I. Introduction

According to the Federal Register of Disabled Persons as of 2022, there are 3.7 million people in Russia with Group 1 or 2 disabilities as a result of their employment or from birth. As of 2019, there are 100,000 cases of ataxia and stiffness each in Russia. According to statistics from the Center for Medical Rehabilitation (CMR) at Sechenov University, there are about 200,000 cases of strokes in Russia in 2021, and in almost all cases patients need rehabilitation. Moreover, according to the UN as of 2018, natural disasters that have occurred around the world in the last 20 years, including the aftermath, have caused more than 1 million deaths [1].

In the workplace, when specialists work on construction sites or in emergency situations, an increase in the quality of work activities and a decrease in the risk of harm to health can be achieved by increasing the weight-bearing capacity of the human arm without severe limitations in their mobility. In addition, rehabilitation of patients with musculoskeletal complications is an essential factor in restoring their quality of life. In this area, the use of arm exoskeletons is also productive.

The use of an exoskeleton in medicine and industry imposes limitations on the speed (no more than 1 s) and accuracy (at least 80%) of trajectory extrapolation. Also, the trajectory must be variable for each user. At the moment, there is no control algorithm that satisfies these criteria.

In 2018, the ExoAtlet 2 motor recovery exoskeleton was developed in Russia for medical purposes, capable of solving the problem of flexion and extension of the foot, knee and lumbar joints. However, its main disadvantages are the remote control of the crutch and the cost ($51,000).

Currently, exoskeletons have the ability to solve various kinds of tasks in both the industrial and rescue sectors. The Russian passive industrial rescue exoskeleton PROEXO is able to transfer loads from arms and back to legs and has no need for a power source. Its disadvantages include low mobility, poor payload compensation. Active exoskeletons HAL (Japan, 2018) and Guardian XO (USA, 2019) have active knee, lumbar and elbow strengthening. However, both devices have high one-year rental costs (over $45,000).

Since the human behavioral system is orders of magnitude superior to all sorts of models of it in the form of neural networks and cognitive models, it is best to use signals from the muscles that humans strain as they perform a task to get the most natural direction of active reinforcement. The most popular noninvasive (i.e., without penetrating inside the body) method for obtaining such signals is electromyography (EMG), which uses measurements of biopotentials to determine muscle activity [2].

The object of the study is to develop a neural network algorithm to control the arm exoskeleton, based on the processing of human motion signals and muscle signals. Potentiometers are used as sensors to measure relative angles of rotation of the arm exoskeleton frame.

The rest of paper is organized as follow. Registration and processing of input signals, neural network architecture and training methods are presented in Section II. Extrapolation results are presented in Section III. Analysis of the results and comparison of implemented neural network architectures are discussed in Section IV. Finally, main conclusions of the study and follow-up research are presented in Section V.

II. Materials and methods

A. Signal registration

EMG is one of the methods of recording movement signals based on reading signals from muscles. This method is the only available and well-proven technology for recording movement signals. MYOstack v1.0 sensors of the Russian company Elemyo, which are a completely finished system of dry electrodes with a built-in unit for amplification, preprocessing and wireless transmission of EMG signals and which read signals in a noninvasive way, were used as sensors of the desired arm position [3].

During muscle work, the transport of charged particles (Ka, Na) through the cell membrane occurs. As a result, there is a redistribution of charge, which leads to a change in the so-called biopotential. The EMG signal is a time series which depicts the dependence amplitude of the muscle's biopotential with time (Fig. 1).

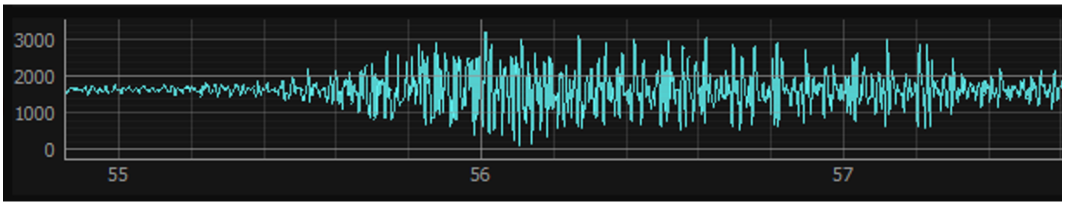


Fig. 1: Example of a read biopotential (EMG signal) of the left biceps muscle during flexion

The signal is read by surface electrodes. The main frequency range of biopotentials recorded by surface electrodes is within 20 to 300 Hz. The maximum power is at 50 - 100 Hz with an amplitude of about 50 mV. The read out data pass a stage of amplification and filtering of high (more than 800 Hz) and low (less than 10 Hz) frequencies. In our case we are taking data from five EMG sensors: 2 on the shoulder and 3 on deltoid muscle.

The data on the actual position of the hand represent the dependence of the joint angle of the hand on time (Fig. 2). On average, the value of the angle lies between 10 and 90, and the frequency is consistently less than 5 Hz. By this criterion, one can successfully filter and remove the high-frequency noise, thereby obtaining a smooth curve. At the end of preprocessing, the EMG and hand position data are stored in a text file for subsequent data set generation. In our case, data from three potentiometers, which measure the angle of rotation of the elbow joint and shoulder joint in two planes, are taken.

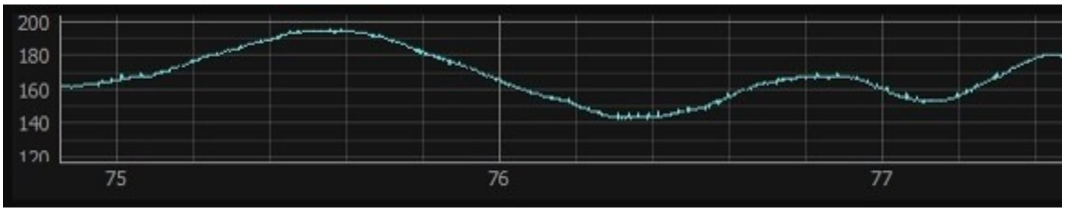


Fig. 2: Dependence of rotation angle of the left forearm

B. Formation of dataset

The trajectory of the arm exoskeleton needs to be extrapolated continuously. In this regard, the dataset structure shown in Fig. 3.

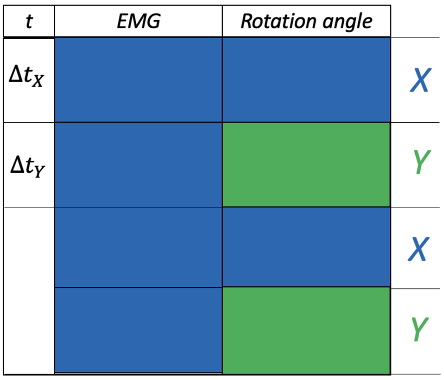
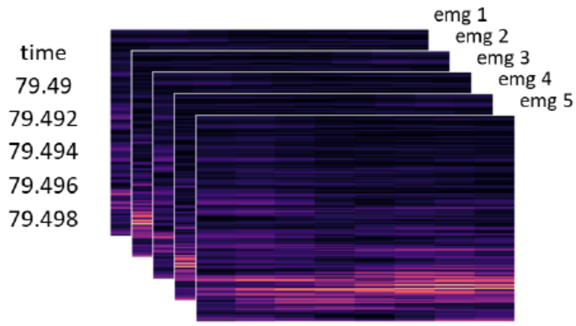


Fig. 3: Structure of the data transmitted to the input of the neural network: input blocks are blue, extrapolated by the neural network - green

First of all, based on the task, we choose the time intervals: Δt_X sec. for the input data, based on which the neural network will make an extrapolation for the next Δt_Y sec. Initially, the time step of data sampling is 2 ms. The input data consists of EMG data from 5 sensors and hand position coordinates from three sensors. Because of the continuity of the extrapolation, the target variable "Y"is extracted from the input data in the next iteration. When choosing the interval Δt_Y, it is important to keep in mind that it must be greater than the sum of $\Delta t_X + \Delta t_{comp}$, where Δt_{comp} is the time it takes the calculator to compute the extrapolation. Otherwise, the extrapolation will be discontinuous.

The signals from the EMG sensors, due to the ability to interpret the EMG signal by a superposition of frequencies, are represented as spectrograms. The input data of the hand positions do not carry frequency information, so their input and output format remains as an array of numerical values in float32 format. They are further processed as a time series. In this case, the input data package is converted to the form shown in Fig. 4.

An alternative way to feed the data into the neural network is to represent both types of signals as a time series. In this case, the time dependence of the amplitude of the biopotential on time is input directly as EMG signals, and the time dependence of the rotation angle is input as hand position data. As a result, the input data can be represented according to the figure 5.



pot 1	pot 2	pot 3
150.0	110.0	142.0
150.0	111.0	142.0
150.0	111.0	143.0
150.0	111.0	142.0
149.0	111.0	141.0
149.0	111.0	141.0
149.0	111.0	140.0
148.0	112.0	140.0

Fig. 4: Input data for the neural network when using spectrograms of EMG signals: spectrograms from the desired position sensors and float32 representation of information from actual and extrapolated position sensors

time	emg 1	emg 2	emg 3	emg 4	emg 5	pot 1	pot 2	pot 3
79.49	1654.03	1586.32	1586.32	1586.32	1586.32	150.0	110.0	142.0
79.492	1657.25	1521.84	1521.84	1521.84	1521.84	150.0	111.0	142.0
79.494	1654.03	1554.08	1554.08	1554.08	1554.08	150.0	111.0	143.0
79.496	1660.48	1741.08	1741.08	1741.08	1741.08	150.0	111.0	142.0
79.498	1660.48	1934.54	1934.54	1934.54	1934.54	149.0	111.0	141.0
						149.0	111.0	141.0
						149.0	111.0	140.0
						148.0	112.0	140.0

Fig. 5: Time stamps, (blue) readings of 5 desired position sensors (EMG) and 3 actual position sensors, as well as neural network extrapolated hand positions (green)

The data is fed to the neural network in batches of 25 time samples, or 50 ms. As output data the neural network extrapolates the next 25 moments of time. Then, there is a shift by 5 samples, and the next time signals are fed to the input of the neural network. With such a structure of data feed into the neural network the depth of the recurrent layers of the neural network is not disturbed.

C. Quality metric

For percentage estimation of the extrapolation quality, a quality metric for extrapolating the motion trajectory was developed and implemented. In consultation with experts from the Three Sisters rehabilitation centers and Sechenov University, the acceptable error (threshold) of extrapolation quality was determined, which equals 3°. The metric algorithm consists of introducing a threshold function, applying it to an array of extrapolation errors, and accepting all values above the threshold as the final error. By dividing the number of non-zero elements of the array by its length, we obtain the percentage extrapolation accuracy.

D. Neural Networks architectures

Based on the selected learning techniques neural network architectures have been developed. The architecture of a neural network is understood as a set and structure of elements, as well as data coding. The implemented architecture, which includes convolutional layers, recurrent blocks and fully connected neural network layers, is shown in Fig. 6.

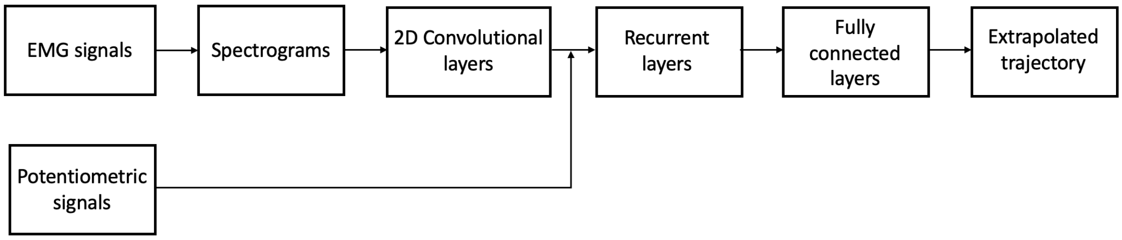


Fig. 6: Neural network architecture with recurrent layers

EMG signals are pre-transformed into spectrograms [4]. As a result, an image is fed to the input from each sensor. Two-dimensional convolution, which is a two-dimensional Fourier transform, is used as a feature extraction operation [5, 6].

After passing through the convolutional layers of the neural network, the data from potentiometers are attached to the features extracted from EMG signals, and then they are fed together to the recurrent layers of the neural network.

Finally, the data is fed to the full-connection layers of the neural network. They allow to achieve high nonlinearity of the extrapolated motion trajectory. The number of output neurons is determined from the number of samples to be extrapolated. Their values are equal to the extrapolated values of rotation angles, which allows to determine the trajectory of exoarm.

An alternative architecture option implies an autoencoder (AE). To apply the considered AE scheme, it is necessary to consider the duality of the desired (EMG) and actual hand position data. EMG data are fed to the convolution layer input in the form of a time series. At the output, the processed data are combined with the input hand position data and together are fed to the encoder, which performs compression of the number of features. The output vector is fed to the decoder which, expanding the parameter space, generates a set of numbers at the output - the coordinates of the hand at a specified time interval. The architecture of a neural network based on an autoencoder, adapted to extrapolate the trajectory of an exoskeleton with EMG signal representation as a time series, is shown in Figure 7.

Moreover, in each of the constructed architectures as input data EMG signals may be presented both in the form of spectrograms and time series.

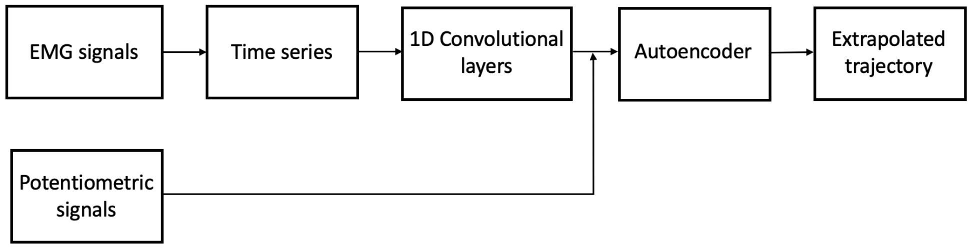


Fig. 7: Neural network architecture with AE

III. Results

The developed neural network architecture is implemented and trained through the PyTorch framework using the CUDA toolkit. The architecture of the neural network, its training algorithm, loss function and quality metric are implemented in the PyTorch library. Testing of the trained neural network algorithm was done in the Spyder development environment ver. 5.1.5. The neural network algorithm was trained using the Tesla K80 GPU. Before being fed into the neural network, the data is split into batches of 500 elements each. Adam [7] was chosen as the optimization algorithm with an optimization step of 0.007.

Neural network training with both architectures took place over 50 epochs. Extrapolated values of left forearm rotation angles for both architectures are shown in Fig. 8 and 9.

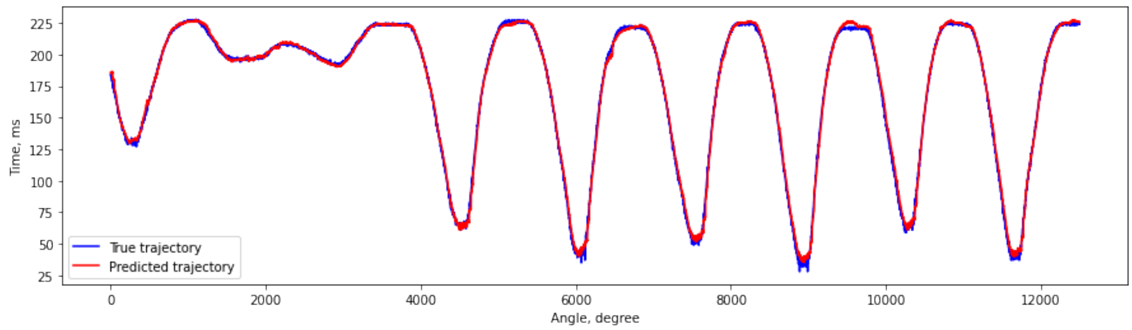


Fig. 8: Extrapolated values of potentiometer 1 in the case of architecture with recurrence layers

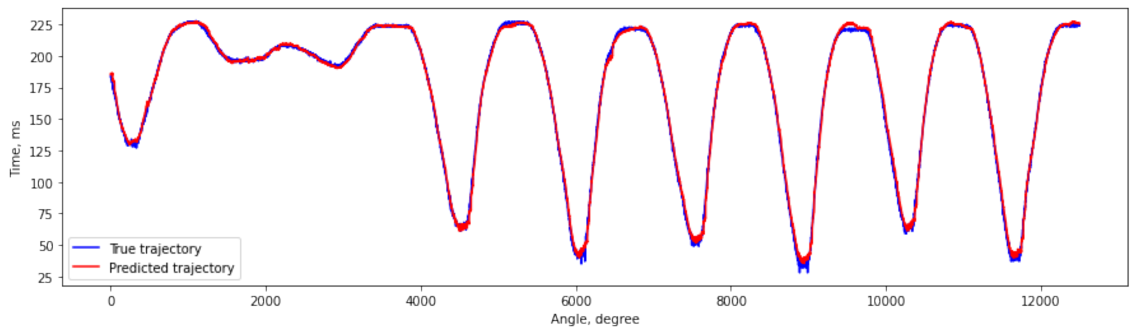


Fig. 9: Extrapolated values of potentiometer 1 in the case of architecture with AE

The extrapolated values for the three potentiometric sensors for both architectures with different types of input data are shown in Table 1.

NN architecture	Accuracy (error less, than 3°), %			Time, ms
	Potentiometer 1	Potentiometer 2	Potentiometer 3	
LSTM (spectrogram)	88.56	88.95	87.49	37.80
LSTM (time series)	87.12	86.89	88.34	19.09
AE (spectrogram)	83.01	84.20	84.42	12.74
AE (time series)	84.38	85.96	85.77	2.38

Table 1: Extrapolation accuracy and execution time per iteration for both architectures with different types of input data

The weights of a trained neural network model with an architecture with recurrent layers and EMG signals in the form of spectrograms, which has the highest accuracy, are retained to implement the real-time extrapolation. Also, the MYOstack-GUI v1.0.1 software was modified to visualize the extrapolated signal. An example output of the program is shown in figure 10.

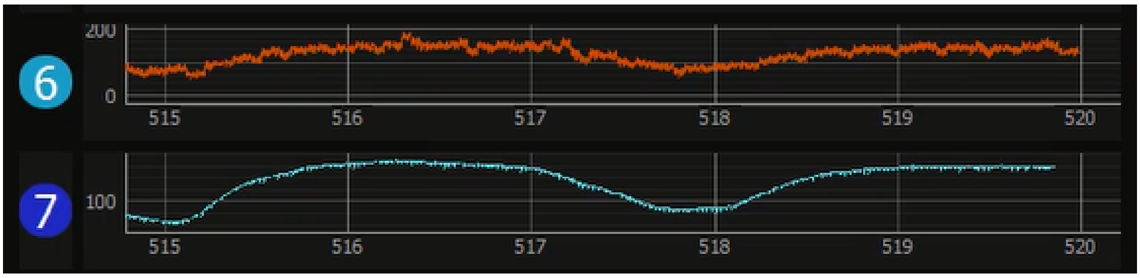


Fig. 10: Real-time extrapolated values of potentiometer 1. Orange - extrapolated trajectory, blue - actual trajectory

IV. Discussion

Accordingly, the architecture with recurrent layers and EMG signals as spectrograms showed the highest accuracy. However, the running time of the neural network algorithm with this architecture is high compared to the others, which is due to the processing time of two-dimensional images using convolutions.

V. Conclusion

The proposed method of using neural network algorithms proved to be practical for solving the problem of extrapolating the trajectory of an arm exoskeleton.

Electromyographic (EMG) muscle signals proved to be sufficiently informative and accessible signals of the desired position for their registration, processing and extrapolation of the arm exoskeleton trajectory. The most accessible registration system was MYOstack v1.1. Despite the presence of the signal interference problem in the practice, this effect did not have a strong influence on the quality of the extrapolation.

The proposed data feed structure with the definition of the predictive time interval showed its suitability for solving the trajectory extrapolation problem. However, more in-depth research is needed to find the optimal data feed and dataset formation scheme.

Recurrent neural networks and autoencoder proved to be sufficiently accurate methods for solving the problem of extrapolating the trajectory of the arm exoskeleton. The best neural network architecture in terms of accuracy was the architecture with EMG signals translated into spectrograms, but in terms of speed it is inferior to architectures using time series.

To extrapolate the exoskeleton trajectory we 1) processed electromyographic and potentiometric signals, 2) generated a data set for training the algorithm, 3) designed and trained a neural network architecture of the algorithm containing recurrent layers and an autoencoder, with different types of EMG signals as input, 4) compared the built architectures, 5) implemented control of the arm exoskeleton by extrapolating its trajectory in real time with the best accuracy neural network architecture.

As further research, we plan to develop a neural network model based on a combined architecture that includes both recurrent layers and AE.

References

[1] Natural disasters in 2018 claim over 10,000 lives. `https://www.aa.com.tr/en/life/natural-disasters-in-2018-claim-over-10-000-lives/1378790`. Accessed: 2019-01-30.

[2] Finsterer J. Emg-interference pattern analysis. *J Electromyogr Kinesiol*, 11(4):231–246, 2001.

[3] Emg signal database nina pro. `http://ninaweb.hevs.ch/`. Accessed: 2021-11-05.

[4] Sebastián Aced López. Design and construction of an emg multichannel acquisition system prototype. 2012.

[5] Xiaolong Zhai and et al. Jelfs. Self-recalibrating surface emg pattern recognition for neuroprosthesis control based on convolutional neural network. *Frontiers in Neuroscience*, 11, 2017.

[6] Wentao Wei, Yongkang Wong, Yu Du, Yu Hu, M. Kankanhalli, and Wei dong Geng. A multi-stream convolutional neural network for semg-based gesture recognition in muscle-computer interface. *Pattern Recognit. Lett.*, 119:131–138, 2019.

[7] Diederik Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *International Conference on Learning Representations*, 12 2014.