People's Democratic Republic of Algeria
Ministry of Higher Education and Scientific Research
University M'Hamed BOUGARA–Boumerdes

Institute of Electrical and Electronic Engineering

Project Report Presented in Partial Fulfillment of the Requirement for the Degree of

Master

In Electrical and Computer Engineering

Title:

Transformer-Based Hand Gesture Recognition

Presented by:
- MESSALIT Rayane
- SIF Yacine

Supervisor:
- [Name of Supervisor]

Co-Supervisor:
- [Name of Co-Supervisor]

2024/2025

Contents

List of Figures

List of Tables

Introduction and Problem Statement

0.1 Introduction

Amputations of the upper limbs significantly affect people's lives by preventing them from performing crucial tasks requiring fine motor skills. Amputees face a number of difficulties adjusting to their condition, including diminished independence, difficulty performing daily tasks, and the need for effective prosthetic substitutes. With the use of electromyographic (EMG) signals generated during muscle activity, myoelectric prosthetics offer a viable substitute for hand function rehabilitation. Their effectiveness, however, is heavily reliant on the control systems' ability to accurately interpret user intent. An important tool for enhancing prosthetics and human-machine interaction is EMG-based hand gesture recognition. By detecting muscle electrical activity, such a system can decode muscle contraction into meaningful hand motion. Despite improvements in the field, current approaches suffer from low classification accuracy, uncertain prediction, and lack of generalizability to new subjects.

0.2 Problem Statement

The major challenge in EMG-based gesture recognition for hands is the complexity and variation in EMG signals. Electrode dislodgment, muscle fatigue, and individual variation in physiology contribute to inconsistencies in signal collection among several factors. Traditional deep and machine learning techniques, such as Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs), have attained a fair level of success. However, even these suffer with regard to representing long-term dependencies and contextual information in the signals.

Transformers, originally designed for application in natural language processing, have proven to have high performance in identifying complex sequential data structures. By employing self-attention, Transformers can accurately extract meaningful features and capture long-term dependencies, thus making a powerful alternative for processing EMG signals. With all that, not much work in the field has considered employing transformer-based architectures in EMG-based hand gesture recognition, and a lack in current work in the field is addressed in this work.

Chapter 1

Literature Review (State of the Art)

1.1 Introduction

This chapter summarizes upper limb anatomy, effects of amputation, and EMG signal acquisition in gesture detection. We address muscle structure, nerve control, and EMG difficulties because of injury, as well as electrode technologies (surface, dry, implanted) and with issues with noise applicable to prosthetic control systems.

1.2 Anatomy and Physiology of Upper Limb Muscles

1.2.1 Structure and Function of Upper Limb Muscles

Numerous muscles that are arranged into anatomical compartments make up the upper limb. These muscles maintain tone, offer stability, and enable precise, fluid movement in the hand, arm, and shoulder joints.

Muscles in the upper limbs and shoulders are specialized for applying pressure and manipulating items. Sarcomeres form each of the myofibers that make up skeletal muscle, which include millions of myofibrils. Skeletal muscle fibers are entirely devoted to producing force since sarcomeres are the contractile unit. Additionally, skeletal muscle performs a variety of tasks, including voluntary movement, internal organ protection, heat production, and support for posture [1].

Muscle mass depends on the equilibrium between protein production and breakdown, and the control of this process is sensitive to several factors such as nutritional status, hormone levels, physical activity, underlying diseases, injuries, among others[2].

1.2.2 Nerve Control and Muscle Activation

The human body contains many skeletal muscles responsible for moving our body in any voluntary movement, starting from a command from the brain, this command goes down through the spinal cord, exits through a motor neuron, and goes to the muscle.

The central nerve system controls every skeletal muscle with an alpha motor neuron. The alpha motor neuron cell is located in the spinal cord in the ventral horn. The axon exits through the ventral root and inserts into muscle fiber, a motor neuron axon branches many times after entering the target muscle and each branch makes its way to different muscle fibers and forms a small cluster of terminal branches when a motor neuron fires all the muscle fibers in the motor unit contract at the same time.

The size of a motor unit varies from just a few fibers in the precise movement of the eye muscles to more than a thousand threads in the large movements of the leg muscles[3].

1.3 Upper Limb Injuries and Amputations

1.3.1 Causes of Upper Limb Amputations

Amputation is an acquired condition that results in the loss of a limb, typically due to injury, disease, or surgery. Congenital limb deficiency, distinct from acquired amputation, occurs when an infant is born without part or all of a limb [4].

The primary causes of acquired upper limb amputations include:

- **Diseases**: Vascular diseases such as peripheral vascular disease (PVD) and diabetes are leading contributors, particularly in cases of severe infection or tissue necrosis. Diabetes alone accounts for approximately 45% of non-traumatic amputations in some cohorts [6]. Chronic infections like osteomyelitis (bone infection) may also necessitate amputation if antibiotic treatment fails [5].
- **Traumatic Injuries**: Trauma is the predominant cause of upper limb amputations, responsible for 75–80% of cases [7]. These injuries often result from industrial accidents, vehicular accidents, or military combat.
- **Surgery**: Oncologic resection of malignant tumors (e.g., osteosarcoma or soft tissue sarcomas) may require amputation to achieve clear margins and prevent metastases [8]. Advances in limb-salvage techniques have reduced this need, but amputation remains critical for aggressive or recurrent tumors [9].

1.3.2 Levels of Amputation and Their Impact on EMG Signals

Upper extremity amputations are classified by level of amputation, from shoulder disarticulation through transhumeral (above elbow), elbow disarticulation, transradial (below elbow), wrist disarticulation, to partial hand amputation at the other extreme.

Different levels of amputation are known to modify the quality and other parameters of electromyographic signals, some facilitating more advanced function in prostheses. EMG signals generated at transradial and wrist disarticulation sites arise from large recruited muscle groups and are characterized by intensity and complexity. Thus, in relation to transradial and wrist disarticulations, more muscle activity allows more accurate and physiological control of the prosthesis. However, proximal limb amputations, such as shoulder disarticulations, damage several posterior muscle groups and produce a less consistent EMG pattern, limiting prosthetic control capabilities, usually forcing the need for signal processing techniques or other control strategies to generate useful motion[10].

Levels of upper amputation include[11]

- Fingers or partial hand (transcarpal).
- At the wrist (wrist disarticulation).
- Below the elbow (transradial).

- At the elbow (elbow disarticulation).
- Above the elbow (transhumeral).
- At the shoulder (shoulder disarticulation).
- Above the shoulder (forequarter).

1.3.3 What is EMG?

Electromyography (EMG) is a diagnostic technique used to record bioelectric activity produced by muscle contraction in skeletal muscles. This technique generally includes stimulation of relevant peripheral and motor neurons. The measurements are made at various levels—ranging from a single muscle fiber or a group of motor units to entire muscles—through either invasive or non-invasive methods. The information obtained with EMG is used in diagnosing neuromuscular and muscular diseases, in planning rehabilitation programs, or in controlling prosthetic devices.

Electrode configurations used in electromyography (EMG) can be monopolar, bipolar, or hybrid ones that combine both surface and intramuscular electrodes. Multi-channel systems are used mainly in surface recordings, where electrodes are systematically placed in stripe or matrix arrangements on a silicone sheet. To reduce interference, a grounding electrode is placed at a significant distance from the point of recording, and a drive configuration of the right leg similar to that used in electrocardiography (ECG) may be used (see Figure 1.1). The frequency band of EMG signals ranges from 0.01 Hz to 10 kHz, with noninvasive methods working mainly within the 50 to 150 Hz range identified as the most clinically relevant [13].

This version maintains technical accuracy with adjustments to sentence structure, word selection, and overall structure. Essential terminology (such as electrode types and frequency ranges) and references to figures or citations are not altered.

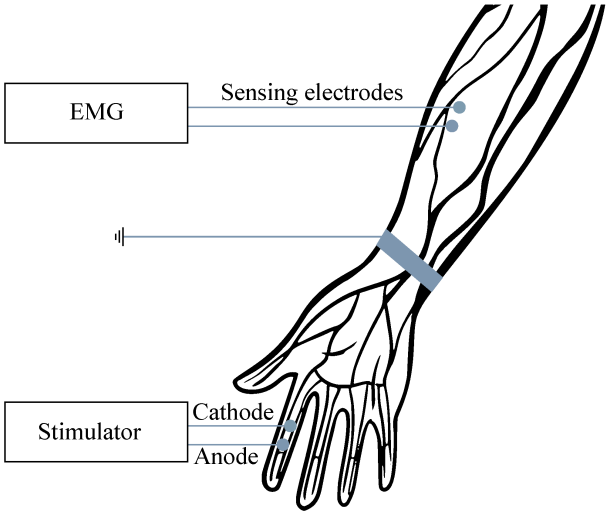

Figure 1.1: Example of the surface EMG signal measurement[13].

1.3.4 Motor Unit Action Potentials

Motor Unit Action Potentials, or MUAPs, are essentially the electrical hum of all the muscle fibers within one motor unit, providing us with some useful information on muscle functioning. Their appearance on a graph is the result of summing individual muscle fiber action potentials, and they're really key to identifying neuromuscular disorders as their alterations in shape and firing rates indicate whether something is amiss[14]. You can capture MUAPs invasively by inserting needles or wires directly into the muscle or non-invasively with surface electrodes that record a mixed signal, generally with a bit of additional noise from adjacent motor units. In EMG signal analysis, we consider MUAPs to be a filtered impulse process, which is a fancy way of saying it allows us to break them down for a closer look to utilize in clinical diagnostics, prosthetic control, grip recognition, and other technology that bridges humans and machines.

1.3.5 Electromyographic Signals Preprocessing

The informations provided by the raw EMG signals are useless. However, this information can be Beneficial if it is quantified. To obtain an accurate and actual EMG signal, a variety of signal-processing techniques are applied on raw EMG signals. A review of EMG signal processing using some different techniques is provided in this section.

1.3.5.1 Wavelet Analysis

Wavelet analysis offers a straightforward approach for handling local aspects of a sig- nal. It is an efficient mathematical tool for local analysis of nonstationary and fast transient signals. Unlike the Fourier Transform (FT), which reveals only frequency components and loses time information, wavelet transforms preserve both time and frequency localization. This is critical for EMG signals, where timing of muscle activations is crucial. The wavelet transform avoids the limitations of FT by decomposing signals into components based on scale instead of time intervals. One key application is denoising: by thresholding wavelet coefficients, noise can be separated from the true EMG signal. While time-frequency methods like the Short-Time Fourier Transform (STFT) address FT's time-localization issues, wavelet transforms belong to the time-scale category, offering a more adaptable framework for processing complex biosignals. Some of the wavelets that are commonly used for denoising biomedical signals include the Daubechies (db2, db8, and db6) wavelets and orthogonal Meyer wavelet[35]. essentially, these waveletes are diveded into are two types of wavelet transform: discrete and continuous. The time taken for processing the signal using Discrete Wavelet Transform (DWT) method is low. However, in Continuous Wavelet Transform (CWT), it is more consistent and less time-consuming due to the absence of down sampling[36]. Figure 1.2 represents the raw sEMG signal from the right rectus femoris muscle during maximum walking speed and its de-noised version using a db2 decomposition level 4.

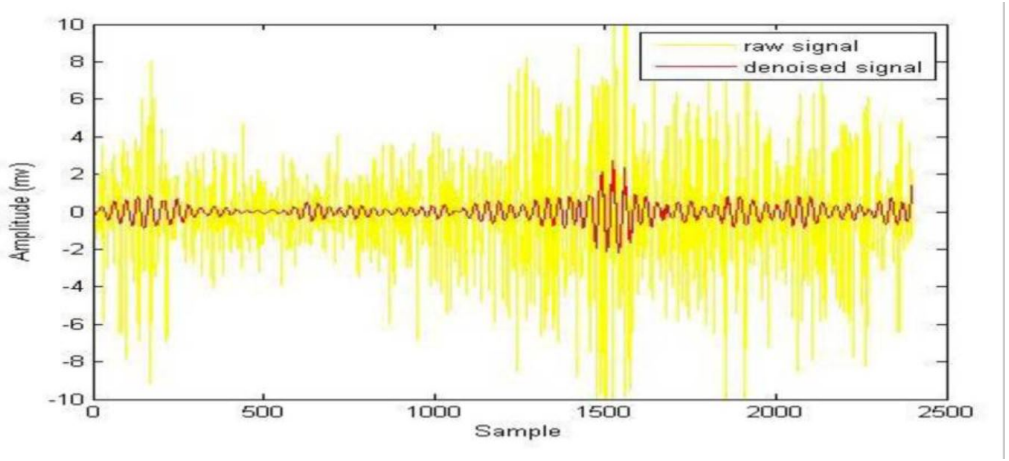

Figure 1.2: Example of the surface EMG signal measurement[36].

In the DWT, the signal is decomposed into approximation and detail signals through repeated filtering, and the approximation signal is further broken down into finer details at each level. This process, which can be iterated to multiple decomposition levels, allows for an analysis of the signal at different scales (resolution levels), providing a multi-resolution view of the signal. The choice of decomposition level affects how much detail and approximation information is preserved during signal analysis [37].

1.3.5.2 Higher Order Statistics (HOS)

Higher-order statistics (HOS) is a technique for analyzing and interpreting the characteristics and nature of a random process. The subject of HOS is based on the theory of expectation (probability theory). Due to the limitations of:

- the detection and characterization existing nonlinearities in the sEMG signal;
- Estimate the phase; and
- Exact information due to derivation from normality. HOS have been introduced in the 1960s and applied in the 1970s

1.3.5.3 Artificial Intelligence

AI techniques are effective for real-time EMG signal processing, especially Neural Networks (NNs). In 1994, Del and Park proposed a real-time application of artificial neural network that can accurately recognize the myoelectric signal (MES). Feature extraction was done through Fourier analysis and clustered using fuzzy c-means algorithm, which is a method of clustering that allows data to belong to two or more clusters. In addition to DSP hardware for fast processing. The feedforward neural network, trained via backpropagation, minimized subject training time while improving adaptability and patient acceptance. Another approach was introduced in 1996 by Cheron et al, which is the Dynamic Recurrent Neural Networks (DRNNs) to model the relationship between EMG activity and limb kinematics. DRNNs

was able to capture complex interactions without predefined control assumptions unlike the traditional models. By adeptly managing the inherent inconsistencies found in biomedical signals, fuzzy logic algorithms have been proven to be effective in EMG processing. fuzzy systems can extract patterns, tolerate contradictions, and integrate expert knowledge through IF-THEN rules, making them more interpretable than NNs. Blind Source Separation (BSS) was another proposed method. It separates a linear mixture of stationary independent sources received by different sensors by the use of higher-order statistical moments in the learning algorithm. Later, spatial time-frequency techniques was used to overcome signal overlap issues. Lastly, ANN-based models were used to compare time-domain, frequency-domain, and wavelet features for EMG classification. [38]

1.3.6 Electromyographic Signals Acquisition

1.3.6.1 EMG Electrodes Types

The detection of the electromyographic signals from the muscles of a human body can be done using different types of electrodes, two main types will be discussed in this section. the surface electrodes(skin electrodes) which are divided into two types: gelled and dry electrodes, and second type is the inserted electrodes which have also further two types: needle and fine wire electrodes.

Surface Electrodes: Surface electrodes are a type of electrode that are placed on the skin of the subject for measurement and detection of EMG signals. Two types of surface electrodes are commonly in use:

- Dry electrodes in direct contact with the skin.
- Gelled electrodes using an electrolytic gel as a chemical interface between the skin and the metallic part of the electrode. [17]

  Gelled Electrodes: For the gelled electrodes an electrolytic gel is used as a chemical interface between the skin and the metallic part of the electrode. Oxidative and reductive chemical reactions occur at the metal surface-gel interface. Silver-silver-chloride (Ag-AgCl) is the most common composite for the metallic part of gelled electrodes. The AgCl layer allows current from the muscle to pass more freely across the junction between the electrolyte and the electrode. This introduces less electrical noise into the measurement, as compared with equivalent metallic electrodes (e.g. Ag). Due to this fact, Ag-AgCl electrodes are used in over 80% of surface EMG applications (Duchene & Goubel, 1993). [17]

  Gelled electrodes can be disposable or reusable. Disposable electrodes are widely used due to their lightweight design. Disposable electrodes can come in a variety of shapes and sizes, as well as for materials used for the patch and conductive gel. Proper application of disposable electrodes reduces the possibility of displacement, especially during rapid movements.

Dry Electrodes: Dry electrodes are used directly in contact with the skin and do not require a gel like the gelled electrodes do. They typically require pre-amplifier circuitry at the electrode site due to their high electrode-skin impedance. Dry electrodes (typically $> 20g$) are considerably heavier than gelled electrodes ($< 1g$). This increased inertial mass increases the difficulty in maintaining electrode fixation, as compared to gelled electrodes. [17]

Here are some devices that use dry electrodes:

- Ottobock MyoBock: Ottobock is a German company that is specialized in producing myoelectric prostheses. One of its products is the MyoBock electrode. These electrodes are more sensitive compared to other electrodes especially for low muscle signals. In the range of high muscle signals the differentiation of the signal level is better since the change in amplification now happens logarithmically. Furthermore, due to modern frequency shielding and filtering technologies it is significantly less sensitive to low and high frequency interferences that are emitted, for example, by mobile phones or shopping center security systems. [18]

Figure 1.3: Ottobock MyoBock Electrodes [18].

- Delsys Trigno: Delsys is an American company that was founded in 1993 and is focused on solving the engineering challenges associated with wearable EMG sensors. These challenges include: low signal artifact, low crosstalk, signal reliability, and signal consistency. Their products serve a critical role in assisting thousands of researchers and educators in 85 countries worldwide to investigate and solve human movement disorder issues. One of its products is Delsys Trigno, which is a wireless EMG and motion sensor system. The sensor is ideal for real-time applications since it is wireless and some models include accelerometers, gyroscopes, and magnetometers for motion tracking. [19]

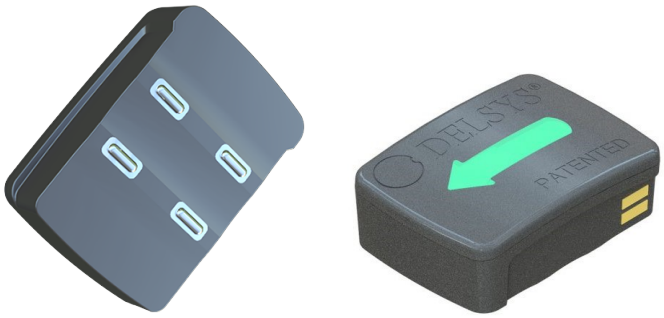

Figure 1.4: Trigno Avanti Sensor [19].

Inserted Electrodes: They are generally made of stainless steel. These electrodes are designed to penetrate the skin surface of the body to some depth to record EMG potentials of a muscle. These electrodes have to be sharp and small like subdermal needles which help them to easily penetrate the scalp. We are going to discuss two types of the inserted electrodes:

- Fine wire electrodes that consist of very thin wires that are inserted into the muscle tissue.
- Needle electrodes which are similar to fine wire electrodes in that they are inserted directly into the muscle but differ in design and application.

Fine Wire Electrodes: In kinesiological studies, human movement may cause problems. Therefore, thin and flexible fine wire electrodes are the preferred choice for invasive electrode application within deeper muscle layers. The wires are inserted by hollow needles and their proper localization can be tested by electrical stimulators or ultrasound imaging. The fine wire is inserted by inserting a needle carefully into the muscle at the selected site. After removing the needle, the distal endings of the wires are connected to steel spring adapters, which again are connected to the regular EMG pre-amplifier lead[20]. Fine wires are easy to insert and to remove from the skeletal muscles and they cause less pain compared to the needle electrodes, which stay within the muscle for the entire test.[21]

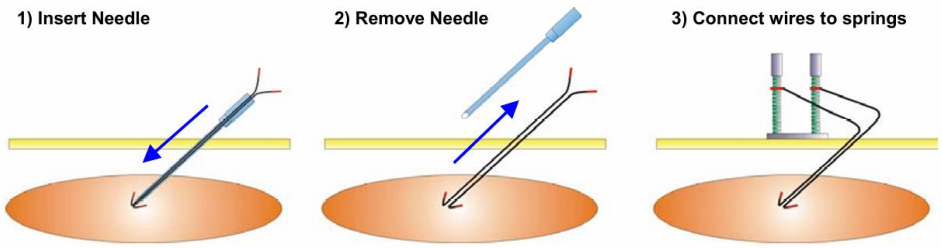

Figure 1.5: Inserting and Removal of the Fine Wire [20].

Needle Electrodes: Needle electrodes can be used to record and amplify the electrical signals generated from resting or contracting muscle fibers after inserting a needle electrode into a muscle and interpreting the signals to determine the function of the muscle fibers and motor units. Needle electromyography is generally a safe procedure. However, since the needle is inserted into the muscle, there are potential risks associated with movements, including pain, bleeding, and hematoma formation, infection, and development of pneumothorax. [22]

1.3.7 Electromyographic Signals Acquisition Challenges

Electromyography (EMG) signal acquisition is fraught with numerous challenges, each impacting the quality and reliability of the recorded data. Below is some examples of these challenges:

- Cross-talk: Cross-talk is commonly encountered when using the surface electrodes. They can only be used effectively on the superficial and large muscles to provide a stable contact area for the electrodes to mount properly, meaning that cross-talk is more problematic when dealing with smaller muscles within a complex mechanical arrangement, such as the forearm. Using surface EMG detection to isolate the activity of a single muscle is challenging. This is due to the fact that the entire limb can be viewed as a volume of conductive tissue. The electrical activity of muscles located anywhere within the limb may be conducted through the intervening tissue to reach the electrode, even if they are some distance away from the skin surface. The signals captured from unrelated muscles, that are "mixed in" with the electrical signals of the muscle of interest, are referred to as cross-talk.(Dumitru and King 1992; Farina et al. 2002). [34]

- Movement Artifact: Motion or movement artifact in EMG signals can arise from cable movement and the shifting interface between electrodes and the skin. These artifacts can have amplitudes similar to the EMG signal and usually occur in the 1–10 Hz frequency range. Recessed electrodes are a type of surface electrode designed to reduce motion artifacts in EMG recordings. small groove or indentation, use a special gel to improve contact with the skin. This reduces undesired signals caused by electrode movement, or motion artifacts. The EMG signal may still be impacted by interference from electrical differences between different layers of the skin, which these electrodes cannot stop. [36]

- Inherent Noise in the Electrode: All electronic devices produce electrical noise, also known as "inherent noise," which spans a frequency range from 0 Hz to several thousand Hz. For recording purposes, electrodes made of silver/silver chloride (10×1 mm) have been found to give adequate signal-to-noise ratio and are electrically very steady. The impedance decreases with increasing electrode size, however the size shouldn't be excessively large. On the other hand, high electrode impedance might reduce the signal-to-noise ratio, so the

impedance and size must be carefully balanced.Higher impedance is suitable for experiments that need a greater number of electrodes or high statistical power. However, lower impedance is preferable for experiments when signal quality is important. Employing advanced circuit design and high-quality instruments can help minimize noise. [36]

1.4 Conclusion

The chapter links amputation levels to EMG viability, compares electrode trade-offs (e.g., sensitivity vs. noise), and underscores signal processing hurdles (cross-talk, impedance). Integrating anatomy, injury dynamics, and EMG tech advances is key to refining responsive, adaptive prosthetics.

Chapter 2

Deep Learning for Hand Gesture Detection Systems

2.1 Introduction

In this chapter, we cover hand gesture detection techniques based on deep learning, focusing on their assistive technology applications. We first explain traditional methods like CNNs and RNNs, then introduce transformers that utilize self-attention to overcome limitations in modeling long-range dependencies. The chapter covers their evolution, advantages, and applications in enhancing the accessibility of users with motor impairments.

2.2 Traditional Approaches

2.2.1 CNN-Based Methods

Convolutional neural networks are distinguished from other neural networks by their superior performance with image, speech or audio signal inputs. They have three main types of layers, which are: [26]

- Convolutional layer.
- Pooling layer.
- Fully-connected (FC) layer.

The convolutional layer is the first layer of a convolutional network. While convolutional layers can be followed by additional convolutional layers or pooling layers, the fully-connected layer is the final layer. With each layer, the CNN increases in its complexity, identifying greater portions of the image. Earlier layers focus on simple features, such as colors and edges. As the image data progresses through the layers of the CNN, it starts to recognize larger elements or shapes of the object until it finally identifies the intended object. [23]

2.2.1.1 Convolution layer

A convolutional layer serves as the core building block of a CNN. Figure 2.1 shows the idea behind the convolution, or in other words, cross-correlation. This process works like a sliding window over a grid of data, where a filter moves across the grid, multiplying its values with the corresponding parts of the data and adding them up as it goes. This helps to extract useful information from the data grid. By learning the appropriate filter weights during training, CNNs can automatically extract meaningful features from the input data, enabling them to perform tasks like image classification. [23]

The figure illustrates the process where a 3x3 kernel is slid across the input signal, which is initially of size 4x4. With each stride, the kernel traverses the input, performing the convolution operation. As a result, we obtain a feature map reduced in size from 4x4 to 2x2.

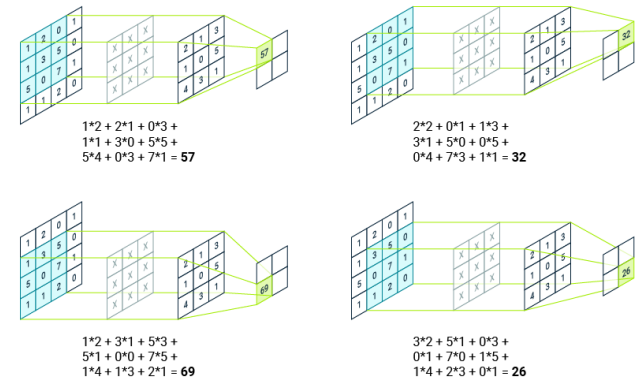

Figure 2.1: Single channel convolution visualization [27]

However, in a CNN, the input signal may consist of multiple channels as well. When a signal represents an image, each channel is a pixel matrix of a base color. Consequently, the convolutional kernel applied in such cases must also be multi-channel, with each channel having its own weight matrix. Figure 2.2 illustrates how the convolution process is carried out in such a scenario for a single kernel position (single stride). For each channel of the input signal, the corresponding channel of the kernel is subjected to element-wise multiplication and summation. The results for each channel are then aggregated. Consequently, at the end of this process, a single reduced matrix is obtained. A typical example of such an input signal is a multi-channel signal of a spectrogram image represented in 4-channel RGBA format. [23]

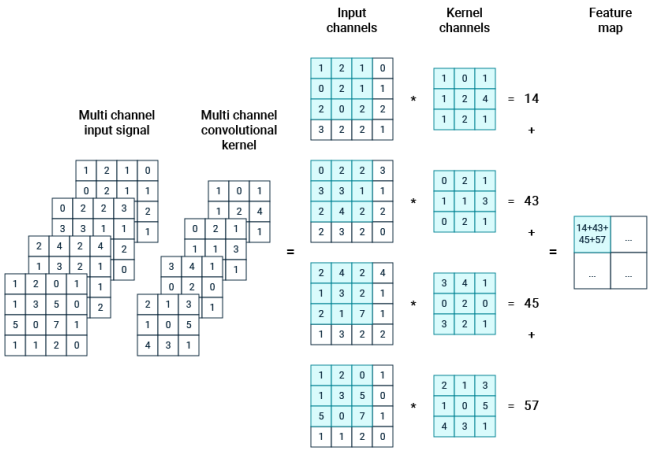

Figure 2.2: Multi-channel convolution visualization[27]

2.2.1.2 Pooling layer

After the convolutional layers, pooling layers come into play to downsample the feature maps, effectively reducing their spatial dimensions while retaining essential information. In Figure 2.3, the pooling layer with a kernel size of 2x2 is depicted

performing a maximum operation on the input layer, which condenses the output matrix dimension to 2x2. With a stride of two, the kernel shifts by two elements both horizontally and vertically while scanning the input layer. In pooling layers, aside from the max operation, other aggregating operations like average pooling, sum pooling, and several others can be used. [23]

Figure 2.3: example pooling layer visualization[27]

2.2.1.3 Fully connected layers

Fully connected layers, also referred to as dense layers, establish connections between every neuron in one layer and every neuron in the subsequent layer. These layers play an important role in classification tasks by combining features learned from preceding layers, typically flattened into a one-dimensional input vector, and associating them with the output classes. While pixel values can constitute the input vector, it more commonly encompasses flattened feature maps derived from the image via convolutional layers. [23]

2.2.1.4 Dropout

Another typical characteristic of CNNs is a Dropout layer. The Dropout layer is a mask that nullifies the contribution of some neurons towards the next layer and leaves unmodified all others. [24]

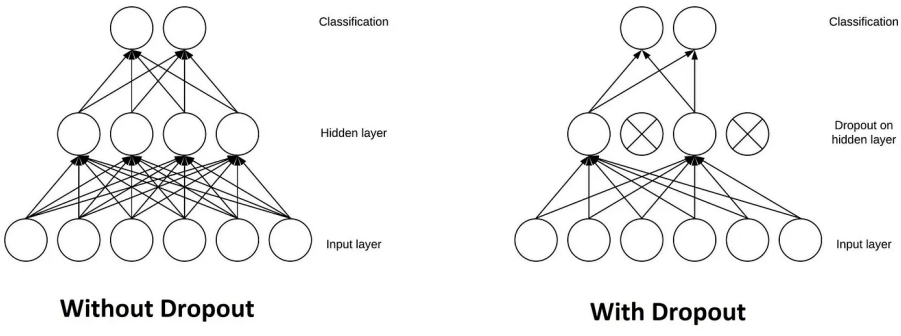

Figure 2.4: With and without dropout[24]

2.2.1.5 Activation function

An Activation Function decides whether a neuron should be activated. This means that it will decide whether the input of the neuron to the network is important or not

in the prediction process. There are several commonly used activation functions such as the ReLU, Softmax, tanH, and the Sigmoid functions. Each of these functions has a specific usage.[25]

2.2.2 RNN-Based Methods

Recurrent Neural Networks (RNNs) were introduced to address the limitations of traditional neural networks, such as FeedForward Neural Networks (FNNs), when it comes to processing sequential data. FNN takes inputs and process each input independently through a number of hidden layers without considering the order and context of other inputs. Due to which it is unable to handle sequential data effectively and capture the dependencies between inputs. As a result, FNNs are not well-suited for sequential processing tasks such as language modeling, machine translation, speech recognition, time series analysis, and many other applications that require sequential processing. To address the limitations posed by traditional neural networks, RNN comes into the picture.

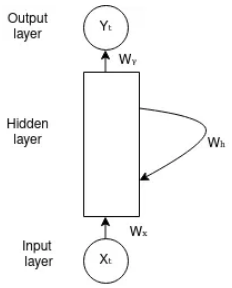

Figure 2.5: Recurrent Neural Network[28]

RNN overcomes these limitations by introducing a recurrent connection that allows information to flow from one time step to the next. This recurrent connection enables RNNs to maintain internal memory, where the output of each step is fed back as an input to the next step, allowing the network to capture the information from previous steps and utilize it in the current step, enabling the model to learn temporal dependencies and handle input of variable length.

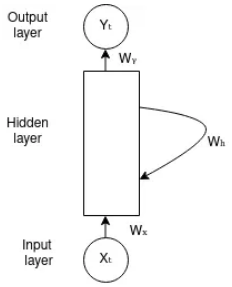

Figure 2.6: Feedforward Neural Network[28]

2.2.3 Architecture Of RNN

The RNN takes an input vector X and the network generates an output vector y by scanning the data sequentially from left to right, with each time step updating the hidden state and producing an output. It shares the same parameters across all time steps. This means that, the same set of parameters, represented by U, V, W is used consistently throughout the network. U represents the weight parameter governing the connection from input layer X to the hidden layer h , W represents the weight associated with the connection between hidden layers, and V for the connection from hidden layer h to output layer y. This sharing of parameters allows the RNN to effectively capture temporal dependencies and process sequential data more efficiently by retaining the information from previous input in its current hidden state.

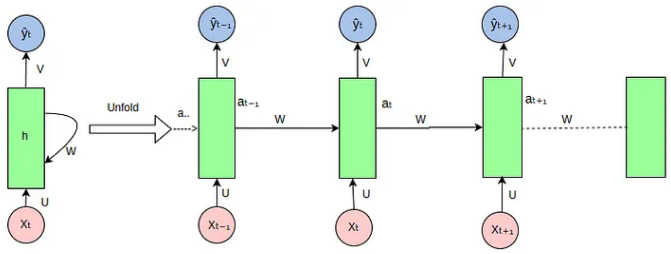

Figure 2.7: RNN Unfolded[28]

2.2.4 Limitations of RNN

During backpropagation, gradients passed in a backwards direction in time can be afflicted with the vanishing gradient problem, whereby gradients are too small to have any influence in updating network weights[28]. This is a result of successive multiplication of gradients (via the chain rule) that produces exponential decay, most typically in deep networks or sequences with long-term dependencies. Models will still eventually reach a point of convergence, but training is ineffective with infinitesimal updates to earlier layers, inhibiting learning in long-range temporal patterns.

In contrast, exploding gradients is a problem where gradients blow to ridiculously large sizes and result in numerical instability in optimization. Large gradients have a tendency to destabilize training, over-shoot optimal optima, and not reach useful minima. This is most problematic in recurrent architectures where gradients accumulate multiplication in steps in time.

To mitigate such issues, techniques such as clipping gradients (in the event of exploding gradients) and architectural innovations such as LSTM or normalization of gradients are usually utilized. Such techniques stabilize training either through clipping sizes of gradients or restructuring network components to provide a smooth flow for sequences with a large range[29].

2.3 Transformer Architectures for Gesture Recognition

2.3.1 The Transformer Revolution

The paper 'Attention Is All You Need' introduces a novel architecture called Transformer. As the title indicates, it uses the attention-mechanism we saw earlier. Like LSTM, Transformer is an architecture for transforming one sequence into another one with the help of two parts (Encoder and Decoder), but it differs from the previously described/existing sequence-to-sequence models because it does not imply any Recurrent Networks (GRU, LSTM, etc.).

Recurrent Networks were, until now, one of the best ways to capture the timely dependencies in sequences. However, the team presenting the paper proved that an architecture with only attention mechanisms without any RNN (Recurrent Neural Networks) can improve the results in translation tasks and other tasks! One improvement on Natural Language Tasks is presented by a team introducing BERT: BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding.[30]

2.3.2 Key Components

2.3.2.1 Encoder-Decoder Structure

An image is worth a thousand words, so we will start with that.

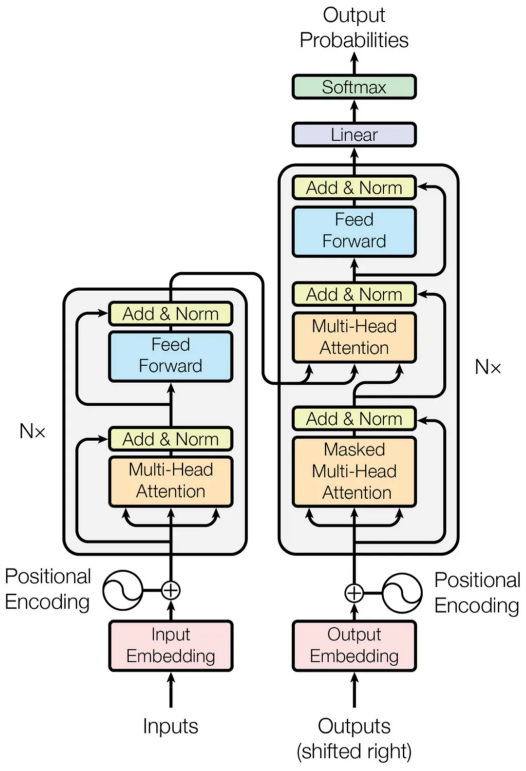

Figure 2.8: The transformer-model architecture[30]

The Encoder is on the left and the Decoder is on the right. Both Encoder and Decoder are composed of modules that can be stacked on top of each other multiple times, which is described by Nx in the figure. We see that the modules consist mainly of Multi-Head Attention and Feed Forward layers. The inputs and outputs (target sentences) are first embedded into an n-dimensional space since we cannot use strings directly.

One slight but important part of the model is the positional encoding of the different words. Since we have no recurrent networks that can remember how sequences are fed into a model, we need to somehow give every word/part in our sequence a relative position since a sequence depends on the order of its elements. These positions are added to the embedded representation (n-dimensional vector) of each word[30].

2.3.2.2 Multi-Head Attention

Let's have a closer look at these Multi-Head Attention bricks in the model:

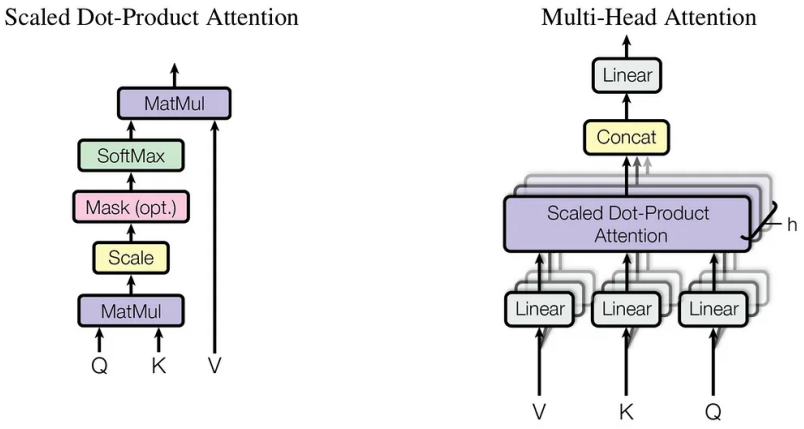

Figure 2.9: (left) scaled Dot-product attention (right) multi-head attention consists of several attention layers running in parallel[30]

We start with the left description of the attention-mechanism. It's not very complicated and can be described by the following equation:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{2.1}$$

Q is a matrix that contains the query (vector representation of one word in the sequence), K are all the keys (vector representations of all the words in the sequence) and V are the values, which are again the vector representations of all the words in the sequence. For the encoder and decoder, multi-head attention modules, V consists of the same word sequence than Q. However, for the attention module that

is taking into account the encoder and the decoder sequences, V is different from the sequence represented by Q.

To simplify this a little bit, we could say that the values in V are multiplied and summed with some attention-weights a, where our weights are defined by:

$$\mathrm{a} = \mathrm{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right) \tag{2.2}$$

This means that the weights a are defined by how each word of the sequence (represented by Q) is influenced by all the other words in the sequence (represented by K). Additionally, the SoftMax function is applied to the weights a to have a distribution between 0 and 1. Those weights are then applied to all the words in the sequence that are introduced in V (same vectors than Q for encoder and decoder but different for the module that has encoder and decoder inputs).

The righthand picture describes how this attention-mechanism can be parallelized into multiple mechanisms that can be used side by side. The attention mechanism is repeated multiple times with linear projections of Q, K and V. This allows the system to learn from different representations of Q, K and V, which is beneficial to the model. These linear representations are done by multiplying Q, K and V by weight matrices W that are learned during the training.

Those matrices Q, K and V are different for each position of the attention modules in the structure depending on whether they are in the encoder, decoder or in-between encoder and decoder. The reason is that we want to attend on either the whole encoder input sequence or a part of the decoder input sequence. The multi-head attention module that connects the encoder and decoder will make sure that the encoder input-sequence is taken into account together with the decoder input-sequence up to a given position.

After the multi-attention heads in both the encoder and decoder, we have a point-wise feed-forward layer. This little feed-forward network has identical parameters for each position, which can be described as a separate, identical linear transformation of each element from the given sequence[30].

2.3.3 Use-Case 'Transformer for Time-Series'

Why doesn't this transformer architecture work for time series? Time series acts like a language in some ways, but it's different than traditional languages. In language, you can express the same idea using vastly different words or sentence orders. Once a language-based transformer such as vanilla has been trained on a language, it can understand the relationship between words, so when you represent an idea in two different inputs, the transformer will still arrive at roughly the same meaning. Time series, however, requires a strict sequence — the order of the data points matter much more. This presents a challenge for using transformers for time series[31].

2.3.3.1 How Transformers Can Improve Time Series?

Transformers have the capability to improve the classification of time series data through self-attention mechanisms that effectively capture both global and local

temporal relationships, thus enabling discriminative feature extraction from complex multivariate data. Compared to recurrent models that process the data sequentially, the use of positional encodings by the transformer preserves the temporal relationship between the input data, an important consideration for the identification of fine class-specific structures. Recent research like ShapeFormer suggests that combining generic feature extraction with class-specific shapelet discovery using the use of transformer modules allows these models to achieve better classification performance on multivariate time series classification tasks[32]. In addition, the use of transformers allows the parallel computation of the whole sequence during training to produce greater efficiency and better representation learning for classification tasks that are dominated by long-range dependencies. These improvements suggest that with certain adaptations like improved positional encoding and classification heads, the use of transformer models is a strong rival to the use of traditional architectures for time series classification.

2.3.3.2 The Quadratic Complexity Issue

The quadratic complexity in Transformer model self-attention is a significant concern for Long Sequence Time-Series Forecasting (LSTF) due to having $O(L^2)$ space and time complexity per layer. This inefficiency is compounded with multi-layer stack memory bottleneck and thus $O(J \cdot L^2)$ overall memory usage with low scalability. Additionally, dynamic decoding with sequence-by-sequence process highly inhibits inference with speed comparable to RNN-based models. While previous approaches such as Sparse Transformer, LogSparse Transformer, Longformer, Reformer, and Linformer have attempted to reduce complexity to O(L logL) or O(L) levels, typically with a cost in real-world usability. To circumvent all such limitations, Informer introduces ProbSparse Self-Attention with complexity reduced to O(L logL), Self-Attention Distilling with space complexity reduced to O((2-)L logL) and a Generative Style Decoder for making predictions in a single forward pass with no cumulative inference error in long sequences[33].

2.4 Conclusion

The chapter contrasts and compares standard CNNs/RNNs and transformer architectures and highlights the latter's capability to globally analyze sequential information with self-attention. Transformers correct gradient instability and wastage in computation and provide scalable solutions to gesture recognition. The trend lies in their capability to enable adaptive, real-time aid-driven applications and open up more accessible human-machine interactions.

Chapter 3

Data Acquisition and Preprocessing

3.1 Datasets Used

- Public datasets (e.g., NinaPro).
- Custom dataset (if applicable).
- Data collection setup (sensors, hardware, signal specifications).

3.2 Preprocessing Techniques

- Noise filtering and signal normalization.
- Data segmentation (sliding windows, feature extraction).
- Handling imbalanced classes.

Chapter 4

Model Design and Implementation

4.1 Transformer-Based Network Architecture

- Detailed explanation of the Transformer architecture.
- Comparison with CNN/LSTM models.

4.2 Feature Representation

- Raw EMG vs. extracted features.
- Choice of input format (time-series, recurrence plots, spectrograms).

4.3 Training and Optimization

- Loss functions, optimization techniques.
- Hyperparameter tuning.

Chapter 5

Conclusion and Future Work

5.1 Summary of Findings

- Key takeaways from the study.

5.2 Limitations

- Challenges faced (dataset limitations, hardware constraints).

5.3 Future Work

- Exploring hybrid Transformer-CNN models.
- Real-time implementation for prosthetics.

Bibliography

[1] *Anatomy, Shoulder and Upper Limb, Muscles.*
Osama Javed; Kenia A. Maldonado; Roman Ashmyan.
Last Update: July 24, 2023.

[2] *The Role of Protein and Amino Acids in Sustaining and Enhancing Performance.*
Committee on Military Nutrition Research, Food and Nutrition Board, Institute of Medicine.
National Academy Press, 1999.

[3] *Neural Contributions to Muscle Fatigue: From the Brain to the Muscle and Back Again.*
Janet L. Taylor; Markus Amann; Jacques Duchateau; Romain Meeusen; Charles L. Rice.
Medicine & Science in Sports & Exercise, 2016.

[4] *Birth Defects Surveillance Toolkit: Limb Deficiencies.*
Centers for Disease Control and Prevention (CDC). (2020).
Retrieved from `https://www.cdc.gov/ncbddd/birthdefects/surveillancemanual/quick-reference-handbook/limb-deficiencies_1.html`.

[5] *Diabetic Foot: Prevention and Management.*
World Health Organization (WHO). (2020).
Retrieved from `https://www.who.int/diabetes/areas-of-work/diabetic-foot-care/en/`.

[6] *Estimating the Prevalence of Limb Loss in the United States: 2005 to 2050.*
Ziegler-Graham, K., MacKenzie, E. J., Ephraim, P. L., Travison, T. G., & Brookmeyer, R. (2008).
Archives of Physical Medicine and Rehabilitation, 89(3), 422–429. DOI: `https://doi.org/10.1016/j.apmr.2007.11.005`.

[7] *Limb Amputation and Limb Deficiency: Epidemiology and Recent Trends in the United States.*
Dillingham, T. R., Pezzin, L. E., & MacKenzie, E. J. (2002).
Southern Medical Journal, 95(8), 875–883. DOI: https://doi.org/10.1097/00007611-200208000-00018.

[8] *SEER Program: Bone and Joint Cancer.*
National Cancer Institute (NCI). (2021).
Retrieved from https://seer.cancer.gov/statfacts/html/bones.html.

[9] *Management of Upper Extremity Tumors.*
Tintle, S. M., Bacchler, M. F., & Levin, L. S. (2010).
Journal of Hand Surgery, 35(10), 1701–1710. DOI: https://doi.org/10.1016/j.jhsa.2010.08.004.

[10] *Epidemiology of Traumatic Upper Limb Amputations.*
G. Pomares; H. Coudane; F. Dap; G. Dautel.
Received: 17 January 2017; Accepted: 1 December 2017;
Available online: 2 February 2018; Version of Record: 27 March 2018.

[11] *Upper Extremity Amputation.*
University of Michigan Health.
Accessed: October 10, 2023.
URL: https://www.uofmhealth.org/conditions-treatments/rehabilitation/upper-extremity-amputation.

[12] *Epidemiology of Traumatic Upper Limb Amputations.*
Radek Martinek; Martina Ladrova; Michaela Sidikova; Rene Jaros; Khosrow Behbehani; Radana Kahankova; Aleksandra Kawala-Sterniuk.
Published: 10 September 2021.

[13] *Advanced Bioelectrical Signal Processing Methods: Past, Present, and Future Approach—Part III: Other Biosignals.*
Radek Martinek; Martina Ladrova; Michaela Sidikova; Rene Jaros; Khosrow Behbehani; Radana Kahankova; Aleksandra Kawala-Sterniuk.
Published: 10 September 2021.

[14] *A novel method for automated EMG decomposition and MUAP classification.*
C.D. Katsis; Y. Goletsis; A. Likas; D.I. Fotiadis; I. Sarmas.
Available online 27 December 2005.

[15] *Anatomy, Shoulder and Upper Limb, Muscles.*
Osama Javed; Kenia A. Maldonado; Roman Ashmyan.
Last Update: July 24, 2023.

[16] *Amputation.*
Stanford Medicine.
Last Update: July 24, 2023.

[17] *Important Factors in Surface EMG Measurement.*
Dr. Scott Day.
Bortec Biomedical Incorporated, n.d. [Additional bibliographic details needed].

[18] *MYOBOCK® Electrodes.*
Otto Bock HealthCare. (n.d.).

[19] *Delsys – Wearable Sensors for Movement Sciences.*
Delsys.
Retrieved on January 2, 2025 from `https://delsys.com/`.

[20] *The ABC of EMG: A Practical Introduction to Kinesiological Electromyography.*
Konrad, Peter. (2005).

[21] *Signal Acquisition Using Surface EMG and Circuit Design Considerations for Robotic Prosthesis.*
M. Zahak.
In: *Computational Intelligence in Electromyography Analysis – A Perspective on Current Applications and Future Challenges.* InTech, October 17, 2012. doi: 10.5772/52556.

[22] *Needle Electromyography: Basic Concepts.*
Rubin, D. I. (2019).
Handbook of Clinical Neurology, 160, 243–256. DOI: `https://doi.org/10.1016/B978-0-444-64032-1.00016-3`. PMID: 31277852.

[23] Shadman Sakib*, Nazib Ahmed#, Ahmed Jawad Kabir@, and Hridon Ahmed$, *An Overview of Convolutional Neural Network: Its Architecture and Applications*

[24] Sirine Amrane, *How do you beat overfitting in machine learning?*, Medium, [Online]. Available: `https://sirineamrane.medium.com/how-do-you-beat-overfitting-in-machine-learning-cf839265b345`.

[25] Sirine Amrane, *Activation Functions in Classification: Part 1 - Softmax and Sigmoid*, Medium, [Online]. Available: `https://sirineamrane.medium.com/activation-functions-in-classification-part-1-softmax-and-sigmoid-324ce6294fbc`

[26] Dharmaraj, *Convolutional Neural Networks (CNN) Architectures Explained*, Medium, [Online]. Available: `https://medium.com/@draj0718/convolutional-neural-networks-cnn-architectures-explained-716fb197b243`.

[27] Tomasz Janaszka, Mariusz Budziński, *Convolutional Networks for Time Series Classification*, CodiLime Blog, [Online]. Available: `https://codilime.com/blog/convolutional-networks-for-time-series-classification/`.

[28] Sirine Amrane, *Recurrent Neural Network (RNN) Architecture Explained*, Medium, [Online]. Available: `https://medium.com/@poudelsushmita878/recurrent-neural-network-rnn-architecture-explained-1d69560541ef`.

[29] *Take A Shortcut Back: Mitigating the Gradient Vanishing for Training Spiking Neural Networks.* Yufei Guo; Yuanpei Chen; Zecheng Hao; Weihang Peng; Zhou Jie; Yuhan Zhang; Xiaode Liu; Zhe Ma. Advances in Neural Information Processing Systems, 2024.

[30] *Attention Is All You Need.*
Ashish Vaswani; Noam Shazeer; Niki Parmar; Jakob Uszkoreit; Llion Jones; Aidan N. Gomez; Lukasz Kaiser; Illia Polosukhin.
Published: June 12, 2017.

[31] Qingsong Wen, Tian Zhou, Chaoli Zhang, Weiqi Chen, Ziqing Ma, Junchi Yan, and Liang Sun, *Transformers in Time Series: A Survey*,

[32] Xingguang Yan, Liqiang Lin, Niloy J. Mitra, Dani Lischinski, Daniel Cohen-Or, and Hui Huang, *ShapeFormer: Transformer-based Shape Completion via Sparse Representation*, arXiv:2201.10326 [cs.CV], 2022.

[33] Zhou, H., Zhang, S., Peng, J., *et al.* "Informer: Beyond Efficient Transformer for Long Sequence Time-Series Forecasting," *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(12), pp. 11106–11115, 2021.

[34] G. Kamen and D. A. Gabriel, Essentials of Electromyography. Human Kinetics, 2010.

[35] R. Thukral and G. Kumar, 'Analysis of EMG Signals Based on Wavelet Transform-A Review', International Journal of Web Engineering and Technology, vol. 2, pp. 3132–3135, 07 2015.

[36] R. H. Chowdhury, M. B. Reaz, M. A. Ali, A. A. Bakar, K. Chellappan, and T. G. Chang, "Surface electromyography signal processing and classification techniques," *Sensors (Basel)*, vol. 13, no. 9, pp. 12431-12466, Sep. 2013, doi: 10.3390/s130912431.

[37] J. Pauk, 'Different techniques for EMG signal processing', Journal of Vibroengineering, vol. 10, pp. 571–576, 12 2008.

[38] M. B. I. Reaz, M. S. Hussain, and F. Mohd-Yasin, "Techniques of EMG signal analysis: detection, processing, classification and applications (Correction)," Biological Procedures Online, vol. 8, no. 1, p. 163, Oct. 2006, doi: 10.1251/bpo124.