applied sciences

MDPI

Article

A Performance Comparison of Japanese Sign Language Recognition with ViT and CNN Using Angular Features

Tamon Kondo [1], Sakura Narumi [1], Zixun He [2], Duk Shin [2] and Yousun Kang [2,*]

[1] Graduate School of Engineering, Tokyo Polytechnic University, Atsugi 243-0218, Kanagawa, Japan; m2363001@st.t-kougei.ac.jp (T.K.); d2481001@st.t-kougei.ac.jp (S.N.)
[2] Faculty of Engineering, Tokyo Polytechnic University, Atsugi 243-0218, Kanagawa, Japan; hezixun94@gmail.com (Z.H.); d.shin@eng.t-kougei.ac.jp (D.S.)
* Correspondence: yskang@eng.t-kougei.ac.jp; Tel.: +81-46-242-9524

Abstract: In recent years, developments in deep learning technology have driven significant advancements in research aimed at facilitating communication with individuals who have hearing impairments. The focus has been on enhancing automatic recognition and translation systems for sign language. This study proposes a novel approach using a vision transformer (ViT) for recognizing Japanese Sign Language. Our method employs a pose estimation library, MediaPipe, to extract the positional coordinates of each finger joint within video frames and generate one-dimensional angular feature data from these coordinates. Then, the code arranges these feature data in a temporal sequence to form a two-dimensional input vector for the ViT model. To determine the optimal configuration, this study evaluated recognition accuracy by manipulating the number of encoder layers within the ViT model and compared against traditional convolutional neural network (CNN) models to evaluate its effectiveness. The experimental results showed 99.7% accuracy for the method using the ViT model and 99.3% for the results using the CNN. We demonstrated the efficacy of our approach through real-time recognition experiments using Japanese sign language videos.

Keywords: Japanese sign language; MediaPipe; vision transformer

Citation: Kondo, T.; Narumi, S.; He, Z.; Shin, D.; Kang, Y. A Performance Comparison of Japanese Sign Language Recognition with ViT and CNN Using Angular Features. *Appl. Sci.* **2024**, *14*, 3228. https://doi.org/10.3390/app14083228

Academic Editors: Sheng Huang and Yongxin Ge

Received: 1 February 2024
Revised: 4 April 2024
Accepted: 9 April 2024
Published: 11 April 2024

1. Introduction

In a recent 2023 study by the World Health Organization (WHO), it was reported that approximately 466 million people are living with disabling hearing loss caused by various factors [1]. Experts estimate that by 2050, hearing loss will affect 700 million people worldwide, which is equivalent to one in ten individuals. A parallel report by JapanTrak [2] suggested that by 2022, 10 percent of Japan's population, 10 million people, will experience hearing loss. A survey conducted by the Committee for the Promotion of Hearing Loss Assistance Programs [2] revealed that about 70% of individuals with hearing loss have experienced job changes because of challenges in relationship building and requesting accommodations.

Advances in deep learning technology have been instrumental in propelling research aimed at enhancing communication with individuals who are deaf or having difficulty hearing. Over the past decade, this technology has seen significant advancements, with research in deep learning-based sign language recognition garnering substantial attention. Compared to traditional video recognition, sign language recognition poses unique challenges. The RGB-based recognition requires an extensive number of samples because of the minimal correlation between sign language movements and background, coupled with limited inter-frame variation. The perceivability of sign language movements is influenced by various factors, such as the signer's speed, body shape, rhythm, and performance location, underscoring the necessity for models exclusively focusing on sign language movements.

Recently, skeleton-based sign language recognition methods have received increasing attention owing to their adaptability in dynamic scenarios and against complex backgrounds. Researchers have recently shifted their focus to developing several skeleton-based approaches [3,4] that incorporate diverse types of feature data into coordinate values derived from pose estimation and categorize them using convolutional neural network (CNN) models.

We introduce a straightforward and accessible method for recognizing all 46 Japanese finger spellings. Our goal is to reduce computational complexity and perform recognition in real time by making input data two-dimensional through feature extraction. This method employs a visual transformer (ViT) [5] model, which has become a topic of interest in computer vision. The first step of our method involves using a posture estimation library, MediaPipe [6], to capture the coordinates of each finger joint from a video frame. These coordinates are then used to calculate one-dimensional angular feature data. We organize the angular feature data to create a two-dimensional input vector as the next step. This vector is essential for the recognition process, which is conducted using the ViT model. An important part of this process is adjusting the number of encoder layers in the ViT. This change allows us to assess which configuration yields the highest recognition accuracy and identify the optimal number of encoder layers. To show the effectiveness of our proposed method, we will compare its performance with that of existing CNN models. This comparison aims to showcase the potential improvements and benefits of using the ViT model for sign language recognition, especially in Japanese sign language.

2. Related Works

Researchers have devised various methods in sign language recognition, including physical methods such as the use of wearable devices [7]. In video-based sign language recognition, the two primary approaches are manual feature extraction and deep learning-based approaches. Manual feature extraction involves picking out and analyzing important features from video footage. Techniques like scale-invariant feature transform (SIFT) [8], histogram of oriented gradients (HOG), and optical flow are used to identify these key features. Once identified, machine learning algorithms like support vector machines (SVM), decision trees, or basic neural networks classify these features.

The deep learning-based approach uses advanced deep learning models, CNNs, and recurrent neural networks (RNNs) to identify and learn features from video data automatically. The models feed on the raw video and autonomously carry out feature extraction and classification. There are two subtypes in this method, continuous sign language recognition (CSLR) and isolated sign language recognition (ISLR). CSLR aims to recognize the flowing sign language from videos. It tackles the challenge of identifying and interpreting sign language in a continuous stream, where signs flow into each other. In CSLR, the study by Lianyu Hu et al. [9] focused on the correlation between frames.

In contrast, ISLR focuses on recognizing individual signs or short phrases that are shown separately. The most common approach in ISLR involves CNN-based models like I3D [10] and R3D [11], which classify sign language using RGB video as input. This is simpler than CSLR, as it does not involve interpreting transitions between signs. Researchers widely use CNN models, sometimes combined with RNNs, in ISLR. Chun Keat Tan et al. [12] also recently proposed an ISLR approach using the ViT classifying hand gesture images as input data. Their results showed accuracies of 99.98% on the American Sign Language (ASL) dataset, 99.36% on the ASL with the Digits dataset, and 99.85% on the National University of Singapore (NUS) hand gesture dataset.

Marcelo et al.'s study [13] explored four transformer-based approaches and four pre-training data regimes, investigating all combinations on the WLASL2000 dataset. The model known as MaskFeat achieved 79.02% accuracy in gloss-based WLASL2000, outperforming pose-based and supervised video models. OpenPose [14] and MediaPipe have led to the development of skeleton-based ISLR. A study by S. Jiang [3] proposed the skeleton-aware multimodal framework (SAM-SLR-v2), achieving 98% accuracy test the AUTSL

dataset. Improving hand pose estimation can significantly enhance the accuracy of sign language recognition.

Recent developments have witnessed emerging hybrid methodologies, amalgamating elements from both manual feature extraction and deep learning paradigms, endeavoring to harness the synergistic strengths of both approaches. Technological advancements in vision transformers (ViTs), OpenPose, and MediaPipe are progressively augmenting the efficacy of these approaches, particularly within the deep learning spectrum. Based on these findings, this study proposes a sign language recognition method using ViT and MediaPipe. While there is research on ASL and other foreign sign languages, studies on Japanese sign language are less advanced in comparison. In their study, Syosaku et al. [15] used OpenPose and MediaPipe to estimate posture and hand shape, and they constructed a vector to characterize sign language expressions. The researchers computed the similarity of sign language expressions for each part of the frame sequence and evaluated it by using characteristic actions as key images for retrieval. As a result, nine pairs were determined to be motion synonyms, seven of which had little semantic relevance and appeared infrequently, indicating that they were new motion synonyms. In their study, Miku et al. [16] used the leap motion controller which is a device to acquire hand skeleton data to estimate finger characters representing the Japanese sign language syllabary. The shape and motion of the fingers were detected, and the characters were estimated from the Euclidean distance. Their experiment revealed that the recognition rate for fixed finger letters was 78%, but it decreased to 75% when finger letters with motion were included. The researchers attribute the accuracy loss to the leap motion controller's inability to recognize overlapping fingers correctly. This study involves direct collaboration with sign language speakers to film 46 phonemes and uses these data for training in sign language recognition.

3. Datasets and Angular Features

In this section, we describe creating a Japanese finger-spelling dataset and the method for extracting angular features from this dataset to generate input vectors. This research ultimately aims at real-time operation. For this reason, the requirements for a posture estimation system are fast processing speed and high accuracy for a single person. In response to this, we used MediaPipe because it combines "fast end-to-end processing" and "high accuracy for a single person", and compared to other methods (OpenPose), we believe MediaPipe is the best system for this study.

3.1. Shooting Data of Japanese Sign Language

We created the Japanese sign language videos used in this study by inviting sign language speakers to record the finger-spelling actions. We present the details of the filming below.

- A total of 46 Japanese syllables and specified words were recorded.
- Five letters with the same consonant were filmed in sequence.
- The sign language speaker was photographed from the waist up in the frame.
- The recording was taken with the subject facing forward, and the left–right tilt was kept within 15 degrees.
- A green screen background was used for the filming.
- The images were taken without masks while the speaker was speaking.
- Filming was conducted in Full HD (1920 × 1080) quality at 60 frames per second, and the data were compiled in mp4 format.
- Each of the 46 different letters was photographed 6 times.

In this study, we categorized the filmed videos as spelling videos (Figure 1), which comprise a series of finger-spelling videos for the syllables 'あ|a|' to 'ん|ŋ|', and word videos, which depict several words in finger spelling. We invited one male and one female for filming. The filming team recorded three takes of each person for the spelling videos and one take for the word videos. We filmed the spelling videos in three different sessions. First time, participants used no special posture to enable natural fingerspelling movements,

second time, filming was conducted with sleeves rolled up to prevent overlapping of body and finger spelling, and third time, filming maintained the same position, without the need for rolled-up sleeves. We conducted the shooting for the word videos in the same posture used in the third session of the spelling videos. Table 1 shows the final composition of the videos. The table shows that the average value for the first five-character video of the woman is smaller than the other data. The reason for this is that the woman was told to "go as fast as usual" during the first shooting. After the second shooting, the value became larger because the female subject made a sign language movement a little slower for the shooting. The average value remained stable because the male subject had extensive experience in sign language and consistently signed at a constant speed.

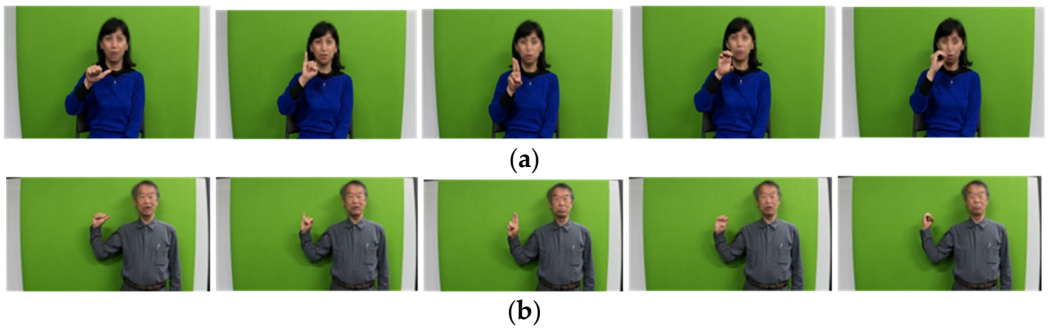

Figure 1. Example of a partially extracted image from the spelling videos: (**a**) images of a female signer expressing the Japanese characters from 'あ|a|' to 'お|o|' (first-time data); (**b**) image of a male signer expressing the Japanese characters from 'あ|a|' to 'お|o|' (third-time data).

Table 1. Information on shooting data.

	Video Type		Position	Rolled-Up Sleeves	Video Duration Average	
					Five Letters	Three Letters
Spelling videos	First-time data	Male	Seating	without	8 s	5 s
		Female	Seating	without	8.6 s	5 s
	Second-time data	Male	Standing	with	8.6 s	5.5 s
		Female	Standing	with	9.1 s	5 s
	Third-time data	Male	Standing	without	8.6 s	5 s
		Female	Standing	without	8.7 s	5 s
					Four letters	
Word videos	Word data	Male	Standing	without	6 s	
		Female	Standing	without	6 s	

The videos taken were manually split. The stage in which the hand shape of the finger character became clear was set as the beginning, and the stage just before the hand shape was broken was set as the end. For 'mo(も)', which is a finger character that changes the finger shape in the middle of the action, we used the finger shape from the beginning of the action to the end of the action.

3.2. Calculation of Angular Features

In this subsection, we explain the method of calculating the tilt of fingers and hands from video data into cosine angles and extracting them as feature vectors. Angle features were extracted from the Japanese finger-spelling video dataset through the following process:

1. MediaPipe was utilized to extract joint position coordinate data from each video data.

2. By utilizing the extracted coordinate data, we obtained 20 angles for each finger joint, as illustrated in Figure 2a, and 20 features that indicate the overall tilt of the hand, as displayed in Figure 2b, employing cosine values. The datasets in our experiments were classified into three types based on the various combinations of these angular features.
3. Since feature data is acquired frame by frame, the size of the input data varied with the length of the video. To address this, we applied an interpolation method, commonly used in object detection techniques, to standardize the size of the uneven input data extracted from the videos. These results generated a two-dimensional matrix with a uniform number of frames.
4. From this standardized two-dimensional matrix, we extracted multiple random frames, which reduced the frame count and created a smaller two-dimensional matrix. This process, when repeated, allowed us to augment the data, resulting in 324 data points per finger-spelling character.

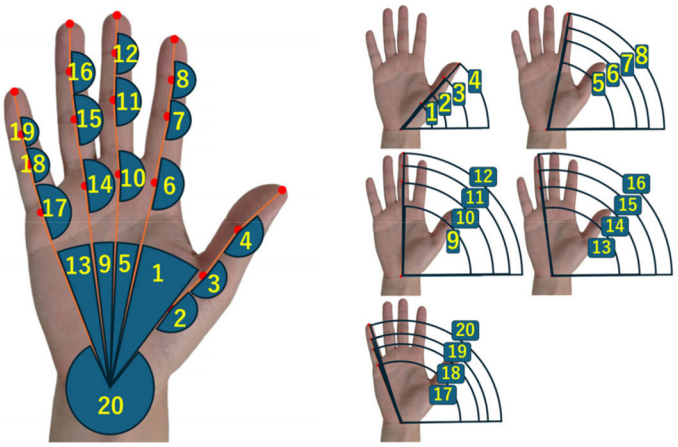

Figure 2. Angular features extracted from finger joint and wrist coordinates: (**a**) the 20 dimensions of angles with each finger joint; (**b**) the 20 dimensions of angles with the overall tilt of the hand.

Two types of angular features were used in this study. The first type focuses on three points centered on the joint of interest and determines the angle of each finger joint, as shown in Figure 2a. The second type evaluates the overall tilt of the hand, as shown in Figure 2b. As shown in Figure 2b, the angle was determined using a reference vector and the tilt from it. In both cases, we calculated the vector from the coordinates of each joint to the adjacent joint and computed its cosine value to measure the opening of each joint. In Figure 3 and Equation (1), we can see the calculation method and the positional relationship of the coordinates. To obtain the angle shown in red in Figure 3, the coordinates of the three points obtained by MediaPipe are (x1, y1) (x2, y2) (x3, y3), as shown in the figure. (xn, yn) indicate the x-coordinate and y-coordinate of each joint obtained by Mediapipe, respectively. These values are substituted into Equation (1) to calculate the cosine value. The overall tilt of the hand depicted in Figure 2b was calculated using a reference vector parallel to the x-axis and moving outward from the wrist and a vector from the wrist to each finger joint. These calculations resulted in 20 features with each finger joint and 20 features with the overall tilt of the hand.

$$cos\theta = \frac{(x2-x1)(x3-x2)+(y2-y1)(y3-y2)}{\sqrt{(x2-x1)^2+(y2-y1)^2}\sqrt{(x3-x2)^2+(y3-y2)^2}} \quad (1)$$

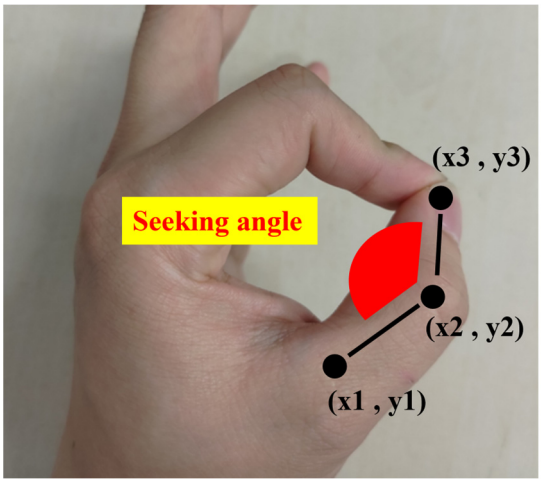

Figure 3. Relationship between each coordinate location and angle location for angle calculation performed for feature extraction.

3.3. Unification of Input Data Size

When the calculated angle features are extracted from consecutive frames, they form a two-dimensional array. As the length of the video affects the length of this two-dimensional array, interpolation methods are used in this study to standardize the input data size. The ROIAlign (region of interest align) method [17], which is used in object recognition, is applied to interpolate the targeted area (region of interest) of the hand more accurately. By employing this approach, videos of different durations can be transformed into sequences with fixed frames, and it is also possible to standardize the size of the input 2D matrix data.

In our experiments, videos of approximately 1 to 2 s were extended to 100 frames using the ROIAlign method. From these 100 frames, a fixed number of frames was randomly extracted while maintaining intervals within 10 frames. As a result, we obtained 54 time-sequential data points for each class. The video data, comprising six videos with three recordings each from both male and female participants, resulted in 54 time-sequential data points per video, amounting to 324 feature vectors. With 46 characters, this led to a total of 14,904 feature vectors.

The feature vectors obtain values ranging from -1 to 1 for calculations using the cosine formula. However, as the ViT is originally a model suited for image-based learning, these values were converted to a pixel-value range of 0 to 255 using the following Equation (2):

$$x_{new} = (x_{old} + 1) \times 125 \quad (2)$$

Based on the calculations of angular features, three datasets were created. These three datasets differ in the features they use, as shown in Table 2. Dataset 1 focused solely on the joint angles (Feature 1), comprising 20 dimensions. Dataset 2 concentrated on the overall tilt of the hand (Feature 2), also encompassing 20 dimensions. Finally, Dataset 3 combined Features 1 and 2, resulting in a total of 40 dimensions.

Table 2. Three datasets by angular feature combination.

Dataset	Used Angular Feature	Dimension
Dataset 1	The angles of each finger joint (Feature 1)	20
Dataset 2	The overall tilts of the hand (Feature 2)	20
Dataset 3	Feature 1 and Feature 2 combined	40

4. Structure of the ViT and CNN Models

The CNN used in this study retains and learns location information. The ViT learns by adding location information. Based on this, we thought it would be possible to perform

3D recognition using a 2D recognition model by converting each frame of the original 3D data into a 1D feature vector and then converting this into 2D data on the time axis. The ViT model was investigated by changing the number of Encoder layers, and data with various dimensions were input to the model with the highest accuracy. In contrast, the CNN model's internal structure changes by changing the size of the input data, so ablation studies were conducted with three types of data input.

The ViT model, based on the transformer's core architecture [5], uses an encoder featuring a multihead self-attention mechanism and fully connected neural networks. The addition of more encoder layers in the ViT amplifies the model's depth, enabling it to perform more intricate feature extraction and facilitating the learning of hierarchical and abstract representations, although this may lead to a higher chance of overfitting. In this study, we investigated the recognition performance for 46 classes of Japanese finger spelling by varying the number of encoder layers in the ViT model.

The ViT model's input comprises the angle features extracted from each frame, fixed in number, and transformed into two-dimensional matrix vectors. Unlike conventional ViT models that involved dividing an image into patches, in our study, as depicted in Figure 4a, we treat each frame as a single vector. Then, a linear projection of the angular features is performed upon data input, and position information is assigned to each vector using embedding by Keras. Once the data are input, they are processed with a transformer encoder for training. The transformer encoder comprises four layers: a multihead attention layer, an MLP layer, and a layer normalization layer in front of each layer. The encoder iterates the process 4 to 9 times, and finally, its output is passed to the MLP head for classification into 46 classes. This approach facilitates the input of sequential frames into the transformer's encoder with added positional encoding.

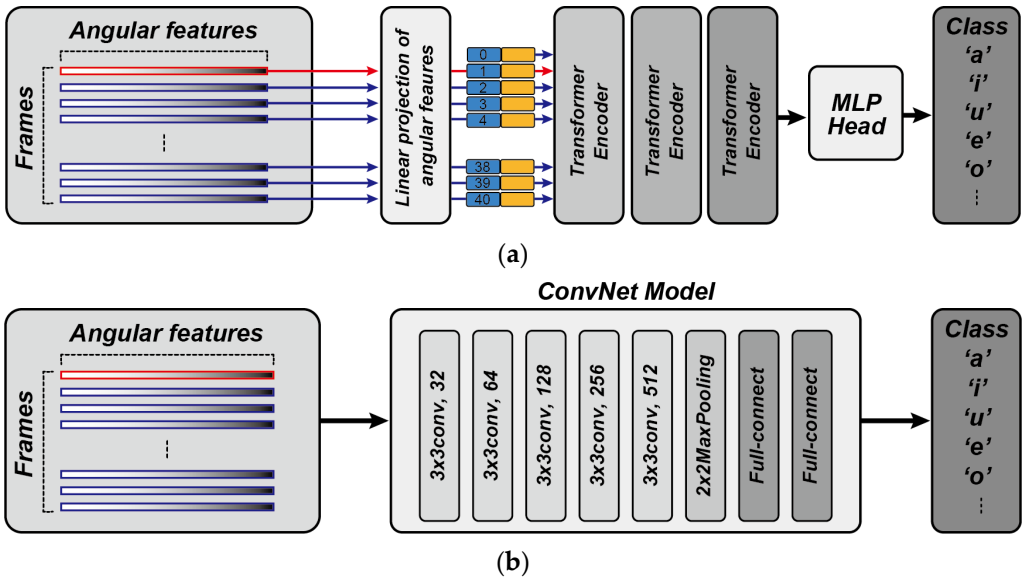

Figure 4. The structure of the ViT model and CNN model: (**a**) example of our ViT model with 3 encoders; (**b**) example of our CNN model with 8 layers.

Our method employs a multilayer perceptron (MLP) with two fully connected layers, two dropout layers, and the Gaussian error linear unit (GELU) function, combined with residual connections. The encoder structure, comprising a normalization layer, a multihead attention layer, and an MLP layer, is repeated for the specified number of encoders, ending with an additional fully connected layer for classification into 46 classes.

The learning rate was initially set at 0.001 and was halved every five epochs. For this implementation of the ViT, we adapted the model to our unique dataset, which comprises either 20 or 40 feature data × 20 frames, by omitting the original ViT's patch-splitting component. This modification ensured the effective use of the dataset's unique character-

istics in our experiment. We employed the ViT model to assess the impact of varying the number of encoders on accuracy. The performance of Japanese sign language recognition was verified by varying the encoder layer by one layer, from 4 to 9 layers. Additionally, four multi-head self-attention mechanisms were included in one encoder.

Meanwhile, as shown in Figure 4b, we developed a CNN model based on the VGG16 framework and evaluated its performance for the ViT. This CNN was designed with eight layers, half of the original VGG16, to handle input frames with angular features. These frames were processed through a series of convolutional layers and then through a maximum pooling layer and a fully connected layer to classify the output.

The CNN model comprises multiple "3 × 3 convolutional" layers, each with a specified number of filters, such as 32, 64, 128, 256, and 512. These layers extract features from the input by applying filters that identify spatial patterns. Afterward, a "2 × 2 MaxPooling" layer is then used to reduce the spatial size of the output, which reduces computation and helps prevent overfitting. The activation function also uses Relu functions. Although not shown in the figure, the model uses regularization techniques, such as dropout and weight decay. With each layer, the depth of the model and the number of filters increase, which is a typical approach for detecting more complex features.

5. Experimental Results

In the experiment, accuracy verification was conducted using the angular features calculated with MediaPipe.

5.1. Verification of VIT Performance for Different Numbers of Encoder Layers

The dataset encompassed video recordings from six distinct finger-spelling videos, with an equal division of three videos, each contributed by male and female participants. Each video was analyzed to extract 54 sequential data points, resulting in a total of 324 feature vectors, arranged in a two-dimensional matrix for each Japanese character. Across the dataset, 46 unique characters were represented, culminating in an aggregate of 14,904 feature vectors.

Of these, 75% were used as training data and 25% as validation data. The training was conducted over 45 epochs, with the loss and accuracy values recorded for each epoch. The dataset used in this process was Dataset 2. The ViT's performance results as a function of the number of layers in the encoder are shown in Figure 5. As shown in Figure 5, the highest accuracy was achieved with four layers, and a decline in accuracy was observed with each additional layer beyond five. The model was trained for 45 epochs, achieving high accuracy by epoch 15. No significant improvement was observed beyond epoch 15, suggesting overfitting. However, there was an increase in accuracy up to about the 20th epoch. In terms of processing time, the ViT model took 100 s per epoch with four layers, and 317 s per epoch with ten layers.

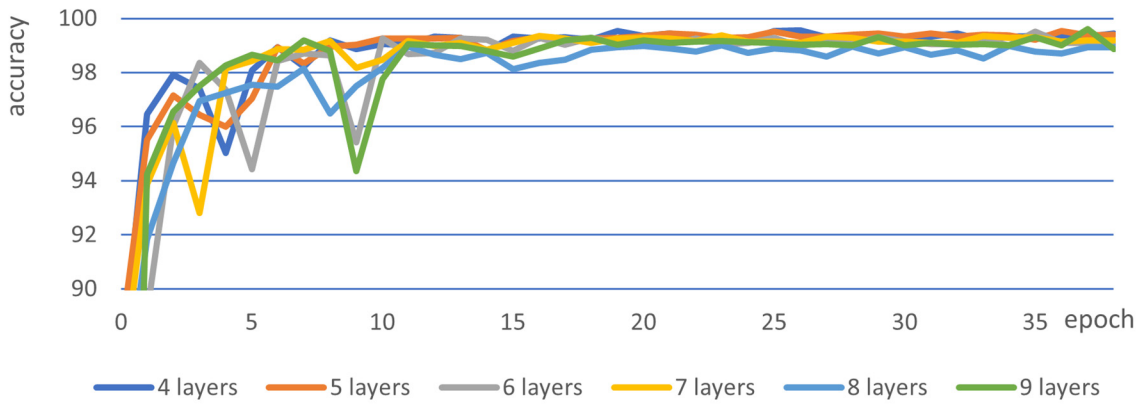

Figure 5. The results of ViT's performance as a function of the number of layers in the encoder.

5.2. Experiments Comparing the Accuracy of CNN and ViT

We conducted an experiment comparing the accuracy of a model with four encoders in the ViT and a CNN based on VGG16 across three datasets: Dataset 1, Dataset 2, and Dataset 3. We present the results in Figure 6. In the comparison between the ViT and CNN, data from Dataset 1 resulted in high accuracy at epoch 25, data from Dataset 2 and Dataset 3 resulted in high accuracy at epoch 15, and no significant improvement was observed even after training up to epoch 45. For the CNN model, the highest accuracy was obtained with Dataset 3, recording an accuracy of 99.6%. The datasets following this were Dataset 1 with 96.7% accuracy and Dataset 2 with 99.3% accuracy. Additionally, the training times for each method were as follows: For the ViT model, it was 37 s per epoch for Dataset 1, 37 s for Dataset 2, and 101 s for Dataset 3. For the CNN model, it took 4 s per epoch for Dataset 1, 4 s for Dataset 2, and 5 s for Dataset 3. Table 3 summarizes the recognition results compared to CNN when the ViT's encoder was four layers.

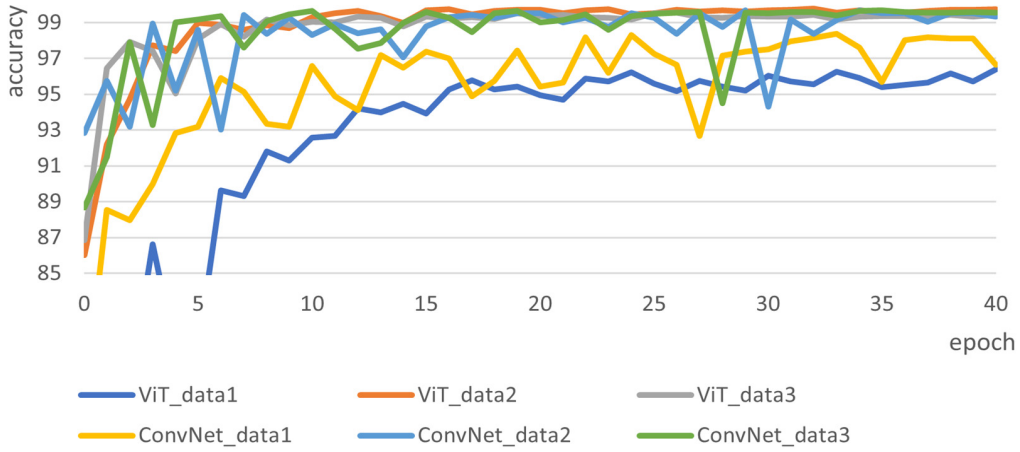

Figure 6. Comparison of recognition performance of ViT and CNN by dataset.

Table 3. The recognition results compared to ViT and CNN.

Dataset	ViT with 4 Encoder Layers	CNN with 8 Convolutional Layers
Dataset 1	96.4%	96.7%
Dataset 2	99.7%	99.3%
Dataset 3	99.4%	99.6%

5.3. Validation of Additional Learning and Real-Time Recognition

The next step involved conducting further experimentation with real-time Japanese sign language recognition using both ViT and CNN models. During the test, the researcher faced forward to the camera and executed sign language movements. The performed movements were eight types of 'あ|a|' to 'お|o|', and 'り|ri|', 'も|mo|', and 'ん|ŋ|'. As a result, the CNN model recognized the finger letters 'あ|a|' to 'お|o|' with very high accuracy. The high recognition rate of these characters is because of their stationary expression with no accompanying finger movements. However, the three types of 'り|ri|', 'も|mo|', and 'ん|ŋ|' were not recognized accurately due to the inclusion of finger movements. In particular, there was a strong tendency to recognize them as different characters at the beginning of the movement and at the end of the finger movement. The ViT model did not show significant differences compared to the CNN model in real-time recognition.

Next, we explain the accuracy of recognizing words composed of a sequence of finger letters. The video we tested uses the *word video* in Table 1, the words are "いいやま|i-i-ya-ma|" and "のりもの|no-ri-mo-no|". Both words consist of four letters. The CNN model correctly recognized 'い|i|', 'や|ya|', and 'ま|ma|', with few false positives when transitioning to the next character. On the other hand, there was misrecognition in the characters

'の|no|', 'り|ri|', and 'も|mo|' which included actions. The characters 'の|no|', 'り|ri|', and 'も|mo|' were frequently misrecognized as 'え|e|' or 'た|ta|', 'し|si|' or 'る|ru|', and 'な|na|' or 'ほ|ho|', respectively. As shown in Figure 7, the percentages of characters recognized as matching word characters for the number of frames in the video were 60.4% for "いいやま|i-i-ya-ma|" and 38.6% for "のりもの|no-ri-mo-no|" in the CNN model, and 24.4% for "いいやま|i-i-ya-ma|" and 27.5% for "のりもの|no-ri-mo-no|" in the ViT model, showing no difference in accuracy between the video without motion and the video with motion. Thus, ViT resulted in a small difference in accuracy between words with and without actions, while CNN resulted in a larger difference in accuracy between words with and without actions.

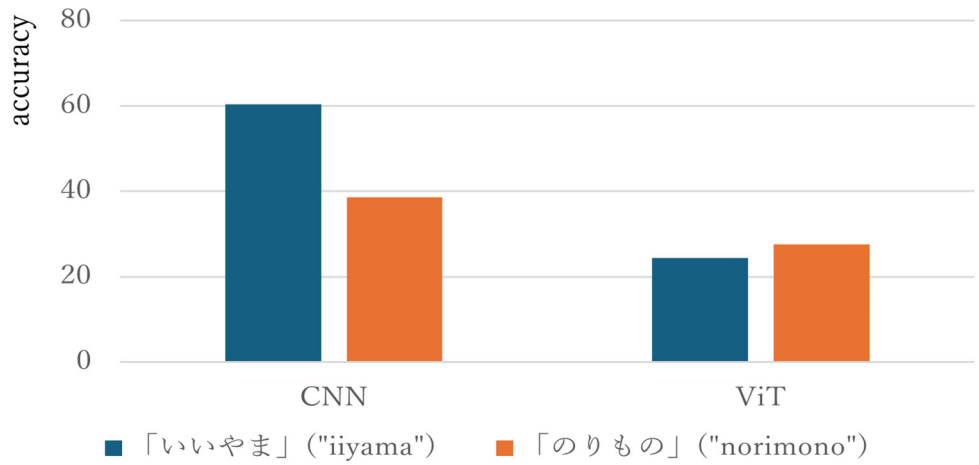

Figure 7. Comparison of ViT and CNN recognition performance on sign languages with and without movement.

6. Conclusions

In this paper, we applied a ViT to a relatively small dataset of 20 × 40 dimensions. As discussed in Section 5, The ViT model achieves its best accuracy with four encoder layers experiences a decrease in accuracy as the number of layers increases beyond four. Particularly, accuracy significantly declined when the number of encoder layers exceeded ten, suggesting that a smaller dataset, like the one used in this experiment, limits the number of features. Accuracy did not significantly change from three to four encoder layers, only seeing subtle improvement. Future research should increase the dataset. Therefore, when creating additional datasets, it is necessary to increase the amount of participants who speak sign language and capture videos of finger spelling movements from various angles.

For finger spelling without actions, both the ViT and CNN models were able to achieve high accuracy with a small amount of training. However, for finger spelling with actions, the accuracy was poor. This is thought to be because the finger shape in finger spelling with actions is very similar to that of finger spelling without actions due to the narrowing of the number of features and the number of frames. For example, the finger shape of 'no(の)' is very similar to that of 'hi(ひ)' at the beginning of the action and 'so(そ)' at the end of the action, and the ViT model misidentified these two finger shapes.

Therefore, in future research, the number of extracted frames and angular features should be increased, and postures and facial expressions should also be included as features to improve the accuracy in real time. Based on the results of this study, the ViT with 20-dimensional data (Dataset 2) showed the highest accuracy. In real-time recognition, the accuracy was lower than that of the CNN model, but the accuracy of each character was significantly different, suggesting that accuracy can be greatly improved by studying the learning method. Therefore, we believe that accuracy can be improved by increasing the number of frames, the number of angular features for each joint of the upper body us-

ing MediaPipe's Holistic, and the number of features for the entire mouth and eye area of the face.

Finally, we discuss some issues for future research. In this experiment, the dataset was small because there were only two people who used sign language on a regular basis that were available. Therefore, we would like to expand the dataset by increasing the number of participants in future experiments.

Author Contributions: Conceptualization, D.S. and Y.K.; Methodology, Y.K.; Validation, Y.K.; Formal analysis, S.N.; Data curation, Z.H.; Writing—original draft, T.K. and Y.K.; Writing—review & editing, D.S.; Visualization, T.K.; Supervision, Y.K.; Project administration, Y.K.; Funding acquisition, Y.K. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by Co-G.E.I. (Cooperative Good Educational Innovation) Challenge 2023 of Tokyo Polytechnic University.

Institutional Review Board Statement: The study was conducted in accordance with the Declaration of Helsinki, and approved by the Ethics Committee of Tokyo Polytechnic University (2022-01).

Informed Consent Statement: Informed consent was obtained from all subjects involved in the study.

Data Availability Statement: The original contributions presented in the study are included in the article, further inquiries can be directed to the corresponding authors.

Acknowledgments: We would like to express our sincere gratitude to the two signers who helped us create the Japanese sign language database.

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

References

1. World Health Organization. *Safe Listening Devices and Systems: A WHO-ITU Standard*; World Health Organization: Geneva, Switzerland, 2019; p. 10.
2. Japan Hearing Instruments Manufacturers Association. *JapanTrak 2022*; JHIMA: Tokyo, Japan, 2022; p. 99.
3. Jiang, S.; Sun, B.; Wang, L.; Bai, Y.; Li, K.; Fu, Y. Skeleton Aware Multi-Modal Sign Language Recognition. In Proceedings of the 2021 Conference on Computer Vision and Pattern Recognition (CVPR), Nashville, TN, USA, 19–25 June 2021.
4. Hezhen, H.; Wengang, Z.; Houqiang, L. Hand-Model-Aware Sign Language Recognition. In Proceedings of the 35th AAAI Conference on Artificial Intelligence, Virtual, 2–9 February 2021; Volume 35, pp. 1558–1566.
5. 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 16×16 Words: Transformers for Image Recognition at Scale. *arXiv* **2020**, arXiv:2010.11929.
6. Lugaresi, C.; Tang, J.; Nash, H.; McClanahan, C.; Uboweja, E.; Hays, M.; Zhang, F.; Chang, C.; Yong, M.; Lee, J.; et al. MediaPipe: A Framework for Building Perception Pipelines. In Proceedings of the Third Workshop on Computer Vision for AR/VR, Long Beach, CA, USA, 16–20 June 2019.
7. Ambar, R.; Fai, C.K.; Wahab, M.H.A.; Jamil, M.M.A.; Ma'radzi, A.A. Development of a Wearable Device for Sign Language Recognition. *J. Phys. Conf. Ser.* **2018**, *1019*, 012017. [CrossRef]
8. Ma, L.; Huang, W. A Static Hand Gesture Recognition Method Based on the Depth Information. In Proceedings of the 8th International Conference on Intelligent Human-Machine Systems and Cybernetics (IHMSC), Hangzhou, China, 27–28 August 2016.
9. Lianyu, H.; Liqing, G.; Zekang, L.; Wei, F. Continuous Sign Language Recognition with Correlation Network. In Proceedings of the 2023 Conference on Computer Vision and Pattern Recognition (CVPR), Vancouver, BC, Canada, 20–22 June 2023; pp. 2529–2539.
10. Carreira, J.; Zwsserman, A. Quo Vadis, Action Recognition? A New Model and the Kinetics Dataset. In Proceedings of the 2017 Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 22–25 July 2017.
11. Kensho, H.; Hirokatsu, K.; Yutaka, S. Learning spatio-temporal features with 3D residual networks for action recognition. In Proceedings of the 2017 IEEE International Conference on Computer Vision Workshops (ICCVW), Venice, Italy, 22–29 October 2017.
12. Chun, K.T.; Kian, M.L.; Roy, K.Y.C.; Chin, P.L.; Ali, A. HGR-ViT: Hand Gesture Recognition with Vision Transformer. *Sensors* **2023**, *23*, 5555. [CrossRef] [PubMed]
13. Marcelo, S.-C.; Yanhong, L.; Diane, B.; Karen, L.; Gregory, S. Self-Supervised Video Transformers for Isolated Sign Language Recognition. In Proceedings of the 2024 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV), Waikoloa, HI, USA, 4–8 January 2024.
14. Cao, Z.; Simon, T.; Wei, S.-E.; Sheikh, Y. Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 22–25 July 2017; No. 121; pp. 1302–1310.

15. Syosaku, T.; Kyuhei, H.; Zacharie, M. A Simple Method to Identify Similar Words with Respect to Motion in Sign Language Using Human Pose and Hand Estimations. *Forum Inf. Technol.* **2022**, *21*, 175–176.
16. Miku, K.; Atsushi, T. Implementation and Evaluation of Sign Language Recognition by using Leap Motion Controller. *IPSJ Tohoku Branch SIG Tech. Rep.* **2015**, *2015-ARC-8*, 1–6.
17. Kaiming, H.; Gkioxari, G.; Dollár, P.; Girshick, R. Mask R-CNN. In Proceedings of the 2017 International Conference on Computer Vision, Venice, Italy, 22–29 October 2017.