A Video-based Taiwan Sign Language Recognition System Using Deep Learning Techniques

Ming-Han Huang

National Yang Ming Chiao Tung University, cwhuangb1321@gmail.com

Hsuan-Min Wang*

National Yang Ming Chiao Tung University, hmwang.cs04g@g2.nctu.edu.tw

CHUEN-TSAI SUN

National Yang Ming Chiao Tung University, ctsun@cs.nctu.edu.tw

Past research on sign language recognition has mostly been based on physical information obtained via wearable devices or depth cameras. However, both types of devices are costly and inconvenient to carry, making it difficult to gain widespread acceptance by potential users. The goal of this research is to use sophisticated and recently developed deep learning technology to build a recognition model for a Taiwanese version of sign language, with a limited focus on RGB images for training and recognition. It is hoped that this research, which makes use of lightweight devices such as mobile phones and webcams, will make a significant contribution to the communication needs of deaf-mute individuals. This research is based on the sign language dictionary approved by the Taiwan Ministry of Education, with 40 commonly used words being featured in short videos. After a training data set was created, DarkPose human pose estimation model[1] was employed to estimate whole-body postures from RGB images. Human body keypoints were extracted from the images prior to sign language recognition training. Predictions from the two models were weighted and averaged to construct a combined model that achieved a 98.6% recognition rate for the 40 sign language terms.

Additional Keywords and Phrases: Deep learning, sign language recognition, Taiwan Sign Language, 3DCNN, human poses, model ensembles

1. MOTIVATION

Today we have many daily opportunities to watch deaf-mute individuals use sign language to communicate with each other, as well as to watch sign language interpreters at work in meetings or in media coverage of press conferences and official announcements. In addition to communicating with each other, deaf-mute individuals use sign language to interact with hearing-enabled people who have learned sign language or have access to interpreters. To communicate with their deaf-mute children, parents and other family members must invest large quantities of time and expense learning a sign language, or hire caregivers who can use and/or teach sign languages. The number of individuals who are sign language-fluent is much smaller than those who speak foreign languages, thus posing challenges for prompt and accurate communication. The primary goal of this research is to use the latest computer vision and deep learning technologies to perform sign language recognition tasks in support of interactions between deaf-mute and hearing

* Place the footnote text for the author (if applicable) here.

individuals who do not have full-time access to interpreters. We believe such a tool will be especially useful for providing appropriate assistance to deaf-mute individuals in emergency situations.

Presently, the most commonly used sign language recognition technologies consist of wearable devices and depth cameras [2]. Although the information obtained by such devices provides great assistance, users must deal with problems tied to universality and lack of portability. To achieve widespread social or personal use, these tools must be made smaller, lighter, cheaper, and easier to maintain and upgrade. Accordingly, the model described in this research uses the DarkPose [1] whole body estimation model to extract sign language information via images, and then integrates it with pre-trained neural networks to achieve sign language recognition. Sign languages are distinctly national or regional, and currently there is no model training data set for the Taiwanese version, which is required to create a body of local sign language support materials. Since DarkPose does not require special equipment for data acquisition, our proposed system can be applied to portable and lightweight devices such as mobile apps or simple webcams that are already in wide use.

2. RELATED WORK

2.1 Sign language recognition

Sign language communication requires gestures and upper body postures that express concepts and ideas, plus facial expressions that convey meaning or tone. Digital sign language recognition presents multiple challenges in terms of computer vision. Users must make many tradeoffs between systems based on a detailed understanding of their advantages and disadvantages [3]. For data acquisition tasks, the most widely used technologies today include the use of gloves with detectors, accelerometers, Microsoft Kinect, Intel RealSense (with depth lenses), or webcam and multi-view cameras [2]. Many systems rely heavily on depth lenses to obtain 3D data [4].

Arguably the greatest challenge for sign language recognition systems is background separation. Successful preprocessing requires the separation of hand and facial information from their backgrounds, using cues such as skin color and the continuous tracking of hand movements [5]. Wren et al. [6] have created a useful method that entails visual “blobs” that separate all or parts of bodies from complex backgrounds, thus removing a large amount of noise and achieving better recognition rates. Other researchers have reported that the combination of Kinect depth information and RGB data supports good background separation for training purposes with convolutional neural networks (CNN) [7, 8]. Note that in terms of perspective, researchers have proposed both third-person and first-person approaches to system design, but problems obtaining sufficient amounts of information indicate a need for additional objects such as wristbands [9].

After background separation, the next major challenge is ensuring image recognition of each individual finger, since understanding finger movement is key to learning a sign language. The most commonly used method for identifying finger extensions is reference points [10]. Since certain identical gestures can look completely different based on different rotations and translations, depth data are required to identify 3D positions and to make predictions for individual fingers. For the next step—recognizing hand positions and making translations [11, 12]—existing

technologies such as Intel RealSense depth cameras can be used to obtain hand data and to identify gestures [13]. Similar to printed text, individual signs with similar-looking gestures can cause confusion. [14] The use of multi-layered random forest (MLRF) classifiers achieves better recognition rates when dealing with this problem, with one layer detecting hand position and another recognizing hand movement.

Starner and Pentland [15] used the Hidden Markov Model (HMM, a standard method for analyzing continuous movement) to track hand motions for purposes of recognizing signs for 40 English words. For deep learning, the direct use of CNN has been shown to achieve a high recognition rate for sign language speakers compared to video images of single words [16]. Since sign language data are continuous, region-based CNN (R-CNN) produces better results, but suffers from a tendency to over-simulate in cases of insufficient data, resulting in performance degradation [17]. When 3D skeletal data are available, a CNN + long short-term memory (CNN+LSTM) model is useful for recognizing continuous 3D+time actions [18]. The 3DCNN model proposed by Ji et al. [19] represents an important improvement to the CNN model limitation of not referring to time series data; today it is widely used for action recognition tasks [21, 22]. The 3DCNN model structure is shown as Figure 1.

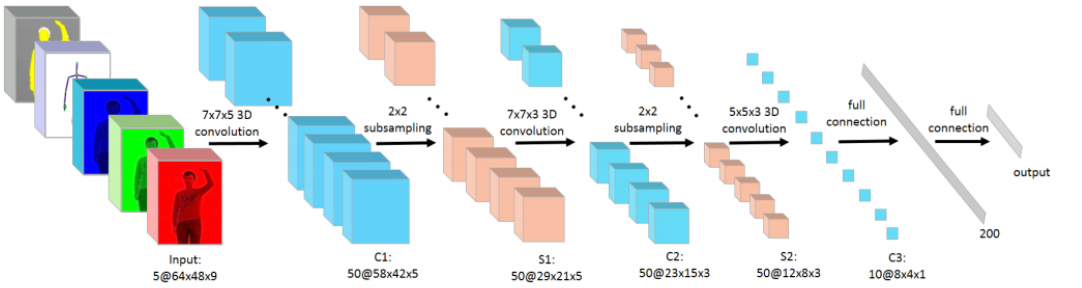


Figure 1. The 3DCNN model structure as applied to sign language recognition tasks. The model uses three 3DCNN layers and two partial sampling layers. Source: Huang et al. [7].

2.2 Human body pose estimation

In early human pose estimation (HPE) research, bodies were treated as combinations of parts rather than joint systems. To obtain binary images, Felzenszwalb et al. [23] separated bodies from their backgrounds and matched individual body parts (represented as boxes) to individual limbs using a posture estimation method. One limitation was that limb object matching was based on binary images that were identified following background separation, therefore the correct positions of covered (overlapping) limbs could not be seen, resulting in many incorrectly matched body parts (Fig. 2).

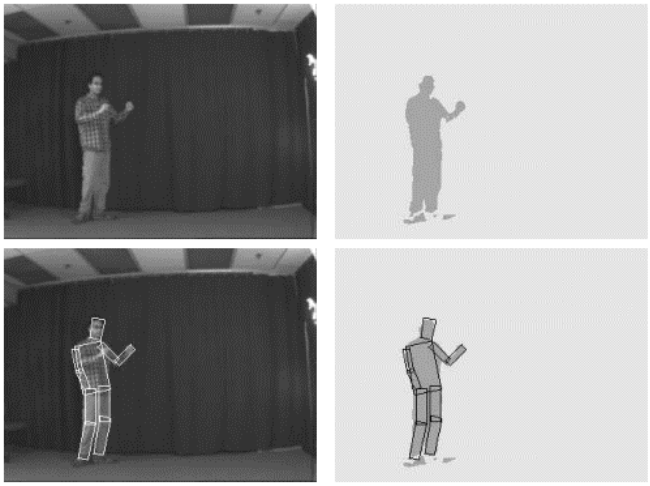


Figure 2. The right arm and hand of the individual in this image is incorrectly matched because it overlaps the torso. Source: Felzenszwalb et al. [23].

DeepPose [24], the first tool to apply deep learning to HPE research, uses a seven-layer deep neural network (DNN) to identify images. The “return method” improves accuracy by connecting a final output layer representing joint point positions as (x, y) coordinates, but it only works with two-dimensional local coordinates; since it lacks spatial and environmental information, it does not perform well with overlapping joints (Fig. 3).

Figure 3. Schematic diagram of the DeepPose DNN network architecture plus regression. Source: [24].

Another HPE research direction is heat map prediction technology [25], which processes images in parallel with multiple resolutions to detect sliding windows and locate targeted joint points. The junction node generates a heat map that forms a two-dimensional Gaussian distribution with the targeted joint position at its center (Fig. 4). A Gaussian distribution allows the model to consider environments around joint points during training, which helps improve model performance in cases of complex backgrounds or joint points that are occluded or overlapping. Heat map prediction technology has been applied to advanced research involving Cascaded Pyramid Networks (CPNs) [19], SimpleBaseline [17], and HRNet [26], among other tools.

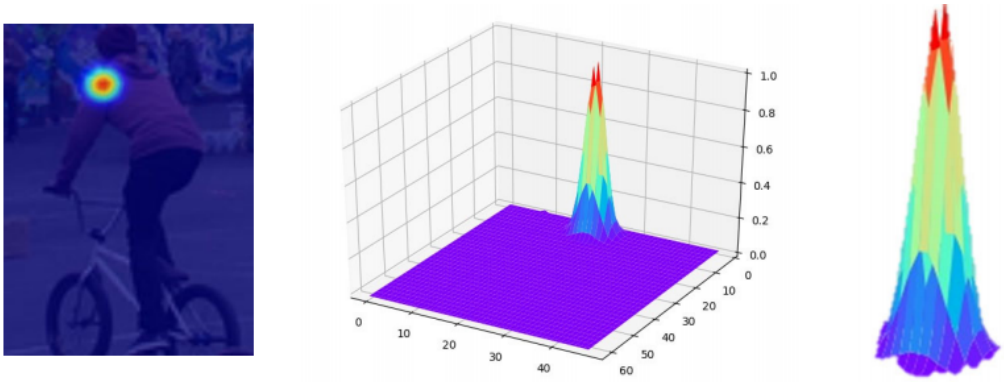


Figure 4. The heat map prediction method generates 2D Gaussian probability maps at joint points. Source: [1].

2.3 DarkPose

DarkPose [1], a model-independent plug-in that optimizes heat map technology and verifies COCO and MPII data sets, was released by the University of Electronic Science and Technology of China in October of 2019. All existing human body pose estimation models were analyzed in terms of effectiveness, with HRNet producing the best results. According to our data, heat map and joint coordinate point conversions exert significant impacts on HPE training and accuracy, and are therefore responsible for decoding heat maps into coordinate points and encoding coordinate points into heat maps. Table 1 presents results from a comparison of DarkPose with other HPE models using a carry out computation optimization (COCO) data set, and the distribution-aware coordinate representation of keypoint (DARK) algorithm.

Table 1. A comparison of DarkPose with other advanced human pose estimate (HPE) models, based on a carry out computational optimization (COCO) data set.

HPE Model	Backbone	Input Size	AP	AP^{50}	AP^{75}	AP^{M}	AP^{L}	AR
Bottom-up								
OpenPose [27]	-	-	61.8	84.9	67.5	57.1	68.2	66.5
MultiPoseNet [28]	-	-	69.6	86.3	76.6	65.0	76.3	73.5
Top-down								
G-RMI [29]	ResNet-101	353×257	64.9	85.5	71.3	62.3	70.0	69.7
CPN [30]	ResNet-Inception	384×288	73.0	91.7	80.9	69.5	78.1	79.0
SimpleBaseline [31]	ResNet-152	384×288	73.7	91.9	81.1	70.3	80.0	79.0
HRNet [32]	HRNet-W48	384×288	77.0	**92.7**	84.5	73.4	83.1	82.0
DARK [1]	HRNet-W48	384×288	**77.4**	92.6	**84.6**	**73.6**	**83.7**	**82.3**

Notes: AP, average precision; AP 50, average success rate when intersection over union > 50%; AP 70, average success rate when intersection over union > 70%; APm, medium-size frame pattern (32*32 < area), regarded as successful and accurate; APL, large-size frame pattern (96*96 < area) regarded as successful and accurate; AR = average recall.

The model-predicted heat map was found to have multiple peaks after decoding into coordinate points, and was therefore convolved with a Gaussian kernel with the same distribution for use as test data in order to obtain a smoothed heat map. Next, Taylor expansion was applied to calculate correct joint point positions (Fig. 5a-b) prior to returning peak heat map calculations to the same space as the original image, and converting them to the correct target joint coordinates (Fig. 5c). However, the part of this process where joint coordinates are encoded into a heat map has the same quantization problem as that observed during the decoding process. In standard encoding methods, whenever original image resolution is reduced, joint point coordinates may be rounded into integers, resulting in errors. DARK solves this problem by directly setting heat map centers in non-quantized positions. Since coordinate point encoding usually refers to ground truth encoding into heat maps for model learning, many model training optimizations are possible.

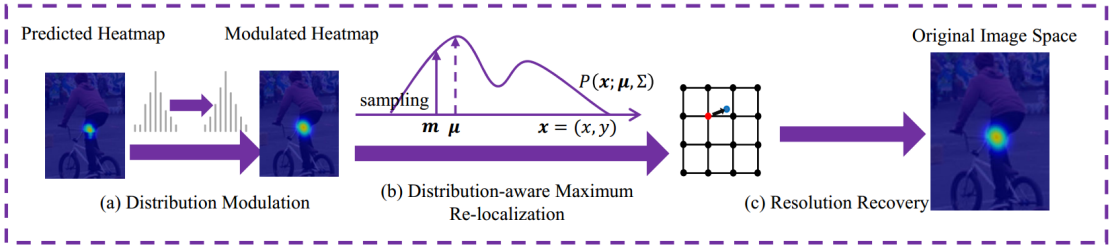


Figure 5. Schematic of DARK-based optimization for decoding heat maps into coordinate points.

3. PROPOSED SOLUTION

3.1 RESEARCH STRUCTURE

The main goal for this research is creating a recognition system for Taiwanese sign language, with video clips of sign language speakers serving as input and prediction results for 40 language terms serving as output. Author-produced Taiwanese sign language videos were used to form a data set, and then used with an HPE model to extract body, hand, and facial keypoints. RGB images from the videos were input into a 3DCNN system for training. Final output prediction was determined as the weighted average of results from the two steps. The video data set was used to separately train two models: a GCN model (explained in detail in section 3.5) with human body keypoints as input, and a 3DCNN model with the original RGB images as input. Last, model prediction results were weighted, averaged, and used as output.

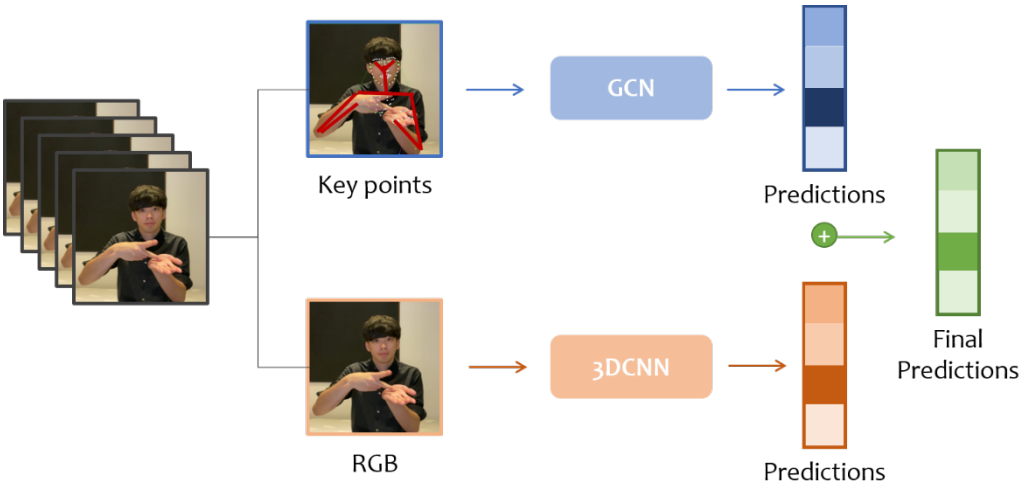


Figure 6. Schematic diagram of the research architecture.

3.2 Data collection

Sign language data sets tend to be scattered due to regional and national differences. Currently the most common sign language data sets are associated with American Sign Language (ASL) and Chinese Sign Language (CSL). There is currently no database available for training for the Taiwanese sign language used in this research, only two online sign language dictionaries compiled by the Taiwan Ministry of Education and National Chung Cheng University. Although both offer demonstration videos, their data are insufficient for training a sign language recognition model. We therefore supplemented the Ministry of Education dictionary with data from the ASLLVD and DEVISIGN sign language databases.

As stated above, our goal is to create a tool that deaf-mute individuals can use for communication during emergency situations. Four terminology categories were chosen for this task: feelings, asking for help, communication, and daily needs. The 40 terms shown in Table 2 served as identification targets; gesture and movement images are presented in Appendix.

Table 2. The 40 Taiwanese sign language terms used in this research and their categories.

Category	Vocabulary Item
Feelings	fear, glad, dislike, painful
Ask for help	disappear, search, rob, headache, hungry, lost, hearing aid, wounded, catch a cold, dizzy, ask for help, danger
Communicate	we, cannot, not right, don't want, don't know, never mind, careful, understand, at once, can, agree, forget, sorry, welcome, request, thank, very, encourage
Daily needs	eat, drink, respirator, rent, telephone, relax

3.3 Data pre-processing

3.3.1 Video-to-RGB Image Conversion

Since the final layer of the 3DCNN model used in this study is a fully connected classification layer, a necessary step is converting the video to RGB images prior to training, making sure that the number of images (frames) is the same in each data set. After using the HPE model to extract a whole body keypoint vector from the video, and after using the whole body keypoint vector to identify the maximum range of motion for the signing individual, images were cropped down to squares with the signing individual as the central focus. Picture size was then reduced to 256 x 256 pixels to facilitate training. To ensure equal numbers of video images during the training process, GPU memory space was calculated and the average number of video frames used as a benchmark for cutting and cropping (70 frames). When the number of video frames exceeded the number of reference frames, images were cropped to emphasize the middle part; when the number of video frames was less than the number of reference frames, sections of the video were duplicated until an equal number was achieved.

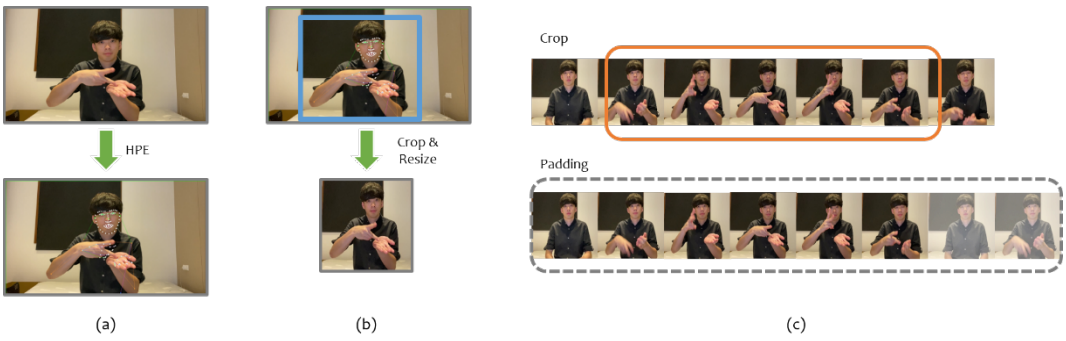


Figure 7. Flow chart of the video-to-RGB picture conversion process. (a) Whole-body keypoints are extracted from the video; (b) cuts are made to reduce video size; (c) if necessary, further cuts are made to achieve an equal number of reference frames or otherwise make corrections.

3.3.2 Training and validation sets

Trained model quality was determined according to whether the model made correct judgments after receiving previously unseen photos or videos. To properly train and test the model, data were divided into training and validation sets (the latter used to verify model effectiveness) prior to the start of each experiment. Randomly dispersed data were divided into training and validation sets at a 4:1 ratio, with sets containing the same amounts of data for each sign language classification. Upon completion of the pre-processing stage there were 619 videos in the training set and 127 in the validation set.

3.4 Human keypoint detection

Currently marketed and open source human keypoint detection or HPE tools for capturing human movement include OpenPose (developed by CMU) and DarkPose, with HRNet serving as a basic model. Although OpenPose features

detailed parameter descriptions and a complete real-time 2D multi-person pose estimation system, it lags behind the latest DarkPose version in terms of performance and accuracy. During testing we noticed that OpenPose did not detect hand positions when the elbow of the signer was not visible on-screen (Fig. 8). Further, OpenPose frame rates were greatly reduced during simultaneous face-hand-body posture detection. Since DarkPose accuracy and performance were not affected by similar conditions, it was chosen for the total body pose estimation tasks in this research.

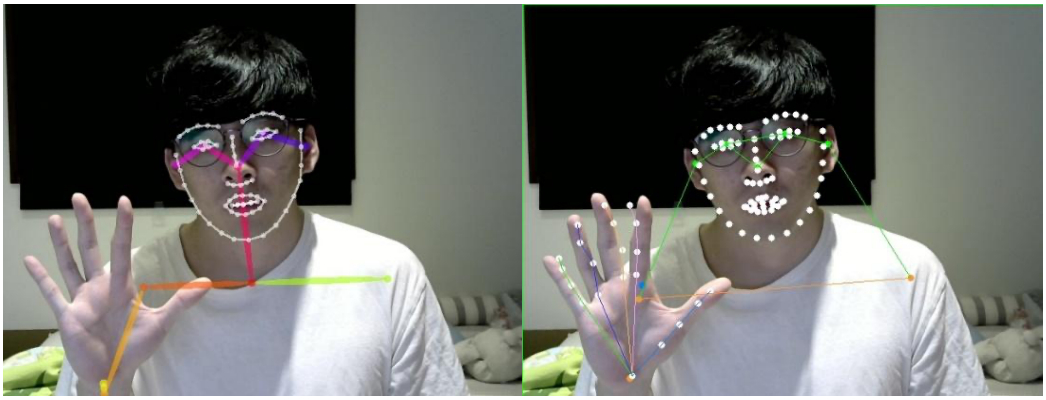


Figure 8. OpenPose (left) did not detect hand positions when the subject's elbow was not visible. DarkPose (right) was not affected.

3.4.1 Keypoint Selection

The COCO WholeBody data set [33] consisted of 133 extracted keypoints—17 limb, 6 foot, 68 face, and 42 hand (21 each for left and right hands). Three-dimensional data sets were generated for each keypoint. The first two numbers represent two-dimensional keypoint coordinates (x, y) indicating horizontal and vertical positions, and the third number the keypoint confidence value (a floating point number between 0 and 1). The data sets did not cover areas below the torso, therefore waist (12, 13), knee (14, 15), ankle (16, 17) and foot keypoints were not included in the model prediction process.

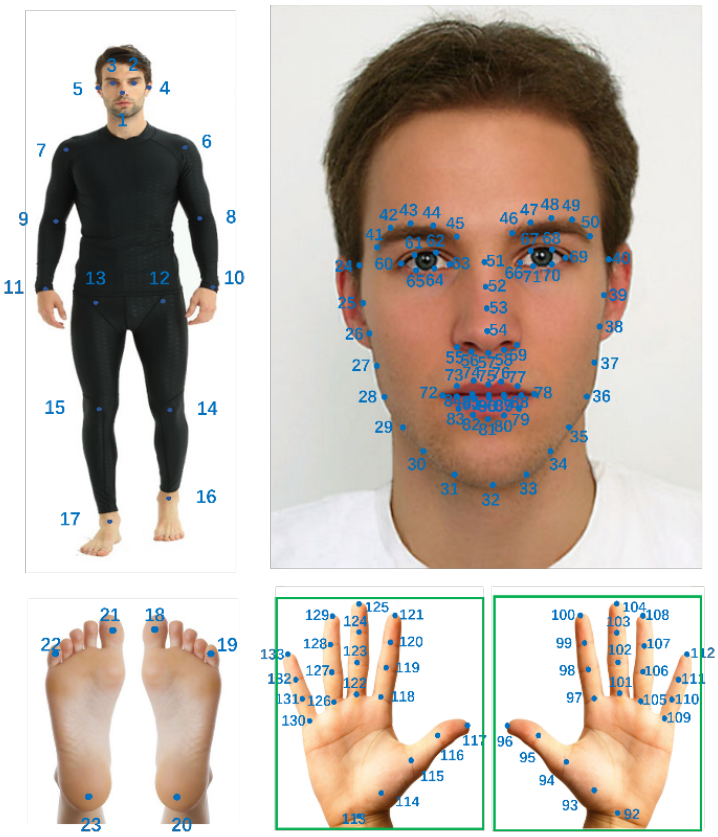


Figure 10. Extracted body keypoints and their numbered positions. Source: [33].

3.4.2 Keypoint Features

The 121 keypoints that were trained together during this part of the experiment were connected to obtain approximate outlines of each body part, which allowed all bodily movements to be captured, even subtle ones. Since the presence of too much information during model training can result in poor model performance, an effort was made to limit keypoint extraction to small numbers of "feature keypoints." Although these decisions affected the capability to capture the most subtle movements of key body parts, the extracted information was sufficient for research purposes. The 39 feature keypoints listed in Table 3 include 7 limb, 22 hand (11 each for left and right), and 11 face.

Table 3. Body keypoints used to test the proposed model.

Category	Keypoints
Limb (7)	nose (1), ears (4, 5), shoulders (6, 7), elbows (8, 9)
Face (10)	eyebrows (41, 43, 45, 46, 48, 50), mouth (84, 86, 88, 90)
Left hand (11)	wrist (92), thumb (94, 96), index finger (97, 100), middle finger (101, 104), ring finger (105, 108), little finger (109, 112)
Right hand (11)	wrist (113), thumb (115, 117), index finger (118, 121), middle finger (122, 125), ring finger (126, 129), little finger (130, 133)

3.5 Forecast model

3.5.1 Pose Estimate Model

Experiments conducted to test the use of keypoints for sign language identification utilized the temporal and spatial graph convolution network (GCN) model proposed by Yan et al. [34]. To obtain body position information, keypoints must be connected to skeletal data in order to construct two-dimensional graphs consisting of points and edges. In order to capture position changes over time, corresponding points in adjacent frames must be connected for use as model input (Fig. 11).

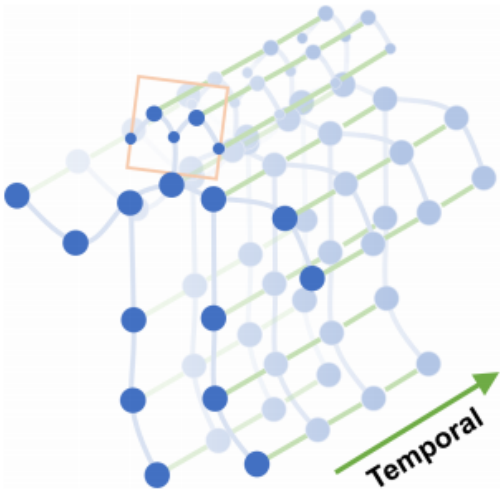


Figure 11. Schematic diagram of model input. Blue dots are body keypoints, blue lines the supporting framework, and green lines connectors between time dimension points. Source: Yan et al. [34].

3.5.2 RGB model

RGB image processing required the use of Tran et al.'s [35] ResNet2+1D convolutions, a variant that applies 1D convolution to 3D ResNet and uses pre-trained weights with a Kinectics data set. This model separates the original TxHxW 3D convolution kernel into a 1xHxW 2D convolution kernel (for dealing with spatial features) and a Tx1x1 1D convolution kernel (for dealing with temporal features). The error rate is reduced by increasing the number of non-linear layers.

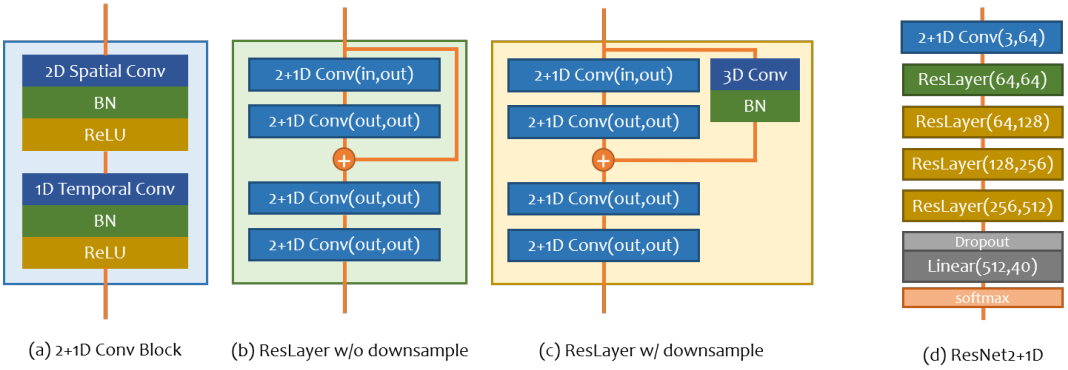


Figure 12. Architecture of the ResNet2+1D model. Source: Tran et al. [35].

3.6 Model ensemble learning

During training, different models focus on different features in the same data set, with different weights given to individual features. This produces distinctly different results across models, which encourages the use of ensembles to vote for individual models or to create weighted averages of model results so that all input data features can be at least partially acknowledged. This is a common deep learning technique. In this study, the model architecture supports the attainment of prediction and RGB image output. GCN and 3DCNN model outputs were expressed as vectors with lengths = 40, indicating the probability of video input being 40 sign language vocabulary items. The two prediction results were given different weights that were summed to achieve the highest possible accuracy:

$$predict_{final} = \alpha \times predict_{pose} + \beta \times predict_{RGB},$$

with predict denoting the model output and α and β the weights of the two models. During the verification step, weight distribution was adjusted according to model accuracy to achieve the best results.

4. EXPERIMENT

4.1 Experiment details

CPU: AMD Ryzen7 3700X

GPU: GeForce RTX 2070

Operating system: Ubuntu 18.04

Programming language: Python 3.7

Deep learning framework: Pytorch 1.8.1

4.2 Model Design

4.2.1 Human Body Keypoints

During the first phase of our experiments, the COCO-WholeBody dataset was used to remove 12 lower-body keypoints (not required for sign language communication), including waist (12, 13), knee (14, 15), ankle (16, 17) and foot (6)

keypoints. The remaining 121 keypoints were used for model training and prediction. Vertical and horizontal coordinates for these keypoints were used as input, followed by 100 epochs of training. Accuracy data for the Top1, Top3 and Top5 verification trends are shown in Figures 13a and b. In the figures, Top 1 refers to the largest final probability vector prediction result; a correct prediction indicates a correct result classification. Top 3 refers to the three largest and Top 5 the five largest probability vectors, with correct predictions indicating correct probabilities.

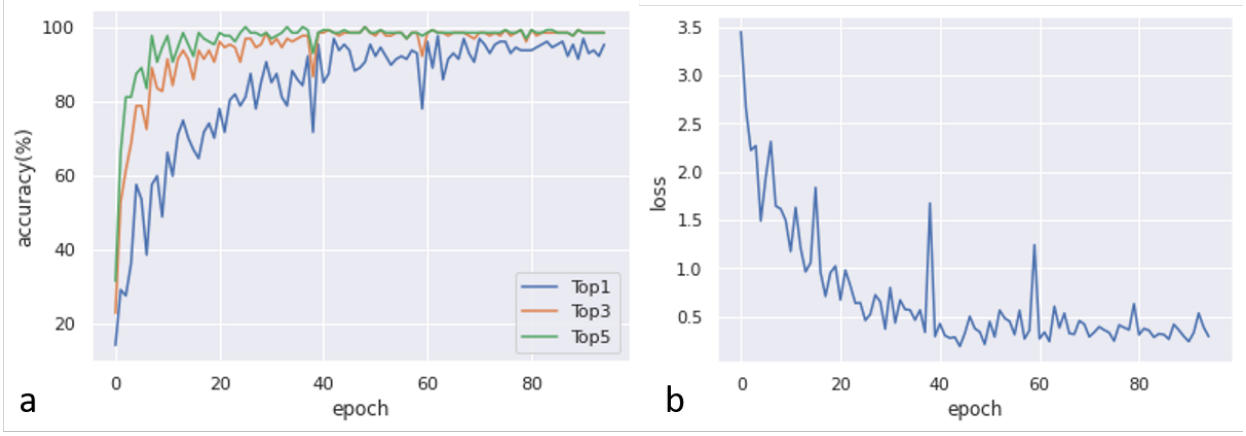


Figure 13. Accuracy data for the Top1, Top3 and Top5 verification data trends from (a) the training of 121 full keypoints and (b) downward loss trends.

The results indicate approximately 95% Top-3 and Top-5 accuracy rates after 20 epochs, and stabilized Top-1 accuracy after 60 epochs. Top-1 accuracy during this phase of our experiments was 94.9%; for both Top-3 and Top-5 it was 99.3%. According to Figure 13b, there was a downward loss trend due to the excessive information produced by the 121 keypoints.

A confusion matrix of experimental results during this stage is presented as Figure 14. According to this matrix, “at once,” “not right” and “thank” were poorly performing categories—their similar actions are distinguished by slightly different gestures. Comparable characteristics were noted for two other poorly performing categories: “don’t know” and “dislike.” A likely explanation for these findings is the presence of excessive feature point noise affecting gesture detection accuracy.

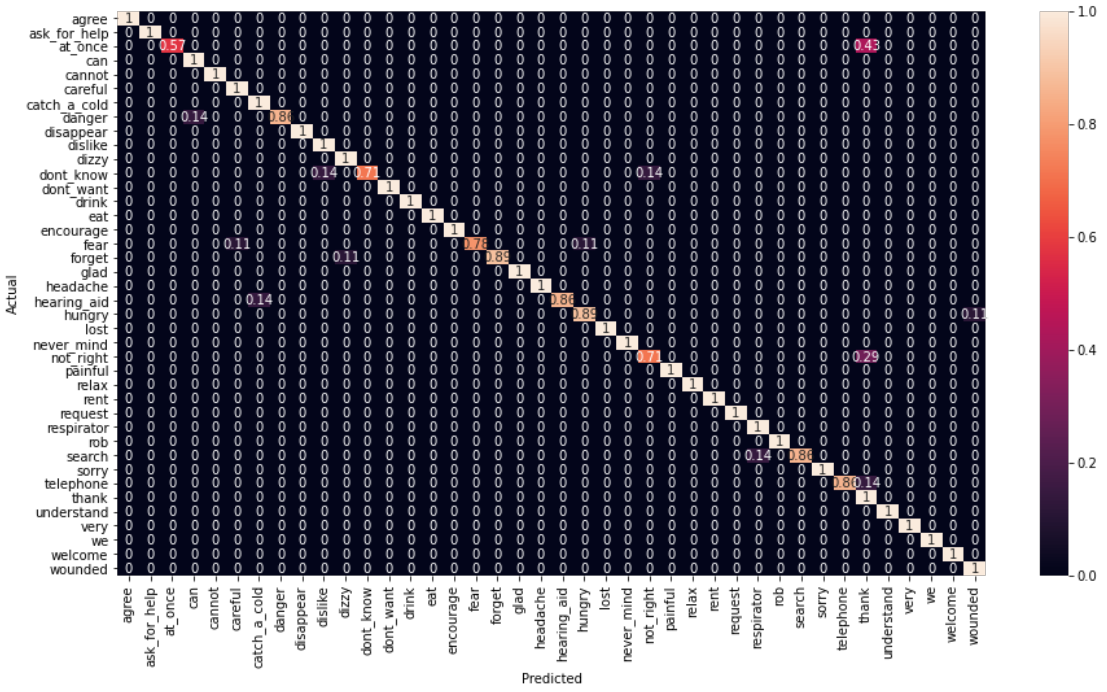


Figure 14. Confusion matrix following full keypoint training.

4.2.2 Feature Keypoints

During the second experiment phase, 39 of the original 121 body keypoints were identified as sufficient representations of key body parts for training and test data purposes. All unnecessary and redundant information was removed to improve model performance. The coordinates of these 39 keypoints were used during training (100 epochs). As shown in Figures 15a and b, Top-3 and Top-5 accuracy results stabilized after reaching approximately 95% after 20 epochs of training; Top-1 accuracy stabilized after 40 epochs—significantly faster than during the first

stage (≈60 epochs). Specific accuracy results were Top-1, 97.9% and Top-3 and Top-5, both 100%. Fewer spikes are noted in Figure 15b, indicating greater stability during the training process.

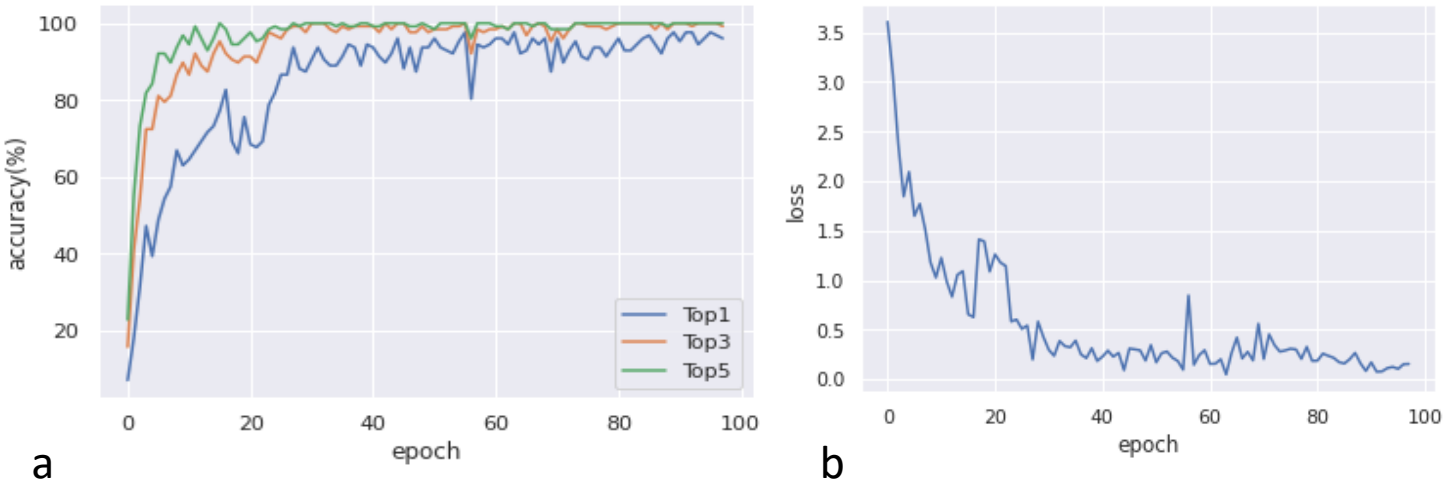


Figure 15. Top1, Top3 and Top5 accuracy data during (a) feature keypoint training and (b) downward loss trends.

A confusion matrix of experimental results during this stage is presented as Figure 16. Note that “don’t know” and “dislike” performed better during this stage compared to the first stage; similar improvements were observed to a lesser degree for “at once,” “not right” and “thank.” Note also the stronger focus on gesture changes.

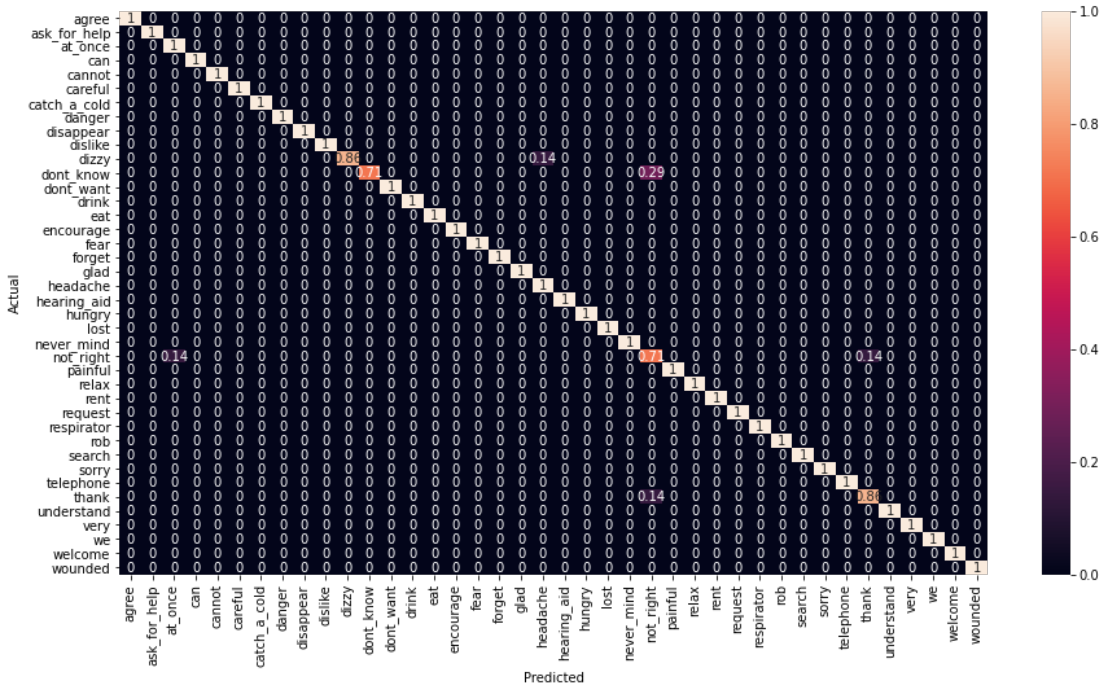


Figure 16: Confusion matrix following feature keypoint training.

4.3 RGB model

In the third experiment phase, 3DCNN was used to identify the continuous RGB graphics (see section 3.3.1). The R(2+1)D model was used to disassemble the 3D convolution kernel in 3D-ResNet, thus creating a 2D + 1D convolution kernel variant capable of separately performing spatial and temporal processing for purposes of optimizing the model's training process. A total of 59,610 pieces of original image information was use for experiment input, with the input dimension expressed as (channel, frame, size_x, size_y) = (3, 70, 64, 64) with a learning rate of 0.001 (100 epochs). Results are shown in Figures 17 and 18. Training and verification accuracy values were approximately 95% after 70 epochs. Accuracy stabilized while verification loss continued to decrease steadily; overall, the training process became relatively stable. Accuracy values during this stage were Top-1, 97.6% and Top-3 and Top-5, both 99.3%.

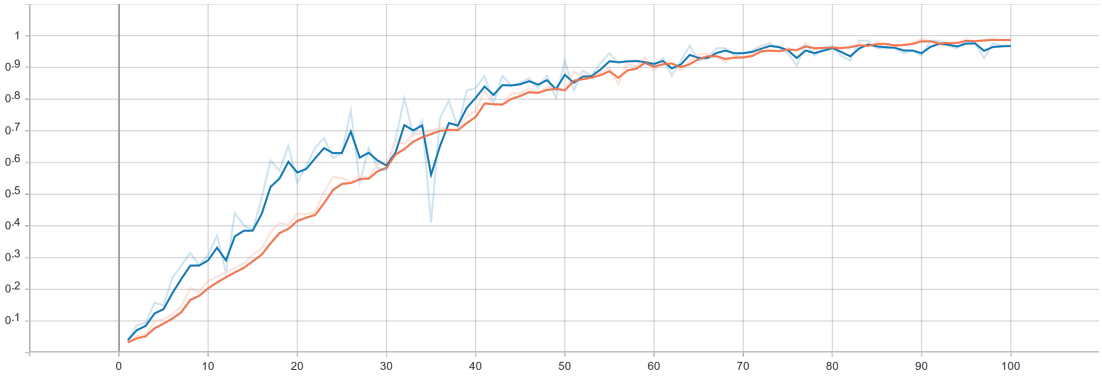


Figure 17. Accuracy trends during 3DCNN training. Blue line, training accuracy; orange line, verification accuracy.

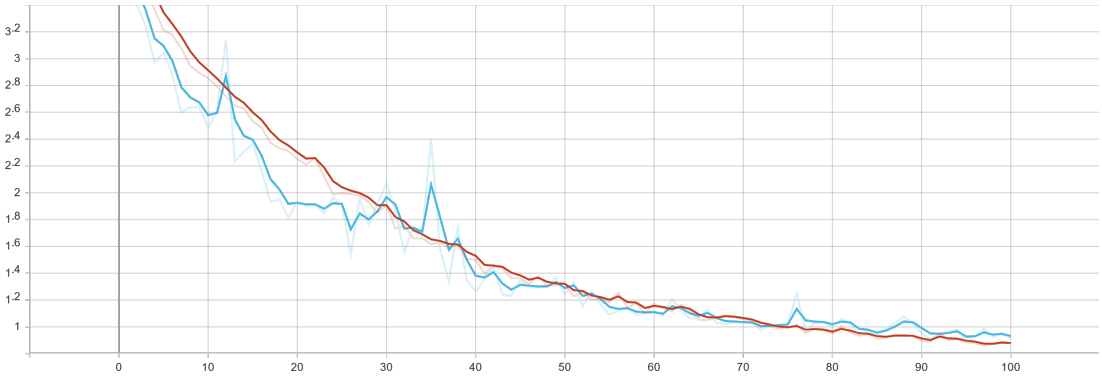


Figure 18. 3DCNN training loss trends. Blue line, training loss; red line, verification loss.

A confusion matrix of experimental results for this stage is presented as Figure 19. Compared to the second stage, the RGB model results were less stable when dealing with vocabulary items, with large differences between individual gestures and expressions. For example, there was a 14% probability of “don't know” being misread as “not right,” “painful,” or “understand.” As shown in Figure 20, these four signs are all expressed with one hand, with very small differences between them. According to the poor performance results for the terms “at once,” “not right” and “thank,” the RGB model is more sensitive to changes in motion. Further, better RGB performance was noted for related vocabulary recognition.

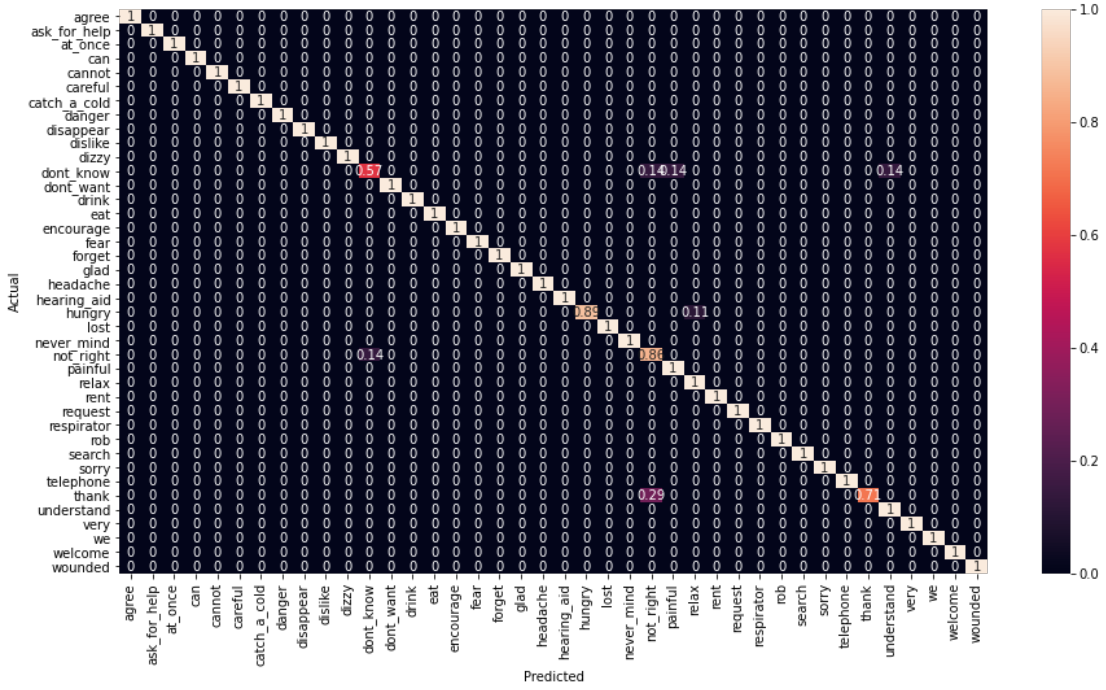
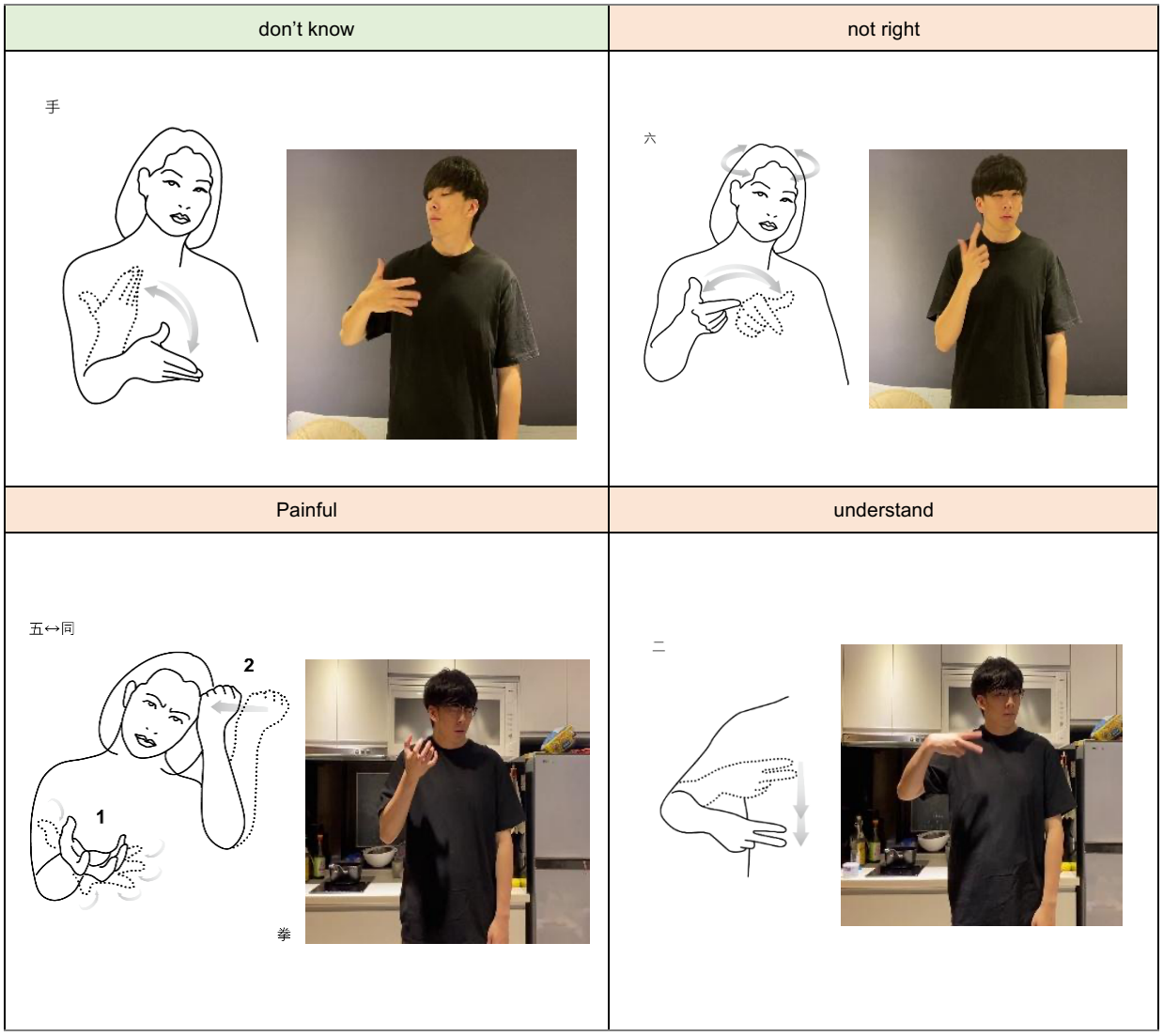


Figure 19. Confusion matrix constructed from data collected following RGB model training.

Figure 20. A comparison of signs for four terms. Source: Taiwan Ministry of Education online dictionary of commonly used sign language terms, located at https://signlanguage.moe.edu.tw/.



4.4 Model Ensemble

For the next stage, the human pose and RGB models were purposefully integrated to determine whether prediction accuracy could be improved. Specifically, an effort was made to determine best performance when model weights were 0.55 and 0.45, respectively, using the formula

$$predict_{final} = 0.55 \times predict_{pose} + 0.45 \times predict_{RGB}$$

Post-model integration results indicate an accuracy rate of 98.6%. Top-3 and Top-5 accuracy rates were both 100% (Table 4).

Table 4. Accuracy rates for individual and integrated models.

Method	Top-1	Top-3	Top-5
Joint-121	94.9%	99.3%	99.3%
Joint-39	97.9%	100%	100%
RGB	97.6%	99.3%	99.3%
Ensemble (RGB + Joint-39)	98.6%	100%	100%

A confusion matrix constructed from the experimental results for this stage is shown as Figure 21. Note that “don’t know,” “thank” and “not right” are shown as having incorrect predictions. Improved stability was noted compared to the second and third stages. Other keypoint model weaknesses also exhibited improvement due to the RGB model’s greater motion sensitivity characteristic. Other errors were the same in both models.

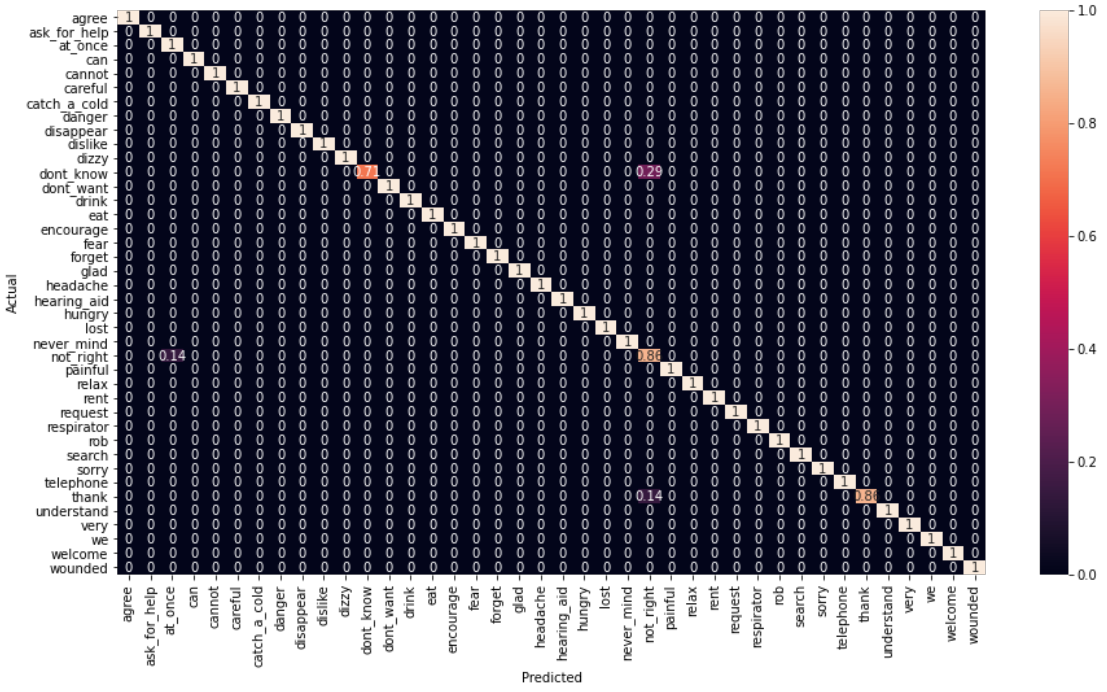


Figure 21. A confusion matrix from data collected following model integration.

5. CONCLUSION

Most sign language recognition systems require wearable devices or depth cameras to capture and analyze signer movement. The goal of our research is to reduce dependency on such equipment in order to help signers communicate more easily with people lacking any knowledge of sign language. The method described in this paper is

based on RGB image data. According to results from a series of experiments, the method is capable of identifying 40 signs in the Taiwanese sign language system. The proposed model consists of two integrated sub-models: a GCN model that uses human body keypoints for prediction purposes, and a 3DCNN model that recognizes RGB images.

First-stage experiment results using 121 upper-body keypoints for model training indicate a Top-1 accuracy rate of 96.8%. During the second stage, redundant information was removed in an effort to improve performance. Data for 39 selected keypoints indicate a higher Top-1 accuracy rate of 97.9% and a 100% Top-3 accuracy rate. Complete screen information was added during the third stage, in which RGB images were used for sign language recognition; here the Top-1 accuracy rate was 97.6%. For the fourth stage, prediction results generated during the second and third stages were weighted and added so that the model could concurrently refer to bodily motion and RGB changes, resulting in recognition accuracy values of 98.6% for Top-1 and 100% for Top-3. According to the confusion matrix constructed from these data, GCN and 3DCNN model integration successfully addressed the problem of identification errors involving similar signs when the RGB model was used alone.

In the absence of a complete Taiwanese sign language database, this research was limited to producing its own videos. Due to manpower and time limitations, only 40 common vocabulary items could be used for training purposes, with each item being the focus of approximately 20 short videos. Thus, even though a recognition accuracy rate of 99% was noted for the final experiment, lack of item diversity must be taken into consideration when interpreting the findings.

6. FUTURE RESEARCH

To overcome limitations associated with a lack of item diversity, it is essential to establish a large-scale Taiwanese sign language database in order to create sufficient training sets. Such a database requires waist-up images of signers standing in front of a variety of backgrounds under different intensities of light. For each vocabulary item, database producers should ask several signers to participate in video production to ensure a diverse body of data, making it possible to create comprehensive and robust models. Further, any successful sign language database should contain combinations of signs that are conducive to creating sentences. When such a database is established, an important next step will be to refine the proposed model in order to create apps that can be used with smart phones and other small-scale devices.

References

[1] F. Zhang, X. Zhu, H. Dai, M. Ye and C. Zhu, "Distribution-Aware Coordinate Representation for Human Pose Estimation," 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 7091-7100, 2020.

[2] R. Anderson, F. Wiryana, M. C. Ariesta and G. P. Kusuma, "Sign language recognition application systems for deaf-mute people: A review based on input-process-output", Procedia Computer Science, vol. 116, pp. 441-448, Oct. 2017.

[3] 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 (IJMLC), vol. 10, no. 1, pp. 131-153, Jan. 2017.

[4] H. Cheng, L. Yang and Z. Liu, "Survey on 3D Hand Gesture Recognition," IEEE Transactions on Circuits and Systems for Video Technology, vol. 26, no. 9, pp. 1659-1673, Sept. 2016.

[5] K. Imagawa, Shan Lu and S. Igi, "Color-based hands tracking system for sign language recognition," Proceedings Third IEEE International Conference on Automatic Face and Gesture Recognition, Nara, pp. 462-467, 1998.

[6] C. R. Wren, A. Azarbayejani, T. Darrell and A. P. Pentland, "Pfinder: real-time tracking of the human body," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 19, no. 7, pp. 780-785, July 1997.

[7] J. Huang, W. Zhou, H. Li and W. Li, "Sign language recognition using 3d convolutional neural networks", ICME, pp. 1-6, 2015.

[8] L. Pigou, S. Dieleman, P.-J. Kindermans and B. Schrauwen, "Sign language recognition using convolutional neural networks", Proc. Eur. Conf. Comput. Vis. Pattern Recog. Workshops, pp. 1-6, 2014.

[9] H. Brashear, T. Starner, P. Lukowicz and H. Junker, "Using multiple sensors for mobile sign language recognition," Seventh IEEE International Symposium on Wearable Computers, pp. 45-52, 2003.

[10] M. Oberweger, P. Wohlhart and V. Lepetit, "Hands deep in deep learning for hand pose estimation", 2015

[11] X. Chen, G. Wang, H. Guo and C. Zhang, 'Pose Guided Structured Region Ensemble Network for Cascaded Hand Pose Estimation,"2017.

[12] P. S. Rajam and G. Balakrishnan, "Real time Indian Sign Language Recognition System to aid deaf-dumb people," 2011 IEEE 13th International Conference on Communication Technology, Jinan, pp. 737-742, 2011.

[13] Q. De Smedt, H. Wannous and J. Vandeborre, "Skeleton-Based Dynamic Hand Gesture Recognition," 2016 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Las Vegas, NV, pp. 1206-1214, 2016.

[14] A. Kuznetsova, L. Leal-Taixé and B. Rosenhahn, "Real-Time Sign Language Recognition Using a Consumer Depth Camera," 2013 IEEE International Conference on Computer Vision Workshops, pp. 83-90, 2013.

[15] T. Starner and A. Pentland, "Real-Time American Sign Language Recognition From Video Using Hidden Markov Models", Proc. Int'l Symp. Computer Vision, 1995.

[16] G. A. Rao, K. Syamala, P. V. V. Kishore and A. S. C. S. Sastry, "Deep convolutional neural networks for sign language recognition," 2018 Conference on Signal Processing And Communication Engineering Systems (SPACES), Vijayawada, pp. 194-197, 2018.

[17] R. Cui, H. Liu and C. Zhang, "Recurrent Convolutional Neural Networks for Continuous Sign Language Recognition by Staged Optimization," 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, pp. 1610-1618, 2017.

[18] J. C. Núnez, R. Cabido, J. J. Pantrigo, A. S. Montemayor and J. F. Vélez, "Convolutional neural networks and long short-term memory for skeleton-

based human activity and hand gesture recognition", Pattern Recognition, vol. 76, pp. 80-94, Apr. 2018.

[19] S. Ji, W. Xu, M. Yang and K. Yu, "3D Convolutional Neural Networks for Human Action Recognition," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 35, no. 1, pp. 221-231, Jan. 2013.

[20] D. Guo, S. Wang, Q. Tian and M. Wang, "Dense temporal convolution network for sign language translation", Proc. 28th Int. Joint Conf. Artif. Intell., pp. 744-750, Aug. 2019.

[21] D. Guo, S. Wang, Q. Tian and M. Wang, "Dense temporal convolution network for sign language translation", Proc. 28th Int. Joint Conf. Artif. Intell., pp. 744-750, Aug. 2019.

[22] J. Kim, S. Mastnik and E. Andr, "EMG-based hand gesture recognition for realtime biosignal interfacing", Proc. 13th Int. Conf. Intell. User Interfaces, pp. 30-39, 2008.

[23] P. Felzenszwalb and D. Huttenlocher, "Pictorial Structures for Object Recognition", International Journal of Computer Vision (IJCV), vol. 61, no. 1, pp. 55-79, 2005.

[24] A. Toshev and C. Szegedy, "DeepPose: Human Pose Estimation via Deep Neural Networks," 2014 IEEE Conference on Computer Vision and Pattern Recognition, pp. 1653-1660, 2014.

[25] J. Tompson, R. Goroshin, A. Jain, Y. LeCun and C. Bregler, "Efficient object localization using Convolutional Networks," 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 648-656, 2015.

[26] J. Wang, K. Sun, T. Cheng, B. Jiang, C. Deng, Y. Zhao, et al., "Deep high-resolution representation learning for visual recognition", IEEE Transactions on Pattern Analysis and Machine Intelligence, Apr. 2020.

[27] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh, “Realtime multi-person 2d pose estimation using part affinity fields,” in CVPR, 2017.

[28] M. Kocabas, S. Karagoz and E. Akbas, "MultiPoseNet: Fast multi-person pose estimation using pose residual network", Proc. ECCV, pp. 417-433, Sep. 2018.

[29] G. Papandreou, T. Zhu, N. Kanazawa, A. Toshev, J. Tompson, C. Bregler, et al., "Towards accurate multi-person pose estimation in the wild", Proceedings of the IEEE conference on computer vision and pattern Recognition (CVPR), pp. 3711-3719, 2017.

[30] Y. Chen, Z. Wang, Y. Peng, Z. Zhang, G. Yu and J. Sun, "Cascaded pyramid network for multi-person pose estimation", Proceedings of the IEEE conference on computer vision and pattern Recognition (CVPR), pp. 7103-7112, 2018.

[31] B. Xiao, H. Wu and Y. Wei, "Simple Baselines for Human Pose Estimation and Tracking", Proceedings of the European Conference on Computer Vision, pp. 466-481, 2018.

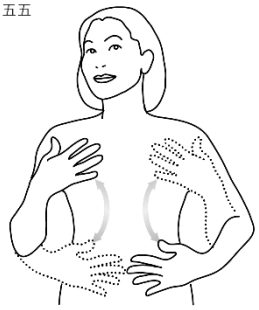
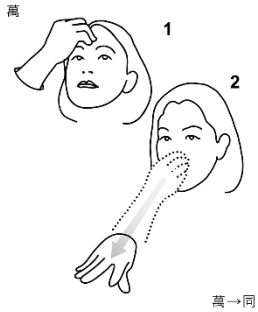
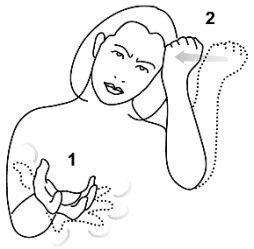
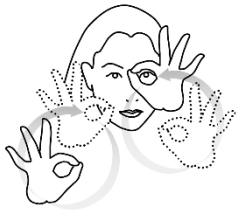
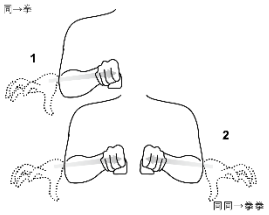
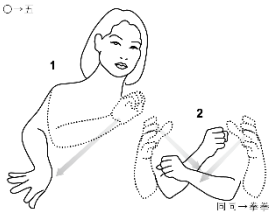
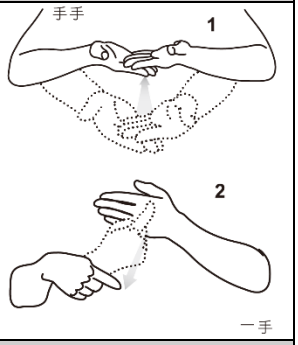
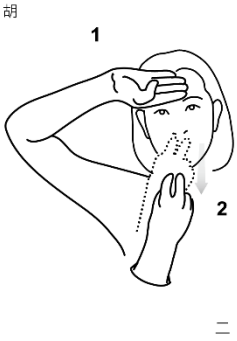
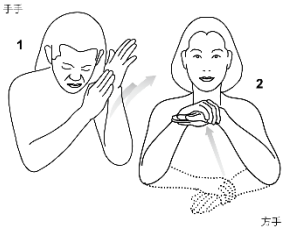
[32] J. Wang, K. Sun, T. Cheng, B. Jiang, C. Deng, Y. Zhao, et al., "Deep high-resolution representation learning for visual recognition", IEEE Transactions on Pattern Analysis and Machine Intelligence, Apr. 2020.

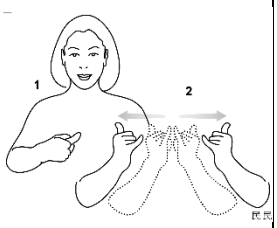
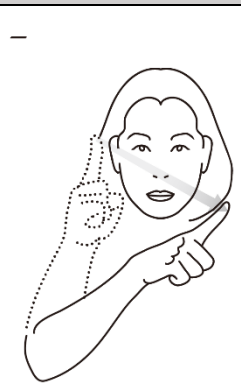
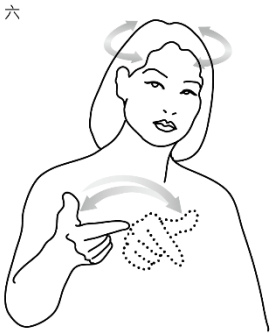
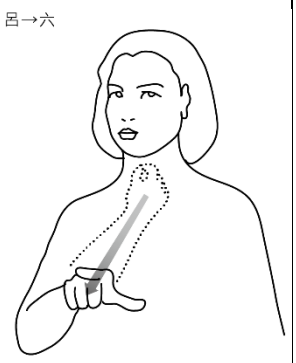
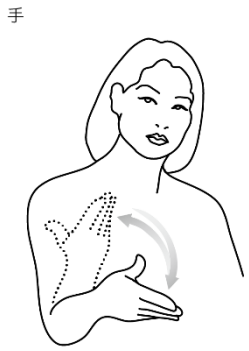
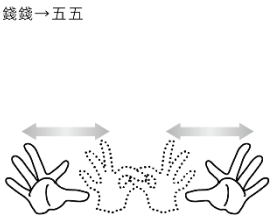
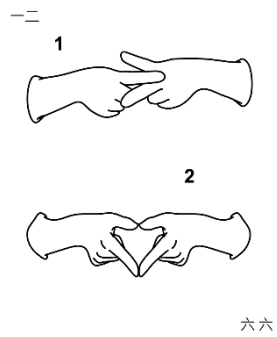
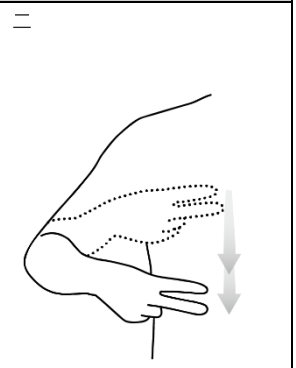
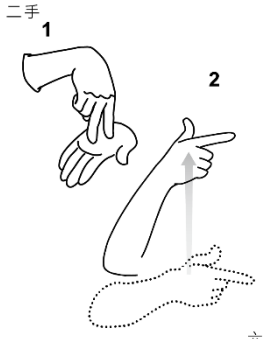
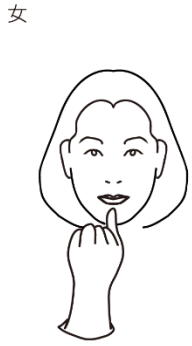
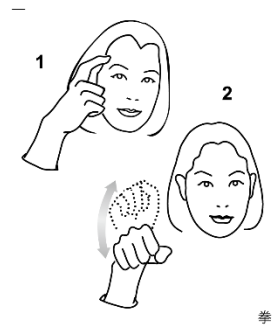
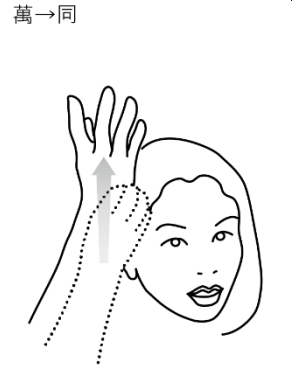
[33] S. Jin, L. Xu, J. Xu, C. Wang, W. Liu, C. Qian, W. Ouyang, and P. Luo, “Whole-body human pose estimation in the wild,” Proceedings of European Conference on Computer Vision, 2020.

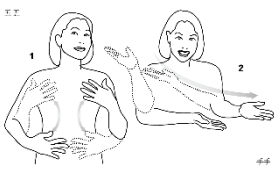
[34] S. Yan, Y. Xiong and D. Lin, "Spatial temporal graph convolutional networks for skeleton-based action recognition", Proc. AAAI, pp. 1-9, 2018.

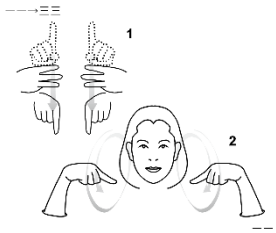
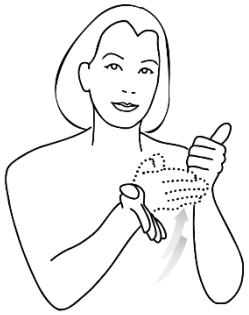
[35] D. Tran, H. Wang, L. Torresani, J. Ray, Y. LeCun and M. Paluri, "A Closer Look at Spatiotemporal Convolutions for Action Recognition," 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 6450-6459, 2018.

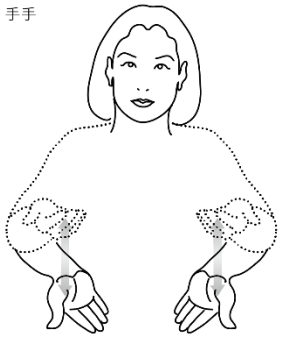
APPENDIX

afraid	happy	disgusted	painful
			
missing	**looking for**	**robbery**	**headache**
			
hunger	**lost**	**hearing aid**	**injured**
			
cold	**dizzy**	**help**	**danger**
			

we	no	not	do not want
			
do not know	**it is ok**	**careful**	**understand**
			
immediately	**can**	**agree**	**forget**
			

sorry	welcome	request	thanks
			

very	encourage	eat	drink
			

mask	rental	phone	rest
			

Source: Taiwan Ministry of Education, https://signlanguage.moe.edu.tw/.