунок 3 - Ввод слова «где» с помощью языка жестов

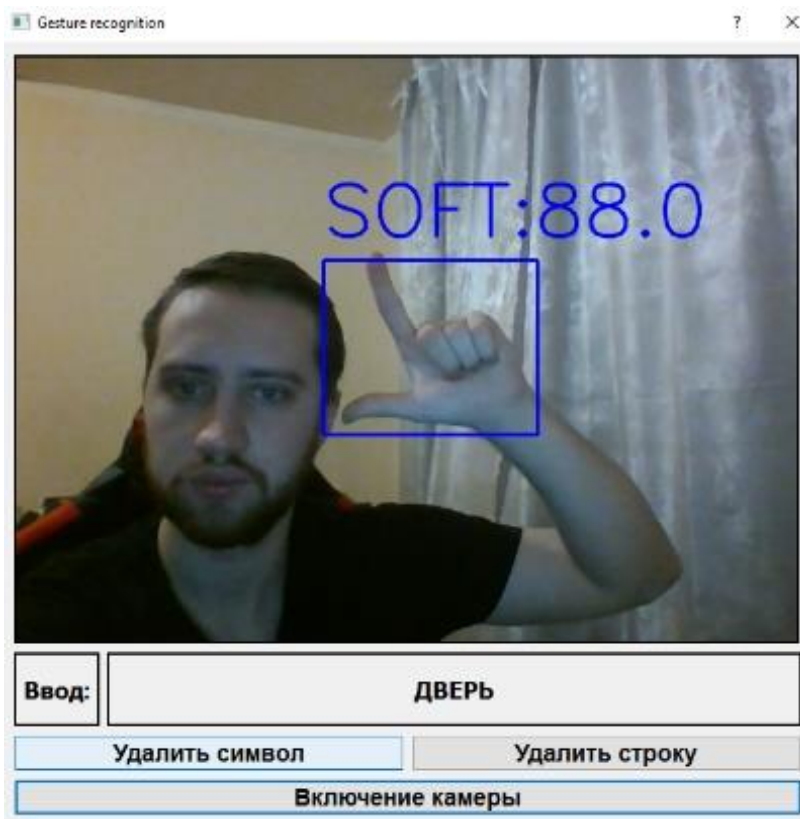


Рисунок 4 - Ввод слова «дверь» с помощью языка жестов

### **Заключение**

Представленные на рисунках 3 и 4 примеры работы разработанного программного обеспечения демонстрируют, как оно находит и распознает жест кисти руки на изображении, преобразует его в текст и записывает в поле вывода на интерфейсе приложения.

Также напротив названия класса найденного объекта можно наблюдать процент вероятности, что нейросети смогла правильно классифицировать жест. На данных примерах минимальным пороговым значением является 98 процентов. Снижение этого значения, может вызвать ложные срабатывания и ошибки в классификации жеста.

В дальнейшем планируется использовать приложение для распознавания жестов, обозначающих отдельные слова или фразы, распознавания латинской азбуки жестов, а также преобразование написанного текста в звуковую запись с помощью голосового робота. Дополнительно, планируется использовать обученную нейронную сеть и отработанные методологии при написании мобильного приложения. Это позволит сделать серьезный шаг к решению проблемы коммуникации, появляющейся при общении с глухонемыми людьми.

### **Список литературы**

1. М. С. Григорьев, А. В. Ляшков. Они нас не слышат (жизнь российских глухих и слабослышащих). Кучково поле, Москва, 2017 г.
2. Базоев В.З., Паленный В.А. Человек из мира тишины. – М.: Академкнига, 2002.
3. Базоев В. З., Гаврилова Е. Н., Егорова И. А., Ежова В. В., Давиденко Т. П., Чаушьян Н. А. Словарь русского жестового языка. М.: Флинта, 2009.
4. Галушкин, А.И. Нейронные сети: основы теории. М.: РиС, 2014.
5. Enis Berk Coban. Neural Networks and Their Applications. Istanbul Sehir University, 2016.
6. Nelson Piedra, Janneth Chicaiza, Jorge López, Jesús García. Study of the application of neural networks in internet traffic engineering. International Book Series "Information Science and Computing", 2008.
7. Adam Oken. An Introduction To and Applications of Neural Networks. 2017.
8. Zijie J. Wang, Robert Turko, Omar Shaikh, Haekyu Park, Nilaksh Das, Fred Hohman, Minsuk Kahng, and Duen Horng (Polo) Chau. CNN EXPLAINER: Learning Convolutional Neural Networks with Interactive Visualization.
9. Николенко С., Кадурич А., Архангельская Е. Глубокое обучение. Погружение в мир нейронных сетей. Питер, 2018 г.
10. Peter Goldsborough. A Tour of TensorFlow. Fakultät für Informatik Technische Universität München, 2016.
11. Tensorflow [Электронный ресурс] – Режим доступа: <https://www.tensorflow.org>
12. Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks.
13. LabelImg [Электронный ресурс] – Режим доступа: <https://github.com/tzutalin/labelImg>
14. Коэльо Л.П., Ричарт В. Построение систем машинного обучения на языке Python. ДМК Пресс, 2016.
15. Daniel Lélis Baggio, Shervin Emami, David Millán Escrivá, Khvedchenia Ievgen, Naureen Mahmood, Jason Saragih, Roy Shilkrot. Mastering OpenCV 3. - Второе издание, Packt Publishing, 2017 г.

## References

1. M. S. Grigoriev, A. V. Lyashkov. They do not hear us (the life of Russian deaf and hard of hearing). Kuchkovo field, Moscow, 2017
  2. Bazoev V.Z., Palenny V.A. A man from the world of silence. Akademkniga, 2002.
  3. Bazoev V.Z., Gavrilova E.N., Egorova I.A., Ezhova V.V., Davidenko T.P., Chaushyan N.A. Dictionary of Russian Sign Language. Flinta, 2009.
  4. Galushkin, A.I. Neural networks: foundations of the theory. Moscow: RiS, 2014.
  5. Enis Berk Coban. Neural Networks and Their Applications. Istanbul Sehir University, 2016.
  6. Nelson Piedra, Janneth Chicaiza, Jorge López, Jesús García. Study of the application of neural networks in internet traffic engineering. International Book Series "Information Science and Computing", 2008.
  7. Adam Oken. An Introduction To and Applications of Neural Networks. 2017.
  8. Zijie J. Wang, Robert Turko, Omar Shaikh, Haekyu Park, Nilaksh Das, Fred Hohman, Minsuk Kahng, and Duen Horng (Polo) Chau. CNN EXPLAINER: Learning Convolutional Neural Networks with Interactive Visualization.
  9. Nikolenko S., Kadurin A., Arkhangelskaya E. Deep learning. Dive into the world of neural networks. Peter, 2018
  10. Peter Goldsborough. A Tour of TensorFlow. Fakultät für Informatik Technische Universität München, 2016
  11. Tensorflow Available at: <https://www.tensorflow.org> (accessed 7 December 2020).
  12. Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks.
  13. LabelImg Available at: <https://github.com/tzutalin/labelImg> (accessed 10 December 2020).
  14. Coelho L.P., Richart V. Building machine learning systems in Python. DMK Press, 2016.
  15. Daniel Lélis Baggio, Shervin Emami, David Millán Escrivá, Khvedchenia Ievgen, Naureen Mahmood, Jason Saragih, Roy Shilkrot. Mastering OpenCV 3. - Второе издание, Packt Publishing, 2017 г.
-