mathematics

MDPI

Article

RFaNet: Receptive Field-Aware Network with Finger Attention for Fingerspelling Recognition Using a Depth Sensor

Shih-Hung Yang [1,*,†], Yao-Mao Cheng [2,†], Jyun-We Huang [1] and Yon-Ping Chen [2]

[1] Department of Mechanical Engineering, National Cheng Kung University, Tainan City 701, Taiwan; z10806026@ncku.edu.tw
[2] Institute of Electrical Control Engineering, National Yang Ming Chiao Tung University, Hsinchu City 300, Taiwan; mark228926.ee08@nycu.edu.tw (Y.-M.C.); ypchen@cc.nctu.edu.tw (Y.-P.C.)
* Correspondence: vssyang@gs.ncku.edu.tw; Tel.: +886-6-27-57575 (ext. 62171)
† These authors contributed equally to this paper.

Abstract: Automatic fingerspelling recognition tackles the communication barrier between deaf and hearing individuals. However, the accuracy of fingerspelling recognition is reduced by high intra-class variability and low inter-class variability. In the existing methods, regular convolutional kernels, which have limited receptive fields (RFs) and often cannot detect subtle discriminative details, are applied to learn features. In this study, we propose a receptive field-aware network with finger attention (RFaNet) that highlights the finger regions and builds inter-finger relations. To highlight the discriminative details of these fingers, RFaNet reweights the low-level features of the hand depth image with those of the non-forearm image and improves finger localization, even when the wrist is occluded. RFaNet captures neighboring and inter-region dependencies between fingers in high-level features. An atrous convolution procedure enlarges the RFs at multiple scales and a non-local operation computes the interactions between multi-scale feature maps, thereby facilitating the building of inter-finger relations. Thus, the representation of a sign is invariant to viewpoint changes, which are primarily responsible for intra-class variability. On an American Sign Language fingerspelling dataset, RFaNet achieved 1.77% higher classification accuracy than state-of-the-art methods. RFaNet achieved effective transfer learning when the number of labeled depth images was insufficient. The fingerspelling representation of a depth image can be effectively transferred from large- to small-scale datasets via highlighting the finger regions and building inter-finger relations, thereby reducing the requirement for expensive fingerspelling annotations.

Keywords: fingerspelling recognition; depth sensor; finger attention; receptive field; inter-finger relation

Citation: Yang, S.-H.; Cheng, Y.-M.; Huang, J.-W.; Chen, Y.-P. RFaNet: Receptive Field-Aware Network with Finger Attention for Fingerspelling Recognition Using a Depth Sensor. *Mathematics* **2021**, *9*, 2815. https://doi.org/10.3390/math9212815

Academic Editors: Grigoreta-Sofia Cojocar and Adriana-Mihaela Guran

Received: 27 September 2021
Accepted: 27 October 2021
Published: 5 November 2021

1. Introduction

For deaf people, sign language is a means to communicate. However, communication between deaf and hearing people remains challenging. Automatic sign language recognition tackles this communication barrier by translating sign language to text or speech. Fingerspelling is a sign language that signals words letter by letter. Fingerspelling enables the communication of technical terms and other terms lacking a representation in sign language. Note that ~35% of words in social interactions refer to technical topics requiring fingerspelling [1].

Vision-based fingerspelling recognition has been widely developed because cameras are inexpensive and ubiquitously available. Fingerspelling recognition systems may benefit from depth images acquired by structured light or time-of-flight sensors, which are robust to illumination variations [2] and enable easy hand detections against a complex background. However, intra-class variability, inter-class similarity, and inter-subject variability hinder vision-based fingerspelling recognition, as shown in Figure 1. The inter-class similarity refers to different fingerspelling signs sharing similar hand postures. The intra-class

variability refers to the various representations of identical signs captured from multiple views. The inter-subject variability refers to the various representations of identical signs performed by different subjects.

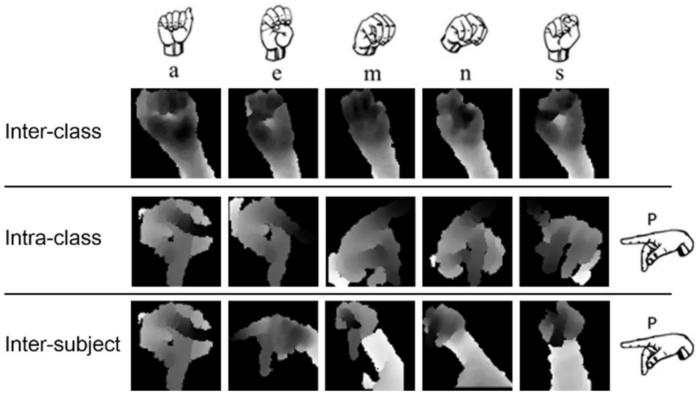

Figure 1. Inter-class similarity, intra-class variability, and inter-subject variability of hand gestures captured by a depth camera [3].

Accurate finger localization can potentially tackle inter-class similarity by detecting small hand posture variations between two signs. To account for finger localization, we propose a finger attention mechanism that enhances the finger regions in the depth image, highlighting discriminative finger features in these regions. Moreover, building inter-finger relations can tackle intra-class variability because inter-finger relations are inherently invariant to viewpoint changes. To account for inter-finger relations, we enlarge the receptive fields (RFs) of the convolutional kernels and model neighboring and inter-region dependencies between fingers. We now present the challenges of inter-class similarity and intra-class variability and illustrate our approach to handling them.

The first challenge is caused by inter-class similarity. A standard way to handle this is to highlight finger regions because accurate finger localization facilitates identifying slight posture variations across signs and further distinguishing signs that show inter-class similarity. The conventional method [4] assumes that finger localization and finger occlusion problems can be solved by the depth level. For this purpose, it manually decomposes the hand image into a few depth levels. Finger localization by this method is affected not only by manual predefinition of the depth level but also by the appearance of the forearm region at the depth level of the fingers. To deal with this issue, we highlighted the finger regions for low-level feature extraction using *depth finger attention* (DFA), as shown in Figure 2. DFA simultaneously considers the hand depth image and the forearm-removed image (non-forearm image) to explore the finger regions. There are two reasons behind simultaneously considering the hand depth image and non-forearm image. The first reason is that as most subjects signal with a preferred posture, the background pattern is sign-dependent. A model may tend to learn the sign representation according to the background pattern, which biases the learning toward the background [5]. When the fingers inside the hand region are not highlighted, the finger localization can be incorrect, and signs with inter-class similarity are poorly recognized. To handle the background-bias problem, we provide the non-forearm image as a reference for the hand depth image. The model then highlights the inside of the hand region rather than the outside region (the sign-dependent background pattern). The second reason is interference by forearm appearance in the hand depth image. Although the forearm can be removed from the hand image by detecting the wrist point [6], this approach is non-robust to occlusion of the wrist point by the fingers. Such occlusions can lead to inaccurate wrist point detection and unexpected finger removal. To tackle this limitation, we provide the hand depth image containing the forearm as a complementary reference for the non-forearm image. DFA can then highlight the finger regions on at least one of the two images.

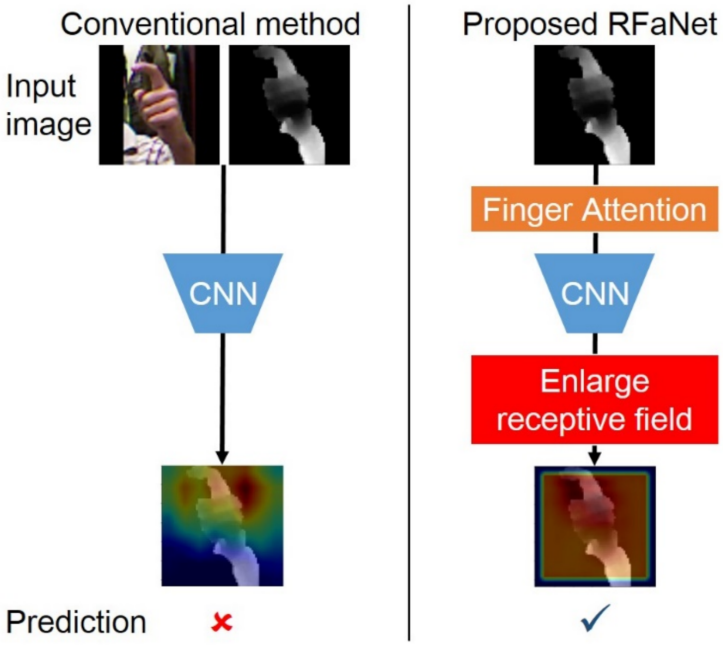

Figure 2. Hand gesture recognition models: (**left**) the conventional method and (**right**) RFaNet. Most conventional methods recognize hand gestures in color and depth input images, whereas RFaNet only processes a depth image to recognize hand gestures. RFaNet employs finger attention to highlight fingers before a CNN extracts the features and enlarges the RF to build long-range connections across finger features for better hand-gesture recognition.

The second challenge is caused by intra-class variability over multiple views. An identical sign viewed from multiple angles can have various representations in a convolutional neural network (CNN), leading to poor recognition. As convolutional operations focus on the local neighborhood, they capture the local finger features within a small RF (or field of view [7]), as shown in in Figure 2. When one sign is viewed from a different perspective, the change in local finger features leads to a variant representation. However, the long-range dependency between the fingers of an identical sign is invariant to viewpoint changes. Capturing the long-range dependency could improve the recognition of signs with intra-class variability. To handle this issue, we designed the second key component, a *non-local receptive field* (NLRF), that captures the neighboring and inter-region dependencies between fingers. The NLRF block employs atrous spatial pyramid pooling (ASPP) [7] to enlarge the field of view on multiple scales, and hence develops the long-range dependencies of distant fingers, as shown in Figure 2. Although ASPP varies the sampling distance from the kernel center, the feature maps from the previous convolutional layer have a uniform resolution. Consequently, the background enhancement is incorrect and the features are rendered less discriminative. Inspired by the receptive field block [8], we employed standard convolutional operations with various kernel sizes followed by the atrous convolution, accounting for the impact of RF eccentricities. However, directly merging the feature maps from various kernel sizes into a spatial pooling may model the dependency between fingers and the neighboring background rather than the dependency between distal fingers. The neighboring and inter-region dependencies are not simultaneously considered. To avoid this problem, we modified the non-local block [9] to further capture the dependencies of the feature maps extracted from various RFs. The non-local operation computed interactions between the multi-scale feature maps, and thus jointly captured the neighboring and inter-region dependencies across distal fingers, facilitating the modeling of inter-finger relations. Because the inter-finger relations of a sign are inherently invariant to viewpoint changes, a representation based on inter-finger relations could reduce intra-class variability.

Fingerspelling recognition systems may experience limited accuracy when the number of labeled images is insufficient. The number of labeled data can be increased by inviting

multiple subjects to perform hand gestures under various conditions, but this approach is expensive. Furthermore, the data annotation of hand gestures often requires specialized domain knowledge, which reduces the scalability of the data. Transfer learning tackles this issue by training a deep neural network model via sufficiently many data in a source domain and fine-tuning the model using small data in a target domain [10]. The source domain does not necessarily require relevance to the target domain but must share certain common representations with it. The representations learned from the large-scale datasets facilitate learning from the small-scale datasets. Nihal *et al*. [11] observed that computer-vision tasks share similar features. They trained a model on ImageNet and transferred the knowledge to Bangla sign alphabet recognition [12]. Observing similar hand gestures in British and American sign languages, Bird *et al*. [13] conducted transfer learning from British to American sign languages, based on color modality and bone modality (finger joints). However, the background of the color modality may affect transfer learning in this method. The depth modality could facilitate the transfer learning of fingerspelling recognition because finger features are robust to illumination changes and background complexity. Therefore, in this study, we only adopted depth modality for fingerspelling recognition and demonstrate its advantage in the application of transfer learning on limited training datasets.

The DFA and NLRF blocks were the key components for mitigating inter-class similarity and intra-class variability, respectively. We assembled the DFA and NLRF blocks to the top and bottom of a backbone network (VGG-9 [14]) and proposed a model—***R**eceptive **F**ield-**a**ware **Net**work with finger attention* (RFaNet)—for fingerspelling recognition, as shown in Figure 3. The primary contributions of the proposed model to fingerspelling recognition are summarized below.

1. We introduce RFaNet for effective fingerspelling recognition.
2. The DFA block on the top of RFaNet highlights the finger regions and facilitates the identification of slight hand-posture variations across signs with inter-class similarity.
3. The NLRF block at the bottom of RFaNet captures inter-finger relations by fusing multi-scale feature maps of various RFs. By learning the representations of inter-finger relations, the NLRF block improves the recognition of signs with intra-class variability because the representation of a sign is invariant to viewpoint changes.
4. RFaNet outperformed state-of-the-art methods on two standard benchmark fingerspelling datasets.
5. RFaNet effectively learned the fingerspelling representations from large- to small-scale datasets by highlighting the finger regions when the training data were insufficient.

The rest of the paper is organized as follows: Section 2 presents a review of related works in the literature; Section 3 describes RFaNet for fingerspelling recognition; Section 4 presents the experimental results, which are compared and analyzed; Section 5 extensively describes the experimental results of the RFaNet in transfer learning applications; and Section 6 concludes the study.

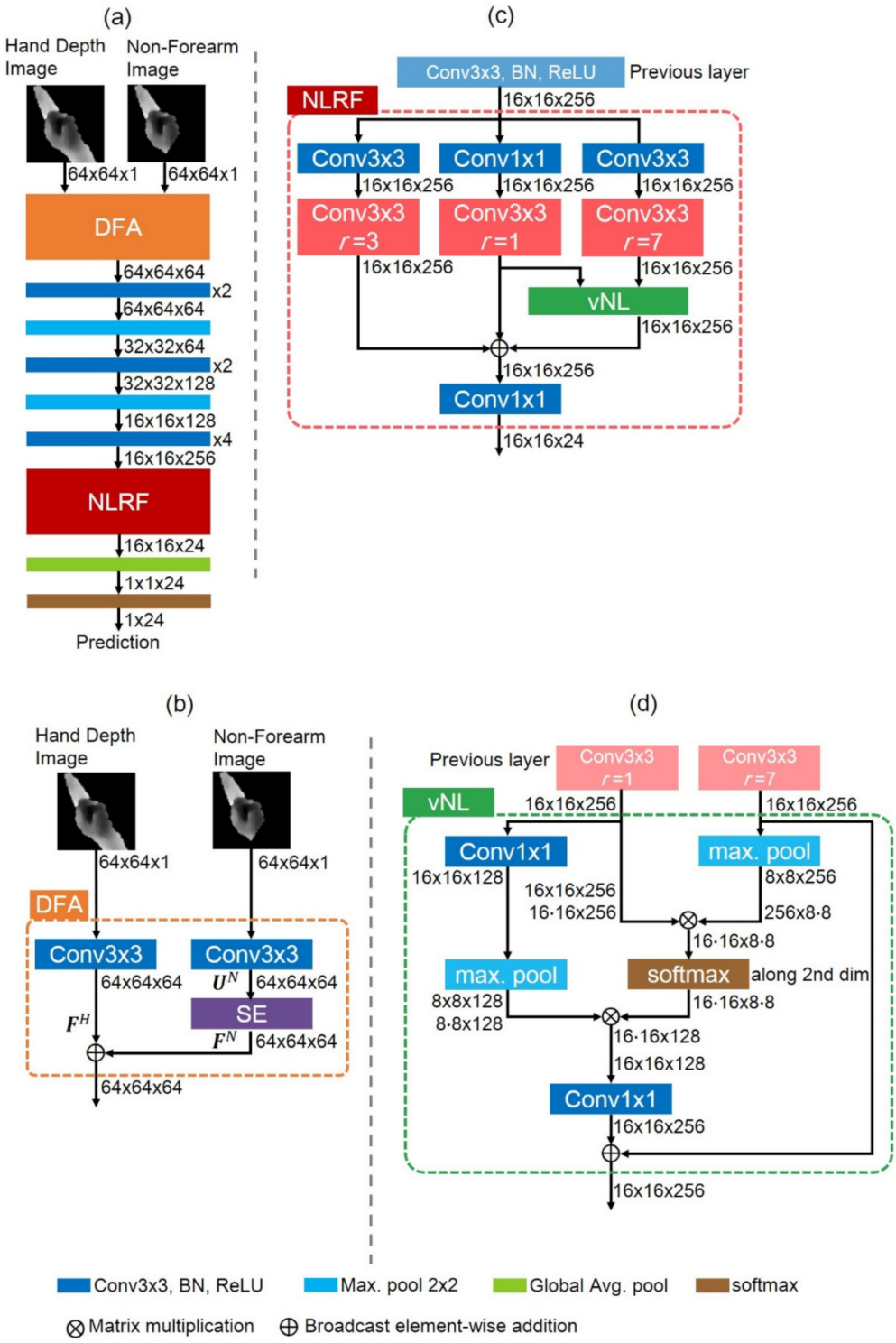

Figure 3. Overview of our hand gesture recognition model. (**a**) RFaNet. (**b**) DFA block. (**c**) NLRF block. (**d**) Variant of non-local block (vNL). SE: squeeze-and-excitation block; non-forearm image: hand depth image after forearm removal process; *r*: atrous sampling rate which corresponds to the stride when sampling the input signal.

2. Related Work

In this section, we describe the relevant recent works on fingerspelling recognition, RF, and attention mechanisms.

2.1. Fingerspelling Recognition

Usually, fingerspelling recognition applies depth modality, which is robust to illumination variations. Hu *et al*. [15] detected hands by assuming them as the closest objects to the sensor in depth images. Zhang and Tian [16] extracted the depth features and integrated them with a three-dimensional point cloud. Wang *et al*. [17] considered not only the depth-modality features (depth and skeleton features) but also the color modality features (color, texture, and contour features). Tao *et al*. [18] applied a CNN that recognizes letter signs captured from different perspectives in the depth modality. Modanwal and Sarawadekar [6] observed that the forearm usually appears in the hand image and is irrelevant to the hand gesture. They suggested removing the forearm from the hand image to improve hand gesture recognition. They developed a robust wrist-point detection algorithm to separate the palm and forearm based on hand anatomy. Removing the forearm is essential for capturing the fingers and extracting finger features in the hand image. Motivated by this result, we removed the forearm from the hand-depth image and extracted the low-level features from the finger and palm regions.

Rioux-Maldague and Giguère [4] decomposed the depth map of the hand into several layers, each representing a depth-level of the hand region. Partial fingers and palm regions at similar depth values appear in the same layer and are represented as depth features at the corresponding depth level. Decomposing a hand into different depth levels can handle finger occlusion. When one finger partially occludes another finger, both fingers belong to two depth levels and appear in two layers. This facilitates the localization of fingers, which is important for distinguishing fingerspelling signs. Accordingly, we were motivated to decompose the hand region into several depth-feature maps containing various depth information and facilitating finger localization.

2.2. Receptive Field

Conventional methods usually employ very deep convolutional networks that recognize objects at multiple scales, leading to huge computational costs. The cost can be reduced by replacing deep backbones with a lightweight model, in which enlarged RFs can potentially increase the field of view at multiple scales. The ASPP [7] enlarges the RFs by changing the sampling distance from the kernel center to capture the long-range dependency. Using ASPP, Wang *et al*. [19] extracted the spatial information around an object occluded by other objects. ASPP exploits and preserves the fine details around occlusions. Tan *et al*. [20] yielded a fixed-length feature representation using spatial pyramid pooling, which recognizes hand gestures regardless of input size. This method facilitates the propagation of gradients from the final fully connected layer to the input layer. The resolution of the input feature maps from the previous convolutional layers is uniform in the ASPP. Lu *et al*. [21] suggested that when inferring occlusion relationships, a sufficient RF is required at different scales for aggregating the cues around the occlusion region. Therefore, they extended the ASPP to different scales of the RF, enabling the complete sensing of foreground and background objects. Liu *et al*. [8] developed a receptive field block (RFB) that considers the relationship between the size and eccentricity of the RF. The RFB improves feature representation and can be equipped on top of a lightweight network for object detection tasks.

2.3. Attention Mechanism

Attention mechanisms are helpful for recalibrating the channel dependency of a computer vision task [22]. They model the long-range dependency of natural language processing [23]. Wang *et al*. [24] designed a residual ASPP block that extracts multiscale features from stereo images and a parallax-attention module that fuses these multiscale features to capture the stereo correspondences. Han *et al*. [25] simultaneously applied an ASPP block and a channel attention module for multiscale context extraction and channel-wise feature recalibration, respectively. The features extracted from the two branches were fused by weighted summation for the semantic labeling of high-resolution remote

sensing images. Liu *et al*. [26] densely connected the branches of an ASPP to cover the dense feature scales of RGB and depth modalities. Using a selective self-mutual attention module, they then integrated the attentions of the RGB and depth modalities to capture the long-range dependencies in RGB-D salient object detection. Yang *et al*. [27] developed a depth-aware attention module to refine the RGB and depth feature maps for suppressing the effect of color–depth misalignment. This module highlights important fingers for fingerspelling recognition. Inspired by the interactive learning of attentions from two modalities, we exploited the merits of ASPP and attention mechanisms to enlarge RFs at multiple scales and build the long-range dependencies of distant fingers. Our idea is to leverage the neighboring and inter-region neighboring dependencies between fingers. The resulting fingerspelling representation is invariant to viewpoint changes and further reduces intra-class variability.

3. Receptive Field-Aware Network with Finger Attention

In this section, we first introduce the overall architecture of the proposed fingerspelling recognition method, RFaNet; then, we describe how the key components of RFaNet facilitate tackling the fingerspelling recognition tasks.

Figure 3a shows the overall architecture of RFaNet. RFaNet was trained to enhance the finger regions and to build inter-finger relations in the depth image. A VGG-9 [14] is adopted as the backbone network. The proposed DFA and NLRF blocks are inserted at the top and bottom of RFaNet, respectively. The DFA block was designed to extract the low-level features from the finger and palm regions rather than the background regions. The NLRF block is designed to fuse the neighboring and inter-region information and extract a fingerspelling representation invariant to viewpoint changes. Experimental results supported the hypothesis that the DFA and NLRF blocks improved the overall fingerspelling recognition performance. We share our code and models at: https://github.com/yaomao-cheng/RFaNet_model/tree/master (accessed on 25 October 2021).

3.1. Depth Finger Attention Block

Fingerspelling recognition is usually hindered by inter-class similarity, i.e., by the similar appearances of more than one sign. Accurate finger localization is crucial for identifying slight hand posture variations. Unlike the method in [4], which manually divides the hand depth image into several depth-level layers for finger localization, the proposed DFA block applies learnable convolutional operations to obtain several depth feature maps from a hand depth image. However, the convolutional model may tend to learn sign-dependent background patterns, because most subjects make signs with a preferred posture, resulting in similar background patterns for identical signs (known as the background-bias phenomenon [5]). To guide the model toward the finger regions, the DFA block jointly processes two depth images: the hand depth image and the same image with the forearm removed (non-forearm image), which provide complementary information, as shown Figure 3b. As the non-forearm image references the hand depth image, the DFA block can highlight inside the hand region rather than the outside, i.e., a sign-dependent background pattern. The forearm was removed by a wrist-point detection algorithm [6], thus creating the non-forearm images. However, when the fingers occlude the wrist point, they can be incorrectly removed by the algorithm. In such cases, the hand depth image (possessing forearm) could provide complementary finger information that enhances the finger region.

In the non-forearm image path, the squeeze-and-excitation (SE) block [22] is employed to adaptively recalibrate relations across feature maps to effectively highlight finger regions, as shown in Figure 3b. These recalibrated feature maps from the non-forearm image are fused (by addition) with the feature maps from the hand depth image for learning to excite finger regions. This fusion ensures that the hand depth image and non-forearm image could provide complementary finger features, leading to finger localization even under wrist occlusion. The DFA block is inserted in the first layer of the proposed model, which

enables the following layers to extract discriminative features in the finger regions, as shown in Figure 3a. It facilitates the identification of slight hand posture variation across signs with inter-class similarity.

Given a feature map of the non-forearm image $\boldsymbol{U}^N = \left[\boldsymbol{u}_1^N, \boldsymbol{u}_2^N, \ldots, \boldsymbol{u}_{\mathrm{C}}^{\mathrm{N}}\right] \in \mathbb{R}^{H \times W \times C}$ extracted by the convolutional kernels, where $H = W = C = 64$ in this study, the SE block first squeezes the global spatial information via a global average pooling to obtain the channel-wise statistics, as follows:

$$z_c^N = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} \boldsymbol{u}_c^N(i,j), \tag{1}$$

where z_c^N represents the channel-wise statistics of the c-th channel. Then, the SE block captures channel-wise dependencies by two fully connected layers, as follows:

$$s^N = \sigma\Big(\boldsymbol{W}_2 \delta\Big(\boldsymbol{W}_1 \boldsymbol{z}^N\Big)\Big), \tag{2}$$

where σ and δ denote the sigmoid activation and rectified linear unit [28] functions, respectively, $\boldsymbol{W}_1 \in \mathbb{R}^{\frac{C}{r} \times C}$, and $\boldsymbol{W}_2 \in \mathbb{R}^{C \times \frac{C}{r}}$. We set the reduction ratio r to 2. The output of the SE block was obtained by recalibrating the channel-wise features, as follows:

$$\boldsymbol{F}^N = \boldsymbol{s}^N \otimes \boldsymbol{U}^N, \tag{3}$$

where $\otimes$ represents the element-wise product implemented by broadcasting the $\boldsymbol{s}^N$ values along the spatial axis. The SE block learns to excite the informative features of the non-forearm image and can potentially boost the finger localization ability.

The DFA block fuses the feature maps of the hand depth image F^H and the non-forearm image F^N. Among several operations in the fusion strategy—addition, product, and concatenation—we empirically reported that the addition operation provides better classification accuracy at less computational cost than the others. Therefore, addition was selected as the fusion strategy of F^H and F^N in the DFA block. By recalibrating the channel-wise dependencies of the features F^N, the DFA exploits the contextual information outside small RFs and enhances the features inside the hand region. The feature map F^N provides a reference for F^H, guiding the model toward the finger regions rather than sign-dependent background patterns. Moreover, the feature map F^H provided complementary information to F^N when the fingers were incorrectly removed in F^N under wrist occlusion. Jointly processing F^H and F^N focuses the attention on fingers in the depth image by highlighting the salient finger regions, thus improving the low-level finger representations.

3.2. Non-Local Receptive Field Block

A fingerspelling sign captured from multiple views may have various representations, resulting in intra-class variability. However, the inter-finger relations of a sign are inherently invariant to viewpoint changes. To capture inter-finger relations, the proposed NLRF block enlarges the RF and field of view to capture the long-range dependencies of distal fingers. Unlike the ASPP [7] and receptive field block (RFB) [8], the NLRF block not only applies standard convolutional operations with various receptive fields, followed by the atrous convolution, but also modifies the non-local block [9] to capture the relation of feature maps with multiple fields of view, which facilitates the modeling of the relations between distal fingers. The NLRF block exploits multi-scale feature maps using three atrous convolutions, with rates r = 1, 3, and 7, as shown in Figure 3c. The reason behind using the three rates is that the atrous convolution with a high rate only samples a region with checkerboard patterns, leading to a gridding problem [29] and the loss of neighboring information. We followed the suggestion in [29] to select rates that did not possess a common factor relationship (i.e., 1, 3, and 7). The rate parameter r represents the stride where the operator sampled the input signal. We applied the maximal atrous sampling rate r = 7 because

the feature map from the previous layer is of spatial resolution 16 × 16. We empirically found that the atrous convolution with $r = 5$ did not significantly improve the classification accuracy, and thus was removed (see Section 4.5 for a detailed analysis). The removal of the atrous convolution with $r = 5$ reduces the computational cost.

To relate the small and large fields of view, the feature maps from the branches of atrous convolution with rates $r = 1$ and 7 are fused by a variant of non-local (vNL) block. The reason underlying this fusion step is shown in Figure 4. The ASPP and RFB directly merge the feature maps with various RFs and fields of view, such that all pixels in the spatial array of RF equally contribute to the output response. Therefore, the relation between finger and background may be modeled rather than that between distal fingers (e.g., index finger and thumb), leading to incomplete inter-finger relations. The branch $r = 1$ captures neighboring information in a local area, whereas the branch $r = 7$ captures inter-region information in a large area. The neighboring information could provide the local relations between neighboring fingers, whereas the inter-region information could provide the non-local relations between distal fingers. Fusing the neighboring and inter-region information could emphasize the most essential regions, according to local and non-local relations, and better model inter-finger relations, as shown in Figure 4.

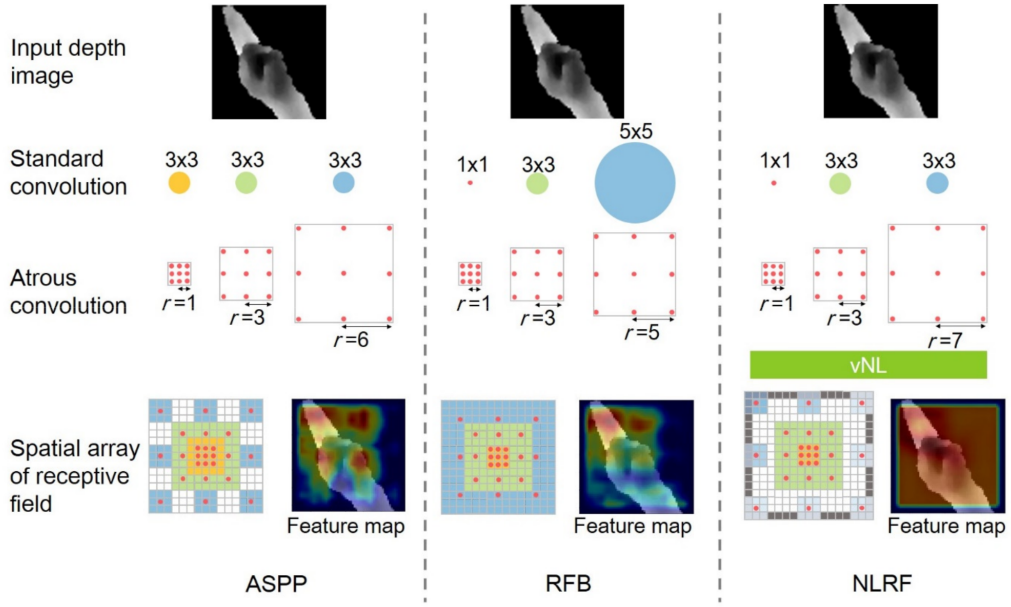

Figure 4. We adjusted the RF sizes in the original ASPP and RFB for a fair comparison. The feature maps are derived from the output of ASPP, RFB, and NLRF. The dark outer region of the feature map of the NLRF is zero-padded to fit the RF of atrous convolution with rate $r = 7$. vNL: variant of non-local block; r: atrous sampling rate.

The non-local block [9] applies a self-attention mechanism to enhance the features at a given position by aggregating the information at other positions of the same input feature vector. Different from the non-local block, which derives the value, key, and query from an identical input, our vNL enhances the features at a position of the atrous convolution with the rate $r = 7$ (a large RF) by aggregating the information at other positions of atrous convolution with the rate $r = 1$ (a small RF). The vNL facilitates the modeling of the long-range dependencies of multi-scale feature maps. Figure 3d shows that the vNL processes the feature maps produced from the branches $r = 1$ and 7 in the previous layer. The vNL shares a similar framework to the non-local block comprising context modeling, transformation, and fusion [30]. The global context features are modeled as the dot-product (matrix multiplication) of the feature embeddings of two positions, in the branches $r = 1$ and 7, respectively. The channel-wise dependencies of the global context features, captured by a 1 × 1 convolution, are as shown at the bottom of Figure 3d. The

global context features are aggregated at the features of each position in the branch $r = 7$ by a broadcast element-wise addition. We employed max-pooling to the feature maps of the branches $r = 1$ and 7 after a linear transformation to reduce the computational cost and extract shift-invariant features. The max-pooling could reduce the background effect because the background feature values were smaller than the hand feature values. Figure 4 shows that the NLRF block effectively captures the neighboring dependency of the index and middle fingers in the small RF and the inter-region dependency of the index finger and thumb within the large RF.

3.3. Optimization

Optimization was performed by summing two loss functions. The first loss function is the categorical cross-entropy loss function for multi-class classification:

$$\mathcal{L}_{CE} = -\frac{1}{NK}\sum_{i=1}^{N}\sum_{k=1}^{K} y_{ik}\log\hat{y}_{ik}, \tag{4}$$

where N is the mini-batch size, K is the number of classes, y_{ik} denotes the ground-truth label, and $\hat{y}_{ik}$ is the network output.

We also considered the sparsity-induced penalty term [31] in the loss function. This penalty term forces the scaling factors to be sparse in the batch normalization layer to improve the generalization ability. The complete loss for training RFaNet is as follows:

$$\mathcal{L} = \mathcal{L}_{CE} + \lambda\sum_{\gamma\in\Gamma}|\gamma|, \tag{5}$$

where γ is the scaling factor, Γ is the set of scaling factors in the network, and λ regulates the tradeoff between the classification accuracy and generalization ability.

4. Experimental Results

4.1. Datasets

We evaluated RFaNet on the following datasets. Each sample in these datasets consists of a pair of RGB and depth images. Figure 5 shows sample depth images from these datasets, where certain signs share similar hand shapes.

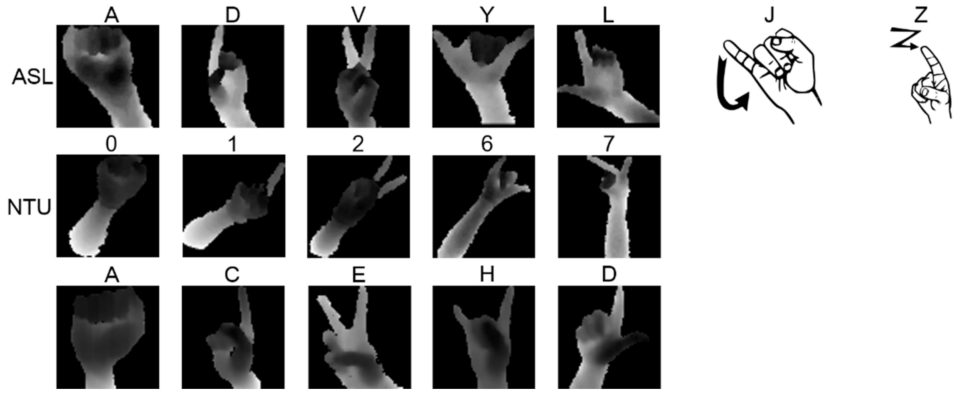

Figure 5. Samples from ASL, NTU, and OUHANDS datasets. The letter on the top of each panel represents the gesture label of the dataset. Each column represents the gestures that share similar hand shapes but different labels across three datasets. The letters *j* and *z* were excluded due to their dynamic characteristics.

ASL Fingerspelling Dataset. The ASL fingerspelling dataset comprises 24 letter signs of the American Sign Language alphabet acquired by the Microsoft Kinect sensor [3]. The dynamic letters *j* and *z* were excluded because RFaNet recognizes fingerspelling from a single depth image, which cannot reveal the dynamic characteristics of the letters,

as shown in Figure 5. These letter signs were performed by five subjects in front of various backgrounds and from different viewpoints. Each letter sign has 500 samples for each subject.

This dataset contains a few invalid samples in which the hand is missing or the letter sign does not belong to the ground-truth label, which was described in Yang *et al*. [27]. Therefore, we manually removed these invalid samples from the training and testing data.

NTU Digit Dataset. The NTU digit dataset comprises 10 digit signs acquired by the Microsoft Kinect sensor [32]. These digit signs were performed by 10 subjects, where each subject performed 10 times for each digit sign.

OUHANDS Dataset. The OUHANDS dataset comprises 10 signs acquired by the Intel RealSense F200 sensor [33]. This dataset includes hand and non-hand samples. Only the hand samples were selected for the present experiment. The hand samples were performed by 23 subjects, and 2150 and 1000 samples were adopted as the training and testing data, respectively.

4.2. Hand Detection and Depth Map Enhancement

The hand is assumed as the closest object to the camera, which is reasonable in practice. We detected the hand and enhanced its corresponding depth map to suppress the noise as well as improve the representation of the hand gesture. The first step applied Otsu's method [34] to select a threshold from the depth image. Pixel values smaller than the threshold were assumed as the background and set to zero. The second step applied the connected-component labeling algorithm [35] to group the non-zero pixels as foreground objects. In the NTU dataset, the objects closest to the camera were sometimes the knee regions of subjects seated on a chair. We thus selected the top foreground object as the hand because the hands are usually above the knees. The third step linearly scaled the pixel values in the hand region to 0–1 to enhance the hand texture.

When cropping the hands, the unequal width–height ratios hindered the batch learning because the image samples in a mini-batch should have identical width–height ratios. If the width was greater than the height, we resized the width to 64 pixels and maintained a constant width–height ratio. The height was expanded to 64 pixels by zero-padding; otherwise, we resized the height and expanded the width. Resizing and zero-padding did not alter the shape of the hand gesture.

4.3. Training and Testing

RFaNet was trained with a 0.9 momentum over 10 epochs and a 10^{-4} weight decay. The initial learning rate was 0.1, which was halved every 10 epochs. The proposed model was trained with a mini-batch size of 64 on an NVIDIA GeForce GTX 1080 Ti GPU using the PyTorch library.

The testing phase was implemented by leave-one-subject-out cross-validation (LOOCV). One subject was adopted as the testing data while the remaining subjects were adopted as the training data. The LOOCV was iterated until each subject was removed once. The LOOCV revealed whether RFaNet could be generalized to an unseen subject and whether RFaNet was robust to inter-subject variability, a common problem in practice.

We evaluated RFaNet in terms of classification accuracy. Furthermore, we computed the precision, recall, and *F*-score in comparison with state-of-the-art methods. We computed these measures as follows:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}, \tag{6}$$

$$Precision = \frac{TP}{TP + FP}, \tag{7}$$

$$Recall = \frac{TP}{TP + FN}, \tag{8}$$

$$F = 2\frac{Precision \cdot Recall}{Precision + Recall}, \quad (9)$$

where TP, TN, FP, and FN represent the numbers of true positives, true negatives, false positives, and false negatives, respectively. The F-score represents the harmonic mean of precision and recall.

4.4. Comparison of Different RF Blocks

To assess the effectiveness of the proposed NLRF block, we inserted NLRF and other blocks of receptive fields, namely ASPP and RFB, into the proposed RFaNet and compared their performances. These blocks processed multi-scale inputs. Table 1 presents the performance of RFaNet using different RF blocks. Note that only the NLRF block in RFaNet was replaced with ASPP or RFB. The NLRF block achieved a significant performance boost on both ASL and NTU datasets compared with ASPP and RFB.

Table 1. Performance comparison of RFaNet with different blocks of receptive fields, evaluated on the ASL and NTU datasets. We simply replaced the NLRF block with ASPP and RFB to evaluate the effects of these blocks. Numbers in parentheses indicate the standard deviation. #FLOPs: number of floating-point operations; #Param: number of parameters in the model. The symbol + indicates that the block was inserted into the proposed RFaNet. Bold values indicate the highest classification accuracy among the three blocks.

Block	#FLOPs (B)	#Param (M)	ASL (%)	NTU (%)
+ASPP	1.67	4.73	94.48(1.91)	95.80(4.61)
+RFB	3.19	10.69	94.60(1.90)	95.50(4.55)
+NLRF	3.06	5.45	**95.20(2.08)**	**96.50(3.63)**

4.5. Effect of Different Receptive Fields in NLRF Block

To examine the effect of varying the RFs in the NLRF block of RFaNet, Table 2 presents the performance of various configurations of the NLRF block on both ASL and NTU datasets. Each row indicates one configuration combining different branches of atrous convolution. Configurations 2–4 applied the vNL block to building non-local (long-range) connections across different branches of atrous convolution. The comparison of Configurations 1 and 2 shows that the vNL block improved the classification accuracy on both the ASL (+0.77%) and NTU (+0.10%) datasets. However, the computational cost increased in terms of the number of FLOPs (+2.80 B) and parameters (+1.96 M) due to the use of two vNL blocks. Notably, the vNL block was not applied to branches r = 1 and 3 because the RFs of these branches have a large overlap.

Table 2. Effects of various RFs in the NLRF block. The configuration r = 1 denotes the branch of atrous convolution with rate 1, and ✓ denotes that the branch was applied. The output feature maps of the branches with symbols * and † were further processed by the vNL block for capturing long-range dependencies. Configuration 3 adopted two vNL blocks to process branches r = 1 and 5 and branches r = 1 and 7, denoted by * and †, respectively. Numbers in parentheses indicate the standard deviation. A graphical illustration of the NLRF configuration is depicted in Figure 3c. Bold values indicate the highest classification accuracy among the four configurations.

	Configuration					#FLOPs	#Params	ASL	NTU
	r = 1	r = 3	r = 5	r = 7	vNL	(B)	(M)	(%)	(%)
1	✓	✓	✓	✓		1.67	4.73	94.48(1.91)	96.20(3.08)
2	✓ *†	✓	✓ *	✓ †	✓	4.47	6.69	95.25(1.80)	96.30(3.43)
3	✓ *	✓	✓ *		✓	3.06	5.45	95.11(1.72)	96.25(3.70)
4	✓ *	✓		✓ *	✓	3.06	5.45	**95.20(2.08)**	**97.00(3.09)**

The comparison of configurations 2 and 3 shows that the classification accuracies slightly decreased on the ASL (–0.14%) and NTU (−0.05%) datasets when removing one

vNL block. However, these classification accuracies were better than that of Configuration 1, which did not apply the vNL block. The comparison of Configurations 3 and 4 shows that connecting the branches $r = 1$ and 7 led to better performance than connecting the branches $r = 1$ and 5. Configuration 4 achieved comparable performance to Configuration 2 but saved computational cost. We selected configuration 4 as the NLRF configuration due to the tradeoff between accuracy and computational cost.

4.6. Qualitative Analysis of Various Receptive Fields in NLRF Block

Next, we analyzed the effect of changing the RFs in the NLRF block of RFaNet. Visual explanations were generated from the NLRF block using gradient-weighted class activation mapping (Grad-CAM) [36]. Grad-CAM can produce localization maps, which highlighted essential regions of the fingerspelling images corresponding to any decision of interest. Therefore, the discriminative features learned by NLRF could be visualized by Grad-CAM. Figure 6 shows the outcomes of each branch of atrous convolution from the NLRF block. The atrous convolution with a large rate captured the long-range dependency, whereas that with a small rate captured neighboring dependency. As the rate increased, the grid effect was observed in the branch of atrous convolution with rates 5 and 7. The localization maps highlighted the essential regions with checkerboard patterns, losing some neighboring information because the regions between two pixels of the convolutional kernel were not considered. Similar results from atrous convolutions with large rates were reported in [29].

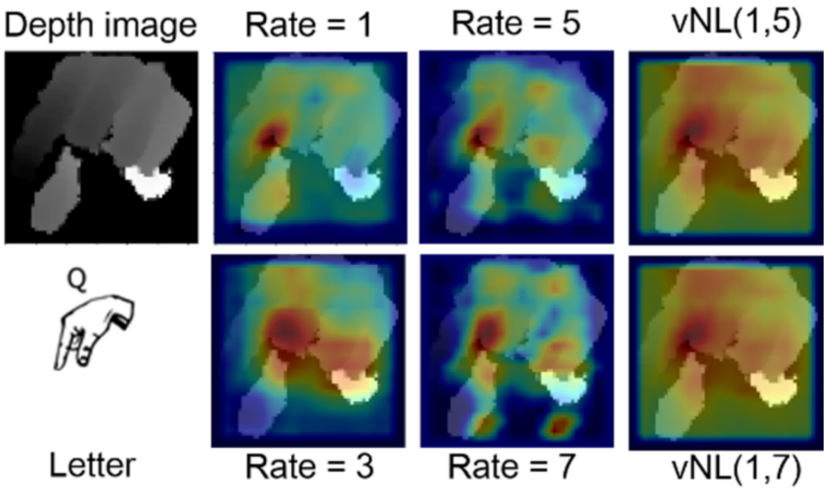

Figure 6. Visualization of outcomes of each branch from the NLRF block. The left-most column shows the depth image and its corresponding label. The middle two columns show the outcome of each branch of atrous convolution. vNL(1,5) and vNL(1,7) represent the outcomes of the vNL block determining the dependencies between the branches of atrous convolutions with rates 1 and 5 and rates 1 and 7, respectively.

The rightmost column of Figure 6 shows that the vNL block integrated the atrous convolution with small and large rates. However, the outcomes of the two vNL blocks NL(1,5) and NL(1,7) were similar (the notation is explained in the caption of Figure 6), which suggested that integrating the atrous convolutions with rates 1 and 5 and rates 1 and 7 gives similar contributions. Therefore, we maintained one of the vNL blocks and selected NL(1,7), i.e., Configuration 4 in Table 2, because it provides slightly higher performance and longer-range dependency than NL(1,5).

4.7. Effects of DFA and NLRF Blocks

To assess the effectiveness of the DFA and NLRF blocks, we performed comprehensive ablation experiments on the ASL and NTU datasets. Table 3 shows the performances of RFaNet in various configurations. Here, a VGG-13 was adopted as the backbone. The comparison of rows 1 and 2 presents a performance boost on both datasets when adopting

the DFA block, which demonstrates that selecting the representative finger regions from the depth image facilitated accuracy improvement. The configuration of row 3 adopted a VGG-9 as backbone instead of a VGG-13 because the insertion of the NLRF block increased three convolutional layers and one vNL block. Therefore, the NLRF block was inserted into the VGG-9 for a fair comparison. Inserting the NLRF block significantly improved the performance on both datasets (cf. rows 1 and 3 of Table 3), demonstrating the effectiveness of building short- and long-range dependencies. Moreover, employing both the DFA and NLRF blocks significantly improved the accuracy for the ASL (+1.7%) and NTU (+7.0%) datasets (cf. rows 1 and 4 of Table 3). For the computational cost, the number of parameters of the model is less than that of the backbone (−5.14 M).

Table 3. Ablation study for various configurations of RFaNet on the ASL and NTU datasets in terms of accuracy (%). Notably, the VGG-13 has four more convolutional layers than the VGG-9. These four convolutional layers were inserted before the global average pooling layer. Numbers in parentheses indicate the standard deviation. #FLOPs: number of floating-point operations; #Param: number of parameters of a model. Bold values indicate the highest classification accuracy among the four configurations.

	Configuration			#FLOPs (B)	#Param (M)	ASL (%)	NTU (%)
	Backbone	DFA	NLRF				
1	VGG-13			1.43	10.59	93.50(2.30)	89.50(6.59)
2	VGG-13	✓		1.59	10.63	94.26(1.94)	91.40(4.72)
3	VGG-9		✓	2.91	5.41	94.87(2.33)	94.90(4.86)
4	VGG-9	✓	✓	3.06	5.45	**95.20(2.08)**	**96.50(3.63)**

4.8. Qualitative Analysis of DFA and NLRF Blocks

We conducted a qualitative analysis of the DFA and NLRF blocks in RFaNet. Figure 7 shows the effects of the DFA and NLRF blocks on the ASL dataset. As shown in the left two columns, RFaNet without the DFA block highlighted only the hand contours. By contrast, RFaNet with the DFA block highlighted the fingers in the depth image while ignoring the wrist, which was irrelevant to the fingerspelling sign. Furthermore, using the DFA block increased the softmax score of the ground-truth class, leading to correct classification.

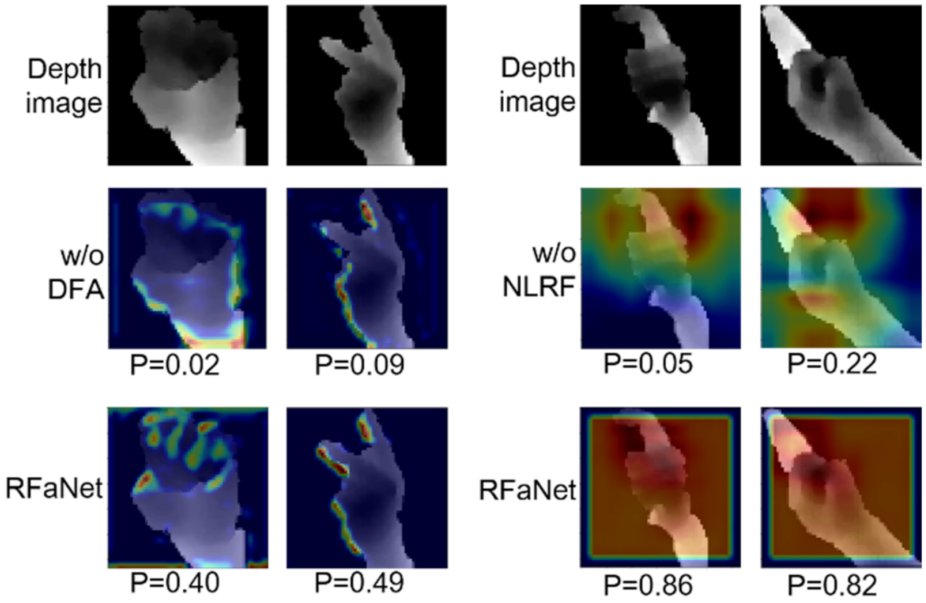

Figure 7. Grad-CAM visualization of the effects of the DFA and NLRF blocks on the ASL dataset. The left two columns show the feature maps of the first bottleneck layer of RFaNet to analyze the effect of the DFA block. The right two columns show the feature maps of the last bottleneck layer of RFaNet to analyze the effect of the NLRF block. w/o: without; P: softmax score of the ground-truth class.

The right two columns of Figure 7 show that RFaNet without the NLRF block emphasized the background rather than the fingers, leading to a low softmax score of the ground-truth class. When the NLRF block was inserted, RFaNet exploited the neighboring and long-range dependencies to emphasize the key fingers related to the fingerspelling sign. Therefore, fingerspelling signs in which the key fingers cover a large area (column 3) and wide posture variation (column 4) were correctly classified.

4.9. Comparison with State-of-The-Art Methods

We compared the performances of RFaNet and state-of-the-art methods on the ASL and NTU datasets. Table 4 lists the classification accuracy. For the ASL dataset, RFaNet outperformed the state-of-the-art methods (95.30%). For the NTU dataset, RFaNet did not outperform the state-of-the-art methods (98.00%). However, the state-of-the-art methods did not simultaneously achieve the highest accuracy on both datasets, and RFaNet was competitive against all compared methods. The high accuracies of RFaNet on both datasets demonstrated the generalization ability of RFaNet for various fingerspelling tasks.

Table 4. Comparisons with state-of-the-art methods using LOOCV evaluation on the ASL and NTU datasets. The "Method" column shows the classifiers used in state-of-the-art methods, where * indicates that the feature descriptor and classifier are jointly trained in the method. Mod: modality; *A*: accuracy; *P*: precision; *R*: recall; *F*: *F*-score. Bold values indicate the highest classification accuracy among the state-of-the-art methods.

			ASL				NTU
Study	**Method**	**Mod.**	***A* (%)**	***P* (%)**	***R* (%)**	***F* (%)**	***A* (%)**
Pugeault *et al.* [3]	RF	D	49.00	–	–	–	–
Kuznetsova *et al.* [37]	RF	D	57.00	–	–	–	–
Wang *et al.* [38]	SVM	D	58.30	–	–	–	91.10
Dong *et al.* [39]	RF	RGBD	70.00	–	–	–	–
Kane *et al.* [40]	SVM	D	71.58	–	–	–	90.75
Wang *et al.* [17]	TM	RGBD	75.80	–	–	–	99.60
Suau *et al.* [41]	RF	RGBD	76.10	–	–	–	–
Feng *et al.* [42]	SVM	D	78.70	–	–	–	**100**
Warchoł *et al.* [43]	HMM	D	78.80	–	–	–	–
Ameen *et al.* [44]	CNN *	RGBD	80.34	82.00	80.00	79.20	–
Nai *et al.* [45]	RF	D	81.10	–	–	–	–
Maqueda *et al.* [46]	SVM	RGB	83.70	–	–	–	95.90
Zhang *et al.* [16]	SVM	D	83.80	–	–	–	94.50
Keskin *et al.* [47]	SCF	D	84.30	–	–	–	–
Rady *et al.* [48]	CNN *	RGBD	84.67	–	–	–	99.85
Aly *et al.* [49]	SVM	D	88.70	–	–	–	–
Rakowski *et al.* [50]	ResNet *	RGBD	90.60	91.80	90.60	90.30	–
Tao *et al.* [18]	CNN *	D	92.70	93.50	92.40	91.71	**100**
Yang *et al.* [27]	DDaNet *	RGBD	93.53	94.10	93.48	93.26	96.10
Ours	RFaNet	D	95.30	95.32	95.70	95.51	98.00

5. Extensive Experimental Results of RFaNet in Transfer Learning

The data annotation and collection of fingerspelling requires specialized domain knowledge and expert interpreters. Thus, large-scale datasets are not commonly available for fingerspelling recognition. Transferring the representation of hand gestures from a large- to a small-scale dataset is always in demand. We evaluated the effectiveness of RFaNet in transferring knowledge from the large-scale ASL dataset to the small-scale NTU and OUHANDS datasets. ASL, NTU, and OUHANDS datasets are commonly used fingerspelling datasets and comprise 60,000, 1000, and 3000 labeled samples, respectively. These three datasets share similar hand gestures, even when they belong to different labels, as shown in Figure 5.

5.1. Implementation Details of Transfer Learning

Transfer learning for fingerspelling recognition was implemented by the following process. First, RFaNet was pre-trained with the ASL dataset (source domain). Second, the last fully connected layers and the corresponding softmax layer were replaced according to the number of classes in the target dataset. Third, the initial two bottleneck layers had their parameters frozen (shared with the source domain) when considering the OUHANDS dataset as the target dataset. The first three bottleneck layers had their parameters frozen when considering the NTU dataset as the target dataset. The number of frozen bottleneck layers differed for the OUHANDS and NTU datasets because the OUHANDS dataset contains more training data (3150) and is larger than the NTU dataset (1000). If the number of parameters requiring fine-tuning and the target dataset were small, the model would result in overfitting [51]. In the fourth step, the remaining model parameters were fine-tuned on the target dataset.

5.2. Quantitative Results of Transfer Learning on NTU Dataset

Table 5 shows the experimental results of transfer learning when considering the NTU dataset as the target dataset. For comparison, we implemented transfer learning on DDaNet [27], a state-of-the-art method for the ASL dataset that adopts the color (RGB) and depth modalities as inputs. The transfer learning protocol for DDaNet was identical to that of RFaNet. Applying the transfer learning to RFaNet improved the accuracy compared with RFaNet without transfer learning (+1.00%). Furthermore, the number of parameters for RFaNet was less than that of DDaNet (−16.37 M), making mobile applications feasible.

Table 5. Results on the transfer learning where the NTU dataset is the target dataset. Numbers in parentheses indicate the standard deviation of the LOOCV across 10 subjects. Mod.: modality; #Param: number of parameters of the model; w/TF: with transfer learning; w/o TF: without transfer learning. Bold values indicate the highest classification accuracy among the four methods.

Study	Method	Mod.	#Param	Accuracy (%)
Yang *et al*. [27]	DDaNet w/TF	RGBD	21.24 M	96.10 (4.12)
Yang *et al*. [27]	DDaNet w/o TF	RGBD	21.24 M	87.90 (4.75)
Ours	RFaNet w/TF	D	5.45 M	97.00 (3.09)
Ours	RFaNet w/o TF	D	5.45 M	**98.00 (1.56)**

5.3. Quantitative Results of Transfer Learning on OUHANDS Dataset

Table 6 shows the experimental results of transfer learning when considering the OUHANDS dataset as the target dataset. After transfer learning, the accuracy and *F*-score of DDaNet were lower than those of DDaNet without transfer learning (−0.80% and −0.85%, respectively). However, after transfer learning, RFaNet showed improved accuracy and *F*-score compared with RFaNet without transfer learning (+2.60% and +2.66%, respectively). Furthermore, RFaNet outperformed the state-of-the-art methods in terms of accuracy and *F*-score (92.90% and 93.00%, respectively), demonstrating the benefits of learning the representations of hand gestures using depth modality from a large-scale dataset (the ASL dataset).

5.4. Qualitative Results of Transfer Learning

For a qualitative analysis of transfer learning by RFaNet, we generated localization maps using Grad-CAM [36] to highlight the essential regions corresponding to any decisions of interest. This analysis visualized the representation of the hand gestures learned by RFaNet during transfer learning. Figure 8 shows the qualitative analysis of transfer learning where NTU and OUHANDS datasets are the target datasets. The localization maps of the NLRF layer revealed that RFaNet without transfer learning emphasized the regions in the background, as shown in the number "6" of NTU and the letter "c" of OUHANDS. Although the ring finger and thumb, respectively, were highlighted in letters "f" and "k"

of OUHANDS, the other key fingers of these hand gestures were not emphasized, leading to a low softmax score of the ground-truth class. After transferring the representation of the hand gestures learned from the ASL dataset, the key fingers of the hand gestures were highlighted, and the softmax score of the ground-truth class was increased, as shown in the third row of Figure 8.

Table 6. Results of the transfer learning where OUHANDS is the target dataset. Numbers in parentheses indicate the performance difference between networks with and without transfer learning. Mod.: modality; #Param: number of parameters of the model; w/TF: with transfer learning; w/o TF: without transfer learning. *A*: accuracy; *F*: *F*-score; HGR-Net: hand gesture recognition network; HOG: histogram of oriented gradients; SVM: support vector machine. Bold values indicate the highest classification accuracy among all methods.

Study	Method	Mod.	#Param	*A*	*F*
He *et al*. [52]	ResNet-50	RGB	23.60 M	–	81.30
Huang *et al*. [53]	DenseNet-121	RGB	7.04 M	–	82.80
Howard *et al*. [54]	MobileNet	RGB	3.22 M	–	86.50
Dadashzadeh *et al*. [55]	HGR-Net	RGB	0.499 M	–	88.10
Matilainen *et al*. [33]	HOG+SVM	RGB	–	83.25	–
Yang *et al*. [27]	DDaNet w/TF	RGBD	21.24 M	88.90	89.10
Yang *et al*. [27]	DDaNet w/o TF	RGBD	21.24 M	88.10 (−0.80)	88.25 (−0.85)
Ours	RFaNet w/TF	D	5.45 M	90.30	90.34
Ours	RFaNet w/o TF	D	5.45 M	**92.90** (+2.60)	**93.00** (+2.66)

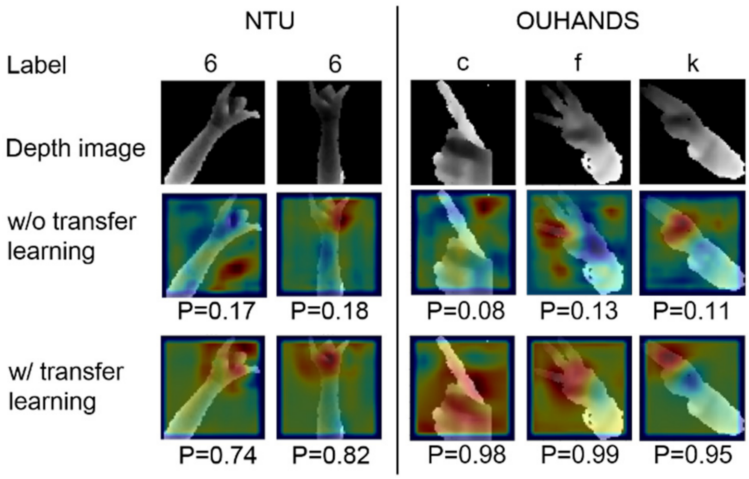

Figure 8. Qualitative analysis of transfer learning on the NTU and OUHANDS datasets where "w/o" and "w/" represent without and with, respectively. The NLRF layer was visualized by Grad-CAM. Transfer learning was implemented by pre-training RFaNet with the ASL dataset and fine-tuning it with the OUHANDS dataset. Because the amount of data in the NTU dataset is 1/3 of that of the OUHANDS dataset, two and three samples were provided for the NTU and OUHANDS datasets, respectively. *P* is the softmax score of the ground-truth class.

5.5. Network Visualization of Transfer Learning

In addition to the NLRF layer, we qualitatively visualized the output of each bottleneck layer to demonstrate the effectiveness of RFaNet during transfer learning. Figure 9 shows the outcomes from the initial three bottleneck layers for three examples. When RFaNet learned the representation of the hand gestures from the ASL dataset, it could more efficiently extract the low-level features in the small-scale target dataset than it could without transfer learning. The key fingers were then accurately localized, leading to correct classification. This result agreed with the empirical evidence showing that the

initial bottleneck layers learned the low-level features that could be shared across different tasks [56].

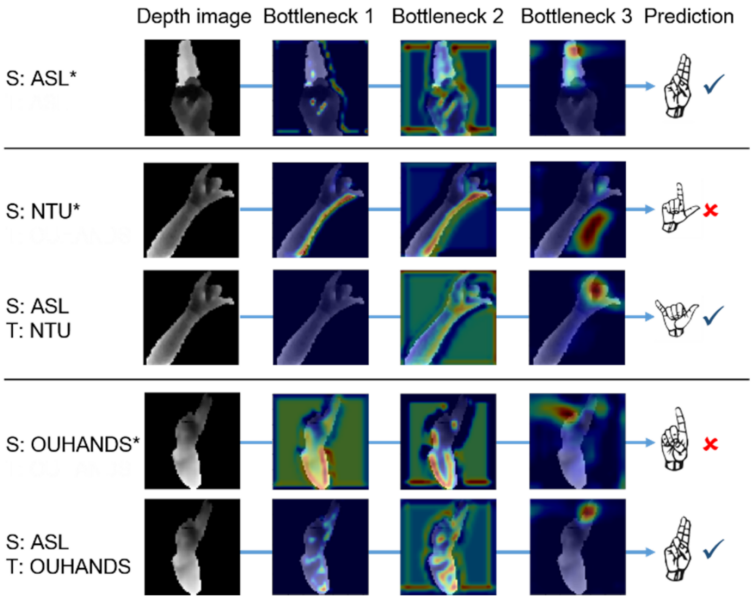

Figure 9. Visualization of the learned features when applying transfer learning to RFaNet, where S and T represent the source and target dataset, respectively. Each row indicates the feature maps of three bottlenecks of RFaNet. The asterisk indicates that RFaNet was trained using only the source data and evaluated on the source data. The third and fifth rows indicate that RFaNet was pre-trained with the source data, fine-tuned with the target data, and evaluated on the target data. This study considered the ASL dataset as the source data due to the sufficiently large training data and considered the OUHANDS and NTU datasets as the target data due to the relatively small amount of training data. The icons in the prediction column were reproduced from [57].

5.6. Failure Modes

Figure 10 shows some failure modes of RFaNet on the ASL and NTU datasets. Our model failed to capture the neighboring and inter-region dependencies of widely variable hand postures. When the fingers extended outside the palm region, they were not correctly highlighted in the localization maps, leading to incorrect classification. Dealing with large hand-posture variations is left for future work.

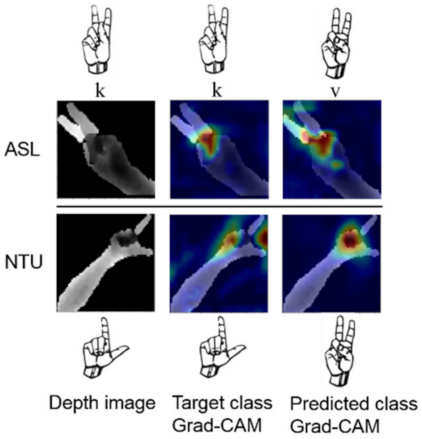

Figure 10. Failure modes of RFaNet on the ASL and NTU datasets. The first and fourth rows represent the labels corresponding to the images and feature maps, respectively. The second and third rows represent the depth images and their Grad-CAM visualizations, respectively, according to their target and predicted classes.

6. Discussion

6.1. Effectiveness of the DFA and NLRF Blocks

The proposed DFA and NLRF blocks are inserted at the top and bottom of RFaNet, respectively. The DFA block highlights the fingers in the depth image. The NLRF block increases the size of the receptive fields and builds long-range connections across the finger features, thus facilitating fingerspelling recognition. This result demonstrates that building long-range connections across the branches of atrous convolutions with rates $r = 1$ and $r = 7$ facilitates the network's learning of discriminative features related to fingerspelling. Furthermore, integrating the small and large receptive fields by the vNL blocks improves fingerspelling recognition. The vNL block integrates the short-range and long-range dependencies and exploits the relation between local and non-local interactions. The NLRF block could effectively capture the fine fingerspelling details and important features across the fingers. Furthermore, this integration allowed RFaNet to recognize letter signs whose important fingers possess long-range dependency and hand shapes with high inter-class similarity. The DFA and NLRF blocks highlight the finger regions and explore the fingers' dependencies, contributing to the performance boost of RFaNet.

6.2. Transfer Learning for Fingerspelling Recognition

The recognition accuracy of RFaNet on small-scale datasets (e.g., the NTU and OUHANDS datasets) can be improved by transferring the representations of hand gestures learned from large-scale datasets (e.g., the ASL dataset). The above experimental results show that the proposed RFaNet learned better representations of the hand gestures from a large-scale dataset than did DDaNet. As DDaNet learns the representation from both color and depth modalities, it may learn to highlight the background information revealed in the color modality corresponding to any decision of interest. This learning can degrade the transfer learning because the low-level features relevant to the background differ across datasets. Therefore, the initial bottleneck layers with frozen parameters may not be shared across the source and target domains. However, RFaNet learns only from the depth modality. As the DFA block of RFaNet facilitates the separation of hand gestures from the background and highlights the fingers, the low-level features hardly involve the background information. Therefore, the initial bottleneck layers pre-trained on the source domain improved the classification accuracy in the target domain. This result demonstrates that during transfer learning, RFaNet can boost fingerspelling recognition on small-scale datasets without the effect of complex background information.

RFaNet efficiently learned the representations of hand gestures from a large-scale dataset and facilitated the learning of a small-scale target dataset. The reasons are explained here. First, as RFaNet processes only depth images, the hand gestures are not easily affected by the complex background. Therefore, RFaNet can effectively transfer the representation of the hand gesture learned from the ASL dataset to the small-scale NTU/OUHANDS datasets, leading to improved recognition performance. Second, the DFA block in the most initial layer emphasizes the fingers and palm regions, indicating that the learning of hand-gesture representations is unaffected by gesture-irrelevant factors. Therefore, RFaNet facilitated transfer learning when the training data of the target domain were insufficient.

6.3. Implementation in Actual Application

The implementation of the proposed fingerspelling recognition system in actual experiments consists of two factors: hardware and software. The hardware factor considers the depth camera and experimental environment. The training datasets were collected by a Microsoft Kinect sensor (ASL and NTU) and a RealSense F200 sensor (OUHANDS). Both depth cameras acquire depth images with a depth resolution of 1 mm and a spatial resolution of 640 $\times$ 480 pixels. The distance from the subject to the depth camera is in a range of 230–800 mm in an indoor environment. The software factor considers hand detection and depth map enhancement. We detected the hand and enhanced its

corresponding depth map to suppress the noise as well as improve the representation of the hand gesture.

When using a new depth camera, the depth image should possess a depth resolution of 1 mm and a spatial resolution of 640 × 480 pixels. Furthermore, the subject is kept at a distance in a range of 230–800 mm from the depth sensor in order to obtain a hand image with quality similar to that of the training datasets. If the hardware meets these requirements in an indoor environment, the proposed fingerspelling recognition system could be implemented using a new depth camera in actual experiments.

7. Conclusions

We proposed and evaluated RFaNet, a network that highlights the finger regions and builds inter-finger relations for fingerspelling recognition. RFaNet aggregates the low-level features in hand depth and non-forearm images to focus on the fingers. It fuses the high-level multi-scale features of various RFs to model the neighboring and inter-region dependencies between fingers, which makes the sign representation invariant to the viewpoint and thus reduces the intra-class variability. In experimental evaluations on the ASL dataset, RFaNet outperformed current state-of-the-art methods. When applied to a small-scale fingerspelling dataset with insufficiently labeled data, RFaNet leverages the depth representations learned from a large-scale dataset to boost the fingerspelling recognition on the small-scale dataset. Using only depth images in RFaNet facilitated transfer learning on limited training datasets without requiring expensive fingerspelling annotations. This technique can improve communication between deaf and hearing people. Large hand posture variations may affect neighboring and inter-region dependencies. Therefore, the question of how to build inter-finger relations under large hand posture variations is left for future work.

Author Contributions: Conceptualization, S.-H.Y., Y.-M.C. and Y.-P.C.; methodology, S.-H.Y. and Y.-M.C.; software, Y.-M.C. and J.-W.H.; validation, Y.-M.C. and J.-W.H.; formal analysis, S.-H.Y. and Y.-M.C.; investigation, S.-H.Y., Y.-M.C., and J.-W.H.; data curation, Y.-M.C. and J.-W.H.; writing—original draft preparation, S.-H.Y. and Y.-M.C.; writing—review and editing, S.-H.Y. and Y.-M.C.; visualization, Y.-M.C. and S.-H.Y.; supervision, S.-H.Y. and Y.-P.C.; project administration, S.-H.Y. and Y.-P.C.; funding acquisition, S.-H.Y. and Y.-P.C. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported in part by the Ministry of Science and Technology of Taiwan MOST 108-2221-E-009-119, 109-2221-E-009-049, 110-2221-E-A49-122, 110-2636-E-006-021 (Young Scholar Fellowship Program); in part by the Headquarters of University Advancement at National Cheng Kung University, Ministry of Education; in part by National Cheng Kung University Hospital, Taiwan.

Data Availability Statement: Publicly available datasets were analyzed in this study. This data can be found here: https://www.kaggle.com/mrgeislinger/asl-rgb-depth-fingerspelling-spelling-it-out for ASL; http://eeeweba.ntu.edu.sg/computervision/people/home/renzhou/HandGesture.htm for NTU; and https://www.kaggle.com/mumuheu/ouhands for OUHANDS.

Acknowledgments: The authors would like to thank Pugeault and Bowden, Ren *et al.*, and Matilainen *et al.*, respectively, for making the ASL Fingerspelling Dataset, NTU Digit Dataset, and OUHANDS Dataset publicly available.

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

References

1. Padden, C.A.; Gunsauls, D.C. How the alphabet came to be used in a sign language. *Sign Lang. Stud.* **2003**, *4*, 10–33. [CrossRef]
2. Tsai, Y.-S.; Hsu, L.-H.; Hsieh, Y.-Z.; Lin, S.-S. The real-time depth estimation for an occluded person based on a single image and OpenPose method. *Mathematics* **2020**, *8*, 1333. [CrossRef]
3. Pugeault, N.; Bowden, R. Spelling it out: Real-time ASL fingerspelling recognition. In Proceedings of the IEEE International Conference on Computer Vision Workshops (ICCV Workshops), Barcelona, Spain, 6–13 November 2011; pp. 1114–1119.

4. Rioux-Maldague, L.; Giguère, P. Sign language fingerspelling classification from depth and color images using a deep belief network. In Proceedings of the IEEE Canadian Conference on Computer and Robot Vision, Montreal, QC, Canada, 6–9 May 2014; pp. 92–97.
5. Tian, M.; Yi, S.; Li, H.; Li, S.; Zhang, X.; Shi, J.; Yan, J.; Wang, X. Eliminating background-bias for robust person re-identification. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 5794–5803.
6. Modanwal, G.; Sarawadekar, K. A robust wrist point detection algorithm using geometric features. *Pattern Recognit. Lett.* **2018**, *110*, 72–78. [CrossRef]
7. Chen, L.-C.; Papandreou, G.; Kokkinos, I.; Murphy, K.; Yuille, A.L. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *40*, 834–848. [CrossRef]
8. Liu, S.; Huang, D. Receptive field block net for accurate and fast object detection. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 385–400.
9. Wang, X.; Girshick, R.; Gupta, A.; He, K. Non-local neural networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 7794–7803.
10. Cui, Y.; Song, Y.; Sun, C.; Howard, A.; Belongie, S. Large scale fine-grained categorization and domain-specific transfer learning. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 4109–4118.
11. Nihal, R.A.; Rahman, S.; Broti, N.M.; Deowan, S.A. Bangla Sign Alphabet Recognition with Zero-shot and Transfer Learning. *Pattern Recognit. Lett.* **2021**, *150*, 84–93. [CrossRef]
12. Deng, J.; Dong, W.; Socher, R.; Li, L.-J.; Li, K.; Fei-Fei, L. Imagenet: A large-scale hierarchical image database. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Miami, FL, USA, 20–25 June 2009; pp. 248–255.
13. Bird, J.J.; Ekárt, A.; Faria, D.R. British sign language recognition via late fusion of computer vision and leap motion with transfer learning to american sign language. *Sensors* **2020**, *20*, 5151. [CrossRef]
14. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv* **2014**, arXiv:1409.1556.
15. Hu, Y.; Zhao, H.-F.; Wang, Z.-G. Sign language fingerspelling recognition using depth information and deep belief networks. *Int. J. Pattern Recognit. Artif. Intell.* **2018**, *32*, 1850018. [CrossRef]
16. Zhang, C.; Tian, Y. Histogram of 3D facets: A depth descriptor for human action and hand gesture recognition. *Comput. Vis. Image Underst.* **2015**, *139*, 29–39. [CrossRef]
17. Wang, C.; Liu, Z.; Chan, S.-C. Superpixel-based hand gesture recognition with kinect depth camera. *IEEE Trans. Multimed.* **2014**, *17*, 29–39. [CrossRef]
18. Tao, W.; Leu, M.C.; Yin, Z. American Sign Language alphabet recognition using Convolutional Neural Networks with multiview augmentation and inference fusion. *Eng. Appl. Artif. Intell.* **2018**, *76*, 202–213. [CrossRef]
19. Wang, Y.; Wu, T.; Yang, J.; Wang, L.; An, W.; Guo, Y. DeOccNet: Learning to see through foreground occlusions in light fields. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, Village, CO, USA, 1–5 March 2020; pp. 118–127.
20. Tan, Y.S.; Lim, K.M.; Tee, C.; Lee, C.P.; Low, C.Y. Convolutional neural network with spatial pyramid pooling for hand gesture recognition. *Neural Comput. Appl.* **2021**, *33*, 5339–5351. [CrossRef]
21. Lu, R.; Xue, F.; Zhou, M.; Ming, A.; Zhou, Y. Occlusion-shared and feature-separated network for occlusion relationship reasoning. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Korea, 27–28 October 2019; pp. 10343–10352.
22. Hu, J.; Shen, L.; Sun, G. Squeeze-and-excitation networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 7132–7141.
23. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. In Proceedings of the Advances in Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017; pp. 5998–6008.
24. Wang, L.; Wang, Y.; Liang, Z.; Lin, Z.; Yang, J.; An, W.; Guo, Y. Learning parallax attention for stereo image super-resolution. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 16–20 June 2019; pp. 12250–12259.
25. Han, B.; Yin, J.; Luo, X.; Jia, X. Multibranch Spatial-Channel Attention for Semantic Labeling of Very High-Resolution Remote Sensing Images. *IEEE Geosci. Remote. Sens. Lett.* **2020**, 1–5. [CrossRef]
26. Liu, N.; Zhang, N.; Han, J. Learning selective self-mutual attention for RGB-D saliency detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 16–18 June 2020; pp. 13756–13765.
27. Yang, S.-H.; Chen, W.-R.; Huang, W.-J.; Chen, Y.-P. DDaNet: Dual-Path Depth-Aware Attention Network for Fingerspelling Recognition Using RGB-D Images. *IEEE Access* **2020**, *9*, 7306–7322. [CrossRef]
28. Nair, V.; Hinton, G.E. Rectified linear units improve restricted boltzmann machines. In Proceedings of the 27th International Conference on International Conference on Machine Learning, Haifa, Israel, 21–24 June 2010; pp. 807–814.
29. Wang, P.; Chen, P.; Yuan, Y.; Liu, D.; Huang, Z.; Hou, X.; Cottrell, G. Understanding convolution for semantic segmentation. In Proceedings of the 2018 IEEE Winter Conference on Applications of Computer Vision (WACV), Lake Tahoe, NV, USA, 12–15 March 2018; pp. 1451–1460.

30. Cao, Y.; Xu, J.; Lin, S.; Wei, F.; Hu, H. Gcnet: Non-local networks meet squeeze-excitation networks and beyond. In Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops, Seoul, Korea, 27–28 October 2019; pp. 1–10.
31. Liu, Z.; Li, J.; Shen, Z.; Huang, G.; Yan, S.; Zhang, C. Learning efficient convolutional networks through network slimming. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2736–2744.
32. Ren, Z.; Yuan, J.; Zhang, Z. Robust hand gesture recognition based on finger-earth mover's distance with a commodity depth camera. In Proceedings of the 19th ACM International Conference on Multimedia, Scottsdale, AZ, USA, 28 November–1 December 2011; pp. 1093–1096.
33. Matilainen, M.; Sangi, P.; Holappa, J.; Silvén, O. OUHANDS database for hand detection and pose recognition. In Proceedings of the Sixth International Conference on Image Processing Theory, Tools and Applications (IPTA), Oulu, Finland, 12–15 December 2016; pp. 1–5.
34. Otsu, N. A threshold selection method from gray-level histograms. *IEEE Trans. Syst. Man Cybern.* **1979**, *9*, 62–66. [CrossRef]
35. Rosenfeld, A.; Pfaltz, J.L. Sequential operations in digital picture processing. *JACM* **1966**, *13*, 471–494. [CrossRef]
36. Selvaraju, R.R.; Cogswell, M.; Das, A.; Vedantam, R.; Parikh, D.; Batra, D. Grad-cam: Visual explanations from deep networks via gradient-based localization. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 618–626.
37. Kuznetsova, A.; Leal-Taixé, L.; Rosenhahn, B. Real-time sign language recognition using a consumer depth camera. In Proceedings of the IEEE International Conference on Computer Vision Workshops, Sydney, Australia, 2–8 December 2013; pp. 83–90.
38. Wang, Y.; Yang, R. Real-time hand posture recognition based on hand dominant line using kinect. In Proceedings of the 2013 IEEE International Conference on Multimedia and Expo Workshops (ICMEW), San Jose, CA, USA, 15–19 July 2013; pp. 1–4.
39. Dong, C.; Leu, M.C.; Yin, Z. American sign language alphabet recognition using microsoft kinect. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, Boston, MA, USA, 7–12 June 2015; pp. 44–52.
40. Kane, L.; Khanna, P. A framework for live and cross platform fingerspelling recognition using modified shape matrix variants on depth silhouettes. *Comput. Vis. Image Underst.* **2015**, *141*, 138–151. [CrossRef]
41. Suau, X.; Alcoverro, M.; López-Méndez, A.; Ruiz-Hidalgo, J.; Casas, J.R. Real-time fingertip localization conditioned on hand gesture classification. *Image Vis. Comput.* **2014**, *32*, 522–532. [CrossRef]
42. Feng, B.; He, F.; Wang, X.; Wu, Y.; Wang, H.; Yi, S.; Liu, W. Depth-projection-map-based bag of contour fragments for robust hand gesture recognition. *IEEE Trans. Hum. Mach. Syst.* **2016**, *47*, 511–523. [CrossRef]
43. Warchoł, D.; Kapuściński, T.; Wysocki, M. Recognition of fingerspelling sequences in polish sign language using point clouds obtained from depth images. *Sensors* **2019**, *19*, 1078. [CrossRef]
44. Ameen, S.; Vadera, S. A convolutional neural network to classify American Sign Language fingerspelling from depth and colour images. *Expert Syst.* **2017**, *34*, e12197. [CrossRef]
45. Nai, W.; Liu, Y.; Rempel, D.; Wang, Y. Fast hand posture classification using depth features extracted from random line segments. *Pattern Recognit.* **2017**, *65*, 1–10. [CrossRef]
46. Maqueda, A.I.; del-Blanco, C.R.; Jaureguizar, F.; García, N. Human–computer interaction based on visual hand-gesture recognition using volumetric spatiograms of local binary patterns. *Comput. Vis. Image Underst.* **2015**, *141*, 126–137. [CrossRef]
47. Keskin, C.; Kıraç, F.; Kara, Y.E.; Akarun, L. Hand pose estimation and hand shape classification using multi-layered randomized decision forests. In Proceedings of the European Conference on Computer Vision, Florence, Italy, 7–13 October 2012; pp. 852–863.
48. Rady, M.A.; Youssef, S.M.; Fayed, S.F. Smart Gesture-based Control in Human Computer Interaction Applications for Special-need People. In Proceedings of the IEEE Novel Intelligent and Leading Emerging Sciences Conference (NILES), Cairo, Egypt, 28–30 October 2019; pp. 244–248.
49. Aly, W.; Aly, S.; Almotairi, S. User-Independent American Sign Language Alphabet Recognition Based on Depth Image and PCANet Features. *IEEE Access* **2019**, *7*, 123138–123150. [CrossRef]
50. Rakowski, A.; Wandzik, L. Hand shape recognition using very deep convolutional neural networks. In Proceedings of the 2018 International Conference on Control and Computer Vision, Singapore, 15–18 June 2018; pp. 8–12.
51. Yosinski, J.; Clune, J.; Bengio, Y.; Lipson, H. How transferable are features in deep neural networks? *Adv. Neural Inf. Process. Syst.* **2014**, *27*, 3320–3328.
52. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
53. Huang, G.; Liu, Z.; Van Der Maaten, L.; Weinberger, K.Q. Densely connected convolutional networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 4700–4708.
54. Howard, A.G.; Zhu, M.; Chen, B.; Kalenichenko, D.; Wang, W.; Weyand, T.; Andreetto, M.; Adam, H. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv* **2017**, arXiv:1704.04861.
55. Dadashzadeh, A.; Targhi, A.T.; Tahmasbi, M.; Mirmehdi, M. HGR-Net: A fusion network for hand gesture segmentation and recognition. *IET Comput. Vis.* **2019**, *13*, 700–707. [CrossRef]
56. Tajbakhsh, N.; Shin, J.Y.; Gurudu, S.R.; Hurst, R.T.; Kendall, C.B.; Gotway, M.B.; Liang, J. Convolutional neural networks for medical image analysis: Full training or fine tuning? *IEEE Trans. Med. Imaging* **2016**, *35*, 1299–1312. [CrossRef]
57. Wikipedia. American Manual Alphabet. Available online: http://en.wikipedia.org/wiki/American_manual_alphabet (accessed on 1 October 2021).