Computer Systems Science & Engineering
DOI: 10.32604/csse.2023.029336
Article

Tech Science Press

Rotation, Translation and Scale Invariant Sign Word Recognition Using Deep Learning

Abu Saleh Musa Miah[1], Jungpil Shin[1,*], Md. Al Mehedi Hasan[1], Md Abdur Rahim[2] and Yuichi Okuyama[1]

[1]School of Computer Science and Engineering, The University of Aizu, Aizuwakamatsu, Fukushima, 965-8580, Japan
[2]Department of Computer Science and Engineering, Pabna University of Science and Technology, Pabna, Bangladesh
*Corresponding Author: Jungpil Shin. Email: jpshin@u-aizu.ac.jp
Received: 01 March 2022; Accepted: 07 April 2022

Abstract: Communication between people with disabilities and people who do not understand sign language is a growing social need and can be a tedious task. One of the main functions of sign language is to communicate with each other through hand gestures. Recognition of hand gestures has become an important challenge for the recognition of sign language. There are many existing models that can produce a good accuracy, but if the model test with rotated or translated images, they may face some difficulties to make good performance accuracy. To resolve these challenges of hand gesture recognition, we proposed a Rotation, Translation and Scale-invariant sign word recognition system using a convolutional neural network (CNN). We have followed three steps in our work: rotated, translated and scaled (RTS) version dataset generation, gesture segmentation, and sign word classification. Firstly, we have enlarged a benchmark dataset of 20 sign words by making different amounts of Rotation, Translation and Scale of the original images to create the RTS version dataset. Then we have applied the gesture segmentation technique. The segmentation consists of three levels, i) Otsu Thresholding with YCbCr, ii) Morphological analysis: dilation through opening morphology and iii) Watershed algorithm. Finally, our designed CNN model has been trained to classify the hand gesture as well as the sign word. Our model has been evaluated using the twenty sign word dataset, five sign word dataset and the RTS version of these datasets. We achieved 99.30% accuracy from the twenty sign word dataset evaluation, 99.10% accuracy from the RTS version of the twenty sign word evolution, 100% accuracy from the five sign word dataset evaluation, and 98.00% accuracy from the RTS version five sign word dataset evolution. Furthermore, the influence of our model exists in competitive results with state-of-the-art methods in sign word recognition.

Keywords: Sign word recognition; convolution neural network (cnn); rotation translation and scaling (rts); otsu segmentation

This work is licensed under a Creative Commons Attribution 4.0 International License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

1 Introduction

Sign language is a nonverbal form of communication for the deaf and hard-of-hearing community. It is essential to help real-time communication among the deaf community, hard of hearing, and speech difficulties without the aid of an interpreter. It can be used in various research fields, including human-robot interaction, computer games, learning, sign word and language recognition, virtual reality, medical diagnostics, and autism analysis [1–4]. In this case, they use body movements parallel with arm, body, head, finger, hand movements, and facial expressions to communicate with humans. In most cases, hand gestures are essential for human-computer interaction and practical application for sign language recognition (SLR), used to establish a communication system among the deaf-mute community [5]. The sign language recognition research field has attracted researchers' attention from the last century. The importance of sign language recognition has increased because of the high growth rate of the deaf and hard-of-hearing population globally and the extended use of vision-based application devices [6,7]. In recent years, many researchers have proposed vision-based sign language recognition by utilizing inputs of the camera, such as 3d camera, web camera and stereo camera [8].

The main cause of the vision-based approach's attractiveness is: it does not need any specialized device, and this method is affordable. At the same time, the sensor-based system needs different specialized devices such as power gloves, accelerometer, kinetic sensor, leap motion controller, and huge wire [9]. However, vision-based systems like SLR become difficult due to the complex background, uncontrolled environment, visual analysis of gesture, lighting illumination, finger occlusion, inter-class variation, constant fatigue, the similarity of the high intraclass, and complexities of different signs. Many researchers have utilized the segmentation approach considering the above problems of vision-based systems [4,10]. In recent years, some researchers employed deep learning-based models used to overcome the above issues, although the redundant background may not be suitable for CNN to train accurately [11]. Apart from the image background issue, translated, rotated, scaled version test image is another important challenge for vision-based systems like SLR. In the real-world application of the SLR system, when a deaf or hard of hearing person tries to make a new sign word to test the system, the test image could be taken from the upper, lower, left, and right-hand positions because of unawareness signer. So, the test image may be rotated, translated or scaled due to an inappropriate hand position. Some researchers have utilized techniques to deal with the test image positions problem [12,13]. Moreover, the existing SLR system may fail to produce correct output for rotated, translated, scaled (RTS) variant test images. The mentioned challenges inspired us to design a segmentation method with RTS invariant deep learning model to solve current problems.

In the study, we have proposed a combined segmentation approach along with an RTS invariant CNN to solve the above problems. For this, the image is collected from the webcam, and the RTS technique is applied to the input image; then, a combined segmentation approach including Otsu [14], YCbCr [15], morphology [16], the watershed [17] is applied on the RTS variant sign language word image. Then a deep learning-based CNN model is used to extract the feature from the segmented image and recognize the sign word based on the collected feature. RTS process is used here to enlarge the datasets with different views, shapes, position images to make it more effective and reduce the potential overfitting of the proposed method. Moreover, RTS techniques are one of the essential techniques for solving the challenges that artificial intelligence faces while capturing the test image for a sign word recognition, especially for overcoming RTS variant test dataset problems. Despite the RTS, there may be an alternative solution if the camera detects the hand position and then normalizes the camera's position concerning the hand to remove the rotation, translation, and scale. Unfortunately, we did not find such a camera yet. However, the RTS technique may be the only solution to the given problem and is very effective for image classification and sign language recognition.

To achieve the goal, the significant contribution of the paper is as follows:

- Firstly, we have developed a Rotated, Translated, and Scaled (RTS) version of the dataset to solve the RTS issue in the test or future images.
- Secondly, we have designed a hybrid segmentation approach to reduce the redundant background of the images by combining Otsu Thresholding with YCbCr, Morphological analysis, and Watershed algorithm.
- Finally, we propose a novel convolutional neural network (CNN) architecture model for extracting features and classifying sign words.

We have organized the paper as follows, relevant research or literature review discussed in Section 2. We described the dataset in Section 3 and discussed the proposed model in Section 4. Experimental results and a brief discussion of the paper we have described in Section 5, and finally, the paper's conclusion we described in Section 6.

2 Literature Review

Over the past few decades, scientists in research activities have opened the door to acquiring sign language knowledge by promoting inspiration and excitement by introducing new methods. In most cases, scientists have used wearable and non-wearable devices for SLR performance. Pradeep Kumar et al. employed a multimodal framework for recognized sign language based on sensor devices [18]. They used the Microsoft Kinect device to collect data on the finger and palm. They used 75,00 Indian sign language images with 50 different symbols to evaluate their model and achieved 97.85% for the hidden markov model (HMM) and 94.55% for bidirectional long short-term memory neural networks (BLSTM-NN). However, they did not explicitly explain the types of their features. Wu et al. introduced an American sign language (ASL) system based on internal and electromyography (EMG) sensors [19]. They extracted features using fusion data, and as a classifier, they used a selected classification model. For validity, they used 80 signs followed by 96.16% for inter-class appropriateness and 85.24% for cross-session evolution. A framework for automatic Chinese SLR at the component level was developed in [20]. They achieved 96% accuracy after extracting hand size, orientation, and movement characteristics from images of EMG sensors. Tubaiz et al. employed a modified K-Nearest neighbor (MKNN) to develop an Arabic sign language recognition system [21]. They collected 40 sentences using an 80-lexicon word and achieved 98.9% accuracy for their model. Rahim et al. developed a hand gesture recognition system based on the Kinect sensor for human-computer interaction (HCI) [22]. They have achieved an average accuracy of 96% in their model based on the theme people communicate verbally without touching the device. Hu et al. used the k-means cluster-based depth segmentation and feature extraction method to develop an ASL recognition system [23]. They achieved 98.49% accuracy after using support vector machine (SVM) as classified with 120, 000 image training for 24 alphabets.

Starner et al. proposed two HMM-based systems based on a single camera image for ASL recognition [24]. They used a wearable device with a sensor, processing, and display module. The main goal of the work is to convert the recognized alphabet to sign-in text on android mobile phones via wireless transmission, and they achieved 98.2% accuracy in their final version model. Nevertheless, wearable technologies are not comfortable for use in daily life activities. This should be progressed because existing smart wearable hand devices have some drawbacks, such as a predefined set of standards and the use of bluetooth modules that only provide short-distance communication. Furthermore, the researchers used smart/depth devices according to different users for the long-range performance of wireless devices. In the same way, Lee BG et al. also applied the wearable device using sensor fusion [25]. Shin et al. proposed a hand tapping gesture system for Japanese hiragana and English characters based on kinetic sensors [26]. The

main concern is that the user's hand gestures on the input characters need to be remembered, requiring considerable computation time. Chong et al. developed an ASL recognition system using a palm-sized Leap motion sensor that is more portable less price compared to the existing solution [27]. They have a combination of 26 letters and ten digits with a recognition rate of 72.79% and 88.79%, respectively. Pisharady et al. compared the red, green, blue (RGB), and RGB-depth images in the field of SLR based on the quantitative and qualitative algorithms [28]. They point out that the model's performance depends on these image measures. They indicate that the performance of the model measures these images. However, the performing tasks need to be memorized in wireless technology, which, if forgotten, will turn into failure, and it will take more time to perform.

Recently, a deep learning strategy has shown outstanding performance in various departments to recognize sign language. Jain et al. applied an SVM and CNN to recognize ASL [29]. After evaluating the model, they calculated single and double optimal convolution filters and achieved 98.58% accuracy. A double-channel convolutional neural network (DC-CNN) was proposed to improve the recognition performance accuracy of the SLR system [30]. However, its hierarchy and scale features need to be improved, and there is much more room for improvement in dynamic gestures. Chevtchenko et al. employed a real-time SLR system by combining a traditional feature extractor with a CNN [31]. They used binary, depth, grayscale images to evaluate their model and sign gesture for real-time recognition. Agrawal et al. have completed a survey to summarize the development of the Indian sign language recognition system and provided a progress report highlighting the field [32]. Sign word recognition based on a CNN and SVM model was proposed in [4]. They segmented their input images using YCbCr and Skinmask segmentation techniques, then calculated feature vectors using CNN and classified their data using multi-kernel SVMs. However, the performance accuracy may be changed depending on the background and RTS variant of the image. To overcome the demonstrated challenges, we propose a combined segmentation approach for solving complex background problems and train the model with the Rotation, Translation and Scale (RTS) version of the training dataset to deal with the RTS variant dependency situation.

3 Dataset Description

This paper uses two benchmark datasets of the Sign language word to evaluate the proposed Sign word recognition model. Then, the RTS version of those two datasets is generated and used to evaluate the proposed model. The Twenty sign word dataset is described in Section 3.1; Section 3.2 describes the Five sign word dataset, and Section 3.3 describes the proposed RTS version for both datasets.

3.1 Twenty Sign Word Dataset [4]

Although the name of the dataset is not given in the original paper, we have given this name to differentiate between the two datasets used in this paper. The twenty sign word dataset consists of twenty static hand gestures, including nine double hand gestures and 11 single-hand gestures. This dataset contains 900 images for each class with 200×200 resolution and 18,000 images for 20 classes. Fig. 1 shows the example of input images per class.

3.2 Five Sign Word Dataset [33]

This dataset contains 216 images for training and 15 images for the test as five sign words [33]. Then they have divided the dataset training dataset into five different sets for five sign words. Each training set of sign words consists of 42 images for each class. They labelled each class, using the You only look once (YOLO) labelling format with a labelling tool. They have used the YOLO format, which labels the data into a text file format and holds the information about the dataset. We have used the five-sign word dataset to evaluate our model in our work.

Figure 1: Example of the input images per class [4]

3.3 Rotated, Translated and Scaled (RTS) Version Dataset

We have created the RTS version for both datasets described in Sections 3.1 and 3.2 to overcome the RTS variant image dependency of the model. Also, a large amount of labelled data is convenient for the proposed method to train accurately. However, the rotation, translation, and scaling techniques are applied here to increase the number of images, including different views, shapes and positions from the existing dataset [34]. We have created the RTS version of the Twenty sign word and Five sign word datasets by using the following techniques:

In the Rotation technique, we randomly applied −30 to +30 degrees to rotate the image. Fig. 2 shows the Rotated images for −30 degree, 15 degrees and 30-degree angles. In the Translation technique, we used x-axis and y-axis translation randomly. We have selected a range between −40 to +40 pixels for translation which is 20% of the original images. We considered that the image could randomly translate a maximum of 40 pixels in the right x-axis and upper y-axis and the same as −40 pixels in the left x-axis and down the y-axis. From the experiment, it is observed that if we translate more than 20%, the image loses its original meaning. Fig. 2 shows the Translated images as sliding by the same distance in the x-axis and y-axis direction, two images for translated only x-axis direction and one image for translated only y-axis direction. In the Scaling technique, we have modified both directions of the object in the

image to reduce and magnify the sign image. We have randomly selected scaling factor ranges between 0.2 to 1.5. We have considered low scaling factor is less than 1 (factor < 1), and the rest is considered as the high scaling factor. We applied zero padding when the scaling factor is low (factor < 1) to make the output image the same size as our input image and clip out (crop) when the scaling factor is high (factor > 1) to make the output image as the same size of the input image.

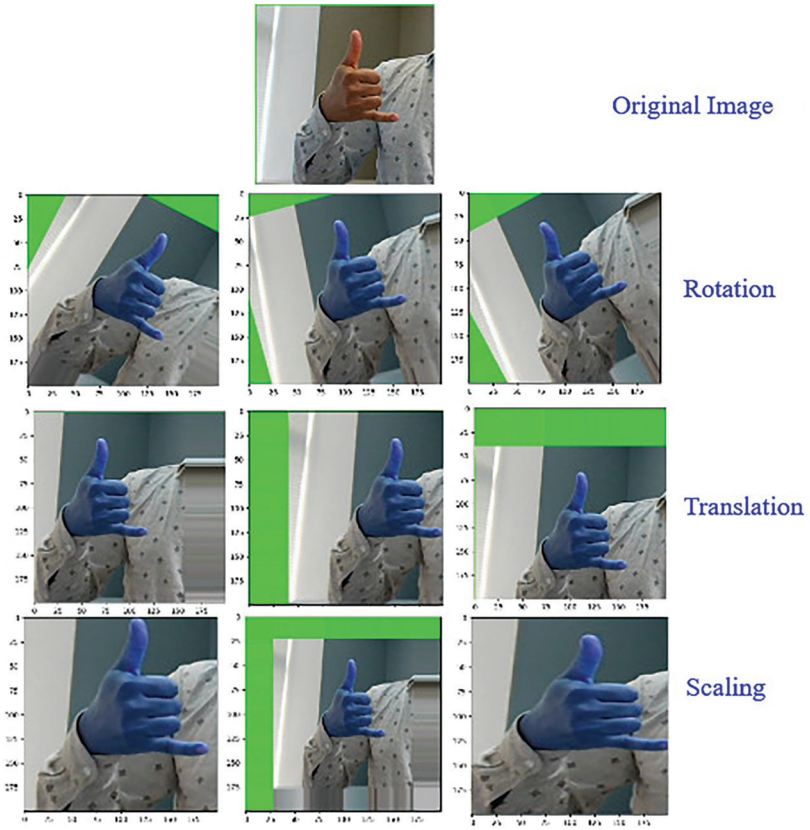

Figure 2: RTS process of an input image

Fig. 2 depicts the RTS process of an input image. Twenty sign word dataset contains 18,000 images; firstly, we have collected more than 50 images from 3 people then added them with each class, and the size of the new dataset is contained 19,000 images. We have generated 10 RTS version images from each of the original images using the RTS technique. However, instead of 19,000 images, a total of 190,000 images are found, whereas ten images are found from each image (1 original, 3 Rotated, 3 Translated, and 3 Scaled). In the same way, We have applied the RTS technique on the Five sign word dataset and found 2,160 images as RTS version training images from 216 original training images. Then 150 images of the RTS version from the 15 original Five sign word test images.

4 Methodology

The conceptual structure of the proposed Sign word recognition system is demonstrated in Fig. 3. Following steps are followed for both training and test dataset. First, this structure divided the dataset into training and test sets. Second, the training dataset's Rotated, Translated, and Scaled (RTS) version is created. Third, a hybrid segmentation process is applied, which is consists of Otsu thresholding with

YCbCr skin color segmentation, morphology analysis, and the watershed algorithm. Finally, a novel CNN architecture is developed to extract the feature map and classification.

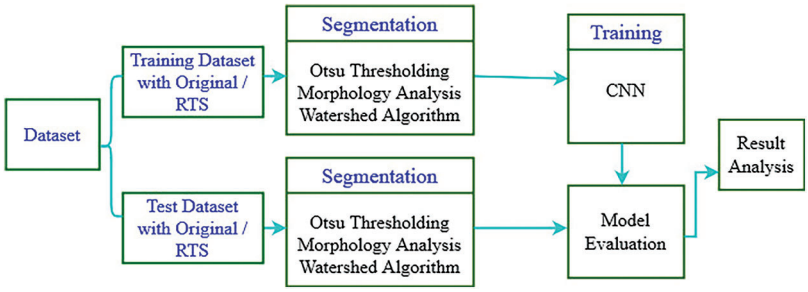

Figure 3: The conceptual structure of the proposed sign word recognition system

RTS version of the dataset is created by applying rotation, translation, scaling techniques on both training and testing datasets. The hybrid segmentation removes the redundant background from the image. Then, we trained the proposed CNN model with a segmented RTS version of the training dataset and tested with the original and RTS version of the test dataset.

4.1 Segmentation Approach

The segmentation process is one of the major steps that can handle the challenges of the vision-based system, especially in the hand gesture recognition system [35]. Sometimes skin color segmentation produces unsatisfied segmented images under the complexity or illumination of the background. The proposed segmentation process combined with three steps to overcome the problem: i) Combining Otsu Thresholding with YCbCr, ii) Morphological analysis: dilation through opening morphology and iii) Watershed algorithm. Fig. 4 presents the process of segmentation.

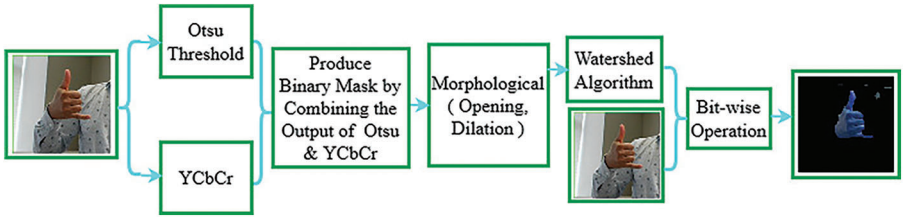

Figure 4: Segmentation process of an input image

4.1.1 Combining Otsu Thresholding with YCbCr

The Otsu method is a well-known threshold technology that varies between two classes and relies on selecting the optimal threshold value [14]. It can automatically set thresholds values based on the image pixel. The basic idea of this method is to minimize within-class variance and maximize between-class variance shown in Eq. (1).

$$\sigma_{\omega}^2(t) = \omega_1(t)\sigma_1^2(t) + \omega_2(t)\sigma_2^2(t) \tag{1}$$

where $\sigma_{\omega}^2(t)$ is the weighted sum of variances of the two classes, $\omega_1(t)$, $\omega_2(t)$ are the probability of the two classes divided by a threshold t, and $\sigma_1^2(t)$, $\sigma_2^2(t)$ are variances of the two classes.

The method of working the Otsu algorithm in this paper is as follows.

i) Input image pixels are divided into foreground and background by a threshold t.
ii) Calculate the mean of the foreground and the background pixels
iii) Calculate the variance between mean
iv) Multiply by the number of pixels between foreground and background pixels

Apart from the Otsu, YCbCr segmentation converted RGB image into YCbCr color space and produced mask by dividing into luminance Y, chrominance Cb and Cr [15]. Finally, we combined the Otsu thresholded output with the luminance Y component of YCbCr and produced a binary mask.

4.1.2 Morphological Analysis: Dilation Through Opening Morphology

Morphological operation and dilation are applied here on the binary mask generated in Section 4.1.1. Morphological operations enhance the efficiency of the segmentation, and it consists of Morphological Analysis: dilation through opening morphology [16]. In dilations, the size of the foreground of the hand sign has been increased and filled in the gaps in the image together. Moreover, we examined the neighbourhood around each pixel and adjusted the size of the segmented image according to the centre pixel p of the structuring element set to white if any pixel of the structuring element is greater than 0.

4.1.3 Watershed Algorithm

Following the process of 4.1.1 and 4.1.2, the waterlogged algorithm is applied for performing the segmentation operation and accurately drawing a border boundary line of the gesture sign word [17]. However, the Watershed algorithm treats the pixel values with brightness as individual spatial topographic maps, finds a boundary line and finally remove the background by applying the bit-wise mask approach, as shown in Fig. 4.

4.2 Classification with Convolutional Neural Network (CNN)

This paper proposed a novel CNN-based classifier model for training and testing on the entire dataset. CNN is a widely used feature extraction and classification algorithm in the Sign language recognition domain [36,37]. We measured different signs of isolated hand gestures using CNN to identify Sign language words. The datasets contain twenty classes, five classes and the RTS version of those datasets where 70% are used for training, and 30% are used for testing the model. The proposed CNN model is shown in

Fig. 5 The CNN is used for extracting the layer feature and classification modules. For this, consider that the N depth image contains $P_n,\ n \in [1,\ N]$, and after normalizing the data, the size of each image is $60 \times 60 \times 3$. Therefore, these features are fed as the input data into three 3×3 convolution layers. We have implemented rectified linear unit (RLU) activation in each convolution. In each convolution level, we used the 2×2 max pooling level, which halves the map of the previous feature. Also, the dropout layer is used to prevent overfitting during the feature extraction process.

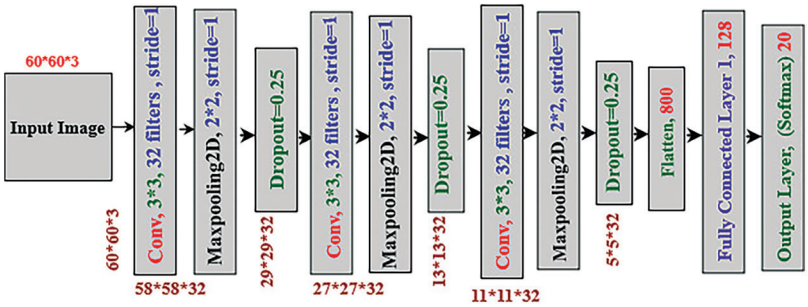

Figure 5: Proposed CNN architecture

A classification module of the dense layer takes a 5 × 5 × 32 feature map which has 128 feature vectors. We then used a fully connected layer to densify the feature vectors into 128 and L sequential dimensions, where L = 20 is the number of sign classes. Therefore, we have used the SoftMax function, which converts the feature vector to the output of the predictable probability by Eq. (2) [38].

$$P(y_n = c|P_n) = \frac{\exp(S_c)}{\sum_{c=1}^{L} \exp(S_c)} \tag{2}$$

where L = 20 dimensional $S([S_1,.., S_c, \ldots S_L])$ and the predicted probability of being class c is $S([S_1,.., S_c, \ldots S_L])$ for sample P_n. Finally, we have used the Adam Optimizer to optimize the predicted value.

5 Result and Discussion

The experimental results are achieved from the analysis of the original and RTS version of the two datasets, which evaluates the feasibility of the proposed system and determines individual class features' contribution to the overall classification accuracy. Also, we have compared it with the state-of-the-art methods based on the same dataset and the same validation scheme through the isolated image analysis

5.1 Environmental Setup and Evaluation Metrics

We resized the segmented image into 60x60x3 for the training and classification. For implementing the proposed experiment, Python programming language is used here on the google colaboratory pro version environment with 25GB GPU, and the name of the GPU is Tesla P100 [39]. As a python package, cv2 [40], numpy, pickle, tensorflow, keras, matplotlib used here where learning rates is 0.001, and optimizer is adam. The number of epochs and batch size are almost the same for all the experiments. Accuracy, Precision, Recall and F1-score are used here as the evaluation metrics, which are calculated based on true positive (Tp), true negative (Tn), false positive (Fp) and false negative (Fn) [32]. The formula of the uses evaluation matrix is given in Tab. 1.

Table 1: Performance evaluation formulas [32]

Name of evaluation matrics	Evaluation matrics formula
Accuracy (%)	$\frac{T_p + F_n}{T_p + F_p + T_n + F_n}$
Precision (%)	$\frac{T_p}{T_p + F_p}$
Recall (%)	$\frac{T_p}{T_p + F_n}$
F1 score (%)	$\frac{2 \times precsion \times recall}{precsion \times recall}$

5.2 Performance Evaluation with the Twenty Sign Word Dataset [4]

To evaluate our model using the original dataset, we trained it with the non-segmented training dataset and tested it with the non-segmented original test dataset. After that, the model is trained with the original segmented training dataset and tested with the original segmented test dataset and the RTS version test dataset.

Tab. 2 shows the classification accuracy of different sign words. Firstly, we trained the proposed model with original non-segmented images, then tested with the original non-segmented test dataset and achieved 97.50% accuracy, 99.00% precision, 97.00% recall and 98.00% f1-score. Secondly, we trained the proposed model with the segmented datasets, then tested with the original segmented test dataset and achieved 98.80% accuracy. After that, again tested the model using the RTS version test dataset and achieved recognition accuracy of 68.00%, precision of 71.00%, recall of 68.00%, and f1-score of 69.00%. We observed that the model's performance is sharply decreased and achieved low accuracy compared to the previous experiment. We have observed that the model's performance depends on the RTS variant test data set. Performance has been achieved very well for the test with the original test dataset; in contrast, very low for the test with the RTS version of the test dataset. The main reason for this situation is we did not train the proposed model using the RTS version training dataset. RTS techniques are being proposed in segmented images to overcome these problems.

Table 2: The proposed method's performance (%) with the twenty sign word dataset

Training Dataset	Segmentation	Test Dataset	Precision	Recall	F1 score	Recognition Accuracy (%)
Twenty sign word	No	30% of Twenty sign word	99.00	97.00	98.00	97.50
Twenty sign word	Yes	30% of Twenty sign word	98.00	98.00	99.00	98.80
Twenty sign word	Yes	30% RTS version of Twenty sign word	71.00	68.00	69.00	68.00

Note: *RTS = Rotated, Translated and Scaled images.

5.3 Performance Evaluation with RTS Version Dataset of [4]

We employed the RTS version of the dataset [4] to make RTS invariant model and improve the model's performance. The RTS version dataset evaluation scenario is as follows: we trained the proposed model with the non-segmented RTS version training dataset then tested it with the non-segmented RTS version test dataset. After that, we trained the model with the segmented RTS version of the training dataset, then tested with the original segmented test dataset and segmented RTS version of the test dataset.

To do the above scenario, After applying the RTS technique, we found 133000 and 57000 RTS versions images for training and testing for non-segmented and segmented cases. The proposed model achieved 98.50% accuracy for training and testing with a non-segmented RTS version dataset. In the case of the segmented RTS version, our proposed system achieved 99.30% accuracy for the original test dataset and got 99.10% accuracy for the RTS version test dataset. Tab. 3 shows the classification accuracy, precision, recall and f1-score. Maximum performance accuracy is 99.30% which is achieved from the training with segmented RTS version of the Twenty sign word dataset and test with the original segmented images. We can decide our model becomes an RTS invariant method because the proposed model can produce good accuracy for the original test dataset and RTS version of the test dataset.

Fig. 6 shows the confusion matrix for the proposed model, which has been trained with the segmented RTS version dataset and tested with the original segmented dataset. Sign word for predicted class represents each row, and each column represents the correct class in the confusion matrix. Our RTS invariant model produced good accuracy for all classes and correctly classified almost more than 95% accuracy except one class, and the maximum misclassification rate is 5% (mentioned in [4,8] location of the confusion matrix). The performance of the proposed model for each class, and there is a minimal amount of true negative and false positive while producing high accuracy.

Table 3: Performance (%) for the RTS version of twenty sign word dataset

Training Dataset	Segmentation	Test Dataset	Precision (%)	Recall	F1 score	Recognition Accuracy (%)
RTS version of Twenty sign word	No	Twenty Sign Word Images	98.00	98.00	99.00	98.50
RTS version of Twenty sign word	Yes	Twenty Sign Word Images	99.00	99.00	99.00	**99.30**
RTS version of Twenty sign word	Yes	RTS version of Twenty Sign Word	99.00	99.00	99.00	99.10

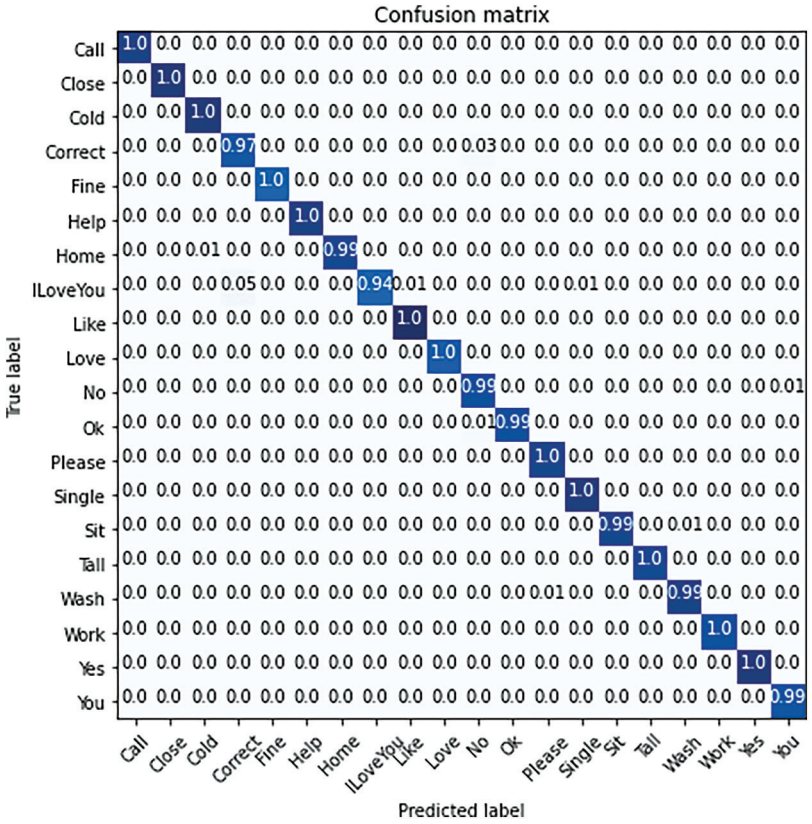

Figure 6: Confusion matrix for best performance of the proposed method

5.4 Performance Evaluation with Five Sign Word Dataset [33]

Tab. 4 shows the performance accuracy for the segmented Five sign word dataset. We did not present the proposed model's non-segmented cases performance accuracy because they are almost identical with segmented images.

Table 4: Performance (%) of the proposed methods with five sign word datasets [33]

Dataset	Segmentation	Test Dataset	Precision (%)	Recall (%)	F1 score (%)	Recognition Accuracy (%)
Five Sign Word	Yes	Original 15 test images	94.00	93.00	93.00	93.33
Five Sign Word	Yes	RTS version images were generated from 15 test images.	71.00	70.00	70.00	70.00

For evaluating the proposed model with Five sign word dataset, we trained it with the original 216 images, tested it with 15 test images, and achieved 93.33% accuracy, 94% precision, 93.00% recall, and 93.00% f1-score. After that, we tested the model with the RTS version of the test dataset and achieved 70.00% accuracy, 71.00% precision, 70% recall, 70% f1-score. We observe that model performance accuracy is sharply down for the RTS version of the Five sign word test dataset compared to the original test dataset. The main reason for low performance is that we did not train the model with the RTS variant training dataset, but we are testing with the RTS version test dataset.

5.5 Performance Evaluation with RTS Version of Five Sign Word Dataset [33]

We have trained the proposed model with the RTS version of the Five sign word training dataset to get good prediction results for the rotated, translated, scaled (RTS) version of Five sign word test datasets. In this case, we trained the proposed model with 2160 images of RTS versions of the Five sign word training dataset. Then, we have tested the model with two cases; firstly, we tested the model with 15 images of the original test dataset and achieved 100% accuracy. Secondly, we tested the proposed model with 150 images of RTS versions of the Five sign word test dataset and achieved 98.00% accuracy. Tab. 5 shows the performance accuracy for the proposed model trained with the RTS version of the Five sign word dataset.

Table 5: Performance (%) with the RTS version dataset of five sign word datasets

Training Dataset	Segmentation	Test Dataset	Precision (%)	Recall (%)	F1 score (%)	Recognition Accuracy (%)
RTS version of Five sign word	Yes	Original 15 test images	99.00	99.00	99.00	**100**
RTS version of Five sign word	Yes	RTS version of 15 test images.	98.00	98.00	98.00	98.00

5.6 Comparison with State-of-the-Art Method

The study proposes a combined segmentation approach along with RTS invariant Convolutional neural network (CNN). According to this, the RTS technique is applied to the input images then the combined segmentation technique is employed on the RTS version images. Our model achieved maximum performance accuracy at the RTS datasets training by testing the original dataset using the CNN model for both datasets.

Tab. 3 shows maximum accuracy is 99.30% achieved for Twenty sign word dataset evaluation, and Tab. 4 shows the maximum accuracy is 100% for Five sign word dataset evaluation. We have compared our proposed method with two existing methods: first one, for twenty classes, we trained the proposed model with RTS version of [4] and tested with the same size of the reference test dataset, and the model

achieved 99.30% accuracy. The second one, for the Five sign word dataset, we trained the model with the RTS version of [33] dataset and tested it with the same size of reference test dataset, and the model produced 100% accuracy. Tab. 6 shows the comparative performance of the proposed and state-of-the-art methods where the proposed method achieved higher classification accuracy than the existing system.

Table 6: Difference between our proposed study and state-of-the-art studies

Method	Dataset	Number of Images	Segmentation	Test Set	Reported Accuracy (%)
Rahim et al. [4], 2019	20-Sign dataset	18000	yes	30% of the Dataset	97.28
Mujahid et al. [33], 2021	5-Sign dataset	216	no	15 Test Images	97.68
Proposed (CNN Train and Test with RTS dataset)	RTS version dataset of [4]	190,000	yes	Same as [4]	99.30
	RTS version dataset of [33]	2160	yes	Same as [33]	100

The authors [4] proposed YCbCr and SkinMask segmentation techniques [15] and CNN to classify the sign word, and their performance accuracy was 97.28%, whereas our proposed model achieved 93.30%. The authors [33] proposed a lightweight model based on You Only Look Once (YOLO) v3 [41] and DarkNet [42] convolutional neural network to classify the five-sign word dataset. The accuracy reported for the five-sign word recognition was 97.68%, whereas our proposed model achieved 100% accuracy.

Fig. 7 depicts sign words' class wise comparison recognition accuracy of the proposed method and Reference [4]. The graph shows that our proposed model produced higher accuracy at 15 labels than the existing state-of-the-art method, with the same accuracy at three classes which are 100% at 'call' label, 100% accuracy at 'fine' labels and 99% accuracy at 'ok' labels. So, we can say our model have produced good performance accuracy compared to [4] model at all labels excludes three labels. The proposed system achieved the ability to recognize the sign word images where test images come from rotation, translation, scaling positions. This ability achieved the proposed system because it is composed of the RTS technique for dealing with the rotated, translated, scaled test images, a combined segmentation technique for correctly segmenting the complex background images and the novel CNN architecture. These components make the system robust in real-time test images and successfully works in a different challenging environment.

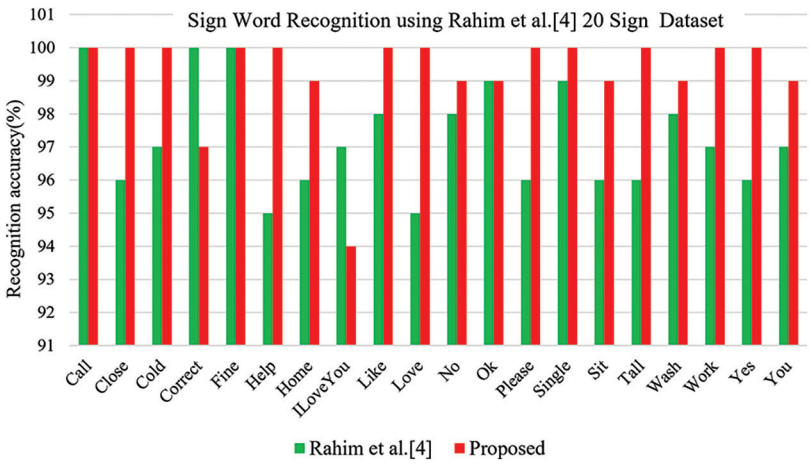

Figure 7: Class wise difference between our proposed study and the existing [4] study

6 Conclusion

In this paper, we have presented a combined segmentation along with RTS invariant convolutional neural network (CNN) method. The method follows the process of segmentation, RTS version of the dataset, and classification. In segmentation, the part of the sign gesture is identified by following the steps, Otsu thresholding with YCbCr, Morphological Analysis: dilation through opening morphology, and Watershed Algorithm. Two benchmark datasets were considered for the performance evaluation of the proposed method. However, rotated, translated, and scaled (RTS) strategies have been applied to the overfitting problem of the method by increasing the dataset's size and making an RTS invariant model. We used the CNN methods for effective classification. We have tested our model during classification using original and RTS test images in both datasets. As a result, the proposed method achieved better performance for both datasets. Our method produced 99.10% accuracy for the RTS version test images and 99.30% accuracy for the original test image in the case of the twenty sign word dataset. In the case of the five-sign dataset, the proposed model has produced 98.00% RTS version test images and 100% accuracy for the original test image. Our experimental results show that RTS images provide optimal performance, and our model has become an RTS invariant model. Finally, the overall results show that the proposed method significantly outperforms the recognition of sign words with or without RTS images compared to state-of-the-art methods. This approach will help users who cannot collect appropriate Sign word images. In future work, we will be collected more continuous and discrete sign word data to evaluate a hybrid method for making mobile applications and other scenarios. Moreover, we will be focused design a data fusion based advanced convolutional neural network on enhancing the sign word recognition precision.

Acknowledgement: We show gratitude to anonymous referees for their useful ideas.

Funding Statement: This work was supported by the Competitive Research Fund of The University of Aizu, Japan.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] P. Neto, M. Simão, N. Mendes and M. Safeea, "Gesture-based human-robot interaction for human assistance in manufacturing," *The International Journal of Advanced Manufacturing Technology*, vol. 101, no. 1, pp. 119–35, 2019.

[2] T. Kamnardsiri, L. O. Hongsit, P. Khuwuthyakorn and N. Wongta, "The effectiveness of the game-based learning system for the improvement of American sign language using kinect," *Electronic Journal of e-Learning*, vol. 15, no. 4, pp. 283–296, 2017.

[3] A. Vaitkevičius, M. Taroza, T. Blažauskas, R. Damaševičius, R. Maskeliūnas *et al.,* "Recognition of American sign language gestures in virtual reality using leap motion," *Applied Sciences*, vol. 9, no. 3, pp. 445, 2019.

[4] M. A. Rahim, M. R. Islam and J. Shin, "Non-touch sign word recognition based on dynamic hand gesture using hybrid segmentation and CNN feature fusion," *Applied Sciences*, vol. 9, no. 18, pp. 3790, 2019.

[5] M. J. Cheok, Z. Omar and M. H. Jaward, "A review of hand gesture and sign language recognition techniques," *International Journal of Machine Learning and Cybernetics*, vol. 10, no. 1, pp. 131–53, 2019.

[6] M. Al-Hammadi, G. Muhammad, W. Abdul, M. Alsulaiman, M. A. Bencherif *et al.,* "Hand gesture recognition for sign language using 3DCNN," *IEEE Access*, vol. 8, no. 79, pp. 491–509, 2020.

[7] M. Jebali, A. Dakhli and M. Jemni, "Vision-based continuous sign language recognition using multimodal sensor fusion," *Evolving Systems*, vol. 12, pp. 1031–1044, 2021.

[8] R. Elakkiya, "Machine learning-based sign language recognition: A review and its research frontier," *Journal of Ambient Intelligence and Humanized Computing*, vol. 12, pp. 7205–7224, 2021.

[9] K. Kudrinko, E. Flavin, X. Zhu and Q. Li, "Wearable sensor-based sign language recognition: A comprehensive review," in *IEEE Reviews in Biomedical Engineering*, vol. 14, pp. 82–97, 2021.

[10] M. A. Rahim, A. S. M. Miah, A. Sayeed and J. Shin, "Hand gesture recognition based on optimal segmentation in human-computer interaction," in *Proc. of the 3rd IEEE Int. Conf. on Knowledge Innovation and Invention (ICKII)*, Taiwan, pp. 163–166, 2020.

[11] N. M. Adaloglou, T. Chatzis, I. Papastratis, A. Stergioulas, G. T. Papadopoulos *et al.,* "A comprehensive study on deep learning-based methods for sign language recognition," *IEEE Transactions on Multimedia*, vol. 24, pp. 1–1, 2021.

[12] S. Zeng, B. Zhang B, J. Gou and Y. Xu, "Regularization on augmented data to diversify sparse representation for robust image classification," *IEEE Transactions on Cybernetics*, pp. 1–14, 2020. https://dx.doi.org/10.1109/TCYB.2020.3025757.

[13] R. Thilahar and R. Sivaramakrishnan, "Fuzzy neuro-genetic approach for feature selection and image classification in augmented reality systems," *International Journal of Robotics and Automation (IJRA)*, vol. 8, no. 3, pp. 194–204, 2019.

[14] N. Otsu, "A threshold selection method from gray-level histograms," in *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 9, no. 1, pp. 62–66, 1979.

[15] K. Kolkur, D. Kalbande, P. Shimpi, C. Bapat and J. Jatakia, "Human skin detection using RGB, HSV and YCbCr color models," in *Proc. of the ICCASP/ICMMD*, India, vol. 137, pp. 324–332, 2016.

[16] K. A. M. Said, A. S. Jambek and N. Sulaiman, "A study of image processing using morphological opening and closing processes," *International Journal of Control Theory and Applications*, vol. 9, pp. 15–21, 2016.

[17] H. Sun, J. Yang and M. Ren, "A fast watershed algorithm based on chain code and its application in image segmentation," *Pattern Recognition Letters*, vol. 26, no. 9, pp. 1266–1274, 2005.

[18] P. Kumar, H. Gauba, P. P. Roy and D. P. Dogra, "A multimodal framework for sensor based sign language recognition," *Neurocomputing*, vol. 259, pp. 21–38, 2017.

[19] J. Wu, L. Sun and R. Jafari, "A wearable system for recognizing American sign language in real-time using IMU and surface EMG sensors," *IEEE Journal of Biomedical and Health Informatics*, vol. 20, no. 5, pp. 1281–1290, 2016.

[20] Y. Li, X. Chen, X. Zhang, K. Wang and Z. J. Wang, "A sign-component-based framework for Chinese sign language recognition using accelerometer and sEMG data," *IEEE Transactions on Biomedical Engineering*, vol. 59, no. 10, pp. 2695–2704, 2012.

[21] N. Tubaiz, T. Shanableh and K. Assaleh, "Glove-based continuous arabic sign language recognition in user-dependent mode," *IEEE Transactions on Human-Machine Systems*, vol. 45, no. 4, pp. 526–533, 2015.

[22] M. A. Rahim, J. Shin and M. R. Islam, "Human-machine interaction based on hand gesture recognition using skeleton information of kinect sensor," in *Proc. of the 3rd Int. Conf. on Applications in Information Technology*, Japan, pp. 75–79, 2018.

[23] Y. Hu "Finger spelling recognition using depth information and support vector machine," *Multimed Tools Application*, vol. 77, pp. 29043–29057, 2018.

[24] T. Starner, J. Weaver and A. Pentland, "Real-time American sign language recognition using desk and wearable computer based video," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 12, pp. 1371–1375, 1998.

[25] B. G. Lee and S. M. Lee, "Smart wearable hand device for sign language interpretation system with sensors fusion," *IEEE Sensors Journal*, vol. 18, no. 3, pp. 1224–1232, 2018.

[26] J. Shin and C. M. Kim, "Non-touch character input system based on hand tapping gestures using kinect sensor," *IEEE Access*, vol. 5, pp. 10496–10505, 2017.

[27] T. W. Chong and B. G. Lee, "American sign language recognition using leap motion controller with machine learning approach," *Sensors*, vol. 18, no. 10, pp. 3554, 2018.

[28] P. K. Pisharady and M. Saerbeck, "Recent methods and databases in vision-based hand gesture recognition: A review," *Computer Vision and Image Understanding*, vol. 141, pp. 152–65, 2015.

[29] V. Jain, A. Jain, A. Chauhan, S. S. Kotla and A. Gautam, "American sign language recognition using support vector machine and convolutional neural network," *International Journal of Information Technology*, vol. 12, no. 3, pp. 1193–200, 2021.

[30] X. Y. Wu, "A hand gesture recognition algorithm based on DC-CNN," *Multimedia Tools and Applications*, vol. 79, no. 13, pp. 9193–205, 2020.

[31] S. F. Chevtchenko, R. F. Vale, V. Macario and F. R. Cordeiro "A convolutional neural network with feature fusion for real-time hand posture recognition,"*Applied Soft Computing*, vol. 73, pp. 748–66, 2018.

[32] S. C. Agrawal, A. S. Jalal and R. K. Tripathi, "A survey on manual and non-manual sign language recognition for isolated and continuous sign," *International Journal of Applied Pattern Recognition*, vol. 3, no. 2, pp. 99–134, 2016.

[33] A. Mujahid, M. J. Awan, A. Yasin, M. A. Mohammed, R. Damaševičius *et al.,* "Real-time hand gesture recognition based on deep learning yolov3 model," *Applied Science*, vol. 11, no. 9, pp. 164, 2021.

[34] W. Tao, M. C. Leu and Z. Yin "American sign language alphabet recognition using convolutional neural networks with multiview augmentation and inference fusion," *Engineering Applications of Artificial Intelligence*, vol. 76, pp. 202–213, 2018 .

[35] Z. Ju, X. Ji, J. Li and H. Liu, "An integrative framework of human hand gesture segmentation for human–robot interaction," *IEEE Systems Journal*, vol. 11, no. 3, pp. 1326–1336, 2017.

[36] S. Sharma and K. Kumar, "ASL-3DCNN: American sign language recognition technique using 3-D convolutional neural networks," *Multimedia Tools and Applications*, vol. 80, pp. 1–3, 2021.

[37] J. Chen, Z. Zhou, Z. Pan and C. Yang, "Instance retrieval using region of interest-based cnn features," *Journal of New Media*, vol. 1, no. 2, pp. 87–99, 2019.

[38] F. Osayamwen and J. Tapamo, "Deep learning class discrimination based on prior probability for human activity recognition," *IEEE Access*, vol. 7, pp. 14747–14756, 2019.

[39] E. Bisong, "Google Colaboratory," in *Building Machine Learning and Deep Learning Models on Google Cloud Platform*, 1st ed., vol. 1. Berkeley, CA: Apress, pp. 59–64, 2019.

[40] S. Gollapudi, "OpenCV with Python," in *Learn Computer Vision Using OpenCV*, 1st ed., vol. 1. Berkeley, CA: Apress, pp. 31–50, 2019.

[41] L. Zhao, and S. Li, "Object detection algorithm based on improved YOLOv3," *Electronics*, vol. 9, no. 3, pp. 537, 2020.

[42] H. Ma, Y. Liu, Y. Ren and J. Yu, "Detection of collapsed buildings in post-earthquake remote sensing images based on the improved YOLOv3," *Remote Sensing,* vol. 12, no. 1, pp. 44, 2020.