Real-Time Recognition System of Sign Language in the Industrial Internet of Things

Artem A. Goncharenko[1], Lilia I. Voronova[2]
Department of Intelligent Systems in Management and Automation
Moscow Technical University of Communication and Informatics
Moscow, Russia
[1]artem@agsys.ru, [2]voronova.lilia@ya.ru

***Abstract*. Industrial Internet of Things implies the concept of combining all electronic devices into a single network in an industrial enterprise, where each device can communicate with each other. The task of people in such an enterprise is to control automatic systems. In addition, enterprises of increased complexity will always need people. The article describes the process of preparing and partially implementation sign language translation in real time that can be embedded in an enterprise. The choice of architecture, recognition methods and programming languages used to create the system is considered. The system was decomposed into functional subsystems.**

***Key words:* IoT, recognition systems, sign language, real-time systems, API interface, Industrial Internet of Things.**

I. INTRODUCTION

The Internet of things implies the concept of combining all electronic devices into a single network, where each device can communicate with each other. Industrial Internet of things is no exception, but do not forget that people will always be present at any enterprises and industries. Their task is and will be to control automatic systems. In addition, enterprises of increased complexity, where the use of the Internet of things can be limited, for example, steelmaking, the aviation industry or military enterprises will always need people [1; 2].

The world community has taken a direction to develop support for people with disabilities. Due to this, more and more appears on the market of various solutions for the comfortable adaptation of people with various health problems. The development of the industry does not stop at creating physical assistants, for example, a smart home or implantable hearing aids. Various software, automatic and automated systems are being developed that perform various roles to support people.

II. DEVELOPMENT OF THE SIGN LANGUAGE RECOGNITION SYSTEM

The problem of creating an automated sign language recognition system in real time is solved at the MTUCI department of the Intelligent Systems in Management and Automation. The development will allow to expand possibilities of dialogue with surrounding world of people with hearing and / or voice violation without the help of sign language interpreter [3–5].

The purpose of this project is to create an accessible tool in mass use that does not require special additional equipment. Expanding the goal for the industrial Internet of things is the creation of ready-made software that can be installed in enterprises and used in translate a sign language in real time.

The tasks necessary to solve the project are to introduce an application into the mobile device or computer of a normal user, allowing effortlessly translating the displayed gestures by the user to the video camera of the device into text. The received text can be transferred to the interlocutor by sending a message, or reproduced, due to widely used voice assistants.

III. SELECTION OF METHODS OF RECOGNITION AND ARCHITECTURE SYSTEM

To recognize gestures using a mobile device, a 2D image processing method is used. Compared with the 3D method, the 2D method is inferior in quality of recognition, however, for the 3D method to be used, additional equipment is needed, which is contrary to the purpose of the project. The essence of the 2D method is to transform the input image, so that there remains a contour or contour of the hand. This transformation can be achieved by dividing the image into color channels, converting it to monochrome, removing the background and highlighting the outline of the hand. By this principle, all the images collected for the training set of neural networks will be transformed.

The system architecture is chosen the most suitable for expansion and for the distribution of computing resources of the system hardware. Using the API architecture, you can combine different recognition technologies based on neural networks. The most popular and fast programming language for implementing neural networks is Python. Applications installed on users' devices work in various programming languages, such as Java, C #, Swift. The browser solution of the client system can be implemented in the programming languages PHP, Java and JavaScript, or in other languages used in the web. Therefore, to ensure the correct interaction of all components of the sign language recognition system in real time, the API architecture was chosen.

One of the most difficult tasks for creating a system is the implementation of the recognition algorithm. This algorithm is divided into several stages:

- detecting a hand in a frame;
- filtration and selection of hands;
- recognition of the gesture in the frame;
- checking and adding the sequence of gestures from the video stream to a single phrase or sentence.

The step of detecting the hand involves the use of a trained neural network that can select a hand at different angles and on different backgrounds. To train this neural network you need a large amount of training data set. All hands of people are

different, skin color, pigmentation, background depends on the direction of the incident light. To obtain the necessary training set, not one hundred images are required.

To recognize the gesture, the largest training set is required, rather than for detecting the hand. All people who communicate in the language of deaf-mutes do it with the speed of speech of a healthy person. Therefore, it is necessary to take into account the lubrication of the hand movement in the frame, and, consequently, the training set should also provide lubrication.

IV. System Architecture

The architecture of the system involves the interaction of two software packages installed on the user's device and on the data processing server. We suppose that our company has a conference room in which tablets with a video camera are installed, information screens that will display the result of the translation and a server that will provide input and output devices and will recognize the sign language. With this distribution of hardware running on different operating systems, the server will provide data in a standardized format for the system. This data will be converted into a display interface on each of the devices due to the client application. All these actions are possible at the expense of the chosen architecture using the API of the subsystem interaction interface.

The life cycle of the system is shown in Fig. 1. Showing gestures to the device's camera, the user starts processing and transferring video to the server through the installed application. The server performs recognition and response generation and sends it to the application, which displays the transfer information on the screen.

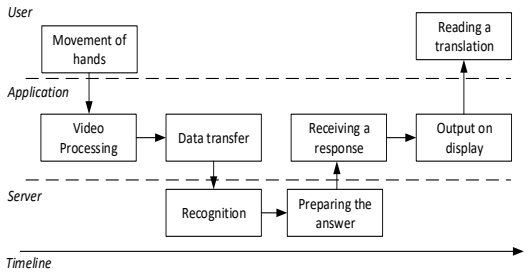

Fig. 1.The life cycle of gesture recognition

To understand the necessary subsystems of the system and their functional features, you need to decompose the system server software and the client application. When decomposing the server software, we have a number of subsystems:

The recognition subsystem is responsible for processing and recognizing the data [6–8]. It includes the following subsystems:

- The image processing and video processing subsystem responds and performs functions for detecting the hand, color processing, framing, bringing the data into a common format for the system, as well as temporarily storing the materials on the server. It has subsystems: compression, detection, storage [9–11].
- The recognition subsystem is responsible for recognizing gestures based on the prepared data, which is provided by the image processing subsystem and video.
- The answer generation subsystem is responsible for combining the received recognition results and converting them into the correct lexical format.

The learning subsystem performs the functions of training the neural networks of the system, as well as collecting data for training. It consists of subsystems:

- The data collection subsystem is responsible for collecting the data of users who have been misinterpreted, as well as collecting data in a special system learning mode, in which users can independently display phrases and write their values in text form.
- The retraining subsystem is responsible for training and retraining neural networks of the system, based on data collected by the data collection subsystem.

The access control subsystem is responsible for limiting and controlling access. Since the system is multi-user and will be used in the live chat mode, this subsystem will ensure the confidentiality of user information. It includes:

- subsystem of registration which makes registration of new users in the system and collecting the necessary information, for example, health data, if you need to urgently call the ambulance service;
- subsystem for authorization which checks the availability of registered user data in the system and the ability to access it;
- authentication Subsystem which performs an access check on every access to the system's actions.

The subsystem of data collection and analysis solves the problems of collecting data used Internet traffic, the amount of load on the computing power of the server. The entire collection of these data will provide an understanding of the need to distribute computing resources, or upgrade the system. This information is necessary, because the system must allow real-time translation and its delays are significant for users.

The operating system for the system server is selected by the Linux OS family. It is freely distributed, has a large community of developers. With the help of community repositories, you can install all necessary software through the Linux console to implement the server. The software for the MySQL database, the PHP and Python language interpreters were installed separately. Installed software for remote access to the server via SSH.

SSH is a network protocol that allows you to create a secure remote connection and manage the server. All transferred data and files are encrypted when connected via SHH, so using SSH you can create a secure connection in an unprotected environment.

V. WEB SERVER AND API

The interaction of the user through the application of the system with the server of the system will be carried out according to the following algorithm (Fig. 2): the application transmits the request to the web server, which in turn uses the interpreters of the PHP or Python languages to process the data and return the result to the client application.

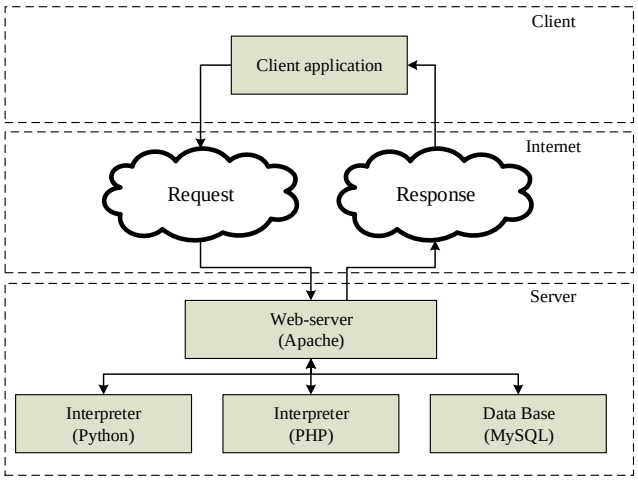

Fig. 2. Process of interaction between the application and the server

Also, for the server to work in the mode of accepting requests from the Internet, you must select and install the web server software. The Apache software is selected as the web server. It provides the use of PHP, Java and Python. Also, this web server is freely distributed, which is convenient for starting development. Python is chosen as the programming language for implementing the execution of neural network algorithms. In this language, the recognition and detection are performed most quickly. Also for Python, there are various libraries that will make it easier to process input images and create neural networks. Interaction with the database and provision of the API work is implemented in the PHP language. In the future, it is planned to translate the development of the interface into the Java programming language. At the moment, the API design of the interface is implemented using the HTTP protocol and looks like this:

https://<domain name>/<API version>/<called method>

To further protect the transmission of data, the HTTPS protocol is used. This protocol is an extension of the HTTP protocol and allows you to use data encryption with the help of special digital certificates. To use the HTTPS protocol, you must obtain a certificate that is issued by specialized certification centers on the Internet. A certificate is issued for each domain name and, depending on the type of certificate, to its subdomains, thereby providing an acknowledgment of access to the domain. There are certificates created independently, they are called “self-signed”, but the responsibility for encrypting the data and security of such a certificate lies with the developer and may be vulnerable. To get a free HTTPS certificate, the global online community offers users a non-profit certification center. Let’s encrypt it. This center issues certificates for a period of 3 months, after which it is necessary to renew the certificate. Its advantage is also that the entire process of obtaining a certificate is automated by installing software on the server to which the domain name belongs.

Recently, all modern browsers prohibit access to the webcam of the device, in the event that the domain name does not have an encryption certificate. Since one of the clients of the system is a web-browser interface, this fact must be taken into account in the development.

In multi-user systems using the API interface, several versions must be considered. With the operation of any program over time, there are errors or the desire for changes, updates from users. In order not to interrupt the system and avoid its inoperability when implementing changes and changes in the current version, it is necessary to separate and conduct version development separately from each other. Therefore, one of the parameters of accessing the interface is its version.

The called method is a pointer to the system, which components and modules need to be connected to perform the requested operation. All methods are described in the structure of the API version of the interface and can be extended or changed when the interface version is updated and changed. In each version of the interface for the sign language recognition system, there is a special router file. This is a kind of router, which describes the interaction and plug-ins, to process the request.

All parameters necessary for processing by subsystems are transmitted to the HTTP server by the method of sending POST. The HTTP protocol supports POST and GET, as well as PUT, DELETE and a number of other methods. However, the most common methods are GET and POST. The difference between them is such that, with the GET method, all the transmitted data is in clear form directly in the URI request. When using the POST method, all transmitted data, including files, is transmitted “hidden” while in the request body.

In response to the request, the API interface connects and calls the necessary methods that ensure the operation of the subsystems and returns a message in the JSON format. The structure of the answer consists of 3 parts. The first part reports on the presence of errors during the execution (error = <0 — no errors, 1 — errors occurred>), the second part transmits a digital number or the number of errors (errorCode = <error description number>), the third parameter transmits the result of the request processing (response = <Processing result>). The received response can be converted for further requests to the system or for performing the output of the results on the device screen by the system client.

VI. INTRODUCTION OF A VIDEO RECORDING AND TRANSMISSION SYSTEM

One of the tasks of implementing the system is to transfer the video stream to the data processing server. The difficulty of solving this problem lies in the fact that if the quality of the Internet connection is poor, the quality of the transmitted video may be degraded or the time delay may occur while data exchange takes place. When solving it, you need to take into account these factors and look for the most acceptable solutions:

- Compress and transfer video. When compressing files, you can achieve a significant difference in the amount of source and compressed files. However, the compression of the images brings quite a few results comparing the output with the input size.
- Conversion to another format is an interesting approach, but it requires large computing resources and speed of the user’s device. It also uses a large amount of time, which in the real-time system is practically nonexistent.
- Cropping video is the most suitable option. To detect the hand and recognize the gesture, a small image size of 80x60 pixels is required. This conversion can be

performed on a user device without significant time delays.

In addition, it is necessary to decide how to transmit the received framed video stream. In the web browser client, there are several ways to implement:

- AJAX technology is built in the JavaScript programming language and allows you to send and receive data without reloading the main page. The downside of this technology lies in the constant multiple access to the server. The results of testing the AJAX technology showed that 7000 calls to the server are created per 1 minute of video transmission, and the server should accept and answer each of them (Fig. 3). This solution is not suitable for a real-time system, because there will be a delay due to frequent connections to the server when transmitting one video stream.

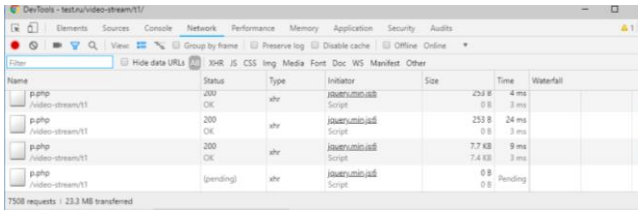

Fig. 3. Test result of AJAX technology

- WebSocket technology allows creating a direct "corridor" between the user and the server and listening to it, which does not create multiple calls to the server (Fig. 4). This technology is widely used in online systems. There are also disadvantages of the technology including being able to connect and "listen" to the server, you need to create a special WebSocket server.

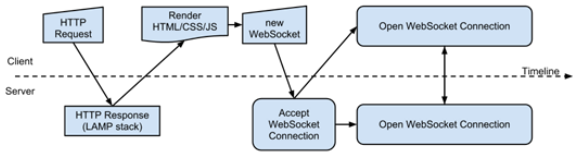

Fig. 4. The scheme of connection of technology WebSocket

VII. CONCLUSION

This article describes a variant of the real-time translation system of sign language that can be implemented in enterprises with industrial Internet of things. When creating this kind of multi-user systems, especially real-time systems using video broadcasting and recognition, a large number of subtasks arise that need to be solved during the implementation of the system and pay attention to various aspects. The article describes the chosen version of the system architecture, decomposition of the subsystems is made, methods for transmitting and processing the video stream are considered.

VIII. REFERENCES

[1] Voronov V.I., Voronova L.I. Features of realization master's program "Automation of technological processes and manufactures" // International Journal of Applied and Fundamental Research. 2016. № 2. URL: www.science-sd.com/464-25196.

[2] Безумнов Д.Н., Воронова Л.И. О развитии и стандартизации технологии интернета вещей // Технологии информационного общества: Материалы XII Международной отраслевой научно-технической конференции. М. 2018. С. 293–294. [Bezumnov D.N., Voronova L.I. On the development and standardization of the Internet of Things technology // Information Society Technologies: Proceedings XII International. branch scientific.-tech. conf. Moscow, 2018. P. 293–294].

[3] Михаеску С.В., Трунов А.С., Воронова Л.И. Анализ предметной области для разработки системы построения скелетной модели человека на основе массива опорных точек, получаемых совокупностью контроллеров Kinect // Международный студенческий научный вестник. 2015. № 3-4. С. 521–522. [Mihaesku S.V., Trunov A.S., Voronova L.I. Analysis of the subject area of system developing for building a human skeletal model based on an array of reference points obtained from a set of Kinect controllers // International Student Scientific Bulletin, 2015. N 3-4. P. 521–522].

[4] Клешнин Н.Г., Воронова Л.И. Применение нейронных сетей в подсистеме распознавания эмоций для проекта "Сурдотелефон" // Телекоммуникации и информационные технологии. 2018. Т. 5. № 1. С. 126–130. [Kleshnin N.G., Voronova L. I. Neural networks application in emotion recognition module for "surdotelephone" project // Telecommunications and information technology. 2018. T. 5. № 1. C. 126–130].

[5] Воронов В.И., Воронова Л.И., Генчель К.В. Применение параллельных алгоритмов в нейронной сети для распознавания жестового языка // Актуальные проблемы инфотелекоммуникаций в науке и образовании (АПИНО 2018). VII Международная научно-техническая и научно-методическая конференция: Сб. науч. ст.: В 4 т. / Под ред. С.В. Бачевского. М., 2018. С. 207–212. [Voronov V.I., Voronova L.I., Genchel K.V. The use of parallel algorithms in the neural network to recognize the sign language // Actual problems of information and telecommunications in science and education (APINO 2018). VII International scientific tech. and scientific method. conf.: In 4 t. / Ed. S.V. Bachevsky. M., 2018. P. 207–212].

[6] Воронова Л.И., Воронов В.И. Big Data. Методы и средства анализа: Учеб. пос. М., 2016. [Voronova K.I., Voronov V.I. Big Data. Methods and tools for analysis. Moscow, 2016].

[7] Усачев В.А., Воронов В.И. Компетенция "машинное обучение и большие данные" // Приоритетные направления развития науки и образования / Под. общ. ред. Г.Ю. Гуляева. Пенза, 2017. С. 97–108. [Usachev V.A., Voronov V.I. The competence "Machine learning and Big Data" // Priorities for development of science and education / Ed. G.Yu. Gulayev. Pensa, 2017. P. 97–108].

[8] Горячев Д.В., Воронов В.И. Большие данные и машинное обучение // Технологии информационного общества: Матер. XII Междунар. отраслевой науч.-техн. конф. 2018. С. 327–328. [Goryachev D.V. Big Data and Machine Learning // Information Society Technologies: Proceedings of 12th the international scientific-practical conference. Moscow, 2018. P. 327–328].

[9] Tsai T.H., Huang C.C., Zhang K.L. Embedded virtual mouse system by using hand gesture recognition // Consumer Electronics-Taiwan (ICCE-TW); 2015 IEEE International Conference on 2015 Jun 6. Taiwan; Taipei; New York: IEEE, 2015. P. 352–353.

[10] Huong T.N., Huu T.V., Le Xuan T. Static hand gesture recognition for vietnamese sign language (VSL) using principle components analysis // Communications, Management and Telecommunications (ComManTel). International Conference on. 2015 Decembee 28. 2015. P. 138–141.

[11] Lee D.H., Hong K.S. A Hand gesture recognition system based on difference image entropy // Advanced Information Management and Service (IMS). 6th International Conference on 2010 Nov 30. Seoul; New York: IEEE, 2010. P. 410–413.