applied sciences

MDPI

Article

Korean Sign Language Recognition Using Transformer-Based Deep Neural Network

Jungpil Shin [1,*], Abu Saleh Musa Miah [1], Md. Al Mehedi Hasan [2], Koki Hirooka [1], Kota Suzuki [1], Hyoun-Sup Lee [3] and Si-Woong Jang [4]

[1] School of Computer Science and Engineering, The University of Aizu, Aizuwakamatsu 965-8580, Fukushima, Japan
[2] Department of Computer Science and Engineering, Rajshahi University of Engineering and Technology (RUET), Rajshahi 6204, Bangladesh
[3] Department of Applied Software Engineering, Dongeui University, Busanjin-Gu, Busan 47340, Republic of Korea
[4] Department of Computer Engineering, Dongeui University, Busanjin-Gu, Busan 47340, Republic of Korea
* Correspondence: jpshin@u-aizu.ac.jp

Abstract: Sign language recognition (SLR) is one of the crucial applications of the hand gesture recognition and computer vision research domain. There are many researchers who have been working to develop a hand gesture-based SLR application for English, Turkey, Arabic, and other sign languages. However, few studies have been conducted on Korean sign language classification because few KSL datasets are publicly available. In addition, the existing Korean sign language recognition work still faces challenges in being conducted efficiently because light illumination and background complexity are the major problems in this field. In the last decade, researchers successfully applied a vision-based transformer for recognizing sign language by extracting long-range dependency within the image. Moreover, there is a significant gap between the CNN and transformer in terms of the performance and efficiency of the model. In addition, we have not found a combination of CNN and transformer-based Korean sign language recognition models yet. To overcome the challenges, we proposed a convolution and transformer-based multi-branch network aiming to take advantage of the long-range dependencies computation of the transformer and local feature calculation of the CNN for sign language recognition. We extracted initial features with the grained model and then parallelly extracted features from the transformer and CNN. After concatenating the local and long-range dependencies features, a new classification module was applied for the classification. We evaluated the proposed model with a KSL benchmark dataset and our lab dataset, where our model achieved 89.00% accuracy for 77 label KSL dataset and 98.30% accuracy for the lab dataset. The higher performance proves that the proposed model can achieve a generalized property with considerably less computational cost.

Keywords: korean sign language (KSL); transformer; convolutional neural network (CNN); sign language recognition (SLR); hand gesture recognition (HGR)

check for updates

Citation: Shin, J.; Musa Miah, A.S.; Hasan, M.A.M.; Hirooka, K.; Suzuki, K.; Lee, H.-S.; Jang, S.-W. Korean Sign Language Recognition Using Transformer-Based Deep Neural Network. *Appl. Sci.* **2023**, *13*, 3029. https://doi.org/10.3390/app13053029

Academic Editor: António J. R. Neves

Received: 13 January 2023
Revised: 17 February 2023
Accepted: 23 February 2023
Published: 27 February 2023

1. Introduction

Communication is inevitable for the deaf or hard of hearing (DHH) community to pass their thoughts, requirements and basic needs to others. In total, 450 million people worldwide belong to the DHH community; they use sign language for communication [1]. According to the DHH community, sign language is not easy and is entirely different from other letters, words and sentences. For example, English and American sign language (ASL) are completely different from Bangla sign language (BSL) and Bangla; the same is true for Korean sign language (KSL) and Korean. In addition, sign language differs from country to country and is even based from state to state. Despite being the same language, sign language is different from American sign language. British sign language is not the same because it is not a motion representation like spoken language. The overall situation is unsuitable for the DHH communities' people to learn sign language, and they often face difficulties in communication due to this variation. Because of this difficulty, there are many obstacles for the DHH community to access their daily requirements, such as public information, social engagement, medical healthcare, education, etc. To ease their lives, we have been lowering the language barrier between the DHH and hearing people community.Many researchers have been working to develop automatic sign language recognition systems with advanced computer vision, robotics and natural language processing to ease their life [2–13]. Some researchers worked to direct a translation system from a sign language into a spoken language [14,15]. Sensor-based [16] and vision-based approaches [17] are the most common domains for developing automated sign language systems [18]. In the last decade, CNNs have made an extraordinary contribution to the vision-based sign language recognition approaches to recording one-handed and two-handed statistical or dynamical movement [4,5,19–22]. In the meantime, self-attention-based architectures have shown an excellent capability of capturing long-distance relationships and have become the de facto most popular approach in natural language processing (NLP) [23]. Inspired by NLP successes, multiple works combine CNN-like architectures with self-attention [24,25], some replacing the convolutions entirely [26,27]. Besides the NLP domain, the transformer model has been applied in different research domains for segmentation, detection, and classification by following the success in NLP [28]. In many works, the CNN architecture is completely replaced by the transformer. Recently, many researchers have proposed a transformer-based method to recognize sign language and classification [29]. For the image processing task, a pure transformer (Vit) is used to replace the CNN backbone [30]. The main concept of transformer-based image processing is firstly to split the original image into a discrete non-overlapping patch by following the word token of the NLP. Then the extracted patches are fed into the transformer for extracting the global relation and feature for the image classification. Inspiration by ViT, many methods have been developed based on the same concept, such as IPT and SETR [31–33]. The technique of the ViT is good for the image-based classification because researchers can directly apply the transformer to the sequence of the patches the same as the NLP sequence. However, some basic themes exist between the image-based vision tasks and the NLP sequence. Moreover, due to the fixed size of the patch, it is difficult to extract multi-scale feature maps and low-resolution features. High computational complexity is a well-known property of the transformer as well. To overcome the problem, researchers recently sequentially combined the transformer feature and the convolution layer, focusing on the computer vision-based task instead of the NLP sequence system CNN meet transformer (CMT) [28]. They employed four stages of the transformer and the CNN, where they mainly considered stage-wise design for extracting multi-scale features, reducing the resolution and increasing the dimension flexibility. However, the CMT method plays a crucial role in overcoming the problem of the pure transformer in the image processing task. However, their main concern is they combined CNN with a transformer sequentially and used it four times in four stages, increasing the time and computational complexity exponentially. In addition, few researchers have extracted CNN and transformer-based features in parallel concepts, and no work has been performed for Korean sign language recognition for the deaf and mute communities. To overcome the challenges, we proposed a convolutional layer-based transformer and CNN-based multi-branch model to recognize sign language recognition to overcome existing challenges. In the method, we parallelly extracted CNN and transformer-based features from the sign image and fed the concatenated feature into a classification module for classifying sign language. To overcome existing patches sequence problems [33], we employed a grain architecture module that firstly employed two-three × three convolutions with a stride, then included a three × three convolution with a stride of 1 and an output channel of 32 to reduce the size of input images for extra tracking and better local information extraction. The grain module mainly consists of a convolution and a layer normalization (LN), which is employed before the first stage to reduce the size of the intermediate feature (2× down-sampling of resolution) and project it to a larger dimension (2× enlargement of dimension). The convolutional layer-based transformer can extract short-range and long-range dependencies. Lastly, we employed a classification module with a global average pooling layer, a fully connected (FC) layer and an n-way classification layer with softmax. In addition, we added position embedding, fed into the attention-based deep learning model. The main contribution of the study is given below.

- We present a deep learning adaptable Korean sign language (KSL) video dataset.
- We proposed a multi-branch attention and CNN-based neural network to recognize sign language, which followed two stages to generate feature maps of different stages. The first stage included two parallel branches to generate feature maps of different scales, which are important for the dense prediction task. The second stage is considered a classification module.
- A grain module was employed to reduce the size of the original image and solve the existing patched processing problems. Then, lightweight and CNN features were concatenated in the first stage and fed into the classification module in the second stage.
- We used KSL video and 3 BSL datasets to evaluate the model and achieved 88.00% accuracy for KSL and 96.88% for our proposed dataset.

The presented work is organized as follows: Section 2 summarizes the existing research work and problems related to the presented work, Section 3 describes the benchmark and proposed Korean sign language datasets, and Section 4 describes the architecture of the proposed system. Section 5 shows evaluation of the performance. In Section 6, we draw the conclusion and future work.

2. Related Work

Sign language classification, continuous sign language recognition (SLR), and sign language translation are the main subdomains of the SLR. Many researchers have been working to develop sign language recognition systems for various languages with several machine learning and deep learning algorithms [34,35]. As different countries have their own sign language based on their mainstream language, researchers have been working to develop a sign language recognizer for individual sign languages, among which the Bengali sign language recognizer was developed with the machine learning and deep learning algorithm [4,5,36–38]. Pitsikalis et al. applied the hidden Markov model to classify the linguistic labeling sub-unit sign. They collected over 961 images with the Kinect TM device and performed well [39]. Additionally, the HMMs model can learn non-discriminatory features, but sometimes it ignores the data coming from the alternate class. In addition, it cannot sell useful features with it. Ong et al. proposed a multi-class sign language classification model using the sequential pattern trees, namely the SP-tree boosting algorithm, where they mainly extracted hand trajectory features from the subunit of the image [40]. To evaluate their model, they used Greek sign language (GSL), which achieved 93.00% accuracy and German sign language, where they achieved 88.00% accuracy, which is better than the hidden Markov model. Almeida et al. proposed a phonological structure based on decomposition and feature extraction with the RGB-D sign language image. After applying a support vector machine (SVM) as a classifier, they achieved 80% accuracy [41]. Fatimi et al. proposed an ASL recognition model using an artificial neural network (ANN) and SVM, where they achieved higher performance accuracy with ANN compared to the HMM and SVM [42]. In addition, 98.2% accuracy was achieved with SVM for the five wearable sign language devices [43], the fuzzy network was used for Chinese sign language (CSL) [44] and KNN, LDA, and SVM were applied for the ASL for 0-9, which achieved 98% higher accuracy [45]. Moreover, some other machine learning-based algorithms were employed to sign language recognition and achieve satisfactory accuracy, such as the multi-stream hidden Markov model [46], hidden conditional random field (HCRF), and random decision forest (RDF) [47,48]. Computational complexity, the inefficiency of selecting potential images and many other drawbacks are faced by researchers in the general machine learning algorithm for image-based sign language recognition. To overcome the problems, researchers used an ANN-based algorithm for sign language recognition [49,50]. In addition, Kim et al. applied ensemble ANN to classify Korean sign language. Based on the 10 labels and 1500 samples, they achieved 97.4% accuracy by considering only finger spelling signs. Camgoz et al. employed a model for making a dataset for KSA [51], then Ko et al. employed translation modeling for exploiting 2D human pose key points and released a KSL dataset [52]. Recently, the computer vision domain has largely switched toward the deep learning-based model due to the performance and various complexity-related issues. Al-Hammadi et al. proposed a single and fusion parallel 3-dimensional convolutional neural network (3DCNN) to recognize three gesture datasets where they achieved 84.38%, 34.9%, and 70% for 40, 23, and 10 class datasets, respectively, for the signer-independent and +10% for the signer-dependent cases [53]. Their performance accuracy is higher than the same previous method. Sincan et al. [37] proposed CNN and the long short-term memory (LSTM) algorithm based on the feature pooling module (FPM) and attention feature, where they evaluated Turkish sign language and achieved good performance. They mainly used an attention module to achieve convergence points as quickly as possible. Yuan et al. proposed a deep convolutional neural network (DCNN) and LSTM to recognize ASL and CSL sign language. The advantage of their model is that they overcome the gradient vanishing and overfitting problem, whereas their model produces long-distance dependency problems in the future feature network [54]. Aly et al. proposed a deep bi-direction LSTM (BiLSTM) to recognize Arabic sign language using a self-organizing map for hand shape feature extraction [55]. The same 2DCNN, 3DCNN, and LSTM models were used to classify skeleton-based sign language recognition [56]. More recently, researchers used different existing CNN architectures for solving specific problems, such as CNN with attention [57–59], AlexNet [21,60] and VGGNet [61], which focus on the convolution and pooling layer, multiple paths of the basic block effectiveness showed by GoogleNet [62] and InceptionNet [63]. ResNet [22] showed better generalization by adding shortcut connections every two layers to the base network. Researchers used an attention module to overcome the existing problem as an operator between adaptable modalities [64]. Stack attention modules used as an intermediate stage between residual networks [65], SENet [24], and GENet [66] were used to analyze the interdependencies among the channels. NLNet employs neural networks as a self-attention mechanism for providing parts interconnection among the spatial position to argue the long-range dependencies [67]. Moreover, MobileNets [68] and EfficientNets [69] were recently used to provide mobile-size portable networks. Transformers are used to achieve remarkably advanced success in natural language processing tasks, and many researchers employed them as effective articles for vision tasks [21–24,26,27]. The main transformer architecture ViT [30] is directly inherited from NLP to the computer vision domain to recognize sign language classification, although it needs a large-scale dataset to achieve satisfactory performance accuracy [70]. To overcome the large dataset problem, another researcher proposed DeiT to introduce a new training procedure with higher efficiency [33]. Another researcher proposed another extended transformer, T2T-ViT [71], to convert recursively neighboring token aggregation into single tokens. Vit used only the patch sequence, which may cause the loss of some potential information. By including pixel-level information with the patch level, another researcher proposed a TNT transformer with inner and outer blocks, respectively. PVT, CPVT, and CvT combined the transformer and CNN to overcome the long-range dependency problem, but their combination utilization was not satisfactory [72–75]. Another researcher employed the CMT method to solve the combination inefficiency problem, which included four stages, a transformer, and the CNN [28]. The main drawback of the CMT is the mix-up of long-term and short dependency in each stage, which increases the computational complexity. Moreover, many transformer-based deep learning models have been developed but have not achieved satisfactory performance. We proposed a simple multi-branch transformer-based architecture to recognize sign language datasets to solve the problem. It is known that for modeling long-range dependency, transformer demonstrated higher efficiency. Moreover, in the paper, we concatenated the transformer-based feature and convolution layers feature, where the overall performance proved the efficiency of the model.

3. Dataset

According to the researchers, a few datasets are available for Korean sign language (KSL) [76]. There is only one dataset we found online for working here: the KSL dataset. We created a new dataset to overcome the lack of a dataset. The KSL dataset is described in Section 3.1, and the proposed dataset is in Section 3.2.

3.1. KSL-Dataset

KSL is the first vision-based dataset for the Korean sign language, where they collected 77 words from 20 people. Although there are many words in the Korean language for daily activities, they tried to cover the most usable words that Koreans use in real life. They included 17 locations for the 20 signers, and they collected facial expressions with hand movement. Moreover, they considered different angles and distances for collecting actual videos in accounting for the real-life situation. They collected a total of 1229 videos and 112,564 frames as well. They extracted frames from the video by considering 30 ps, and to avoid noise and empty information, they also discarded some initial and end images [76]. Figure 1 shows the sample sign of this dataset.

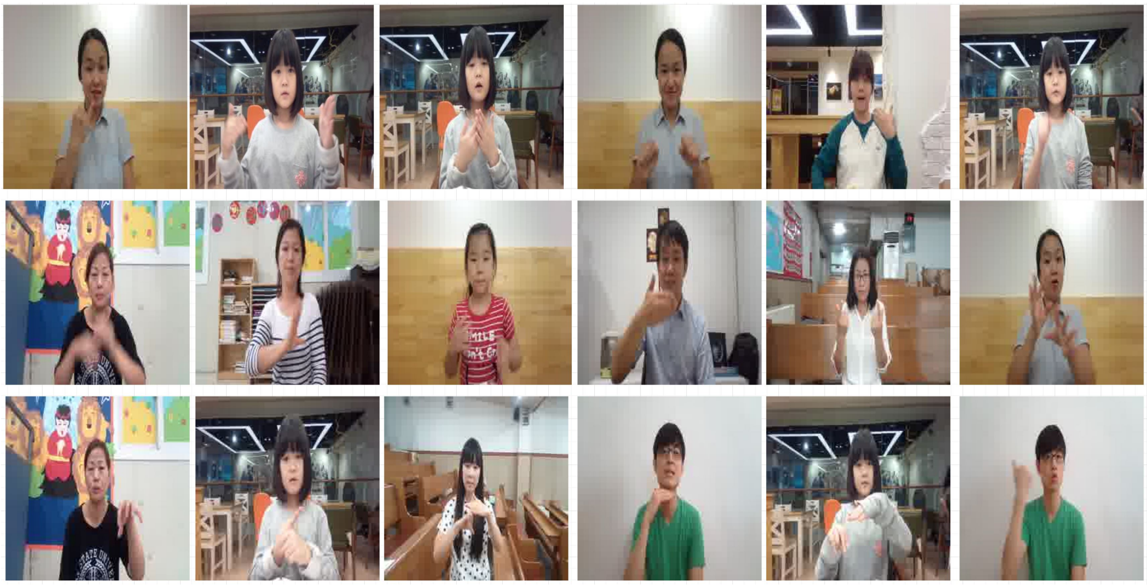

Figure 1. Example of KSL image from 77 class KSL dataset.

3.2. Proposed Dataset

As there are very few datasets available online for Korean sign language, we created an independent Korean sign language dataset using the 20 most significant words in the study. The selected Korean sign language words are thanks, okay, love, happy, sorry, no, hello, late, meet, shame, effort, regrettable, give, yes, help me, welcome, by, who, what, who, and why [75–77]. Also, the author selected 77 words that Korean people use to perform daily activities, but all 77 are not the most usable. Some of them are the most important. We selected the 5 most usable words from the 77 words [75]. The five words are as follows: no, who, what, thanks, and sorry. The photographs of each word are shown in Figure 2. Using the webcam, we collected 20 words in the KSL dataset. Twenty-one people participated in collecting the videos. Sample photographs of each significant word gesture are shown in Figure 2. We collected a 4-second video or 120 frames for each action from a person, and 20 people willingly participated in the data collection procedure. In addition, the background of the video appears in different scenarios as much as possible, such as a natural background.

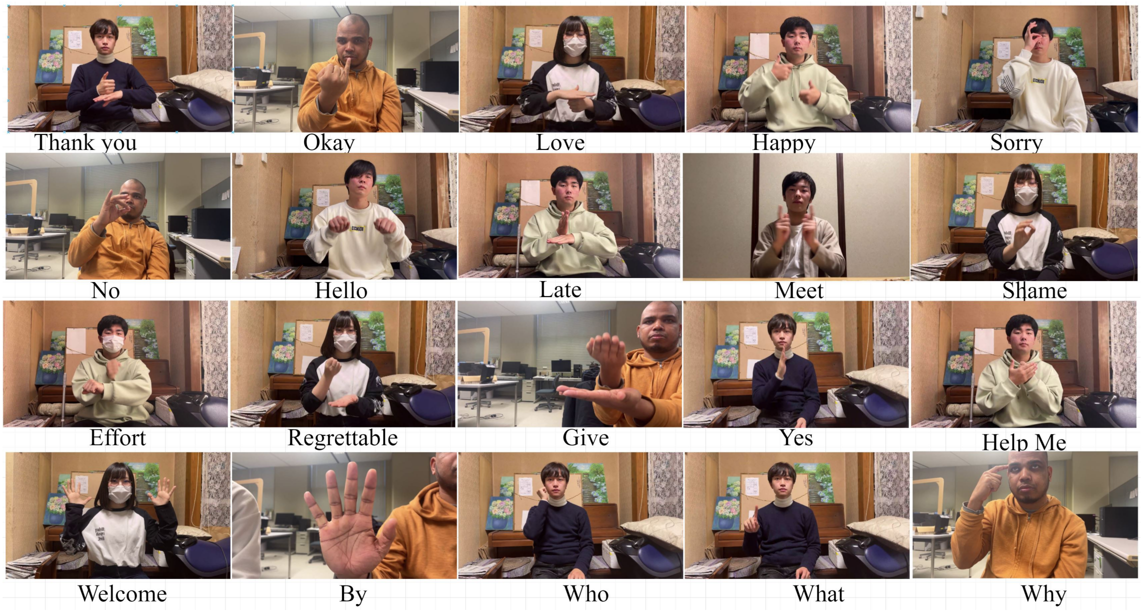

Figure 2. Proposed Korean sign language word.

4. Proposed Methodology

In the study, we proposed a convolutional layer-based transformer and CNN as a multi-branch hybrid network to include the transformer and convolutional neural network (CNN) feature by following the architecture of CMT [28], ResNet-50 [22], and DeiT [33,78]. Most of the existing transformer-based systems directly split the input image into multiple patches, and the main drawback of the patch system is that it can be poorly modeled with a linear projection. In the study, though we followed the patches concept, we employed a new architecture, grain architecture, to implement the patches idea instead of splitting the image. In the grain architecture, firstly, we included two 3 × 3 convolutions with stride one. Then we used a 3 × 3 convolution layer with a 2-stride layer where 32 was used as an output channel for reducing the size of the image to establish the patch concept of the transformer and modern CNNs [5,22]. Our model has three stages—initial grain, feature extraction, and classification module, shown in Figure 3a. The initial stage is used to perform the patches mechanism; the second stage is a feature extraction consisting of two parallel branches for generating different scales of feature maps to achieve dense prediction tasks. Finally, the stage is considered a classification module. After concatenating the two branches of features, we produced the final features fed into the classification module. In the convolutional layer-based transformer, we included an initial module, convolutional layer, layer normalization and lightweight multi-head attention layer and MLP convolution. In the initial module, we combined a convolutional layer with a normalization layer to produce the hierarchical representation of the grained features [5,33]. Convolutional layer-based transformer extracted short-range and long-range dependencies among the pixels. Lastly, we employed a classification module, including some convolutional layer with GELU activation and then a global average pooling layer, a fully connected (FC) layer and an n-way classification layer with a SoftMax activation function. Algorithm 1 shows the overall working flow where we included every step we followed in our work.

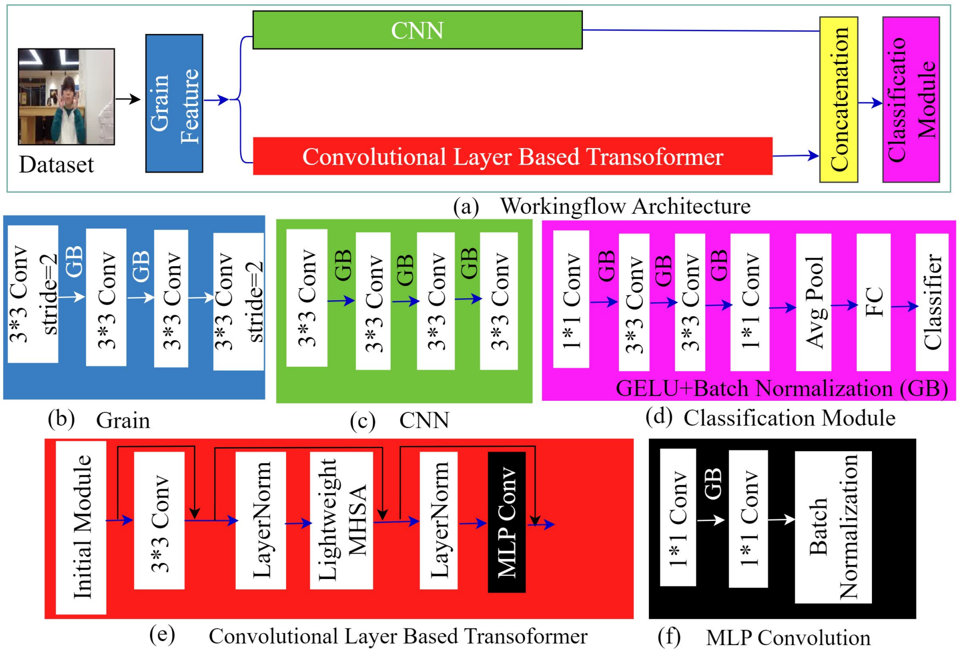

Figure 3. (**a**) Working flow architecture. (**b**) Grain model. (**c**) CNN feature extraction module. (**d**) Classification module (**e**) Convolutional layer-based transformer. (**f**) MLP convolution.

Algorithm 1 Pseudocode of the proposed system.

```
Input: Set of Input Dataset $P_i \in P(n)$
Number of Samples: N, 70% for Training and 30% for Test
Output: Set of vector $s_i$
define Model(input=InpuLayer, outputs=ClassificationLayer):

$GrainFeature \leftarrow GrainModule(D)$
$FirstBranchFeature \leftarrow ConvolutionalLayerBasedTransformer(GrainFeature)$
$SecondBranchFeature \leftarrow CNN(GrainFeature)$
$FinalFeature \leftarrow Concatenate(FirstBranchFeature, SecondBranch - Feature)$
$PredictedClass \leftarrow Classification - Module(Final - Feature)$
return PredictedClass

while$i \neq NumEpochs$ do
    // For Training
    while$Batch \neq NumberBatchTraining$ do
        $PredictedClass \leftarrow Model(Batch)$
        $Loss \leftarrow Criterion(PredictedClass, Train_Class)$
        $Updatetheloss \leftarrow Loss.backward(), Optimizer.Step()$
        // For Testing
    while$Batch \neq NumberBatchTesting$ do
        $PredictedClass \leftarrow Model(Batch)$
        $Output \leftarrow CPerformanceMatrix(PredictedClass, TestClass)$
```

4.1. Grain Module

First, we fed the original image into the grain module for fine-grained initial feature extraction—this module we designed by following modern CNNs (e.g., ResNet [33]). We divided this module into two blocks. The first block includes 3 × 3 convolutions with a stride of two and an output channel of thirty-two to reduce the size of input images, followed by another two 3 × 3 convolutions layer by considering one as a stride value. The second stage performs the patch aggregation approach by including the convolution layer and layer normalization.

4.2. Convolutional Layer-Based Transformer

The convolutional layer-based transformer block consists of an initial module, a lightweight multi-head self-attention (LMHSA) module and MLP convolution as described below.

4.2.1. Initialization Module

We applied the initial module to extract local information from the dataset as position encoding, also known as local perception unit (LPU) [25]. One of the main goals of this module is to consider rotation and shift augmentation operation, which are two of the most important manners in the visual task, and these should not alter the rest of the system. In other words, to overcome the image translation dependency on the system [5,73,79], researchers used absolute positional encoding in previously developed transformers to initially leverage the order of tokens. The main concept adds unique position encoding to individual patches [73], but they ignore the local relation [80] and the structure information inside the patch. To overcome the problem we proposed here, which can be defined as the following equations,

$$IM(X) = EWConv(X) + X \tag{1}$$

where the initial module features denote IM, $EWConv(.)$ represents the element-wise convolution. $X \in R^{H \times W \times d}$, where the resolution is represented by $H \times W$, and the dimension of the feature is represented by d.

4.2.2. Lightweight Multihead Self Attention

Self-attention (SA) is a popular, effective model in the neural network [23]. Generally, the input of the self-attention module can be written as $X \in R^{n \times d}$, which is transformed into three matrices, namely query, key and value defined by $Q \in R^{n \times d}$, $K \in R^{n \times d}$, and $V \in R^{n \times d}$ respectively. Here, several patches are represented by $n = H \times W$. The dimension of the input, key and value can be written as d, d_k and d_v, respectively. The self-attention module can be written as the following Equation (2):

$$SA = softmax(\frac{qk^T}{\sqrt{d_k}}) \times v \tag{2}$$

Here, SA is the self-attention, and q, k, v, and d_k are the query, key, value and key dimension. To make self-attention lighter, we employed an element-wise convolutional neural network with stride k to reduce the dimension of key and value before the attention model [28]. Besides this, relative position bias B was added in each of the self-attention modules [81,82]. We can rewrite Equation (2) for the lightweight self-attention (LSA) as follows:

$$LSA = softmax(\frac{q\overline{k}^T}{\sqrt{d_k}} + B) \times \overline{v} \tag{3}$$

Here, LSA is the lightweight self-attention, and q, k, v, and B are the query, key, value, key dimension and relative position. B can be written as $B \in R^{n \times \frac{n}{k^2}}$, which is randomly initialized and learnable. The relative position B can be transferred into other dimensions $\overline{B}$ in bicubic interpolation with a different size $m_1 \times m_2$. The transformation equation can be written as $\overline{B} = Bicubic(B)$. This procedure is excreted for one head, the same way it will be completed for h heads, defined as lightweight multi-head self-attention (LMHSA). An h number of lightweight attention modules are employed to produce attention features. Each individual head outputs a sequence of size $n \times \frac{d}{h}$. These h sequences of attention feature are then concatenated into an $n \times \frac{d}{h}$ sequence. Figure 4 shows the steps of the LMHSA. Because of the parallelization, we used the lightweight self-attention mechanism because of the computational complexity per layer and the minimum number of operations required in the sequence. Convolutional layer-based transformer block computation can be written as the following formula:

$$\overline{X_i} = IM(X_{i-1}) \tag{4}$$

$$\overline{\overline{X_i}} = LMHSA(LN(\overline{X_i}) + \overline{X_i}) \tag{5}$$

Here, $\overline{X_i}$ and $\overline{\overline{X_i}}$ represent the IM and LMHSA block feature for the individual module i, consequently. Moreover, layer normalization is denoted by the term LN.

4.2.3. MLP Convolution

We employed the multilayer perception convolution block after the attention in the MHSA, which include a batch normalization layer, which can be defined by the following Equation (6):

$$\overline{\overline{\overline{X_i}}} = MLPConv(conv(\overline{\overline{X_i}})) \tag{6}$$

The MLP convolutional module contained a single block consisting of two 1 × 1 convolution layers [83]. This book also looks like a normal convolution layer, where GELU activation and batch normalization are used after the first 1 × 1 convolution layer and only batch normalization after the second convolutional layer. Still, their kernel size is 1, which is worked for 1 pixel for the input images. The working procedure of this convolution layer is almost the same as the position-wise dense layer. Usually, normal convolution layers detect local patterns based on spatial information, whereas the MLP convolutional layer does not follow this but uses windows of action in a single position. The main difference between the 2D convolution and the MLP convolution is in the channels or the embedding dimensionality. If channel c is from the 2D convolution, $\bar{c}$ is the channel of the MLP convolution, and w and h are the width and height of both types, then $c \neq \bar{c}$. The main purpose of using CNN is to extract two-dimensional neighborhood structures, whereas MLPConv, after MHSA, converts the global MHSA into local pixel information.

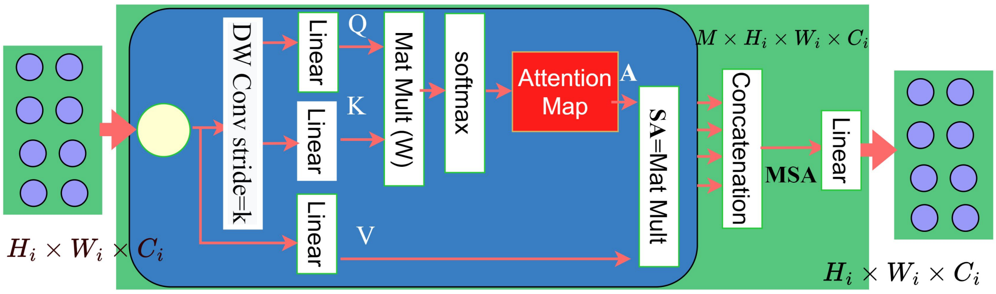

Figure 4. Lightweight multi-head attention.

4.3. Convolutional Neural Network (CNN) Module

We applied a four-block of 3 × 3 convolution layer with GELU activation and batch normalization to extract local features from the grain feature. Although there have been many sign language recognition models that only applied CNN to extract local features from the image dataset, we aim to include this advantage in the transformer.

4.4. Classification Module

In the classification module, we used the modified version of FFN of the ViT [84], which described two linear layers separated by an activation function name GELU [29]. They explained that expanding the dimension with a factor of 4 in the first layer and dimension reduces it in the same ways in the second layer.Here, the weight matrices of the two layers can be written as $W_1 \in R^{d \times 4d}$ and $W_2 \in R^{4d \times d}$, respectively. In addition, the biases of the two layers are represented by b_1, and b_2, respectively. The schematic diagram of the proposed classification module is shown in Figure 3c, which is similar to the inverted residual feed-forward network (IRFFN) [28,31] consisting of an expansion layer through the element-wise convolution and two projection layers. This can be written as the following Equation (7):

$$F_{avg} = Avg(Fconv(conv(conv(conv(x))))) \tag{7}$$

$$F(x) = F(F_{avg}) \tag{8}$$

where F_{avg} represents the output of the averaging pooling layer, and F(x) represents the output of the fully connected layer. In each layer, we included GELU activation and batch normalization layer. The element-wise convolutional neural network is employed to calculate the local information with a minimum cost and value. After that, we applied a global average pooling layer to average the features into a vector, a fully connected (FC) layer and an n-way classification layer with softmax.

5. Experimental Evaluation

We investigated the effectiveness and superiority of the proposed model in the section by conducting experiments on sign language classification with large-scale Korean sign language datasets. Firstly, we demonstrated the performance of the proposed model with multiple datasets, and then we showed the state-of-the-art comparison of the proposed model.

5.1. Training Setting

To train the model, we divided the dataset into 70% as training and 30% as testing. In the training process, we used a learning rate of 0.001, a weight decay of 0.0001, a dropout rate of 0.1, GELU activation, and a batch size of 32. We used a GPU machine with CUDA version 11.7, NVIDIA-SMI 515, GPU name Persistence-M, and RAM 32 GB to implement the system. Models were run for 300 epochs with the optimizer Adam [85] with the Persistence-M GPU.

5.2. Performance with the Benchmark KSL and Proposed Dataset

We considered two benchmark datasets in the experiment: KSL and our proposed dataset. KSL is the famous Korean sign language dataset we used to evaluate the proposed model. Figure 5 shows the confusion matrix showing that most of the Korean alphabet produces 100% accuracy; two signs achieved more than 95%, and two signs achieved in the range 84% to 90% accuracy. Figure 6 demonstrates the proposed model's precision, recall, F1-score and accuracy with the proposed 20 classes of Korean sign language datasets. The figure demonstrates 98.50%, 98.35%, 98.40%, and 98.30% for precision, recall, F1-score and accuracy, respectively, which proves the effectiveness of the proposed model. Figure 7 visualizes the proposed model's precision, recall, F1-score and accuracy with the KSL dataset with 20 classes. The figure demonstrates 94.80%, 87.20%, 90.25%, and 89.05% for precision, recall, F1-score and accuracy, respectively, which proves the effectiveness of the proposed model.

Table 1 shows the performance accuracy of the proposed model with the proposed and KSL datasets. For the KSL dataset, our model produced 89.00% accuracy, and for the proposed dataset, our model achieved 98.30% accuracy.

Table 1. Performance accuracy of the proposed model.

Dataset Name	Accuracy [%]	Parameters	Flops
Proposed Dataset	98.30	1.52 M	1.17 GMac
KSL	89.00	1.5 M	245.5 MMac

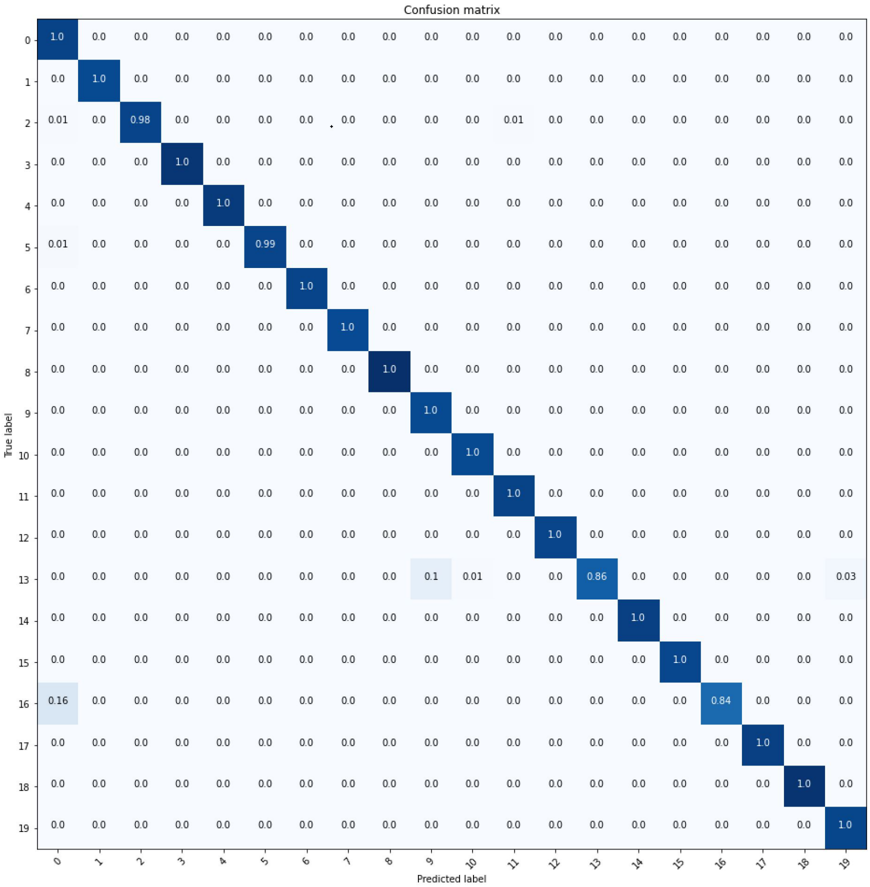

Figure 5. Confusion matrix of the proposed dataset.

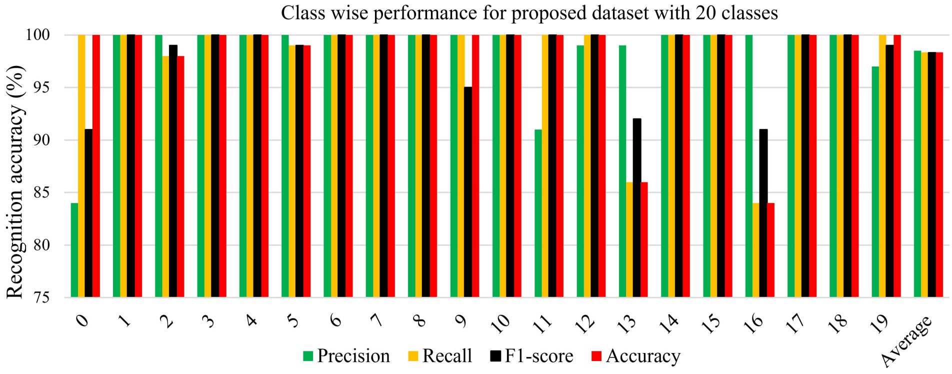

Figure 6. Precision, recall, F1-score and accuracy of the proposed dataset.

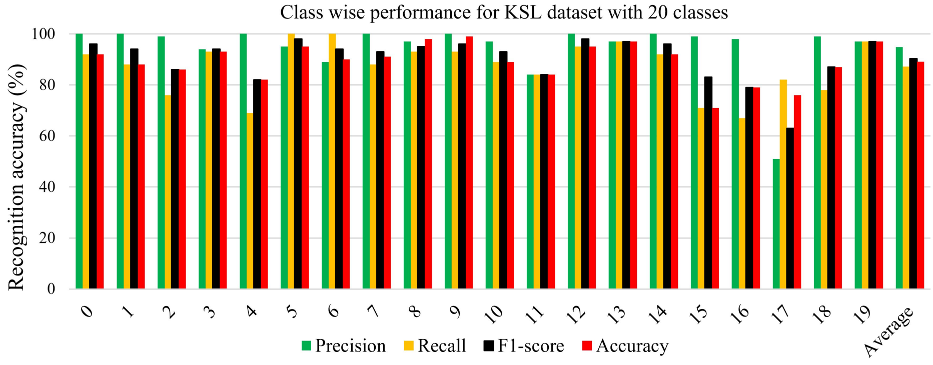

Figure 7. Precision, recall, F1-score and accuracy of the KSL dataset.

5.3. Comparison for the State-of-the-Art Method for KSL Dataset

In this section, we include the state-of-the-art comparison for the KSL dataset. Table 2 shows the state-of-the-art method performance for the KSL dataset. Authors of the [55] employed a deep learning model, which recently achieved high performance and 79% accuracy. Our proposed model gained 89.00% accuracy, which is higher than the existing state-of-the-art model.

Table 2. Comparison of the state-of-the-art model for the KSL dataset.

Dataset Name	Model Name	Accuracy [%]
KSL	TSN [76]	79.80
KSL	TSN [86]	83.66
KSL	Proposed Model	89.00

There are some well-known architectures from which to extract hierarchical feature maps based on different resolutions from the RGB image, such as typical convolutional neural networks [76] and fully convolutional networks [86]. The proposed study can be extracted as a multi-scale representation of the input, where we combined CNN and attention-based features, which proved the model's effectiveness.

6. Conclusions

In the study, we proposed a novel multi-branch architecture for sign language recognition to address the computational complexity and transformer limitations utilized in a brute-force manner. The proposed model takes advantage of both transformers and CNNs to generate global and local information to increase the representation ability of the architecture. Our study achieved good performance compared to the existing Korean sign language model. Extensive experimentation performance visualized the superiority and effectiveness of the proposed model. In the future, we will add more data to the datasets and compare the results with the other latest benchmark methods. In addition, we plan to reduce the number of parameters of the model by considering the speed of the system and apply it in other domains for multi-modal applications.

Author Contributions: Conceptualization, A.S.M.M., M.A.M.H. and J.S.; methodology, A.S.M.M., M.A.M.H., H.-S.L., S.-W.J., and J.S.; investigation, A.S.M.M., M.A.M.H. and J.S.; data collection, A.S.M.M., J.S., K.H. and K.S., data curation, A.S.M.M., M.A.M.H., H.-S.L., S.-W.J. and J.S.; writing—original draft preparation, A.S.M.M. and J.S.; writing—review and editing, A.S.M.M., J.S. and M.A.M.H.; visualization, A.S.M.M. and M.A.M.H.; supervision, J.S.; funding acquisition, H.-S.L., S.-W.J. and J.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by the MSIT (Ministry of Science and ICT), Korea, under the Grand Information Technology Research Center support program (IITP-2023-2020-0-01791) supervised by the IITP (Institute for Information & communications Technology Planning & Evaluation). The Competitive Research Fund of The University of Aizu, Japan supported this work.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

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

References

1. Rafi, A.M.; Nawal, N.; Bayev, N.S.N.; Nima, L.; Shahnaz, C.; Fattah, S.A. Image-based bengali sign language alphabet recognition for deaf and dumb community. In Proceedings of the 2019 IEEE Global Humanitarian Technology Conference (GHTC), Seattle, WA, USA, 17–20 October 2019; Volume 30, pp. 1–7.
2. Musa Miah, A.S.; Hasan, M.A.M.; Shin, J. Dynamic Hand Gesture Recognition using Multi-Branch Attention Based Graph and General Deep Learning Model. *IEEE Access* **2023**, *11*, 4703–4716. [CrossRef]

3. Miah, A.S.M.; Hasan, M.A.M.; Shin, J.; Okuyama, Y.; Tomioka, Y. Multistage Spatial Attention-Based Neural Network for Hand Gesture Recognition. *Computers* **2023**, *12*, 13. [CrossRef]
4. Miah, Abu Saleh Musa, SHin, J.P; Hasan, M.A.M.; Rahim, M.A.; Okuyama, Y. Rotation, Translation And Scale Invariant Sign Word Recognition Using Deep Learning. *Comput. Syst. Sci. Eng.* **2023**, *44*, 2521–2536. [CrossRef]
5. Miah, A.S.M.; Shin, J.; Hasan, M.A.M.; Rahim, M.A. BenSignNet: Bengali Sign Language Alphabet Recognition Using Concatenated Segmentation and Convolutional Neural Network. *Appl. Sci.* **2022**, *12*, 3933. [CrossRef]
6. Miah, A.S.M.; Rahim, M.A.; Shin, J. Motor-imagery classification using riemannian geometry with median absolute deviation. *Electronics* **2020**, *9*, 1584. [CrossRef]
7. Miah, A.S.M.; Shin, J.; Islam, M.M.; Molla, M.K.I. Natural Human Emotion Recognition Based on Various Mixed Reality (MR) Games and Electroencephalography (EEG) Signals. In Proceedings of the 2022 IEEE 5th Eurasian Conference on Educational Innovation (ECEI), Taipei, Taiwan,10–12 February 2022; pp. 408–411.
8. Rahim, M.A.; Miah, A.S.M.; Sayeed, A.; Shin, J. Hand Gesture Recognition Based on Optimal Segmentation in Human-Computer Interaction. In Proceedings of the 2020 3rd IEEE International Conference on Knowledge Innovation and Invention (ICKII), Hualien, Taiwan, 22 July 2020; pp. 163–166.
9. Miah, A.S.M.; Mamunur Rashid, M.; Rahman, R.; Hossain, T.; Sujon, S.; Nawal, N.; Hasan, M.; Shin, J. Alzheimer's disease detection using CNN based on effective dimensionality reduction approach. In *Proceedings of the International Conference on Intelligent Computing & Optimization*; Springer: Berlin/Heidelberg, Germany, 2021; pp. 801–811.
10. Kibria, K.A.; Chowdhury, A.A.; Miah, A.S.M.; Shahriar, M.R.; Pervin, S.; Shin, J.; Rashid, M.M.; Sarkar, A.R. Bangladeshi Land Cover Change Detection with Satelite Image Using GIS Techniques. In *Proceedings of the Machine Intelligence and Data Science Applications*; Skala, V., Singh, T.P., Choudhury, T., Tomar, R., Abul Bashar, M., Eds.; Springer Nature: Singapore, 2022; pp. 125–143.
11. Miah, A.S.M.; Mouly, M.A.; Debnath, C.; Shin, J.; Sadakatul Bari, S. Event-Related Potential Classification Based on EEG Data Using xDWAN with MDM and KNN. In *Proceedings of the International Conference on Computing Science, Communication and Security*; Springer: Berlin/Heidelberg, Germany, 2021; pp. 112–126.
12. Cui, R.; Liu, H.; Zhang, C. Recurrent convolutional neural networks for continuous sign language recognition by staged optimization. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 7361–7369.
13. Koller, O.; Zargaran, S.; Ney, H. Re-sign: Re-aligned end-to-end sequence modelling with deep recurrent CNN-HMMs. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 4297–4305.
14. Camgoz, N.C.; Hadfield, S.; Koller, O.; Ney, H.; Bowden, R. Neural sign language translation. In Proceedings of the Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 7784–7793.
15. Camgoz, N.C.; Koller, O.; Hadfield, S.; Bowden, R. Sign language transformers: Joint end-to-end sign language recognition and translation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 14–19 June 2020; pp. 10023–10033.
16. Kudrinko, K.; Flavin, E.; Zhu, X.; Li, Q. Wearable sensor-based sign language recognition: A comprehensive review. *IEEE Rev. Biomed. Eng.* **2020**, *14*, 82–97. [CrossRef]
17. Sharma, S.; Singh, S. Vision-based sign language recognition system: A Comprehensive Review. In Proceedings of the 2020 International Conference on Inventive Computation Technologies (ICICT), Lalitpur, Nepal, 20–22 July 2022; pp. 140–144.
18. Rajan, R.G.; Leo, M.J. American sign language alphabets recognition using hand-crafted and deep learning features. In Proceedings of the 2020 International Conference on Inventive Computation Technologies (ICICT), Coimbatore, India, 26–28 February 2020; pp. 430–434.
19. Podder, K.K.; Chowdhury, M.E.; Tahir, A.M.; Mahbub, Z.B.; Khandakar, A.; Hossain, M.S.; Kadir, M.A. Bangla sign language (bdsl) alphabets and numerals classification using a deep learning model. *Sensors* **2022**, *22*, 574. [CrossRef]
20. Awan, M.J.; Rahim, M.S.M.; Salim, N.; Rehman, A.; Nobanee, H.; Shabir, H. Improved deep convolutional neural network to classify osteoarthritis from anterior cruciate ligament tear using magnetic resonance imaging. *J. Pers. Med.* **2021**, *11*, 1163. [CrossRef]
21. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. *Commun. ACM* **2017**, *60*, 84–90. [CrossRef]
22. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 26 June–1 July 2016; pp. 770–778.
23. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. In Proceedings of the 31st Conference on Neural Information Processing Systems (NIPS 2017), Long Beach, CA, USA, 4–9 December 2017; Volume 30.
24. Wang, X.; Girshick, R.; Gupta, A.; He, K. Non-local neural networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 7794–7803.
25. Carion, N.; Massa, F.; Synnaeve, G.; Usunier, N.; Kirillov, A.; Zagoruyko, S. End-to-end object detection with transformers. In *Proceedings of the European Conference on Computer Vision*; Springer: Berlin/Heidelberg, Germany, 2020; pp. 213–229.
26. Ramachandran, P.; Parmar, N.; Vaswani, A.; Bello, I.; Levskaya, A.; Shlens, J. Stand-alone self-attention in vision models. In Proceedings of the Advances in Neural Information Processing Systems 32 (NeurIPS 2019), Vancouver, BC, Canada, 8–14 December 2019; Volume 32.

27. Wang, H.; Zhu, Y.; Green, B.; Adam, H.; Yuille, A.; Chen, L.C. Axial-deeplab: Stand-alone axial-attention for panoptic segmentation. In *Proceedings of the European Conference on Computer Vision*; Springer: Berlin/Heidelberg, Germany, 2020; pp. 108–126.
28. Guo, J.; Han, K.; Wu, H.; Tang, Y.; Chen, X.; Wang, Y.; Xu, C. Cmt: Convolutional neural networks meet vision transformers. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 19–26 June 2022; pp. 12175–12185.
29. De Coster, M.; Van Herreweghe, M.; Dambre, J. Sign language recognition with transformer networks. In Proceedings of the 12th International Conference on Language Resources and Evaluation, Marseille, France, 11–16 May 2020; European Language Resources Association (ELRA): Paris, France, 2020; pp. 6018–6024.
30. Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv* **2020**, arXiv:2010.11929.
31. Chen, H.; Wang, Y.; Guo, T.; Xu, C.; Deng, Y.; Liu, Z.; Ma, S.; Xu, C.; Xu, C.; Gao, W. Pre-trained image processing transformer. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 19–25 June 2021; pp. 12299–12310.
32. Zheng, S.; Lu, J.; Zhao, H.; Zhu, X.; Luo, Z.; Wang, Y.; Fu, Y.; Feng, J.; Xiang, T.; Torr, P.H.; et al. Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 19–25 June 2021; pp. 6881–6890.
33. Touvron, H.; Cord, M.; Douze, M.; Massa, F.; Sablayrolles, A.; Jégou, H. Training data-efficient image transformers & distillation through attention. In Proceedings of the International Conference on Machine Learning, Online, 18–24 July 2021; pp. 10347–10357.
34. Ong, S.C.; Ranganath, S. Automatic sign language analysis: A survey and the future beyond lexical meaning. *IEEE Trans. Pattern Anal. Mach. Intell.* **2005**, *27*, 873–891. [CrossRef] [PubMed]
35. Madhiarasan, D.M.; Roy, P.; Pratim, P. A Comprehensive Review of Sign Language Recognition: Different Types, Modalities, and Datasets. *arXiv* **2022**, arXiv:2204.03328.
36. Uddin, M.A.; Chowdhury, S.A. Hand sign language recognition for bangla alphabet using support vector machine. In Proceedings of the 2016 International Conference on Innovations in Science, Engineering and Technology (ICISET), Chittagong, Bangladesh, 28–29 October 2016; pp. 1–4.
37. Yasir, F.; Prasad, P.; Alsadoon, A.; Elchouemi, A.; Sreedharan, S. Bangla Sign Language recognition using convolutional neural network. In Proceedings of the 2017 International Conference on Intelligent Computing, Instrumentation and Control Technologies (ICICICT), Kerala, India, 6–7 July 2017; pp. 49–53.
38. Abedin, T.; Prottoy, K.S.; Moshruba, A.; Hakim, S.B. Bangla sign language recognition using concatenated BdSL network. *arXiv* **2021**, arXiv:2107.11818.
39. Pitsikalis, V.; Theodorakis, S.; Vogler, C.; Maragos, P. Advances in phonetics-based sub-unit modeling for transcription alignment and sign language recognition. In Proceedings of the CVPR 2011 WORKSHOPS, Colorado Springs, CO, USA, 20–25 June 2011; pp. 1–6.
40. Ong, E.J.; Cooper, H.; Pugeault, N.; Bowden, R. Sign language recognition using sequential pattern trees. In Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition, Providence, RI, USA, 16–21 June 2012; pp. 2200–2207.
41. Moreira Almeida, S.G.; Guimarães, F.G.; Arturo Ramírez, J. Feature extraction in Brazilian Sign Language Recognition based on phonological structure and using RGB-D sensors. *Expert Syst. Appl.* **2014**, *41*, 7259–7271. [CrossRef]
42. Fatmi, R.; Rashad, S.; Integlia, R. Comparing ANN, SVM, and HMM based Machine Learning Methods for American Sign Language Recognition using Wearable Motion Sensors. In Proceedings of the 2019 IEEE 9th Annual Computing and Communication Workshop and Conference (CCWC), Las Vegas, NV, USA, 7–9 January 2019; pp. 0290–0297. [CrossRef]
43. Lee, B.G.; Lee, S.M. Smart Wearable Hand Device for Sign Language Interpretation System With Sensors Fusion. *IEEE Sens. J.* **2018**, *18*, 1224–1232. [CrossRef]
44. Wei, S.; Chen, X.; Yang, X.; Cao, S.; Zhang, X. A Component-Based Vocabulary-Extensible Sign Language Gesture Recognition Framework. *Sensors* **2016**, *16*. [CrossRef]
45. Li, L.; Jiang, S.; Gu, G. SkinGest: Artificial skin for gesture recognition via filmy stretchable strain sensors. *Adv. Robot.* **2018**, *32*, 1–10. [CrossRef]
46. Yang, X.; Chen, X.; Cao, X.; Wei, S.; Zhang, X. Chinese Sign Language Recognition Based on an Optimized Tree-Structure Framework. *IEEE J. Biomed. Health Inform.* **2017**, *21*, 994–1004. [CrossRef]
47. Dawod, A.Y.; Chakpitak, N. Novel Technique for Isolated Sign Language Based on Fingerspelling Recognition. In Proceedings of the 2019 13th International Conference on Software, Knowledge, Information Management and Applications (SKIMA), Island of Ulkulhas, Maldives, 26–28 August 2019; pp. 1–8. [CrossRef]
48. Hoang, V.T. HGM-4: A new multi-cameras dataset for hand gesture recognition. *Data Brief* **2020**, *30*, 105676. [CrossRef]
49. Chansri, C.; Srinonchat, J. Hand Gesture Recognition for Thai Sign Language in Complex Background Using Fusion of Depth and Color Video. *Procedia Comput. Sci.* **2016**, *86*, 257–260.
50. Jane, S.P.Y.; Sasidhar, S. Sign Language Interpreter: Classification of Forearm EMG and IMU Signals for Signing Exact English. In Proceedings of the 2018 IEEE 14th International Conference on Control and Automation (ICCA), Anchorage, Ak, USA, 12–15 June 2018; pp. 947–952. [CrossRef]
51. Liu, T.; Liu, H.; Li, Y.F.; Chen, Z.; Zhang, Z.; Liu, S. Flexible FTIR Spectral Imaging Enhancement for Industrial Robot Infrared Vision Sensing. *IEEE Trans. Ind. Informatics* **2020**, *16*, 544–554. [CrossRef]

52. Mahajan, D.; Girshick, R.; Ramanathan, V.; He, K.; Paluri, M.; Li, Y.; Bharambe, A.; van der Maaten, L. Exploring the Limits of Weakly Supervised Pretraining. In *Proceedings of the Computer Vision—ECCV 2018*; Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y., Eds.; Springer: Cham, Switzerland, 2018; pp. 185–201.
53. Al-Hammadi, M.; Muhammad, G.; Abdul, W.; Alsulaiman, M.; Bencherif, M.A.; Mekhtiche, M.A. Hand Gesture Recognition for Sign Language Using 3DCNN. *IEEE Access* **2020**, *8*, 79491–79509. [CrossRef]
54. Yuan, G.; Liu, X.; Yan, Q.; Qiao, S.; Wang, Z.; Yuan, L. Hand gesture recognition using deep feature fusion network based on wearable sensors. *IEEE Sens. J.* **2020**, *21*, 539–547. [CrossRef]
55. Aly, S.; Aly, W. DeepArSLR: A novel signer-independent deep learning framework for isolated arabic sign language gestures recognition. *IEEE Access* **2020**, *8*, 83199–83212. [CrossRef]
56. Rastgoo, R.; Kiani, K.; Escalera, S. Hand sign language recognition using multi-view hand skeleton. *Expert Syst. Appl.* **2020**, *150*, 113336. [CrossRef]
57. Barbhuiya, A.A.; Karsh, R.K.; Jain, R. Gesture recognition from RGB images using convolutional neural network-attention based system. *Concurr. Comput. Pract. Exp.* **2022**, *34*, e7230. [CrossRef]
58. Barbhuiya, A.A.; Karsh, R.K.; Jain, R. A convolutional neural network and classical moments-based feature fusion model for gesture recognition. *Multimed. Syst.* **2022**, *28*, 1779–1792. [CrossRef]
59. Barbhuiya, A.A.; Karsh, R.K.; Jain, R. CNN based feature extraction and classification for sign language. *Multimed. Tools Appl.* **2021**, *80*, 3051–3069. [CrossRef]
60. Barbhuiya, A.A.; Karsh, R.K.; Dutta, S. AlexNet-CNN Based Feature Extraction and Classification of Multiclass ASL Hand Gestures. In *MCCS, Proceedings of the Fifth International Conference on Microelectronics, Computing and Communication Systems, 2020*; Springer: Berlin/Heidelberg, Germany, 2021; pp. 77–89.
61. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv* **2014**, arXiv:1409.1556.
62. Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; Rabinovich, A. Going deeper with convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 1–9.
63. Szegedy, C.; Vanhoucke, V.; Ioffe, S.; Shlens, J.; Wojna, Z. Rethinking the inception architecture for computer vision. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 2818–2826.
64. Wang, F.; Jiang, M.; Qian, C.; Yang, S.; Li, C.; Zhang, H.; Wang, X.; Tang, X. Residual attention network for image classification. In Proceedings of the Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 3156–3164.
65. Hu, J.; Shen, L.; Albanie, S.; Sun, G.; Vedaldi, A. Gather-excite: Exploiting feature context in convolutional neural networks. In Proceedings of the Advances in Neural Information Processing Systems, Montreal, QC, Canada, 3–8 December 2018; Volume 31.
66. Hu, J.; Shen, L.; Sun, G. Squeeze-and-excitation networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 7132–7141.
67. Sandler, M.; Howard, A.; Zhu, M.; Zhmoginov, A.; Chen, L.C. Mobilenetv2: Inverted residuals and linear bottlenecks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 4510–4520.
68. Howard, A.; Sandler, M.; Chu, G.; Chen, L.C.; Chen, B.; Tan, M.; Wang, W.; Zhu, Y.; Pang, R.; Vasudevan, V.; et al. Searching for mobilenetv3. In Proceedings of the IEEE/CVF International Conference on Computer Vision,Seoul, Korea, 27 October–2 November 2019; pp. 1314–1324.
69. Tan, M.; Le, Q. Efficientnet: Rethinking model scaling for convolutional neural networks. In Proceedings of the International Conference on Machine Learning, Long Beach, Long Beach Convention & Entertainment Center, Long Beach, CA, USA, 10–15 June 2019; pp. 6105–6114.
70. Sun, C.; Shrivastava, A.; Singh, S.; Gupta, A. Revisiting unreasonable effectiveness of data in deep learning era. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 843–852.
71. Yuan, L.; Chen, Y.; Wang, T.; Yu, W.; Shi, Y.; Jiang, Z.H.; Tay, F.E.; Feng, J.; Yan, S. Tokens-to-token vit: Training vision transformers from scratch on imagenet. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, QC, Canada, 11–17 October 2021; pp. 558–567.
72. Wang, W.; Xie, E.; Li, X.; Fan, D.P.; Song, K.; Liang, D.; Lu, T.; Luo, P.; Shao, L. Pyramid vision transformer: A versatile backbone for dense prediction without convolutions. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, QC, Canada, 11–17 October 2021; pp. 568–578.
73. Chu, X.; Tian, Z.; Zhang, B.; Wang, X.; Wei, X.; Xia, H.; Shen, C. Conditional positional encodings for vision transformers. *arXiv* **2021**, arXiv:2102.10882.
74. Wu, H.; Xiao, B.; Codella, N.; Liu, M.; Dai, X.; Yuan, L.; Zhang, L. Cvt: Introducing convolutions to vision transformers. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, QC, Canada, 11–17 October 2021; pp. 22–31.
75. Ji, Y.; Kim, S.; Lee, K.B. Sign language learning system with image sampling and convolutional neural network. In Proceedings of the 2017 First IEEE International Conference on Robotic Computing (IRC), Taichung, Taiwan, 10–12 April 2017; pp. 371–375.
76. Yang, S.; Jung, S.; Kang, H.; Kim, C. The korean sign language dataset for action recognition. In *Proceedings of the International Conference on Multimedia Modeling*; Springer: Berlin/Heidelberg, Germany, 2020; pp. 532–542.

77. Shin, H.; Kim, W.J.; Jang, K.a. Korean sign language recognition based on image and convolution neural network. In Proceedings of the 2nd International Conference on Image and Graphics Processing, Singapore, 23–25 February 2019; pp. 52–55.
78. Cui, R.; Liu, H.; Zhang, C. A deep neural framework for continuous sign language recognition by iterative training. *IEEE Trans. Multimed.* **2019**, *21*, 1880–1891. [CrossRef]
79. Kayhan, O.S.; Gemert, J.C.v. On translation invariance in cnns: Convolutional layers can exploit absolute spatial location. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA 13–19 June 2020; pp. 14274–14285.
80. Lowe, D.G. Object recognition from local scale-invariant features. In Proceedings of the Seventh IEEE International Conference on Computer Vision, Corfu, Greece, 20–25 September 1999; Volume 2, pp. 1150–1157.
81. Shaw, P.; Uszkoreit, J.; Vaswani, A. Self-attention with relative position representations. *arXiv* **2018**, arXiv:1803.02155.
82. Liu, Z.; Lin, Y.; Cao, Y.; Hu, H.; Wei, Y.; Zhang, Z.; Lin, S.; Guo, B. Swin transformer: Hierarchical vision transformer using shifted windows. In Proceedings of the IEEE/CVF International Conference on Computer Vision,Kerkyra, Greece, 11–17 October 2021; pp. 10012–10022.
83. Li, J.; Hassani, A.; Walton, S.; Shi, H. Convmlp: Hierarchical convolutional mlps for vision. *arXiv* **2021**, arXiv:2109.04454.
84. Loshchilov, I.; Hutter, F. Decoupled weight decay regularization. *arXiv* **2017**, arXiv:1711.05101.
85. Dozat, T. Incorporating Nesterov Momentum into Adam. In Proceedings of the ICLR 2016 Workshop, San Juan, PR, USA, 2–4 May 2016.
86. Ham, S.; Park, K.; Jang, Y.; Oh, Y.; Yun, S.; Yoon, S.; Kim, C.J.; Park, H.M.; Kweon, I.S. KSL-Guide: A Large-scale Korean Sign Language Dataset Including Interrogative Sentences for Guiding the Deaf and Hard-of-Hearing. In Proceedings of the 2021 16th IEEE International Conference on Automatic Face and Gesture Recognition (FG 2021), Jodhpur, India, 15–18 December 2021; pp. 1–8. [CrossRef]

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.