Chapter

Using Mobile Phone to Assist DHH Individuals

Ming-Han Huang, Hsuan-Min Wang and Chuen-Tsai Sun

Abstract

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. This research aims 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 and hard-of-hearing (DHH) individuals.

Keywords: deep learning, sign language recognition, Taiwan sign language, 3DCNN, human poses, model ensembles

1. Introduction

Today we have many daily opportunities to watch deaf and hard-of-hearing (DHH) 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, DHH individuals use sign language to interact with hearing-enabled people who have learned sign language or have access to interpreters. To communicate with their DHH 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 language. 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 DHH individuals who do not have full-time access to interpreters. We believe such a tool will be especially useful for providing appropriate assistance to DHH individuals in emergency situations.

Presently, the most commonly used sign language recognition technologies consist of wearable devices and depth cameras [1]. 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 [2] whole-body estimation model to extract sign language information via images, and then integrates it with pretrained neural networks to achieve sign language recognition. Sign languages are distinctly national or regional, and currently, there is no model training dataset 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 already widely used.

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 [1]. 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 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 [20, 21].

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 and Huttenlocher [22] 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.

DeepPose [23], 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.

Another HPE research direction is heat map prediction technology [24], 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. 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 [25], among other tools.

2.3 DarkPose

DarkPose [2], a model-independent plug-in that optimizes heat map technology and verifies carry out computation optimization (COCO) and MPII datasets, was released by the University of Electronic Science and Technology of China in October 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 carryout computation optimization (COCO) dataset and the distribution-aware coordinate representation of keypoint (DARK) algorithm.

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

HPE model	Backbone	Input size	AP	AP^{50}	AP^{75}	AP^{M}	AP^{L}	AR
Bottom-up								
OpenPose [26]	–	–	61.8	84.9	67.5	57.1	68.2	66.5
MultiPoseNet [27]	–	–	69.6	86.3	76.6	65.0	76.3	73.5
Top-down								
G-RMI [28]	ResNet-101	353×257	64.9	85.5	71.3	62.3	70.0	69.7
CPN [29]	ResNet-Inception	384×288	73.0	91.7	80.9	69.5	78.1	79.0
SimpleBaseline [30]	ResNet-152	384×288	73.7	91.9	81.1	70.3	80.0	79.0
HRNet [25]	HRNet-W48	384×288	77.0	**92.7**	84.5	73.4	83.1	82.0
DARK [2]	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. Bold text indicates the best performance among all methods.*

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

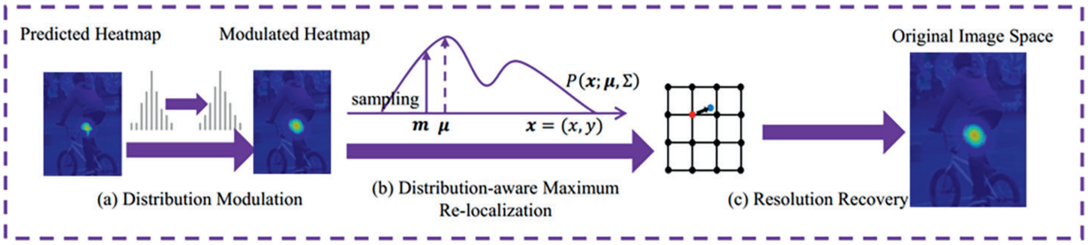

Figure 1.
Schematic of DARK-based optimization for decoding heat maps into coordinate points.

expansion was applied to calculate correct joint point positions (**Figure 1a** and **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 (**Figure 1c**). 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 the 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 nonquantized positions. Since coordinate point encoding usually refers to ground truth encoding into heat maps for model learning, many model training optimizations are possible.

3. Proposed solution

This study conducts experiments using GNN (Graph Neural Network) and ResNet models, which exhibit the following main advantages and characteristics:

1. GNN Model: The GNN model can effectively model the relationships between human keypoints, thereby better capturing spatial features of sign language gestures. Additionally, the GNN model can model the relationships between human keypoints at different time points, thus better capturing temporal features of sign language gestures. These characteristics make the GNN model perform excellently in sign language recognition tasks.

2. ResNet Model: The ResNet model is a deep residual network that effectively addresses the problem of vanishing gradients in deep neural networks, thus facilitating better training of deep models. In this study, the ResNet model is utilized for RGB image recognition, efficiently extracting image features and achieving superior recognition performance.

By combining the advantages and unique features of both models, this study can better capture spatial and temporal features of sign language gestures, resulting in improved sign language recognition performance. Moreover, by employing multiple models for sign language recognition, this research maximizes the benefits of each model and achieves superior recognition results.

3.1 Research structure

The main goal of this research is to create 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 dataset 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 dataset was used to separately train two models: a graph convolution network (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.

3.2 Data collection

Sign language datasets tend to be scattered due to regional and national differences. Currently, the most common sign language datasets 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 DHH 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.

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

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

3.3 Data preprocessing

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 dataset. 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 × 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.

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 preprocessing 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 (**Figure 2**). Further, OpenPose frame rates

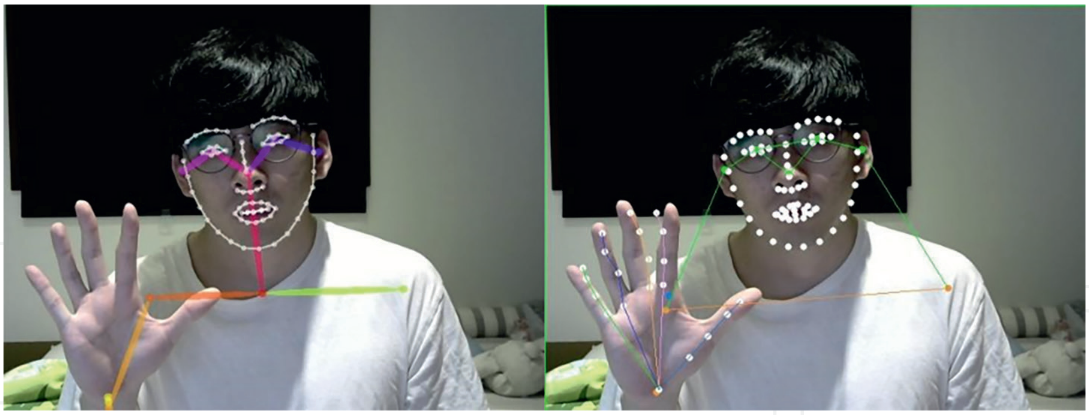

Figure 2.
OpenPose (left) did not detect hand positions when the subject's elbow was not visible. DarkPose (right) was not affected.

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.

3.4.1 Keypoint selection

The COCO WholeBody dataset [31] consisted of 133 extracted keypoints—17 limbs, 6 feet, 68 faces, and 42 hands (21 each for left and right hands). Three-dimensional datasets 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 is the keypoint confidence value (a floating point number between 0 and 1). The datasets 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.

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 limbs, 22 hands (11 each for left and right), and 11 faces.

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. [32]. To obtain body position information, keypoints must be connected to skeletal data in order to construct two-dimensional graphs consisting of points and

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)

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

edges. In order to capture position changes over time, corresponding points in adjacent frames must be connected for use as model input.

3.5.2 RGB model

RGB image processing required the use of Tran et al.'s [33] ResNet2+1D convolutions, a variant that applies 1D convolution to 3D ResNet and uses pretrained weights with a Kinectics dataset. 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 nonlinear layers.

3.6 Model ensemble learning

During training, different models focus on different features in the same dataset, 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:

$$predict_{final} = \alpha \times predict_{pose} + \beta \times predict_{RGB}, \tag{1}$$

with “predict” denotes the model output, and α and β denote 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 Top 1, Top 3, and Top 5 verification trends are shown in **Figure 3a** 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.

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 3b**, 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 4**. 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.

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

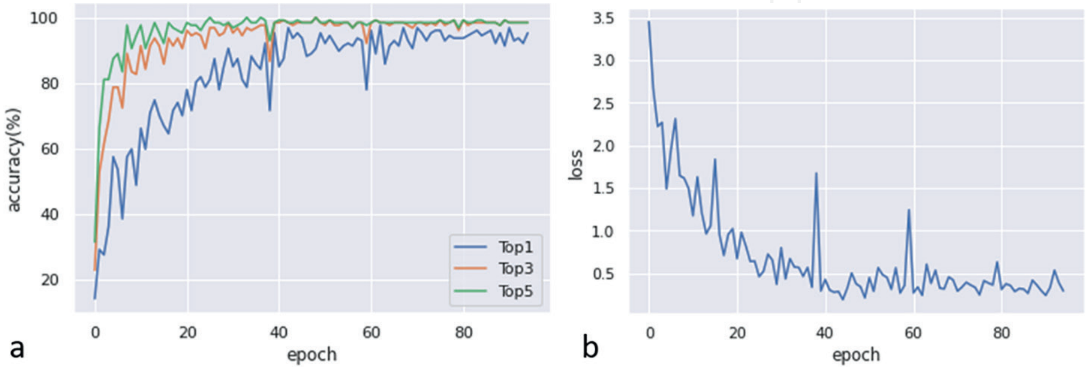

Figure 3.
Accuracy data for the Top 1, Top 3, and Top 5 verification data trends from (a) the training of 121 full keypoints and (b) downward loss trends.

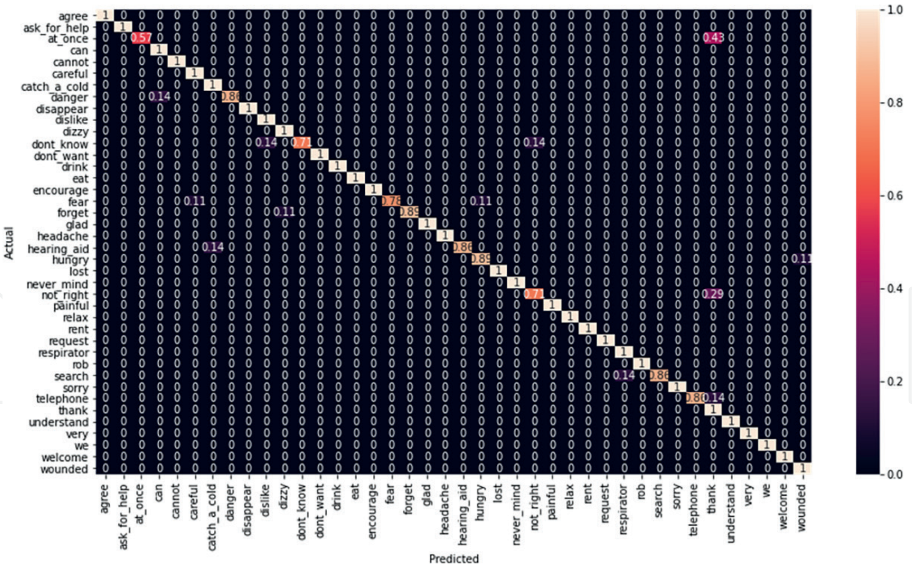

Figure 4.
Confusion matrix following full keypoint training.

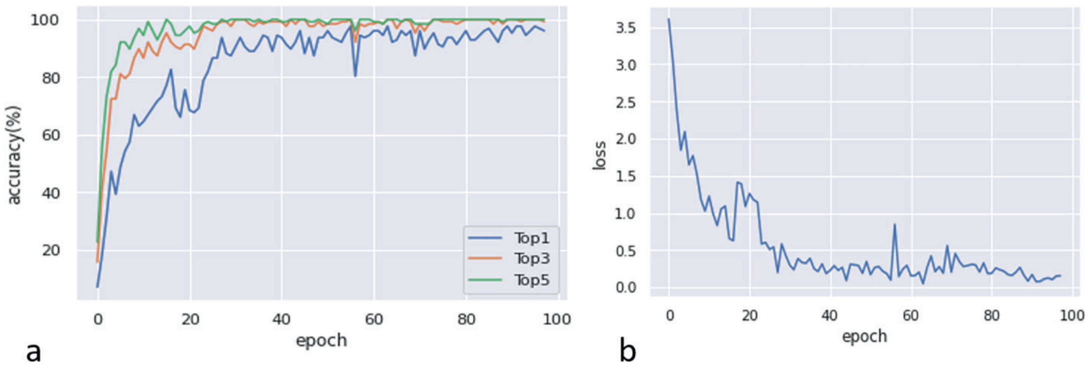

Figure 5.
Top 1, Top 3, and Top 5 accuracy data during (a) feature keypoint training and (b) downward loss trends.

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 **Figure 5a** 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 5b**, indicating greater stability during the training process.

A confusion matrix of experimental results during this stage is presented in **Figure 6**. 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.

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

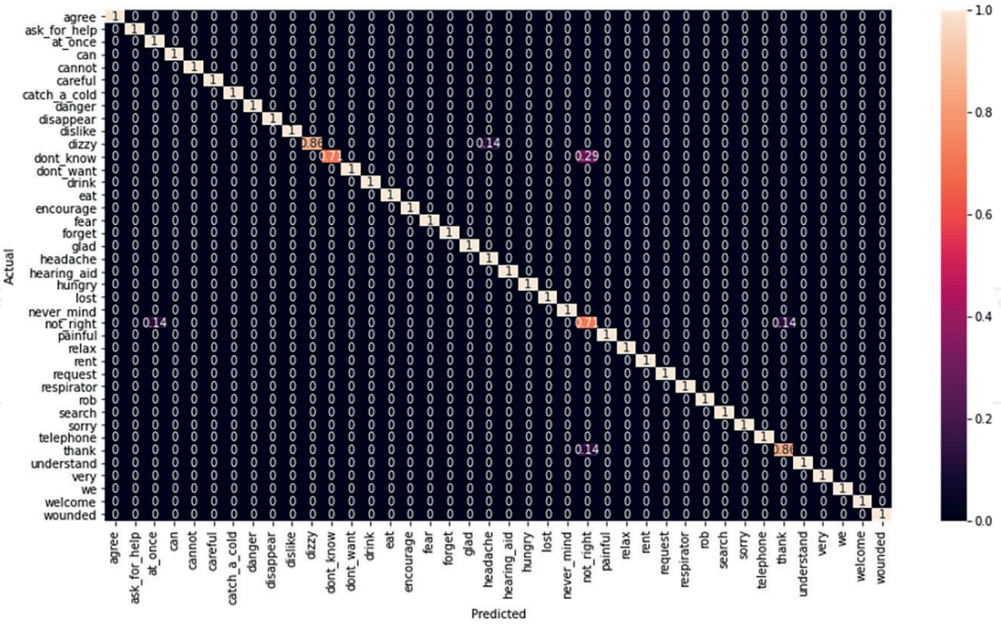

Figure 6.
Confusion matrix following feature keypoint training.

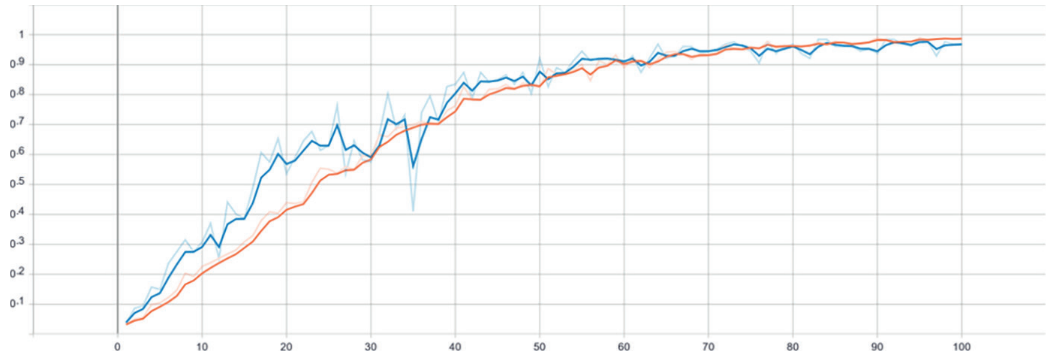

Figure 7.
Accuracy trends during 3DCNN training. Blue line, training accuracy; orange line, verification accuracy.

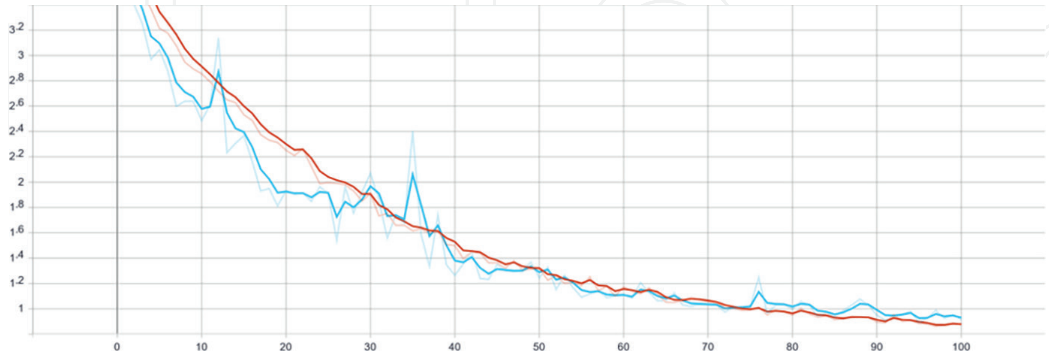

Figure 8.
3DCNN training loss trends. Blue line, training loss; red line, verification loss.

of optimizing the model's training process. A total of 59,610 pieces of original image information was used 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** 7 and **8**. 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%.

A confusion matrix of experimental results for this stage is presented as **Figure 9**. 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 10**, 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.

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 the best performance when model weights were 0.55 and 0.45, respectively, using the formula

$$predict_{final} = 0.55 \times predict_{pose} + 0.45 \times predict_{RGB} \qquad (2)$$

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

A confusion matrix constructed from the experimental results for this stage is shown in **Figure 11**. Note that "don't know," "thank," and "not right" are shown as having incorrect predictions. Improved stability was noted compared to the second

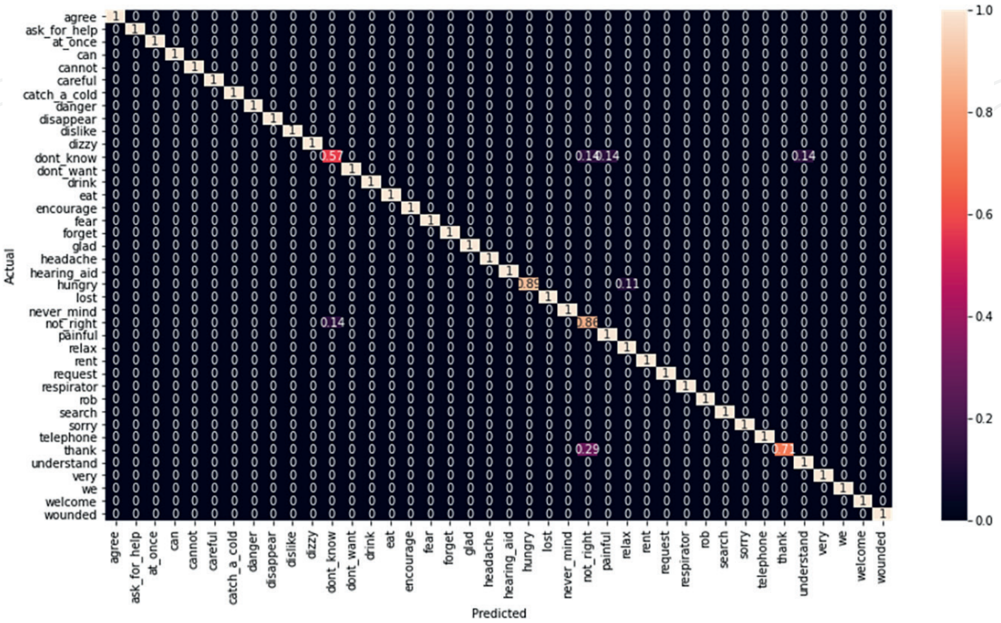

Figure 9.
Confusion matrix constructed from data collected following RGB model training.

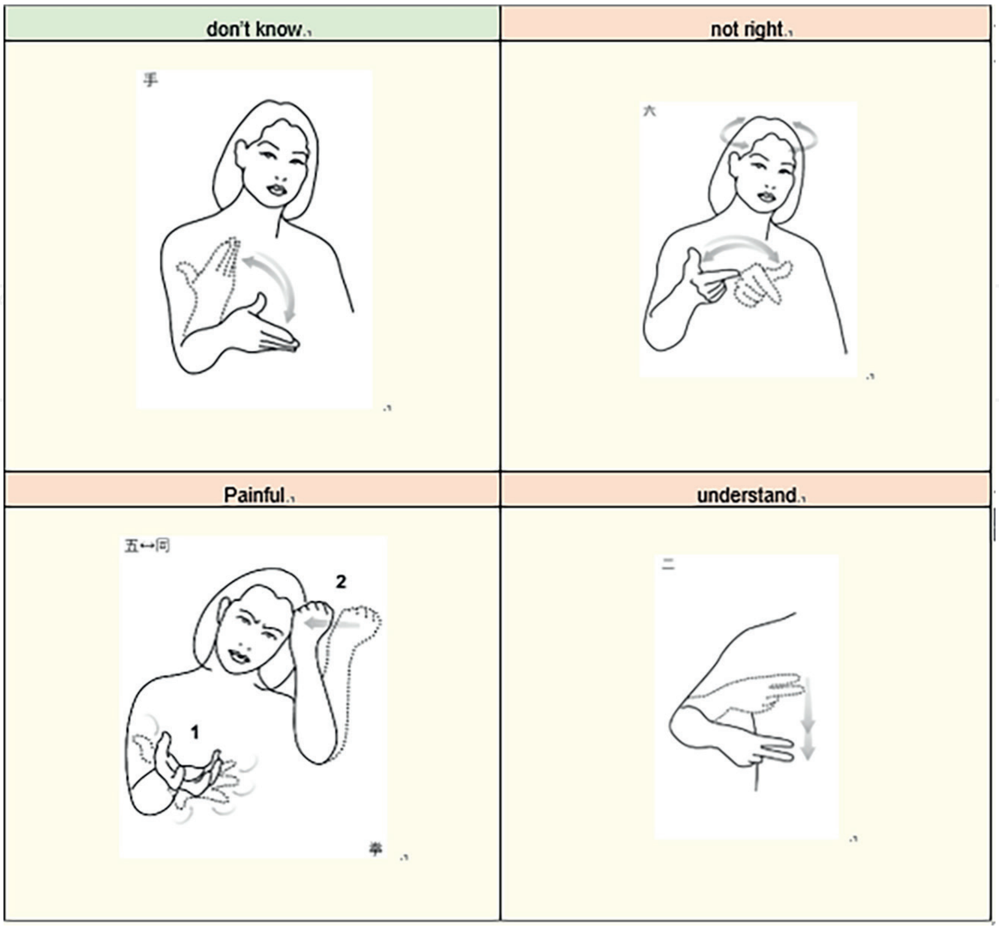

Figure 10.
A comparison of signs for four terms. Source: Taiwan Ministry of Education online dictionary of commonly used sign language terms, located at https://signlanguage.moe.edu.tw/.

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

Table 4.
Accuracy rates for individual and integrated models.

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.

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

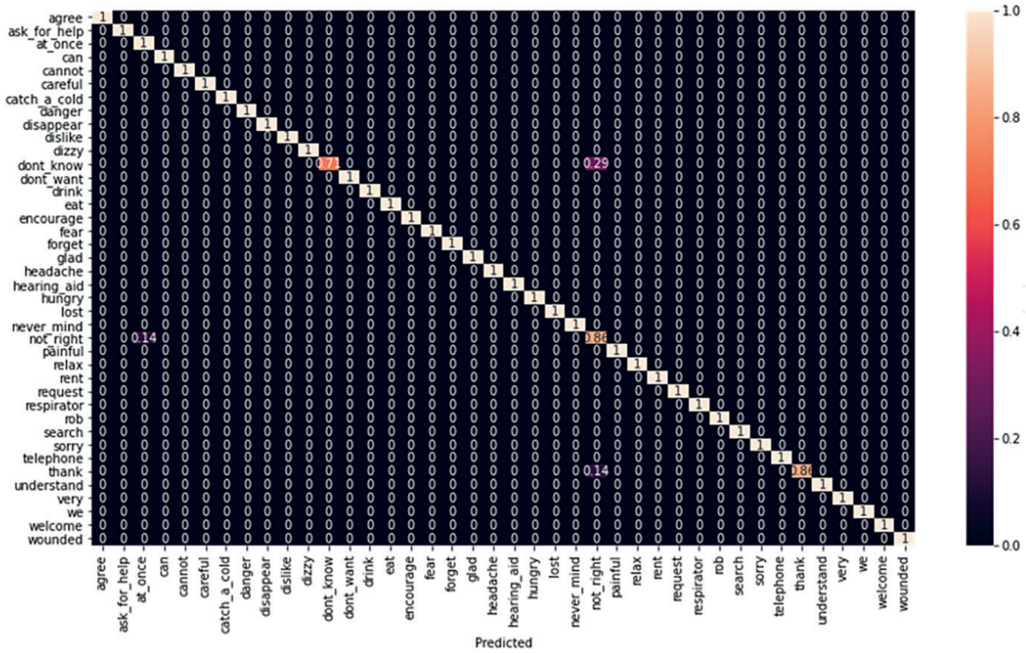

Figure 11.
A confusion matrix from data collected following model integration.

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 submodels: a GCN model that uses human body keypoints for prediction purposes, and a 3DCNN model that recognizes RGB images. For the GCN model, the batch size used in this experiment was 32, and the learning rate was set to 0.001. The model was trained for 60 epochs in the first stage of the experiment and 100 epochs in the second stage. As for the 3DCNN model, a batch size of 64 was used in the experiment, with a learning rate of 0.001. The model was trained for 100 epochs.

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. Compared to the work of DS Chen and SC Lo [33] using the YOLO model for Taiwanese Sign Language recognition, there is a clear improvement in recognition accuracy in this study.

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 smartphones and other small-scale devices.

Conflict of interest

The authors declare no conflict of interest.

Author details

Ming-Han Huang[1]*, Hsuan-Min Wang[2] and Chuen-Tsai Sun[2]

1 National Chiao Tung University, Hsinchu, Taiwan

2 National Yang Ming Chiao Tung University, Hsinchu, Taiwan

*Address all correspondence to: cwhuangbl32l@gmail.com

References

[1] Anderson R, Wiryana F, Ariesta MC, Kusuma GP. Sign language recognition application systems for deaf-mute people: A review based on input-process-output. Procedia Computer Science. 2017;**116**:441-448

[2] Zhang F, Zhu X, Dai H, Ye M, Zhu C. Distribution-aware coordinate representation for human pose estimation. In: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). Salt Lake City, Utah: IEEE; 2020. pp. 7091-7100

[3] Cheok MJ, Omar Z, Jaward MH. A review of hand gesture and sign language recognition techniques. International Journal of Machine Learning and Cybernetics (IJMLC). 2017;**10**(1):131-153

[4] Cheng H, Yang L, Liu Z. Survey on 3D hand gesture recognition. IEEE Transactions on Circuits and Systems for Video Technology. Sept. 2016;**26**(9):1659-1673

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

[6] Wren CR, Azarbayejani A, Darrell T, Pentland AP. Pfinder: Real-time tracking of the human body. IEEE Transactions on Pattern Analysis and Machine Intelligence. July 1997;**19**(7):780-785

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

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

[9] Brashear H, Starner T, Lukowicz P, Junker H. Using multiple sensors for mobile sign language recognition. In: Seventh IEEE International Symposium on Wearable Computers. Washington, DC: IEEE Computer Society; 2003. pp. 45-52

[10] Oberweger M, Wohlhart P, Lepetit V. Hands deep in deep learning for hand pose estimation. Czech Republic: Czech Society for Cybernetics and Informatics; 2015

[11] Chen X, Wang G, Guo H, Zhang C. Pose Guided Structured Region Ensemble Network for Cascaded Hand Pose Estimation. Amsterdam: Elsevier; 2017

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

[13] De Smedt Q, Wannous H, Vandeborre J. Skeleton-based dynamic hand gesture recognition. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW); Las Vegas, NV. 2016. pp. 1206-1214

[14] Kuznetsova A, Leal-Taixé L, Rosenhahn B. Real-time sign language recognition using a consumer depth camera. In: 2013 IEEE International Conference on Computer Vision Workshops. New York City: IEEE; 2013. pp. 83-90

[15] Starner T, Pentland A. Real-time American sign language recognition

from video using hidden Markov models. In: Proc. Int'l Symp. Computer Vision. Berlin, Heidelberg: Springer-Verlag; 1995

[16] Rao GA, Syamala K, Kishore PVV, Sastry ASCS. Deep convolutional neural networks for sign language recognition. In: 2018 Conference on Signal Processing and Communication Engineering Systems (SPACES). Vijayawada; 2018. pp. 194-197

[17] Cui R, Liu H, Zhang C. Recurrent convolutional neural networks for continuous sign language recognition by staged optimization. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI. 2017. pp. 1610-1618

[18] Núnez JC, Cabido R, Pantrigo JJ, Montemayor AS, Vélez JF. Convolutional neural networks and long short-term memory for skeleton-based human activity and hand gesture recognition. Pattern Recognition. 2018;**76**:80-94

[19] Ji S, Xu W, Yang M, Yu K. 3D convolutional neural networks for human action recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence. 2013;**35**(1):221-231

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

[21] Kim J, Mastnik S, Andr E. EMG-based hand gesture recognition for realtime biosignal interfacing. In: Proc. 13th Int. Conf. Intell. User Interfaces. New York: Association for Computing Machinery; 2008. pp. 30-39

[22] Felzenszwalb P, Huttenlocher D. Pictorial structures for object recognition. International Journal of Computer Vision (IJCV). 2005;**61**(1):55-79

[23] Toshev A, Szegedy C. DeepPose: Human pose estimation via deep neural networks. In: 2014 IEEE Conference on Computer Vision and Pattern Recognition. Salt Lake City, Utah: IEEE; 2014. pp. 1653-1660

[24] Tompson J, Goroshin R, Jain A, LeCun Y, Bregler C. Efficient object localization using convolutional networks. In: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Salt Lake City, Utah: IEEE; 2015. pp. 648-656

[25] Wang J, Sun K, Cheng T, Jiang B, Deng C, Zhao Y, et al. Deep high-resolution representation learning for visual recognition. In: IEEE Transactions on Pattern Analysis and Machine Intelligence. New York: IEEE Computer Society; 2020

[26] Cao Z, Simon T, Wei S-E, Sheikh Y. Realtime multi-person 2d pose estimation using part affinity fields. In: CVPR. Salt Lake City, Utah: IEEE; 2017

[27] Kocabas M, Karagoz S, Akbas E. MultiPoseNet: Fast multi-person pose estimation using pose residual network. In: Proc. ECCV. Berlin, Heidelberg: Springer-Verlag; 2018. pp. 417-433

[28] Papandreou G, Zhu T, Kanazawa N, Toshev A, Tompson J, Bregler C, et al. Towards accurate multi-person pose estimation in the wild. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Salt Lake City, Utah: IEEE; 2017. pp. 3711-3719

[29] Chen Y, Wang Z, Peng Y, Zhang Z, Yu G, Sun J. Cascaded pyramid network for multi-person pose estimation. In: Proceedings of the IEEE conference on

computer vision and pattern Recognition (CVPR). Salt Lake City, Utah: IEEE; 2018. pp. 7103-7112

[30] Xiao B, Wu H, Wei Y. Simple baselines for human pose estimation and tracking. In: Proceedings of the European Conference on Computer Vision. Berlin, Heidelberg: Springer-Verlag; 2018. pp. 466-481

[31] Jin S, Xu L, Xu J, Wang C, Liu W, Qian C, et al. Whole-body human pose estimation in the wild. In: Proceedings of European Conference on Computer Vision. Berlin, Heidelberg: Springer-Verlag; 2020

[32] Yan S, Xiong Y, Lin D. Spatial temporal graph convolutional networks for skeleton-based action recognition. In: Proc. AAAI. New York: Association for Computing Machinery; 2018. pp. 1-9

[33] Chen DS, Lo SC. Research of Taiwanese Sign Language Recognition Based on Deep Learning. Taiwan: National Dong Hwa University; 2022