ADVANCES IN CYBER-PHYSICAL SYSTEMS

Vol. 6, No. 1, 2021

SOFTWARE IMPLEMENTATION OF GESTURE RECOGNITION ALGORITHM USING COMPUTER VISION

Vladyslav Kotyk, Oksana Lashko

Lviv Polytechnic National University, 12, Bandery Str, Lviv, 79013, Ukraine.
Authors' e-mail*: vladyslav.kotyk.ki.2017@lpnu.ua*

https://doi.org/10.23939/acps2021.01.021

Submitted on 01.05.2021

Abstract: **This paper examines the main methods and principles of image formation, display of the sign language recognition algorithm using computer vision to improve communication between people with hearing and speech impairments. This algorithm allows to effectively recognize gestures and display information in the form of labels. A system that includes the main modules for implementing this algorithm has been designed. The modules include the implementation of perception, transformation and image processing, the creation of a neural network using artificial intelligence tools to train a model for predicting input gesture labels. The aim of this work is to create a full-fledged program for implementing a real-time gesture recognition algorithm using computer vision and machine learning.**

***Index Terms*: Computer Vision, Deep Learnig, Gaussian Blur, GrayScale, Image Classification, Image Segmenation, ML.NET, Thresholding.**

I. INTRODUCTION

Today, in the age of digital communications and with the transience of technology, information is increasingly becoming the main resource. The more information a person has, the more likely they are to succeed in a particular area of life. One of the ways to transmit information is through communication. For most people, communication is not particularly difficult, but not for people with limited properties such as hearing, vision, and speech loss. Every day they face certain problems – from contacting and making phone calls with others to receiving relevant services in government agencies. It is extremely important that these people have the opportunity to communicate with both other people and emergency workers in order to receive timely assistance. These difficulties will not allow these people to experience life to the fullest and enjoy all the opportunities. Given all the scale of this problem, humanity is trying to help them become full-fledged participants in human society, creating certain systems and means to facilitate communication between people. However, these inventions do not always help and faciitate communication. They all contain certain disadvanages and do not take into account all the features.

In particular, there are not many systems that can effectively translate into sign language, and the existing ones do not meet all the requirements and needs of people with disabilities. These systems are used as dictionaries and are mainly used for learning gestures and the most popular phrases. For a person who does not know or understand sign language, it becomes difficult to work with these software tools and they cannot always get complete information and meet the needs of others. The main reason for these difficulties is the complexity and incomprehensibility of the created programs and systems. When creating a sign language recognition system, methods and algorithms for processing multi-segment images, formed as a result of splitting the video stream into frames, play a special role. In addition, this system requires the involvement of artificial intelligence tools, namely machine learning and its main approaches [1].

II. FORMULATION OF THE PROBLEMS

Currently, software development is aimed at incresing the functionality of the application. Increasing the functionality of the program requires new more powerful electronic computers, which in turn requires significant material resources to update them. Every day the amount of information is growing exponentially and analyzing and converting this information requires high data transfer rates and large amounts of memory to store it.

One solution to this problem is to simplify and improve methods to reduce the amount of information without losing its important features. A gesture recognition system requires a large number of images to prepare the program and significant computing resources to use them. All this affects the efficiency and speed of the system. Therefore, one of the stages of implementing this gesture recognition algorithm is the use of the information preprocessing method. The purpose of creating this system is to simplify the initial information to speed up the gesture recognition process and further improve it.

III. THE NEED TO USE GESTURE RECOGNITION

Today, information is spreading so fast that humanity does not have time to perceive and process this data. And with the development of Information Technologies, this process is spreading even faster. People with hearing or speech impairments do not always have the opportunity to quickly and efficiently get the necessary information. According to the World Health Organization, more than 400 million people worldwide have speech and hearing problems. Every day the number of such people is growing more and more. All of them cannot fully exercise their right to a full life. Most of these people receive secondary education in boarding schools. Further into adulthood, it is very difficult

for them to assimilate, because over the years, people with hearing or speech impairments are considered inferior and deprived of proper living conditions. In addition, the number of people who know sign language is also limited, and no higher education institution trains qualified sign language interpreters for such a large number of people.

The essence of this development is to create a full-fledged gesture recognition system that will help people with hearing impairments adapt to normal living conditions and be full-fledged citizens. With this program, people can get proper working conditions, training, health care and simple human communication. In addition, it allows the other side – people who do not know sign language-to understand and perceive deaf-mute people at the proper level. That is, the program allows a person to feel fully in the family, society and the state, to realize themselves as a person, as an individual and as a person.

IV. SYSTEM COMPONENTS

A. IMAGE PRE-PROCESSING

Image preprocessing plays an extremely important role in this gesture recognition system.

Image conversion helps to reduce the amount of detail in the photo, highlight useful information while discarding unnecessary ones, and reduce the size of saved images. All these features help to improve and speed up the operation of the program itself, get greater system accuracy with less resources spent.

At the image preprocessing stage, filtering is applied using sets of filters, such as Euclidean, grayscale, Gaussian Blur, and others [4]. The image filtering sequence is shown in Fig.1.

At the first stage of filtering, a EuclideanColorFilter is used, which finds the nearest color using the Euclidean distance.

EuclideanColorFiltering(CenterColor, Radius);

A Euclidean distance filter is a rectangular filter where the value of each block is its Euclidean distance from the center of the filter. The center of the filter is set by the CenterColor parameter, and the distance is set by Radius.

The filter filters pixels whose color is inside or outside the RGB sphere with the specified center and radius – it stores pixels with colors inside/outside the specified sphere, and fills the rest with the specified color.

In the second step, the color-filtered image is converted to grayscale using the GrayScale filter. A color image is converted to grayscale when three primary colors of the same intensity are mixed. Each color has its own weight when mixing, and depending on this, the resulting image will contain more information from the channel that has the highest weight.

GrayScale(cR, cG, cB);

This filter accepts three parameters:

- cR – red color coefficient;
- cG – green color coefficient;
- cB – blue color coefficient.

In this case, the value of the coefficients is 0.25 for red and blue colors and 0.5 for green and is calculated using the formula:

$$GrayColor = (0.25 \cdot Red + 0.5 \cdot Green + 0.25 \cdot Blue) \quad (1)$$

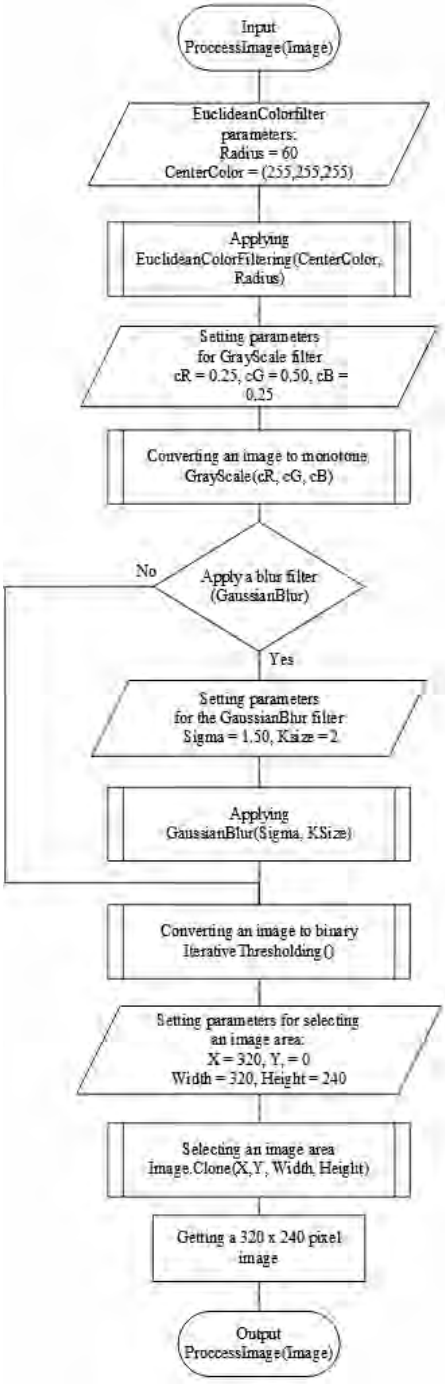

Fig. 1. Filtering algorithm

This choice of coefficients is due to the fact that modern digital images are mainly formed using a Bayer filter [2]. This filter is an array of color filters in the matrix cells that cover the LEDs. A special feature of this filter is that there are twice as many green elements as red or blue ones. In other words, this filter array consists of 25 % red elements, 25 % blue elements, and 50 % green elements.

The third stage of filtering converts a grayscale image to binary. In the binary image, there are only two colors – black or white. Each color is defined through a specific threshold value. In this system the IterativeThreshold method is used to determine the threshold value.

IterativeThreshold works as follows:

1) select any threshold value;

2) calculating the average values of the background (μB) and object (μO):

- all pixels with values below the threshold value belong to the background values;
- all pixels that are larger or equal to the threshold belong to the object's values;

3) calculating a new threshold value:

$$(\mu B + \mu O)/2 \quad (2)$$

4) If the difference between the old and new values is less than the specified minimum allowable error, then the process stops and a binary image with the new threshold is created [6].

At their own request, the user can use the Gaussian Blur filter. This filter is intended to reduce noise on images by blurring the image.

The filter performs convolution filtering using the kernel, which is calculated using the Kernel2D (Int32) method, and then converted to an integer kernel by dividing all elements by the element with the lowest value. A convolution filter that uses the kernel is known as Gaussian Blur.

Gaussian Blur is a type of image blur filter that uses the Gaussian function to calculate the transformation that is applied to each pixel in the image.

For two dimensions, this function is described by the formula:

$$G(x,y) = \frac{1}{2\pi\sigma^2} e^{\frac{x^2+y^2}{2\sigma^2}} \quad (3)$$

σ – standard deviation – affects how much neighboring pixels of the center Pixel affect the result of calculations.

GaussianBlur(Sigma, KSize);

The GaussianBlur method accepts two parameters:

- Sigma – a double variable representing the standard deviation of the Gaussian kernel in the X direction;
- KSize – a size object representing the size of the kernel.

The image area is selected to select the region of interest (ROI) [8]. This operation is performed using the Bitmap.Clone (X, Y, Width, Height), where:

- X – X-axis coordinate;
- Y – Y-axis coordinate;
- Width – width of the region of interest;
- Height – height of the region of interest.

Fig. 2. Filtering algorithm (grayscale and binary images)

B. NEURAL NETWORK TRAINING

Computer vision is one of the areas of artificial intelligence that teaches computers to interpret and understand the visual world. This area focuses on recreating parts of the complex human vision system, which allows computers to identify and process objects in images in the same way that humans do. In this work, computer vision plays an extremely important role, because it is used to obtain images necessary for deep learning [9].

Neural network training takes place using technology ML.NET, which provides a wide range of machine learning algorithms and their classification. To implement this task, we selected the ImageClassificationTrainer algorithm, which belongs to multiclass classification algorithms and performs deep neural network (DNN) training based on an existing pre-trained model, such as Resnet50, for image classification purposes [12].

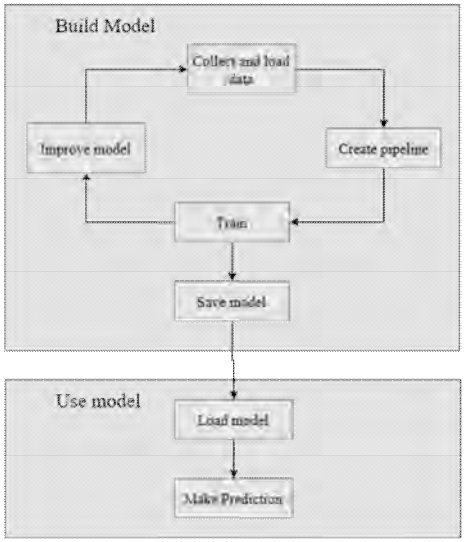

Fig. 3. BuildModel

The steps for creating and using the model (Fig.3):

- Creating a data set involves a certain number of pre-identified images that will be used for the training process.
- Building a model includes the following:
- Loading data.
- Creating a pipeline.
- Defining the main characteristics of training (architecture, number of epochs, data columns for training, etc.).
- Training-launch a pre-created pipeline.
- Accuracy assessment-setting metrics for analyzing results. At this stage, parameters such as micro-accuracy, macro-accuracy, and logarithmic losses are checked.
- The model is saved to allow further use of this model in other programs.
- Uploading is the use of a pre-created model to classify images without the need for additional training.
- Predicting involves obtaining results when using the model and evaluating their accuracy.

V. FUNCTIONS AND FUNCTIONAL PRINCIPLES OF THE SOFTWARE IMPLEMENTATION SOFTWARE IMPLEMENTATION OF GESTURE RECOGNITION AGLORITHM

The software implementation of this algorithm assumes the presence of such architectural modules of the program:

- Image input module – designed to identify graphic information input devices, generate a video stream, and create an image based on a video frame.
- Image processing module converts the resulting image to grayscale, and then to binary, applies a filtering algorithm to reduce noise.
- Neural network training module performs neural network training based on gesture images and labels that are located in the database.
- Gesture recognition module is responsible for correctly loading the trained model and displaying results.

The program assumes the presence of a graphic information input device for generating a video stream and capturing a frame.

Preprocessing of information is performed using computer vision methods, namely using image segmentation.

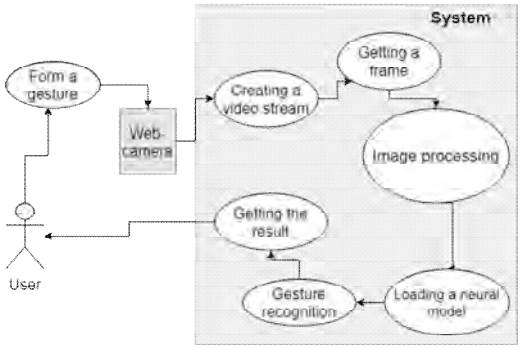

Fig. 4. Principles of work of the software implementation of gesture recognition algorithm using computer vision

The process of training a model and creating a data set takes place using another self-created program. This program is not intended for users, but is used only for creating, training, evaluating, and saving a model. The main program implements only the necessary interfaces and classes to use the pre trained model. It has a graphical interface and an extended menu for the user.

The software implementation contains two modes of operation of the program: normal and advanced.

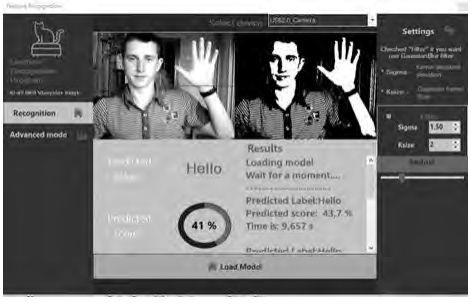

Fig. 5. Normal mode

The normal mode is shown in Fig. 5, and advanced in Fig. 6.

Fig. 6. Advanced mode

The system is tested using Unit Testing [18]. For the test data, about 2–3 images were taken for each gesture. The total number is 24 images (Fig. 7).

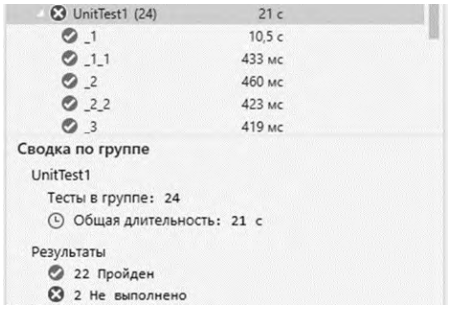

Fig. 7. Unit testing

In general, gesture recognition is more than 90% correct. Of the 24 gestures, the system correctly recognized 22. Incorrect gesture recognition is to some extent due to incoming images. The level of light, noise, and the presence of other objects in the frame make adjustments to the gesture recognition system and to the correctness of the results obtained.

VI. COMPARATIVE ANALYSIS OF EXISTING GESTURE RECOGNITION PROGRAMS

The main disadvantage of a significant part of existing sign language recognition and translation systems is the inability to provide fast translation of arbitrary gestures, but only the selection of available ones from the list. Among the available systems, you can name such as ASL Translator, Talking Hands, MotionSavvy, Mimix3D Sign Language. Some of the systems listed above are paid and require a monthly subscription to use the program. Other programs contain limited functionality and require funds to expand their own capabilities. But the main disadvantage of these products is the inability to understand people who communicate in sign language.

American Sign Language Translator is an online service created using the LingoJam platform that allows to automatically translate words and phrases in Latin into Sign Language [19]. This service generates an image of American Sign Language from the text. This translator converts text only to fingers (and numbers). Basically, it translates the regular alphabet into the American “hand Alphabet” or “finger Alphabet”, which is the hand Alphabet.

The advantages of this service are its interactive design, free use, among the disadvantages – translates only single words, requires access to the Internet.

ASL Translator – an app for translating letters, words, and public expressions from Latin to sign language. It runs on the Android operating system. It has the ability to display the translated text as a video. It recognizes more than 3.000 words and about 1.000 common phrases, translates English text into ASL characters, and generates sentences in word order in English. It has the ability to create a video in real time from the entered text or sentence. Only up to 50 words can be entered simultaneously. The app requires internet access and a paid subscription to work.

UNI – one of the means of communication for the Deaf, which allows both transmitting and receiving information, is developed by MotionSavvy [20]. The device is a tablet with a special case for it and a gesture recognition module, which is attached in the case. The program has the ability not only to pick up hand gestures and translate them into speech, but also to recognize the other person's speech and translate it into gestures. Allows you to add your own gestures to the dictionary and even share them with other users.

Fig. 8. Principles

Fig. 9. Principles

Fig. 10. Principles

VII. CONCLUSIONS

The software implementation of the gesture recognition algorithm has a great social impact. It can help solve the problem of communication between people with hearing or speech impairments and others. Countries that implement this technology will be able to exercise the right to a full life of the deaf and dumb to a certain extent. Help them get qualified medical care, education, proper working conditions, and become full-fledged participants in social and cultural life. Thanks to the use of image conversion algorithms and the correct choice of training algorithm, it will be possible to reduce the cost of computing resources, speed up the speed of program execution and make it accessible to a wide range of users.

Reducing the complexity of input data will allow to perform training in a short period of time, increase the amount of data and analyze it to get better results in program execution.

In addition, this system does not require a significant number of elements and devices to ensure its functioning. Put-ting the algorithm into action will improve the social status of people with disabilities and make their lives a little better.

REFERENCES

[1] Stenger, B., Thayananthan, A., Torr, P. and Cipolla, R., (2006). Model-based hand tracking using a hierarchical Bayesian filter. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(9), pp. 1372–1384.

[2] Wang, H., Chai, X. and Chen, X. (2016). Sparse Observation (SO) Alignment for Sign Language Recognition. *Neurocomputing*, 175, pp. 674–685.

[3] Wang, Q., Chen, X., Zhang, L., Wang, C. and Gao, W. (2007). Viewpoint invariant sign language recognition. *Computer Vision and Image Understanding*, 108(1–2).

[4] Nixon, M. and Aguado, A. (2019). *Feature extraction and image processing for computer vision*. 4th ed. New York: Academic Press, p. 650.

[5] Barghout, L. (2016). Image Segmentation Using Fuzzy Spatial-Taxon Cut: Comparison of Two Different Stage One Perception Based Input Models of Color (Bayesian Classifier and Fuzzy Constraint). *Electronic Imaging*, 2016(16), p. 1-6.

[6] Zhang, Y. and Wu, L. (2011). Optimal Multi-Level Thresholding Based on Maximum Tsallis Entropy via an Artificial Bee Colony Approach. *Entropy*, 13(4), pp. 841-859.

[7] Lai, Y. and Rosin, P. (2014). Efficient Circular Thresholding. *IEEE Transactions on Image Processing*, 23(3), pp. 992–1001.

[8] Brinkmann, R. (1999). *The Art and science of digital compositing*. San Diego, Calif.: Morgan Kaufmann, p. 184.

[9] Shapiro, L. and Stockman, G., (2001). *Computer vision*. Upper Saddle River, NJ: Prentice Hall, p. 137, 150.

[10] Morris, T. (2004). *Computer vision and image processing*. Basingstoke: Palgrave Macmillan.

[11] Vandoni, C. and Huang, T. (1996). *Proceedings / 1996 CERN School of Computing*. Geneva: CERN.

[12] Schmidhuber, J. (2015). Deep learning in neural networks: An overview. *Neural Networks*, 61, pp. 85–117.

[13] Bengio, Y. (2009). Learning Deep Architectures for AI. *Foundations and Trends® in Machine Learning*, 2(1), pp. 1–127.

[14] Cireşan, D., Meier, U., Masci, J. and Schmidhuber, J. (2012). Multi-column deep neural network for traffic sign classification. *Neural Networks*, 32, pp. 333–338.

[15] Capellman, J. (2020). *Hands-On Machine Learning with ML.NET*. [S.l.]: Packt Publishing.

[16] Esposito, D. and Esposito, F. (2020). *Introducing Machine Learning*. 1st ed. Microsoft Press, p. 256.

[17] Asthana, A. (2021). *Introducing ML.NET: Cross-platform, Proven and Open Source Machine Learning Framework | .NET Blog*. [online] Available at: https://devblogs.microsoft.com/dotnet/introducing-ml-net-cross-platform-proven-and-open-source-machine-learning-framework/.

[18] Hamill, P. (2009). Unit Test Frameworks for High-Quality Software Development. Sebastopol: O'Reilly Media, Inc.

[19] Lingojam.com. 2021. *American Sign Language Translator (ASL) — LingoJam*. [online] Available at: https://lingojam. com/AmericanSignLanguageTranslator/

[20] Techcrunch.com. (2021). *TechCrunch is now a part of Verizon Media*. [online] Available at: https://techcrunch.com/2014/06/06/motionsavvy-is-a-tablet-app-that-understands-sign-language/.

Vladyslav Kotyk is a fourth-year computer engineering student at Lviv Polytechnic National University.

His research interests include image processing and segmentation, object identification using neural networks.

Oksana Lashko was born in 1976 in Lviv, Ukraine. She received the B.S. and M.S. degree in Lviv Polytechnic State University, Lviv, in 1999. From 1999 to 2002, she was postgraduate at Lviv Polytechnic National University. Since 2007, she is a senior lecturer at the Computer Engineering Department of Lviv Polytechnic National University. Her research interests include the development of signal processing tools at the algorithmic and software levels, research of image encoding and compression problems.