

Article

# Myoelectric Pattern Recognition Using Gramian Angular Field and Convolutional Neural Networks for Muscle–Computer Interface

Junjun Fan <sup>1,2,3</sup>, Jiajun Wen <sup>1,2,3,\*</sup> and Zhihui Lai <sup>1,2,3</sup><sup>1</sup> College of Computer Science & Software Engineering, Shenzhen University, Shenzhen 518060, China<sup>2</sup> Guangdong Key Laboratory of Intelligent Information Processing, Shenzhen University, Shenzhen 518060, China<sup>3</sup> Guangdong Laboratory of Artificial-Intelligence and Cyber-Economics (SZ), Shenzhen University, Shenzhen 518060, China

\* Correspondence: enjoy\_world@163.com

**Abstract:** In the field of the muscle–computer interface, the most challenging task is extracting patterns from complex surface electromyography (sEMG) signals to improve the performance of myoelectric pattern recognition. To address this problem, a two-stage architecture, consisting of Gramian angular field (GAF)-based 2D representation and convolutional neural network (CNN)-based classification (GAF-CNN), is proposed. To explore discriminant channel features from sEMG signals, sEMG-GAF transformation is proposed for time sequence signal representation and feature modeling, in which the instantaneous values of multichannel sEMG signals are encoded in image form. A deep CNN model is introduced to extract high-level semantic features lying in image-form-based time sequence signals concerning instantaneous values for image classification. An insight analysis explains the rationale behind the advantages of the proposed method. Extensive experiments are conducted on benchmark publicly available sEMG datasets, i.e., NinaPro and CagpMyo, whose experimental results validate that the proposed GAF-CNN method is comparable to the state-of-the-art methods, as reported by previous work incorporating CNN models.

**Keywords:** muscle–computer interface; surface electromyography; myoelectric pattern recognition; Gramian angular field; convolutional neural networks



**Citation:** Fan, J.; Wen, J.; Lai, Z. Myoelectric Pattern Recognition Using Gramian Angular Field and Convolutional Neural Networks for Muscle–Computer Interface. *Sensors* **2023**, *23*, 2715. <https://doi.org/10.3390/s23052715>

Academic Editor: Javier Ramírez

Received: 11 December 2022

Revised: 19 February 2023

Accepted: 24 February 2023

Published: 1 March 2023



**Copyright:** © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

Electromyographic signals (EMGs), which provide representations of electrical potential fields, are crucial biomedical signals produced by membrane depolarization of the muscle fibers. The well-established surface electromyography (sEMG) is a kind of EMG signal acquired from the skin surface [1]. Traditionally, sEMG signals are used to analyze muscle activities and assess functional diagnosis in clinical settings [2]. Recent studies have pointed out that sEMG can be exploited as a feasible input modality for a computer to generate a muscle–computer interface (MCI) with a wide range of applications, such as controlling prosthetics [3], human–computer interfaces [4,5], and sign language recognition [6]. Saponas et al. achieved an accuracy of 78.0% when classifying wrist and finger flexion via forearm sEMG sensing [7]. Huang et al. used an sEMG wristband to detect hand–smartphone interaction, obtaining an accuracy of 82.9% for classifying fine-grained thumb gestures such as left swipe, right swipe, tap, and long press [5]. The above studies have demonstrated the feasibility of using forearm electromyography for MCI. As the performance of traditional vision-based human–computer interaction (HCI) methods depend on the constrained environment, e.g., indoor environment, stable lighting conditions, and simple backgrounds [8], myoelectric pattern recognition is a ubiquitous HCI method that has strong robustness to the environment.

The identification of sEMG signals with regard to various motion patterns is the most important task for MCI. Meanwhile, conventional machine learning techniques have shown promising performance in a variety of applications [9]. However, difficulties in extracting the meaningful handcrafted and domain-dependent features still exist in the application of sEMG signal analysis. Recent studies in deep learning (DL) [10] show that deep networks are able to intrinsically extract important features and better performances on large-scale datasets [11–13] can be achieved. With advances in DL, many studies have explored the possibility of myoelectric pattern recognition without handcraft features, which overcomes the drawbacks of conventional machine learning methods by allowing the transformation of data to a more abstract representation [14–16]. The main popular DL architectures that are applied to myoelectric pattern recognition include deep neural networks (DNNs), deep recurrent networks (DRNNs), convolutional neural networks (CNNs), autoencoders (AEs), and deep belief networks (DBNs) [17]. Park et al. [15] built a CNN-based classification model that obtained promising results for six common gestures, which were 12–18% higher than the conventional machine learning techniques. Most of the approaches are data-driven and employ the sEMG data directly as input [17]. However, sEMG is a kind of nonstationary micro-electrical signal, which makes it hard to extract the patterns when directly using the raw sEMG signals as the model input in deep learning. Therefore, it is especially important to adjust the input of the deep model appropriately according to the characteristics of electromyographic signals.

To address the problem caused by using sEMG data as the direct input for CNN learning [18,19], we propose conducting sEMG signal feature extraction after signal morphology conversion under a DL framework.

Gramian angular field (GAF) encodes one-dimensional data into a two-dimensional symmetrical matrix with the matrix elements defined in polar coordinates [20]. Existing works exploited the potential of using GAF encoding time-series data into 2D images for CNN classification in many applications, such as multivariate time-series image forecasting [21], fiber Bragg grating sensing [22], and electromagnetic modulation signal classification [23]. Inspired by these works, we found that the GAF method can reflect the combination of signals at different positions of one-dimensional data in the form of two-dimensional data. Therefore, it would be feasible to use the GAF method to encode multichannel sEMG signals as images, which can make the feature space more expressive by representing relationships between the values in different sensor channels rather than treating each channel independently. Based on the GAF method and a CNN-based model, we designed a two-stage architecture, GAF-CNN, consisting of a 2D representation stage and a CNN-based classification stage, to improve the accuracy of myoelectric pattern recognition. GAF-CNN employs the GAF method to encode the instantaneous values of multichannel sEMG signals to generate sEMG-GAF images. A deep convolutional neural network (DCNN) model is proposed for automatic sEMG-GAF image feature extraction and classification. It is well known that DCNN can preserve the neighborhood connection and spatial characteristics among local regions [24]. Considering the small scale of the dataset, a DCNN framework is constructed using an input layer, five convolutional layers, and three fully connected layers. Batch normalization is employed instead of using max-pooling layers after convolution to optimize the network. A dropout strategy is adopted on the two fully connected layers to avoid overfitting. This work makes the following contributions:

1. A two-stage architecture consisting of a 2D presentation stage and a CNN-based classification stage is proposed for myoelectric pattern recognition, where the raw sEMG signal is converted to a suitable 2D representation for better CNN feature extraction.
2. A 2D presentation method employing GAF is proposed to encode instantaneous multichannel sEMG signals in image form, and the rationale of the proposed method is given.
3. Evaluations on four benchmark sEMG datasets show that GAF-CNN has outstanding classification performance compared with the state-of-the-art CNN-based myoelectric pattern recognition methods.

The remainder of this paper is organized as follows. Section 2 presents the related work. Section 3 describes the GAF-CNN method for gesture recognition and explains its rationale. Section 4 presents our experimental results. Section 5 states our conclusions.

## 2. Related Work

Recent studies show the promise of machine learning in feature extraction, which has frequently been used in myoelectric pattern recognition for human movement identification [17]. Conventional machine learning pipelines include data acquisition, feature extraction, mathematical modeling, and inference. Frequently used feature extraction methods for myoelectric pattern recognition fall into three categories: (1) time-domain (TD) features, such as mean absolute value (MAV), root mean square (RMS), and zero crossings (ZCs); (2) frequency domain (FD) features, such as power spectrum (PS) and autoregressive coefficients; and (3) time–frequency domain (TFD) features, such as discrete wavelet transform (DWT) and wavelet packet transform (WPT) [25]. These features have been well explored and proved effective in myoelectric pattern recognition [26]. For example, Fan et al. employed 12 types of commonly used EMG features to represent object interaction and investigated their classification performance [27]. New features for gesture recognition have been proposed to improve the classification performance, such as active muscle regions based on the mapping relationship between hand movements and forearm active muscle regions [11]. Conventional classification algorithms such as support vector machine (SVM), k-nearest neighbors (k-NNs), hidden Markov models (HMMs), decision trees, random forest, linear discriminant analysis, and artificial neural networks (ANNs) have been explored to improve the performance of gesture recognition [12,13]. However, these methods cannot achieve high accuracy in real-world applications due to their poor generalization ability and complex parameter adjustment process, which depends on artificially designed extractors and professional knowledge.

In the past decade, the DL technique has undergone significant development. The deep network can intrinsically extract effective features, enabling competitive performance on challenging datasets [28,29]. With the advancements of deep learning, the in-depth features of sEMG signals have been explored and are well-studied for gesture recognition [17,26]. To use the advantages of sEMG signals and deep architecture, Zhang et al. proposed a long short-term memory (LSTM) algorithm to recognize hand gestures through multimodal sEMG data and obtain competitive classification performance [16]. Considering the pros and cons of CNN and LSTM, LSTM-CNN (LCNN) models were proposed to construct autoencoders for automatic feature extraction [30]. Notably, deep convolutional neural networks (DCNNs) can automatically extract crucial features from signals and have been used for myoelectric pattern recognition without handcraft features. Atzori et al. [18] proposed a concise CNN architecture based on five blocks of convolutional and pooling layers with better classification accuracy than classical methods. Geng et al. [19] used instantaneous sEMG images for gesture classification with 89.3% accuracy on eight types of movements.

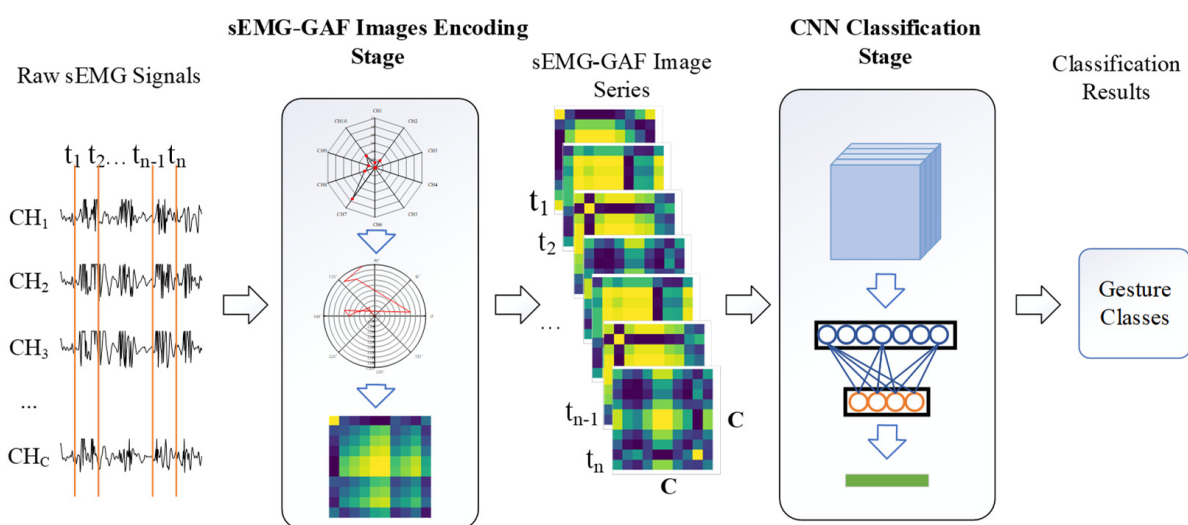
To further explore the correlation between specific gestures and the related sEMG signals of muscular activities, Wei et al. proposed a multi-stream CNN architecture, which divides the input data into small-size images and extracts effective features by convolutional layers before further processing with fully connected layers [31]. Considering that two-dimensional (2D) CNNs with 2D kernels cannot handle a sequence of images that carries signals over time, Chen et al. presented a three-dimensional (3D) CNN with 3D kernels to capture both spatial and temporal structures from sequential sEMG images. Experiments showed that the accuracies of 3D CNN were 18.6% higher than those of 2D CNN when the time window duration was 150 ms on a high-density sEMG dataset [32].

Recently, novel DL frameworks have been designed for myoelectric pattern recognition. Demir et al. [33] proposed a deep-transfer-learning-based approach with sEMG signals for human action classification, obtaining 98.65% accuracy, better than conventional methods, by fine-tuning the AlexNet model. Most of the above methods focus on improv-

ing DL models, achieving significant improvements in classification accuracy. Nevertheless, they usually use raw sEMG signals as image input directly, lacking the exploration of a good representation of the raw sEMG signals [17].

### 3. GAF-CNN

We propose a GAF-CNN method that formulates the sEMG-based gesture recognition as the 2D data representation and CNN-based classification problems. The pipeline includes two stages: sEMG-GAF image encoding and CNN-based classification. In the first stage, the multi-channel sEMG values at instant sampling moments are encoded in image form based on GAF features. Hence, the value of each point in the sEMG-GAF image indicates the correlation between signals from two sEMG sensors. In the classification stage, a deep CNN model is adopted to classify the sEMG-GAF images. Figure 1 shows the pipeline of the proposed GAF-CNN method.



**Figure 1.** Pipeline of the proposed two-stage architecture for myoelectric pattern recognition.

#### 3.1. Transformation from sEMG Signals to sEMG-GAF Image

After the sEMG signals have been divided into small segments with an overlapped time window, building suitable network inputs is essential and is highly dependent on the preferred DL architecture. The CNN-based method traditionally requires an  $M \times N$  gray image or an  $M \times N \times 3$  RGB image as input. To this end, the sEMG signals must be reshaped as 2D or 3D matrix form. Intuitively, reshaping the sEMG signals as an image could be a solution, where a pixel of the image can be regarded as one instantaneous value from the sEMG sensor electrode [31]. This works when high-density sEMG signals are employed, since the construction of an sEMG image requires an adequate number of instantaneous values acquired by high-density sEMG sensors. On the other hand, sparse multichannel sEMG signals can be reshaped as an image form with the dimensionality of  $N \times L$ , where  $N$  is the number of electrodes and  $L$  is the duration of the sliding time window [34]. Those mentioned works provide a method to generate sEMG images as input for the CNN-based method. However, the construction of an appropriate sEMG-based image that can be suitable for both high-density sEMG signals and sparse sEMG signals remains an unsolved problem.

We present an effective approach to encode sEMG signals as images via the GAF method [20]. Unlike the use of the GAF method to convert time-series data to images, we use it for imaging instantaneous sEMG signals on sensor channels, i.e., at each sampling moment. The sEMG signals of  $C$  sensor channels are converted to an image with the size

of  $C \times C$ , which we call an sEMG-GAF image. In the sEMG-GAF encoding stage, one segment of sEMG signals is given by

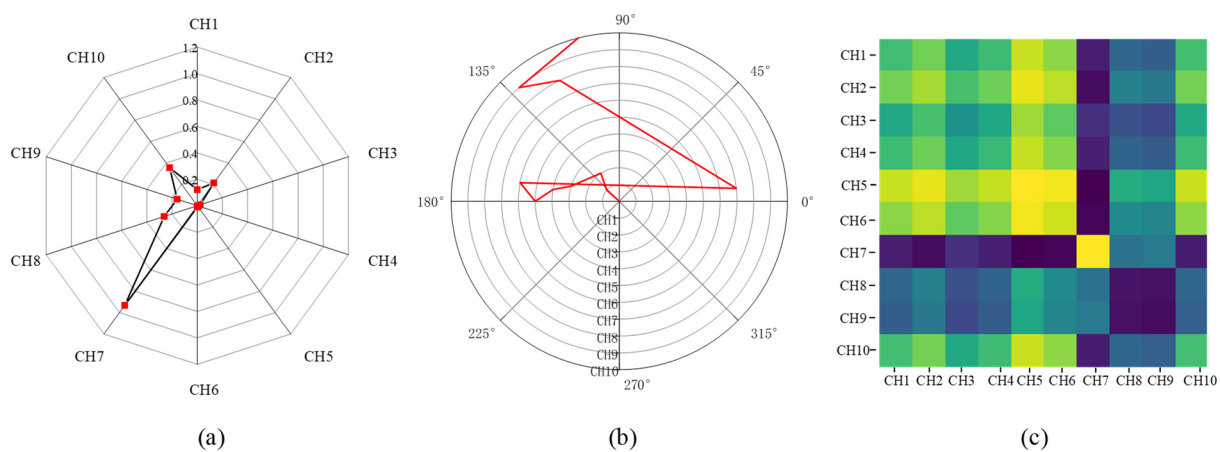
$$X = \begin{bmatrix} x_1^1 & x_2^1 & \dots & x_L^1 \\ x_1^2 & x_2^2 & \dots & x_L^2 \\ \vdots & \vdots & \ddots & \vdots \\ x_1^C & x_2^C & \dots & x_L^C \end{bmatrix}, \quad (1)$$

where  $C$  is the number of sEMG sensor channels and  $L$  is the length of sEMG segments.

We rescaled  $X$  into the interval  $[-1, 1]$ :

$$\tilde{x}_i^c = \frac{(x_i^c - \max(X)) + (x_i^c - \min(X))}{\max(X) - \min(X)}, \quad (2)$$

where  $c$  is the channel index and  $i$  is the sampling moment. Figure 2a demonstrates the radar map of the normalized 10-channel sEMG values at the selected moment.



**Figure 2.** The proposed encoding procedure of GAF conversion on a selected moment: (a) normalized 10-channel sEMG values in radar map; (b) represented angular values in the polar coordinate system; and (c) demonstration of sEMG-GAF image after signal conversion. The different colors in (c) indicate relationships between the instant values in different sensors.

In this way, the rescaled sEMG signals  $X$  can be represented in polar coordinates by encoding  $\tilde{x}_i^c$  as the angular cosine:

$$\begin{cases} \theta_i^c = \arccos(\tilde{x}_i^c), & -1 \leq \tilde{x}_i^c \leq 1, \tilde{x}_i^c \in \tilde{X}, \\ r = \frac{c}{C}, & 1 \leq c \leq C \end{cases}, \quad (3)$$

where  $r$  is the radius that regularizes the span of the polar coordinate system. As  $c$  increases,  $\theta_i^c$ , which can be considered a novel representation of multichannel sEMG signals at moment  $i$ , warps among different angular points on the spanning circles in the polar coordinate. Figure 2b demonstrates the represented angular values in the polar coordinate system at a selected moment. Each angular point indicates the represented angular value of one sEMG sensor. The increment of the angular value can be used to present the relationship between two instant sEMG values. It is clear that the angular increment is bijective, as  $\cos(\phi_i^c)$  is monotonic when  $\phi_i^c \subseteq [0, \pi]$ , which means that the proposed mapping produces only one result in the polar coordinate system and has a unique inverse mapping at a given channel and moment. Additionally, unlike Cartesian coordinates, polar coordinates preserve absolute temporal relationships. Hence, polar coordinates are a good choice for the 2D representation of sEMG values.



After transforming the rescaled sEMG signal to polar coordinates, the angular increment can be calculated by the trigonometric sum between each point to identify the channel correlation of different values in sensor channels. Therefore, we define the inner product as  $\cos(\theta_i^p + \theta_j^q)$ , where  $p$  and  $q$  are the channel indices. In this way, the sEMG-GAF image at moment  $i$  can be defined using a Gram matrix:

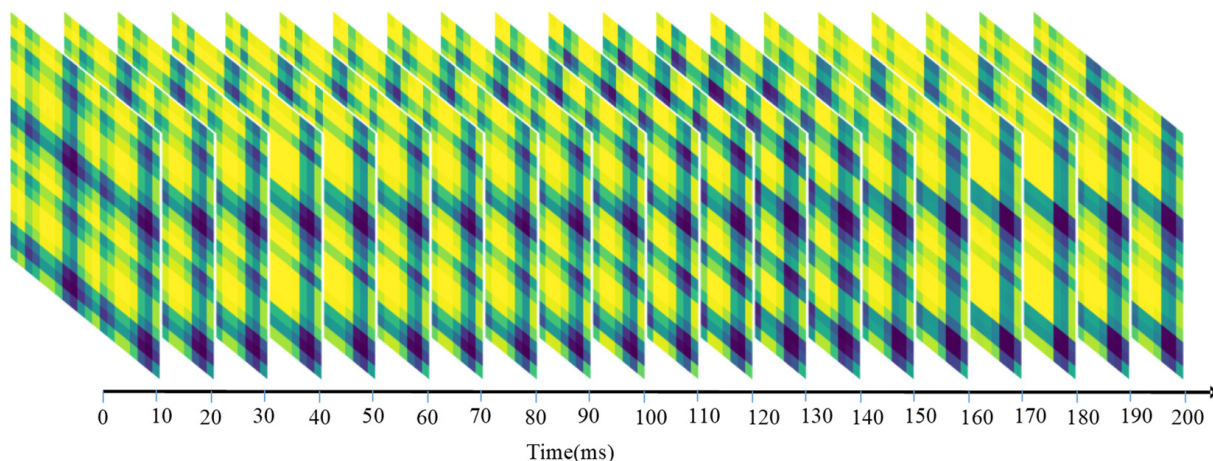
$$\text{sEMG - GAF}_i = \begin{bmatrix} \cos(\theta_i^1 + \theta_i^1) & \dots & \cos(\theta_i^C + \theta_i^1) \\ \cos(\theta_i^1 + \theta_i^2) & \dots & \cos(\theta_i^C + \theta_i^2) \\ \vdots & \ddots & \vdots \\ \cos(\theta_i^1 + \theta_i^C) & \dots & \cos(\theta_i^C + \theta_i^C) \end{bmatrix}. \tag{4}$$

Figure 2c demonstrates a sEMG-GAF image and different colors in the image indicate relationships between the instant values in different sensors.

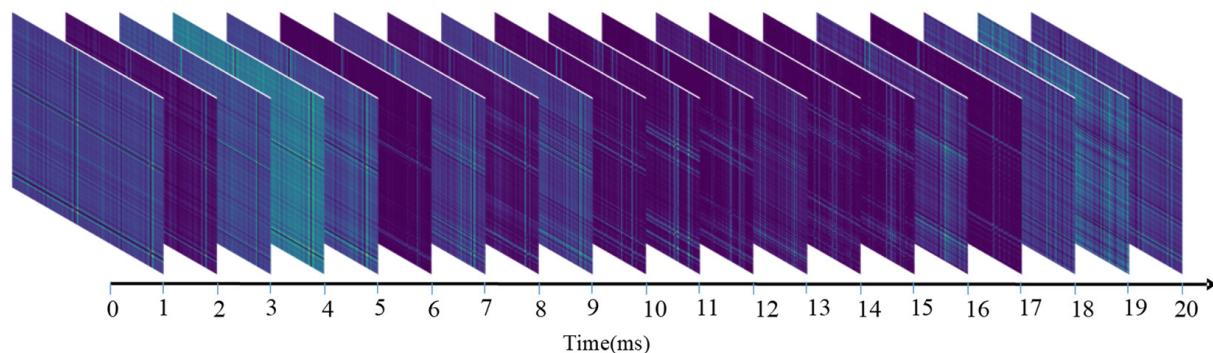
To fulfill the data preprocessing requirement at the training and classification stages, the sEMG signals are formed into segments by a sliding time window. One segment of sEMG signals  $X$  with duration time  $L$  can be represented as the sequence

$$\text{sEMG - GAFs} = \{\text{sEMG - GAF}_1, \dots, \text{sEMG - GAF}_i, \dots, \text{GAF}_L\}. \tag{5}$$

Figures 3 and 4 demonstrate sequential sEMG-GAF images from the NinaPro DB1 and CagpMyo DB-a datasets, respectively. Slight changes between adjacent frames of the sEMG image represent changes in the sEMG signal over time.



**Figure 3.** Sequential sEMG-GAF images obtained from the NinaPro DB1 dataset; 10-channel sEMG signals were sampled at 100 frames/s.

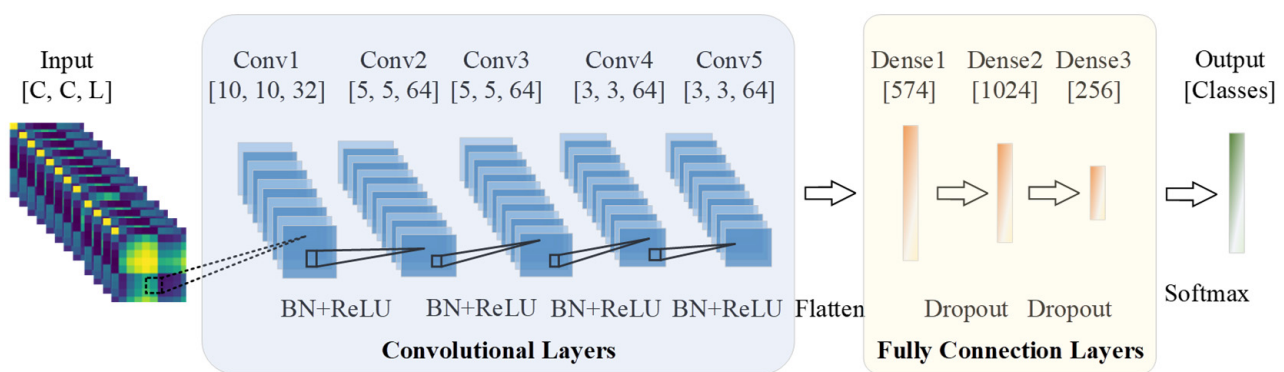


**Figure 4.** Sequential sEMG-GAF images obtained from the CagpMyo DB-a dataset; 144-channel sEMG signals were sampled at 1000 frames/s.

### 3.2. Construction of ConvNet Architecture

DCNN can effectively extract crucial spatial features from the data and has shown exemplary performance in several computer vision and pattern recognition competitions [35]. It is also a competitive classifier in many applications. A CNN-based architecture is essential to achieve good classification accuracy. Krizhevsky et al. [36] proposed AlexNet, which can enhance the learning capacity of a CNN by making the network deeper through various parameter learning strategies. The AlexNet architecture includes five convolutional layers and three fully connected layers, which make a CNN applicable to classification tasks with diverse categories of images [37].

To design a classification network for the sEMG-GAF image, an AlexNet architecture is introduced to the proposed framework to construct an appropriate DL model for extracting crucial features from sEMG-GAF images. As shown in Figure 5, the proposed network consists of an input layer, five convolutional layers, and three fully connected layers. The size of the input layer should be adjusted according to the size of the input data. In our work, the input data size is  $C \times C \times L$ , where  $C$  is the number of sEMG sensor channels and  $L$  is the length of the sliding time window. The kernel size and stride applied on the input layer are resized to adapt to the following convolutional layers. For example, the size of the input is  $10 \times 10 \times 20$  under a sliding time window with a length of 200 ms from a dataset with 10 channels and a sampling rate of 100 Hz. Therefore, the convolutional kernel size will be set to  $3 \times 3$  with a zero-padding size of  $1 \times 1$ , and the stride should be set to 1. In addition, a dropout strategy with a probability of 0.5 on the two fully connected layers is adopted to prevent overfitting. With AlexNet, we employ batch normalization instead of using max-pooling layers after the convolution operation on each convolutional layer to optimize the network. Batch normalization can enhance the model's generalization ability to reduce the dependence on parameter initialization, accelerate network training, and reduce the computational cost.



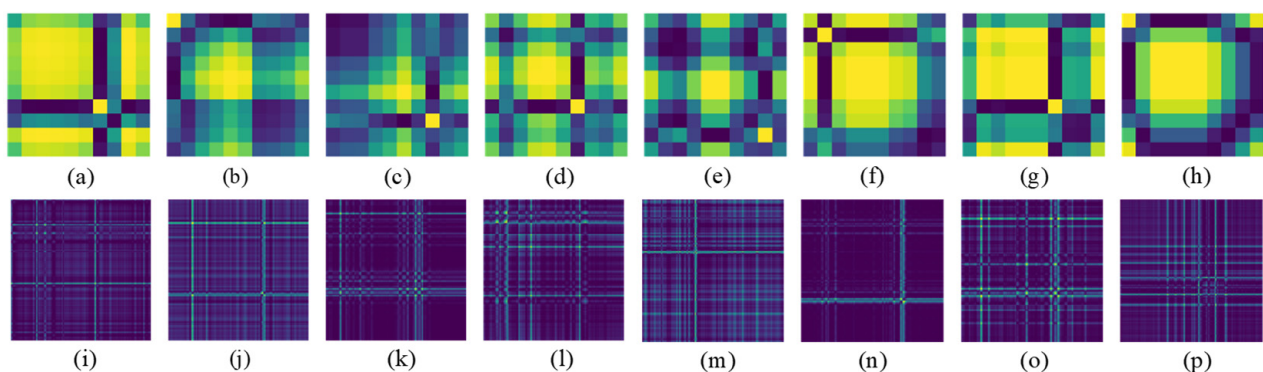
**Figure 5.** The architecture of the proposed five-layer convolutional neural network consists of an input layer, five convolutional layers, and three fully connected layers. The network's input is a clip of a sequence of sEMG images with size  $C \times C \times L$ , where  $C$  is the number of sEMG sensor channels, and  $L$  is the duration of the sliding time window. Outputs of network are class labels of hand gestures. The dotted line between the input layer and the first convolutional layer denotes the kernel size. Therefore, the stride applied to the input layer should differ according to the size of an input image.

### 3.3. Rationale of the GAF-CNN Method

We discuss the rationale of the GAF-CNN method. It is well known that CNNs can not only preserve the neighborhood relations and spatial locality of an input image in latent higher-level feature representations but also can scale well to realistic-sized high-dimensional images in terms of computational complexity, since the number of free parameters that describe their shared weights does not depend on the input dimensionality [20]. Although some works [18,19,31] have shown the effectiveness of CNN models

on myoelectric pattern recognition, those methods directly rearrange the sEMG signals in an image form. However, they are not able to fully leverage the advantages of CNNs. Consequently, how to construct an excellent 2D representation of multichannel sEMG signals for CNN learning is the essential point.

We propose a GAF-CNN method to formulate sEMG-based gesture recognition as a multichannel sEMG image CNN-based classification problem. The GAF method is introduced for the 2D representation of multichannel sEMG signals. Unlike conventional GAF-based applications that encode time-series signals as images, we encode the multichannel sEMG signals on the channel dimension for two reasons. On one hand, the raw sEMG signals in each channel are nonstationary, nonlinear, stochastic, and unpredictable, making the correlations of time stamps during the process of raw sEMG signals not suitable for CNN-based gesture recognition. Our preliminary experimental results show that the performance of the CNN model is extremely poor and even cannot converge in most cases while applying the GAF method to convert sEMG signals on time series. On the other hand, the activation levels of different muscles will maintain relatively stable values while keeping the same hand gesture. Patterns of multichannel sEMG signals between instant values in different sensor channels should be easier to extract compared to the case in time series. Encoding multichannel signals as images can make the feature space more expressive by representing relationships between the values in different sensor channels rather than treating each channel independently. Figure 6 shows sEMG-GAF images obtained from different gestures at a specified moment. Different image texture patterns in sEMG images can be extracted easily. Therefore, encoding the instantaneous values as the image could be a good 2D representation of multichannel sEMG signals for CNN-based classification.



**Figure 6.** Demonstration of sEMG-GAF images of different gestures from NinaPro DB1 dataset (a–h) and CapgMyo DB-a dataset (i–p) on the specified moment. The gestures include thumb up as shown in (a) and (i), extension of index and middle, flexion of the others as shown in (b) and (j), flexion of ring and little finger, extension of the others as shown in (c) and (k), thumb opposing base of little finger as shown in (d) and (l), abduction of all fingers as shown in (e) and (m), fingers flexed together in a fist as shown in (f) and (n), pointing index as shown in (g) and (o), and adduction of extended fingers as shown in (h) and (p). The different colors in sEMG-GAF images indicate relationships between the instant values in different sensors. The color difference between the images of different gestures indicates the degree of differentiation of gestures.

In addition, we consider sEMG-GAF image construction from a mathematical rationale. As described in Section 3.1, we use the Gram matrix to construct the sEMG-GAF image. This is a helpful tool in linear algebra and geometry, which has been frequently used to compute the linear dependence of a set of vectors [20]. The mathematics of this method is intrinsically linked to the inner product and the corresponding Gram matrix. The inner product is an operation between two vectors to measure their similarity. In this work, the Gram matrix is an sEMG-GAF image. Since the channel index increases as the position moves from top-left to bottom-right, the correlation patterns between global channel information are encoded



in the geometry of the matrix. As a result, patterns between gestures and the related muscles in sEMG-GAF images can be well extracted and represented by CNN models. It is noteworthy that the geometric characters of the sEMG-GAF image meet the classification capability of a CNN, which enables it to easily extract the patterns in the image.

To summarize, GAF-CNN employs the GAF method to exploit a 2D representation of multichannel sEMG signals for CNN-based classification. Compared with existing work [18,19,31] that directly rearranges the values of sEMG signals for classification, GAF representation has several advantages. First, it provides a 2D representation and makes the multichannel sEMG signals suitable as input for the CNN model. Second, it encodes the global channel correlations of sensors into local relationships and preserves the temporal dependency of the input image, so as to facilitate extraction by the CNN of gesture patterns in sEMG-GAF images. Finally, it is worth noting that instant values in the sensor channels are highly correlated with the collaboration of muscles, which are used for gesture control.

## 4. Experiments

### 4.1. Dataset Description

We discuss the four publicly available benchmark datasets used in the experiment: NinaPro DB1 [2], NinaPro DB2 [2], CapgMyo DB-a [38], and CSL-HDEMG [39]. For all datasets, the sEMG sensors were placed on various muscle locations on the upper limbs. We describe each dataset as follows:

1. NinaPro DB1 [2] consists of sEMG signals extracted from 27 intact subjects, captured with 10 sEMG electrodes (8 placed around the forearm, and 1 each on the main activity spots of the large flexor and extensor muscles of the forearm) at a 100 Hz sampling rate. The dataset has 52 gestures, each executed in 10 trials for 3–4 s.
2. NinaPro DB2 [2] consists of sEMG signals extracted from 40 intact subjects while performing finger gestures and grasping objects. The signals were sampled at a rate of 2000 Hz. During acquisition, subjects were asked to repeat movements with the right hand, each repetition lasting 5 s, followed by 3 s of rest. The protocol included six repetitions of 49 different movements (plus rest).
3. CapgMyo Db-a [38] is a frequently used high-density sEMG dataset, which was recorded with an  $8 \times 16$  electrode grid wrapped around the right forearm, including eight isometric and isotonic finger gestures acquired from 18 healthy subjects. The 128-channel signals were band-pass filtered at 20–380 Hz with a sampling rate of 1000 Hz. Each subject performed 10 trials of each gesture, holding each for 3–5 s.
4. CSL-HDEMG [39] consists of high-density sEMG signals recorded at a sampling rate of 2048 Hz using an electrode array arranged in an  $8 \times 24$  grid. The dataset includes 27 finger gestures performed by five subjects in five recording sessions, recording each gesture 10 times in each session.

### 4.2. Data sEMG-GAF Transformation

As discussed in Section 3.1, instantaneous sEMG signals with  $C$  sensor channels on each sampling moment were converted to an sEMG-GAF image with a size of  $C \times C$  before training and prediction. Table 1 shows the sizes of the converted sEMG-GAF image for each dataset. Taking one instance, sEMG signals in NinaPro DB1 were captured with 10-channel sEMG electrodes. Therefore, each converted sEMG-GAF image had a size of  $10 \times 10$ , and the input size for the proposed model was  $10 \times 10 \times 1$ . For sequential sEMG-GAF images, the interval of each frame was equal to the time segmentation of the sEMG signal series.

**Table 1.** Size specification of the converted sEMG-GAF image for each dataset.

Datasets	Sensor Channels	sEMG-GAF Image Sizes	Frame Interval
NinaPro DB1	10	10 × 10	10 ms
NinaPro DB2	12	12 × 12	0.5 ms
CapgMyo Db-a	128	128 × 128	1 ms
CSL-HDEMG	192	192 × 192	0.49 ms

#### 4.3. Experimental Setup

For each subject, we evaluated inter-session classification accuracies, where the proposed model was trained on 70% of the samples and evaluated on the remaining 30%. For the training samples, we used a data augmentation strategy that replaces 10% of each sample with data from other samples with a probability of 50%. In this way, random disturbance could be added to the training data to prevent overfitting. Moreover, since the network training requires a large-scale labeled dataset, we executed the data augmentation strategy three times.

To evaluate the performance of the GAF-CNN method, we followed the accuracy calculation rule applied in [29] and [30], i.e., using the instantaneous sEMG images for training and classification first, and then employing a majority voting strategy to calculate classification accuracies over the specified length time window, e.g., 150 ms and 300 ms. Therefore, the input sizes of NinaPro DB1, NinaPro DB2, CapgMyo Db-a, and CSL-HDEMG were  $10 \times 10 \times 1$ ,  $12 \times 12 \times 1$ ,  $128 \times 128 \times 1$ , and  $192 \times 192 \times 1$ , respectively.

Since the GAF-CNN method can handle the input size of sEMG signals with different sensor channel amounts and various sliding windows as presented in Section 3.2, we also conducted experiments both using a sliding window strategy and a voting strategy for classification accuracy calculation on datasets captured with low sampling rates, i.e., NinaPro DB1. For NinaPro DB1, we used an input size of  $10 \times 10 \times L$ , where the amount of the sensor channel is 10 and the  $L$  is the length of the sliding window of 10 ms, 100 ms, and 200 ms, corresponding to a single frame, 10 frames, and 20 frames of sEMG-GAF images, respectively.

Based on the experimental results, the classification accuracy was calculated for each dataset as

$$\text{Classification Accuracy} = \frac{\text{Numbers of correct classifications}}{\text{Total number of classifications}} \times 100. \quad (6)$$

The proposed GAF-CNN method was implemented based on the Keras 1.18.5 platform, written in Python and capable of running on top of TensorFlow. Experiments were conducted on a computer equipped with one Nvidia GPU GTX 3090 with 64 GB RAM, and an Intel CPU i9 9900 k.

## 5. Results and Discussion

### 5.1. Role of Input Sizes

The proposed GAF-CNN method was analyzed with respect to the performance using sliding window and voting strategies.

For the low-density dataset, we tried both sliding windows and voting strategies. Figure 7 shows the recognition accuracy comparison between using sliding windows and voting strategies in NinaPro DB1. When using sliding windows of 10 ms, 50 ms, 100 ms, 150 ms, and 200 ms, the recognition accuracies of 71.2%, 83.9%, 84.0%, 84.6%, and 85.5% were achieved. Since the sampling frequency is 100 Hz, the recognition accuracy of 71.2% on a sliding window of 10 ms can be considered as instantaneous recognition accuracy. Based on the instantaneous recognition accuracy, we employed a majority voting strategy to re-calculate the accuracies over time windows of 50 ms, 100 ms, 150 ms, and 200 ms, by which we achieved accuracies of 75.3%, 79.6%, 80.7%, and 82.2%, respectively. Figure 8 further compares the accuracy of each subject when using a single frame, a majority voting

time window of 200 ms, and a sliding segmented time window of 200 ms in NinaPro DB1. The average classification accuracy of 71.2% was achieved when using every single frame as one sample. Based on this result, we applied a majority voting strategy over 200 ms to calculate the accuracy for each subject, and an average accuracy of 80.1% was obtained. As a comparison, the classification accuracy achieved by the sliding segmentation time window of 200 ms was 85.5%, which is significantly higher than the case calculated by a voting time window of 200 ms using a single frame. As a result, we believe that the majority voting strategy based on a single frame could lose information on the time domain of the myoelectric pattern. Therefore, we only report the recognition accuracies based on the sliding window strategy for NinaPro DB1.

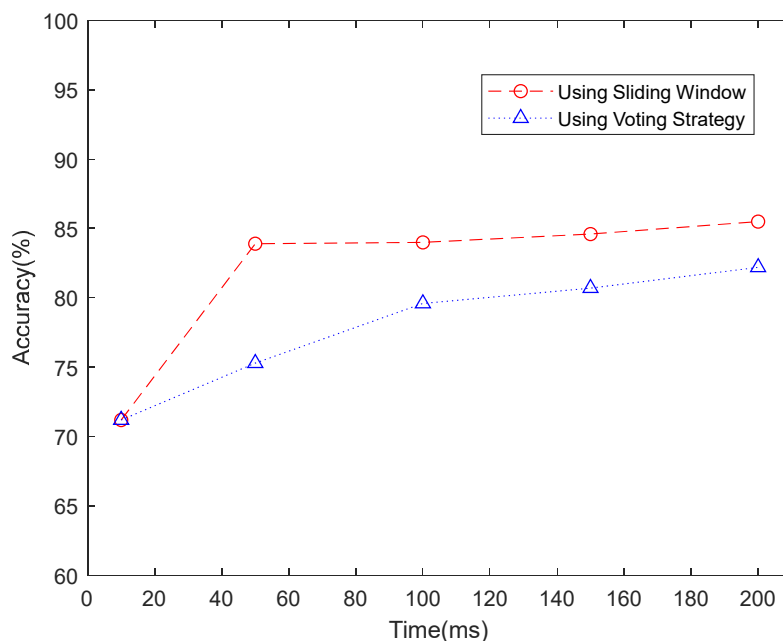


Figure 7. Recognition accuracy comparison between using sliding window and voting strategy in NinaPro DB1.

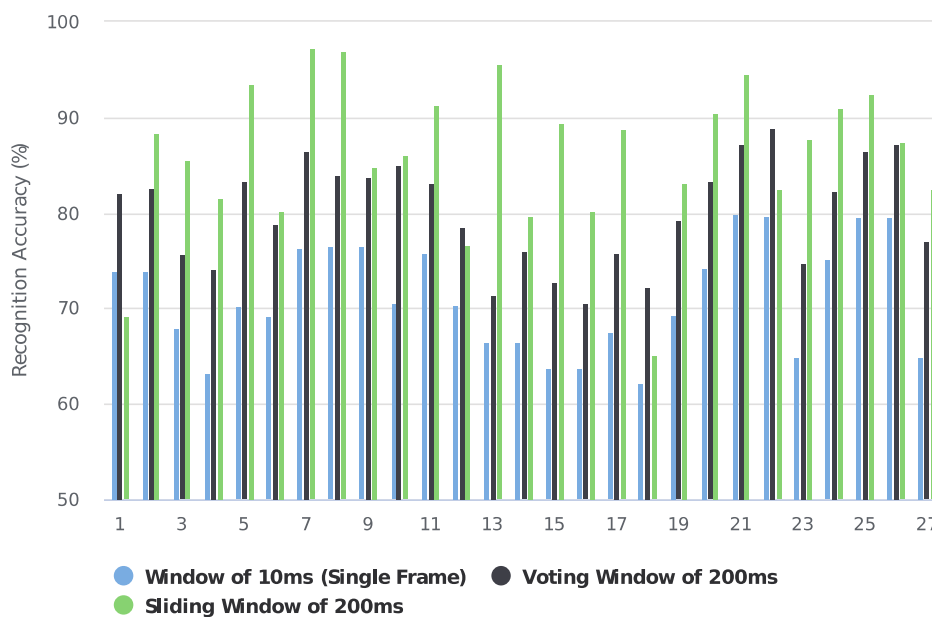
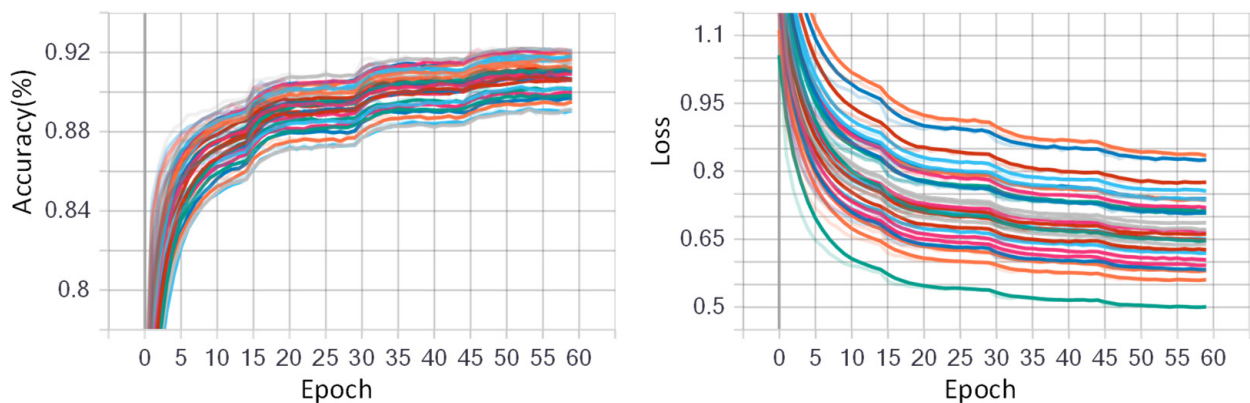


Figure 8. Recognition performances of 27 subjects when using single frame of 10 ms, voting time window of 200 ms, and sliding time window of 200 ms in NinaPro DB1.

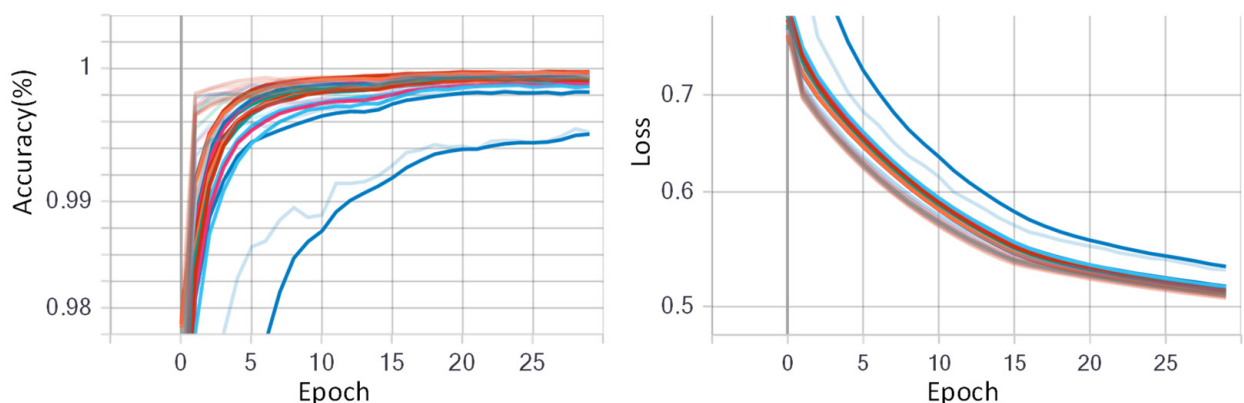
For the high-density datasets, we employed a voting strategy for recognition accuracy calculation as in the existing works [19,31], since there are too many frames for the input of a CNN model even in a small sliding window. For instance, the window length of 100 ms contains 100 frames of sEMG-GAF images in the CapgMyo DB-a dataset, which could be an unbearable burden for the input of the CNN model.

### 5.2. Training Process

The CNN model was initialized with randomized weights and trained using stochastic gradient descent (SGD) with a data batch size of 512 for all experiments. For fine-tuning the CNN model, the initial learning rate was assigned as 0.05 with a decay rate of 0.5. The initial learning rate was chosen large enough to accelerate the learning process in the beginning. The epoch number was set to 60 for dataset NinaPro DB1, and 30 for other datasets. Figures 9 and 10 show the training progress of the fine-tuned CNN model for the low-density dataset NinaPro DB1 and the high-density dataset CapgMyo DB-a, respectively. The left shows the deviation in the accuracies on different training epochs, and the right shows the loss deviations in different training epochs. It is worth mentioning that 70% data were used for fine-tuning the AlexNet model, and the remaining 30% was used for testing the fine-tuned CNN model.



**Figure 9.** Training progress of fine-tuned GAF-CNN model for low-density dataset NinaPro DB1. The epoch number was set to 60 and different colors denote the 27 different subjects. The different color lines indicate different subjects.



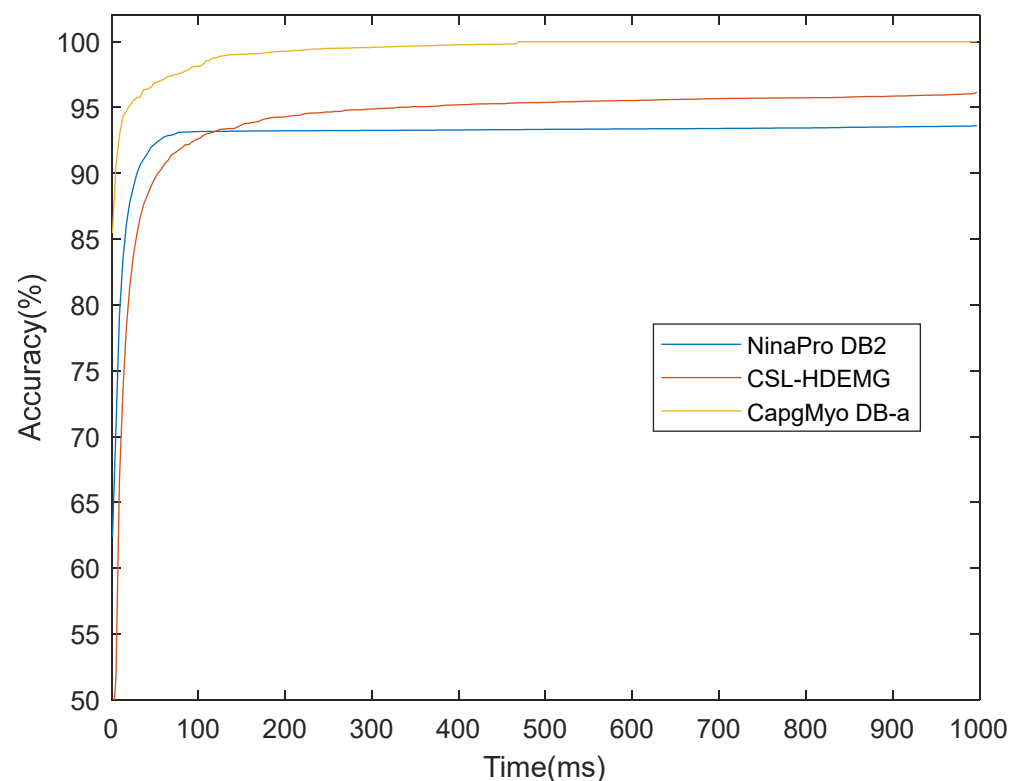
**Figure 10.** Training progress of fine-tuned GAF-CNN model for high-density dataset CapgMyo DB-a. The epoch number was set to 30 and different colors denote the 18 different subjects. The different color lines indicate different subjects.



### 5.3. Evaluation on Recognition Accuracy

We evaluated the proposed GAF-CNN method on four publicly available datasets. The classification accuracies on a single frame of an instantaneous sEMG image were 71.0% on NinaPro DB1, 77.8% on NinaPro DB2, 45.0% on CSL-HDEMG, and 90.2% on CapgMyo DB-a, respectively. We found that data augmentation could significantly improve classification accuracies, achieving 71.2% on NinaPro DB1, 79.8% on NinaPro DB2, 46.3% on CSL-HDEMG, and 93.3% on CapgMyo DB-a, respectively, as well as an average improvement of 1.7%.

Based on the instantaneous classification results, the recognition accuracies could be improved with a majority voting strategy. Figure 11 shows the recognition accuracies over different voting windows on high-density datasets of NinaPro DB2, CSL-HDEMG, and CapgMyo DB-a. It is clear that the recognition accuracies grew when larger voting windows were applied. For example, a classification accuracy of 93.3% was achieved on an instantaneous sEMG image on CapgMyo DB-a, and 99.5%, 99.7%, and 99.8% by majority voting over 40 ms, 150 ms, and 300 ms, respectively.



**Figure 11.** The recognition accuracies over different voting windows on high-density datasets NinaPro DB2, CSL-HDEMG, and CapgMyo DB-a. The recognition accuracies grow up when larger voting windows are applied.

### 5.4. Performance Comparison

To evaluate the performance of the proposed GAF-CNN method, we compared the experimental results with the state of the art as reported by previous work incorporating CNN models, which include a benchmark classifier with a handcrafted feature set, single-stream CNN by Atzori et al. [2], multi-stream CNN by Wei et al. [31], and 3D CNN by Chen et al. [32]. Single-stream CNN by Atzori et al. [2] and multi-stream CNN by Wei et al. [31] are popular 2D feature-based CNN methods, which employ the sEMG data as the direct input for the 2D CNN model. Table 2 presents the classification accuracies obtained by training and subsequently using the majority voting strategy to calculate the accuracies of the specified lengths of time windows on different sources of sEMG data.

**Table 2.** Classification accuracies achieved by GAF-CNN when evaluated on publicly available sEMG datasets, along with the state of the art as reported by previous work incorporating CNN models. The first row indicates the in-use datasets, and the second row indicates the experimental conditions for sliding window lengths. The bold font indicates the best result under the same conditions in comparisons.

	NinaPro DB1		NinaPro DB2		CSL-HDEMG		CapgMyo DB-a	
	100 ms	200 ms	100 ms	200 ms	150 ms	300 ms	150 ms	300 ms
Benchmark classifier with handcrafted feature set	-	75.3% from [2]	-	75.2% from [2]	-	-	99.0% from [40]	-
Single-stream CNN by Atzori et al. [2]	-	66.6%	-	60.3%	-	-	99.5%	-
Multi-stream CNN by Wei et al. [31]	83.4%	85.0%	-	-	93.6%	95.4%	<b>99.7%</b>	<b>99.8%</b>
3D CNN by Chen et al. [32]	-	-	-	-	90.7%	-	98.6%	-
Proposed GAF-CNN	<b>83.9%</b>	<b>85.5%</b>	<b>79.0%</b>	<b>80.2%</b>	<b>94.8%</b>	<b>95.9%</b>	<b>99.7%</b>	<b>99.8%</b>

Evaluations were also performed on the sparse sEMG datasets. The proposed GAF-CNN achieved classification accuracies of 83.9% over the window size of 100 ms and 85.5% over the window size of 200 ms, respectively, on NinaPro DB1. For NinaPro DB2, GAF-CNN achieved the classification accuracies of 79.0% and 80.2%, respectively. The experimental results show that the GAF-CNN method outperformed the CNN-based methods, including single-stream CNN [2] and multi-stream CNN by Wei et al. [31].

Regarding the evaluations performed on the high-density sEMG datasets, GAF-CNN showed comparable performances to all compared methods. Classification accuracies of 95.0% and 96.1% were achieved on CSL-HDEMG over window sizes of 150 ms and 300 ms, respectively. For CapgMyo DB-a, corresponding classification accuracies of 99.7% and 99.8% were achieved.

We also compared the proposed method with a 3D CNN-based method [32] on CSL-HDEMG and CapgMyo DB-a. The 3D CNN-based method [28] exploited the raw sEMG data with a cube size of  $l \times 8 \times 16$  as the model input, where  $l$  was the sliding window length. Therefore, it could not identify the instantaneous sEMG signals, and it used significantly more computation when the sliding window became longer, while GAF-CNN achieved higher classification accuracy on the reported voting windows. For example, while using a sliding window length of 150 ms, GAF-CNN achieved a 4.3% (95.0% vs. 90.7%) higher accuracy on CSL-HDEMG, 1.1% (98.6% vs. 99.7%) higher on CSL-HDEMG, and 1.1% (98.6% vs. 99.7%) higher on CapgMyo DB-a, respectively, than those reported by Chen et al. [32].

## 6. Conclusions

We present a two-stage CNN-based architecture to effectively extract patterns from complex sEMG signals to improve the accuracy of myoelectric pattern recognition. Multichannel sEMG signals are encoded to sEMG-GAF image series by the GAF method so as to map the relationship between hand movements and active forearm muscle regions. Thus, the correlation patterns between individual muscles of specified hand movements could be thoroughly learned by the defined CNN model. To evaluate the effectiveness of the proposed method, we conducted extensive experiments on sparse and high-density sEMG datasets under sliding time windows and majority voting time windows, which showed that GAF-CNN achieves great performance on classification accuracy. Evaluation results on four publicly available sEMG datasets show that GAF-CNN outperforms the state-of-the-art CNN-based myoelectric pattern recognition methods.

In the future, we will improve the GAF-CNN-based myoelectric pattern recognition method in two ways: (1) A concise and practical deep CNN model for training and

classification can be employed. We believe the classification accuracy can be improved by advanced DL techniques, such as deep transfer learning. (2) A CNN-based model can be employed to focus on the processing of sEMG data, which are temporal signals. Novel temporal models, such as R-CNN and LSTM [41], can be employed to exploit more time-domain features in myoelectric pattern recognition.

**Author Contributions:** Conceptualization, J.F. and J.W.; methodology, J.F. and J.W.; writing—original draft preparation, J.F.; writing—review and editing, J.F., J.W. and Z.L.; funding acquisition, J.W. All authors have read and agreed to the published version of the manuscript.

**Funding:** This work was supported in part by the Guangdong Basic and Applied Basic Research Foundation 2020A1515110602 and 2021A1515011318, and in part by the Shenzhen Municipal Science and Technology Innovation Council under the grant JCYJ20190808113411274.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Publicly available datasets were analyzed in this study. This data can be found here: <http://ninaweb.hevs.ch/>, [http://zju-capg.org/research\\_en\\_electro\\_capgmyo.html](http://zju-capg.org/research_en_electro_capgmyo.html), and <https://www.uni-bremen.de/csl/forschung/gestenerkennung> (all accessed on 10 December 2022).

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. De Luca, C.J. The use of surface electromyography in biomechanics. *J. Appl. Biomech.* **1997**, *13*, 135–163. [CrossRef]
2. Atzori, M.; Gijsberts, A.; Castellini, C.; Caputo, B.; Hager, A.G.; Elsig, S.; Giatsidis, G.; Bassetto, F.; Müller, H. Electromyography data for non-invasive naturally controlled robotic hand prostheses. *Sci. Data* **2014**, *1*, 140053. [CrossRef] [PubMed]
3. Li, C.; Li, G.; Jiang, G.; Chen, D.; Liu, H. Surface EMG data aggregation processing for intelligent prosthetic action recognition. *Neural Comput. Appl.* **2020**, *32*, 16795–16806. [CrossRef]
4. Disselhorst-Klug, C.; Schmitz-Rode, T.; Rau, G. Surface electromyography and muscle force: Limits in sEMG-force relationship and new approaches for applications. *Clin. Biomech.* **2009**, *24*, 225–235. [CrossRef] [PubMed]
5. Huang, D.; Zhang, X.; Saponas, T.S.; Fogarty, J.; Gollakota, S. Leveraging dual-observable input for fine-grained thumb interaction using forearm EMG. In Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology, Bend, OR, USA, 8–11 November 2015; pp. 523–528.
6. Kumar, P.; Gauba, H.; Roy, P.P.; Dogra, D.P. Coupled HMM-based multi-sensor data fusion for sign language recognition. *Pattern Recognit. Lett.* **2017**, *86*, 1–8. [CrossRef]
7. Saponas, T.S.; Tan, D.S.; Morris, D.; Balakrishnan, R. Demonstrating the feasibility of using forearm electromyography for muscle-computer interfaces. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, Florence, Italy, 5–10 April 2008; pp. 515–524.
8. Rautaray, S.S.; Agrawal, A. Vision based hand gesture recognition for human computer interaction: A survey. *Artif. Intell. Rev.* **2015**, *43*, 1–54. [CrossRef]
9. Nazmi, N.; Abdul Rahman, M.A.; Yamamoto, S.I.; Ahmad, S.A.; Zamzuri, H.; Mazlan, S.A. A review of classification techniques of EMG signals during isotonic and isometric contractions. *Sensors* **2016**, *16*, 1304. [CrossRef]
10. Lecun, Y.; Bengio, Y.; Hinton, G. Deep learning. *Nature* **2015**, *521*, 436–444. [CrossRef]
11. Li, G.; Li, J.; Ju, Z.; Sun, Y.; Kong, J. A novel feature extraction method for machine learning based on surface electromyography from healthy brain. *Neural Comput. Appl.* **2019**, *31*, 9013–9022. [CrossRef]
12. Zhang, Z.; Yang, K.; Qian, J.; Zhang, L. Real-time surface EMG pattern recognition for hand gestures based on an artificial neural network. *Sensors* **2019**, *19*, 3170. [CrossRef]
13. Waris, A.; Niazi, I.K.; Jamil, M.; Englehart, K.; Jensen, W.; Kamavuako, E.N. Multiday evaluation of techniques for EMG-based classification of hand motions. *IEEE J. Biomed. Health Inform.* **2018**, *23*, 1526–1534. [CrossRef] [PubMed]
14. Phinyomark, A.; Scheme, E. EMG pattern recognition in the era of big data and deep learning. *Big Data Cogn. Comput.* **2018**, *2*, 21. [CrossRef]
15. Park, K.H.; Lee, S.W. Movement intention decoding based on deep learning for multiuser myoelectric interfaces. In Proceedings of the 2016 4th international winter conference on brain-computer Interface (BCI), Gangwon, Republic of Korea, 22–24 February 2016; pp. 1–2. [CrossRef]
16. Zhang, X.; Yang, Z.; Chen, T.; Chen, D.; Huang, M.C. Cooperative sensing and wearable computing for sequential hand gesture recognition. *IEEE Sens. J.* **2019**, *19*, 5775–5783. [CrossRef]
17. Buongiorno, D.; Cascarano, G.D.; De Feudis, I.; Brunetti, A.; Carnimeo, L.; Dimauro, G.; Bevilacqua, V. Deep learning for processing electromyographic signals: A taxonomy-based survey. *Neurocomputing* **2021**, *452*, 549–565. [CrossRef]

18. Atzori, M.; Cognolato, M.; Müller, H. Deep learning with convolutional neural networks applied to electromyography data: A resource for the classification of movements for prosthetic hands. *Front. Neurobotics* **2016**, *10*, 9. [[CrossRef](#)]
19. Geng, W.; Du, Y.; Jin, W.; Wei, W.; Hu, Y.; Li, J. Gesture recognition by instantaneous surface EMG images. *Sci. Rep.* **2016**, *6*, 1–8. [[CrossRef](#)]
20. Wang, Z.; Oates, T. Imaging time-series to improve classification and imputation. In Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence, Buenos Aires, Argentina, 25–31 June 2015.
21. Mamani, E.F.L.; del Alamo, C.L. GAF-CNN-LSTM for Multivariate Time-Series Images Forecasting. In Proceedings of the LatinX in AI Research at ICML, Long Beach, CA, USA, 10 June 2019.
22. Lyu, C.; Yang, B.; Chang, X.; Tian, J.; Deng, Y.; Jin, J. FBG Tactile Sensing System Based on GAF and CNN. *IEEE Sens. J.* **2022**, *22*, 18841–18849. [[CrossRef](#)]
23. Bai, J.; Yao, J.; Qi, J.; Wang, L. Electromagnetic Modulation Signal Classification Using Dual-Modal Feature Fusion CNN. *Entropy* **2022**, *24*, 700. [[CrossRef](#)]
24. Masci, J.; Meier, U.; Cireşan, D.; Schmidhuber, J. Stacked convolutional auto-encoders for hierarchical feature extraction. In Proceedings of the International Conference on Artificial Neural Networks, Espoo, Finland, 14–17 June 2011; Springer: Berlin/Heidelberg, Germany; pp. 52–59.
25. Oskoei, M.A.; Hu, H. Support vector machine-based classification scheme for myoelectric control applied to upper limb. *IEEE Trans. Biomed. Eng.* **2008**, *55*, 1956–1965. [[CrossRef](#)]
26. Phinyomark, A.; Quaine, F.; Charbonnier, S.; Serviere, C.; Tarpin-Bernard, F.; Laurillau, Y. EMG feature evaluation for improving myoelectric pattern recognition robustness. *Expert Syst. Appl.* **2013**, *40*, 4832–4840. [[CrossRef](#)]
27. Fan, J.; Fan, X.; Tian, F.; Li, Y.; Liu, Z.; Sun, W.; Wang, H. What is that in your hand? recognizing grasped objects via forearm electromyography sensing. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* **2018**, *2*, 1–24. [[CrossRef](#)]
28. Guo, Y.; Liu, Y.; Oerlemans, A.; Lao, S.; Wu, S.; Lew, M.S. Deep learning for visual understanding: A review. *Neurocomputing* **2016**, *187*, 27–48. [[CrossRef](#)]
29. Min, S.; Lee, B.; Yoon, S. Deep learning in bioinformatics. *Brief. Bioinform.* **2017**, *18*, 851–869. [[CrossRef](#)] [[PubMed](#)]
30. Wu, Y.; Zheng, B.; Zhao, Y. Dynamic gesture recognition based on LSTM-CNN. In Proceedings of the 2018 Chinese Automation Congress (CAC), Xi'an, China, 30 November–2 December 2018; pp. 2446–2450.
31. Wei, W.; Wong, Y.; Du, Y.; Hu, Y.; Kankanhalli, M.; Geng, W. A multi-stream convolutional neural network for sEMG-based gesture recognition in muscle-computer interface. *Pattern Recognit. Lett.* **2019**, *119*, 131–138. [[CrossRef](#)]
32. Chen, J.; Bi, S.; Zhang, G.; Cao, G. High-density surface EMG-based gesture recognition using a 3D convolutional neural network. *Sensors* **2020**, *20*, 1201. [[CrossRef](#)]
33. Demir, F.; Bajaj, V.; Ince, M.C.; Taran, S.; Şengür, A. Surface EMG signals and deep transfer learning-based physical action classification. *Neural Comput. Appl.* **2019**, *31*, 8455–8462. [[CrossRef](#)]
34. Asif, A.R.; Waris, A.; Gilani, S.O.; Jamil, M.; Ashraf, H.; Shafique, M.; Niazi, I.K. Performance evaluation of convolutional neural network for hand gesture recognition using EMG. *Sensors* **2020**, *20*, 1642. [[CrossRef](#)]
35. Khan, A.; Sohail, A.; Zahoor, U.; Qureshi, A.S. A survey of the recent architectures of deep convolutional neural networks. *Artif. Intell. Rev.* **2020**, *53*, 5455–5516. [[CrossRef](#)]
36. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. *Adv. Neural Inf. Process. Syst.* **2017**, *60*, 84–89. [[CrossRef](#)]
37. Alom, M.Z.; Taha, T.M.; Yakopcic, C.; Westberg, S.; Sidike, P.; Nasrin, M.S.; Van Esesn, B.; Awwal, A.S.; Asari, V.K. The history began from alexnet: A comprehensive survey on deep learning approaches. *arXiv* **2018**, arXiv:1803.01164.
38. Du, Y.; Jin, W.; Wei, W.; Hu, Y.; Geng, W. Surface EMG-based inter-session gesture recognition enhanced by deep domain adaptation. *Sensors* **2017**, *17*, 458. [[CrossRef](#)] [[PubMed](#)]
39. Amma, C.; Krings, T.; Böer, J.; Schultz, T. Advancing muscle-computer interfaces with high-density electromyography. In Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems, Seoul, Republic of China, 18–23 April 2015; pp. 929–938.
40. Khushaba, R.N.; Al-Timemy, A.H.; Al-Ani, A.; Al-Jumaily, A. A framework of temporal-spatial descriptors-based feature extraction for improved myoelectric pattern recognition. *IEEE Trans. Neural Syst. Rehabil. Eng.* **2017**, *25*, 1821–1831. [[CrossRef](#)] [[PubMed](#)]
41. Hu, Y.; Wong, Y.; Wei, W.; Du, Y.; Kankanhalli, M.; Geng, W. A novel attention-based hybrid CNN-RNN architecture for sEMG-based gesture recognition. *PLoS ONE* **2018**, *13*, e0206049. [[CrossRef](#)] [[PubMed](#)]

**Disclaimer/Publisher’s Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.