Hindawi
Journal of Sensors
Volume 2021, Article ID 4312245, 8 pages
https://doi.org/10.1155/2021/4312245

Research Article

SqueezeFace: Integrative Face Recognition Methods with LiDAR Sensors

Kyoungmin Ko,[1] **Hyunmin Gwak**,[1] **Nalinh Thoummala**,[2] **Hyun Kwon**,[3] **and SungHwan Kim**[1]

[1] *Department of Applied Statistics, Konkuk University, Seoul, Republic of Korea*
[2] *AI Analytics Team, DeepVisions, Seoul, Republic of Korea*
[3] *Department of Electrical Engineering, Korea Military Academy, Seoul, Republic of Korea*

Correspondence should be addressed to Hyun Kwon; hkwon.cs@gmail.com and SungHwan Kim; shkim1213@konkuk.ac.kr

Received 20 May 2021; Revised 10 August 2021; Accepted 19 August 2021; Published 28 September 2021

Academic Editor: Haibin Lv

In this paper, we propose a robust and reliable face recognition model that incorporates depth information such as data from point clouds and depth maps into RGB image data to avoid false facial verification caused by face spoofing attacks while increasing the model's performance. The proposed model is driven by the spatially adaptive convolution (SAC) block of SqueezeSegv3; this is the attention block that enables the model to weight features according to their importance of spatial location. We also utilize large-margin loss instead of softmax loss as a supervision signal for the proposed method, to enforce high discriminatory power. In the experiment, the proposed model, which incorporates depth information, had 99.88% accuracy and an $F1$ score of 93.45%, outperforming the baseline models, which used RGB data alone.

1. Introduction

LiDAR, short for light detection and ranging, is a remote sensing technology similar to radar. The difference is that radar uses radio waves to detect its surroundings, whereas LiDAR uses laser energy. When a LiDAR sensor directs a laser beam at an object, it can calculate the distance to the object by measuring the delay before the light is reflected back to it, making it possible to extract depth information for an object and display it in the form of a point cloud or depth map. Not only can LiDAR sensors estimate an object's range but also they can measure its shape with high accuracy and spatial resolution. Furthermore, LiDAR sensors are robust under various lighting conditions (day or night, with or without glare and shadows), thereby overcoming the disadvantages of other sensor types. Because of its superiority, LiDAR has been widely used in a variety of applications, including autonomous vehicles, river surveys, and pollution modeling. Recently, products launched by technology companies often come equipped with a LiDAR scanner, making it more convenient to obtain depth information for objects in the form of 3D point clouds, as shown in Figure 1.

A face recognition system is a computer-assisted application that automatically determines or verifies an individual's identity using digital images. In practice, the system verifies the person's identity by comparing intensity images of the face captured by a camera with prestored images. It can be used for biometric authentication and is emerging as a critical authentication method for information and communications technology (ICT) services. Security-based applications are spreading to various fields; they include employee attendance checks, airport surveillance, and bank transactions. A face recognition system can provide a straightforward yet convenient authentication process, as it can operate using just an RGB image captured from a person's face. However, this simplicity makes it vulnerable to

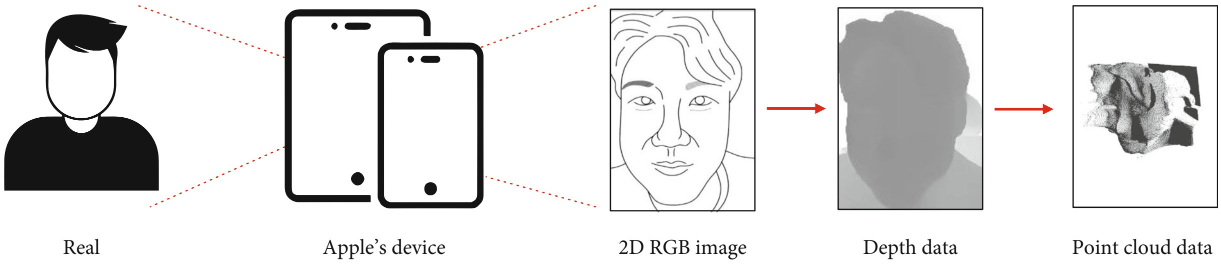


Figure 1: Capture of RGB image, depth, and point cloud data using a LIDAR scanner-equipped device.

spoofing attacks [1, 2] because pictures of people's faces can easily be obtained on social media platforms without their consent, and these can be used by someone with malicious intent to steal a person's identity. To prevent such face spoofing attacks, we propose a robust face recognition method that uses both RGB images and depth information such as those extracted from point clouds and depth maps produced by a LiDAR scanner.

Face recognition based on RGB images is already widely acknowledged for its promising performance. However, the determination of whether a face is real or fake, known as liveness detection, cannot be performed simultaneously. Distinguishing in terms of liveness between RGB images captured directly from people's faces using a camera and digital images from other sources used for face spoofing attacks remains challenging because the two images are just one type of input used by the recognition system. A point cloud and depth map, however, can be obtained only by direct capture from people's faces using sensors such as LiDAR. In addition, depth information is three-dimensional. In other words, spoofing attacks using 2D digital images are immediately identifiable by their lack of 3D information.

The main feature of the proposed method is a face recognition model that incorporates depth information into RGB images. The method uses a device equipped with a LiDAR sensor to collect the supplementary data. Because the method utilizes point cloud and depth data, it solves the liveness detection problem of the existing 2D face recognition method. We also hypothesize that a deep learning framework using depth information can demonstrate higher performance on the classification model for face recognition systems.

According to the developers of the SqueezeSeg3 model [3], point cloud data present strong spatial priors, and their feature distributions vary according to spatial location. Thus, we built an attention-based deep convolutional model based on SqueezeSeg3, called SqueezeFace. Its architecture is shown in Figure 2.

Based on previous studies [4–9], we additionally adopted large-margin loss as a supervision signal that enables the model to learn highly discriminative deep features for face recognition by maximizing interclass variance and minimizing intraclass variance during the training phase. In the test phase, facial embedding features are extracted 5using our proposed convolution network for face verification. The method can then verify an identity by calculating the cosine similarity between embedding features. The proposed method delivers performance superior to that of existing methods that use only RGB images. The remainder of this paper is organized as follows. In Section 2, related work is reviewed. The structure of the proposed method is described in detail in Section 3. The experimental results are discussed in Section 4. Finally, we conclude the paper in Section 5.

2. Related Work

Convolutional neural networks (CNNs) are powerful models that play an essential role in learning feature representations that best describe the given domain while maintaining the spatial information of an image. Because of their excellence in learning important patterns, CNNs have achieved breakthroughs on a variety of computer vision tasks such as those involved in image classification, object detection, and semantic segmentation [10–16].

Attention-based CNNs in particular have attracted considerable interest and have been extensively exploited to improve a model's performance on numerous computer vision tasks by integrating attention modules with the existing CNN architecture [3, 17–20]. The attention module allows the model to selectively emphasize important features and discard less informative ones. Hu et al. [17] proposed the Squeeze-and-Excitation (SE) block, which learns the relationship between the channels of its convolutional features and adaptively recalibrates channel weights according to the relationship learned. Specifically, the SE block extracts a representative scalar value for each channel using global average pooling (GAP) and assigns a weight for each channel based on the interdependency between channels through the excitation process. Park et al. [19] introduced the simple yet efficient Bottleneck Attention Module (BAM), which generates attention maps by separating the process of inferring a attention map into a channel attention module and a spatial attention module and configures them in parallel. Woo et al. [20] presented the lightweight Convolutional Block Attention Module (CBAM), which sequentially applies channel and spatial attention modules to emphasize important elements in both the channel and spatial axes.

Exploiting face representation embedding features extracted using a deep CNN is one of several methods used in face recognition tasks [9, 21–24]. Face recognition using a deep CNN involves two essential preprocessing steps: face detection and face alignment. These two tasks should be performed jointly because they are inherently correlated [25].

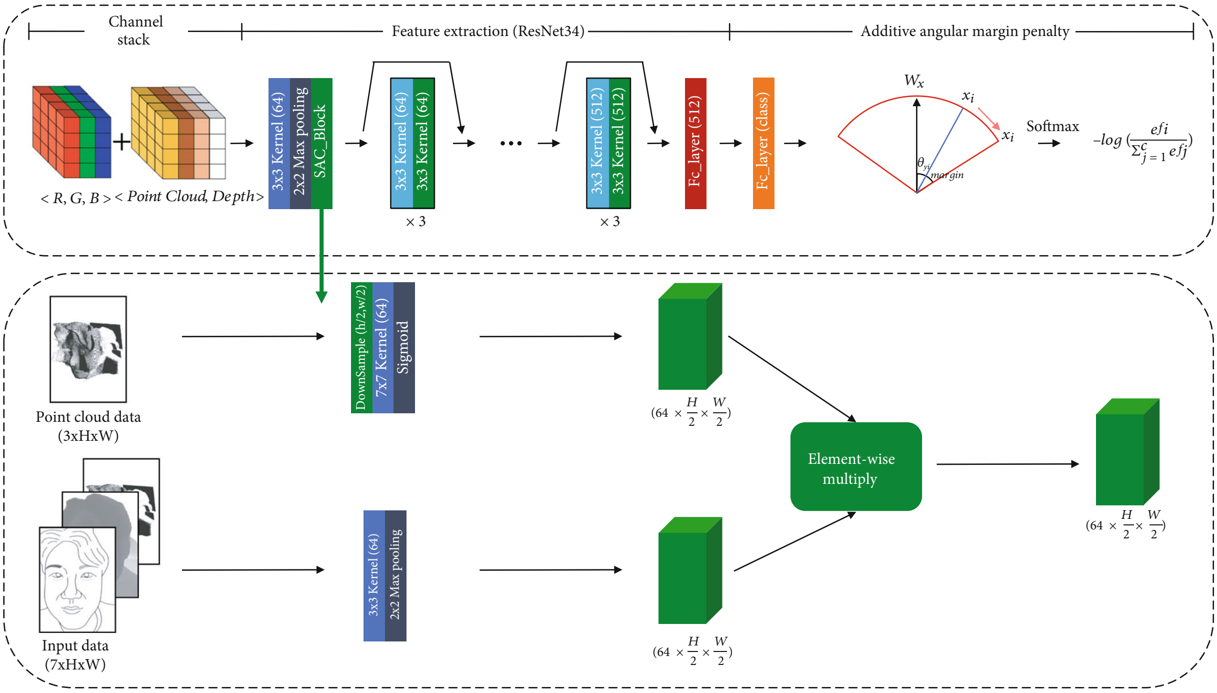


Figure 2: Integration of the attention block in the SqueezeFace architecture.

Softmax loss [26] is commonly used as a loss function to supervise the face recognition model and was used in DeepID [21] and DeepFace [22]. However, recent studies have indicated that softmax loss is not suitable for face recognition tasks owing to its inability to optimize the feature embedding to enforce strong similarity within positive class samples and diversity across negative class samples, which can deteriorate model performance on face recognition. Suggested alternatives included functions based on Euclidean distance, such as contrastive loss, triplet loss, and center loss, to alleviate such constraints while strengthening discriminative features.

Contrastive loss was proposed as the loss function in DeepID2 [21] and DeepID3 [27]. Generally, this loss requires pairs of inputs, and it will adjust the distances between embedding features differently depending on whether the pair belongs to the positive class (for an intraclass pair) or the negative class (for an interclass pair). To increase the learning efficiency of contrastive loss, triplet loss was proposed in FaceNet [23]. Unlike contrastive loss, triplet loss requires three inputs, two of which are in the same class and the third belongs to a different class. This loss function reduces the distance between the intraclass pairs and increases the distance between the interclass pairs. Despite being used in many metric learning methods because of its excellent performance, triplet loss requires an expensive preprocessing step in constructing input data for the distance comparison. Thus, center loss was proposed to learn the centroid of the features of each class and penalize the distances between the centroids and their corresponding class features. This loss not only handles the complicated input data preprocessing step but also boosts performance.

In addition to the losses described above, there exists a series of losses that incorporate a large angular margin to strengthen discriminatory power on classification, decrease the distance between features within the same class, and increase the distance between features from different classes [7–9]. We discuss these losses in detail in Section 3.

Traditional face recognition methods utilize only RGB data as the input. Such methods perform relatively well, but they present a disadvantage with regard to liveness in that the model cannot distinguish whether an image has been captured directly from a person's face or is a digital image obtained from other sources. This characteristic makes such methods vulnerable to face spoofing attacks. Recent studies have sought to mitigate this problem by adding depth information in the form of point cloud and depth data as inputs. Fuseseg [28], Fusenet [29], and Chinet [30] have been proposed for boosting model performance by effectively fusing such data collected from various sensors. Each model has different methods for data fusion, and each embedding feature created is fused at the layer level.

3. Proposed Method

In this section, we describe the proposed face recognition method, which uses not only RGB images but also depth and point cloud data (3D coordinates) extracted from LiDAR sensors. We constructed the proposed model with a data integration network that processes data serially from different sensors. Because it is imperative to emphasize features that will influence the model's performance, the attention mechanism was adopted to allow the model to capture and best exploit important features from the point cloud. For the operational technique, we incorporated the spatially adaptive convolution (SAC) block of SqueezeSegv3 into a data integration network to process our data and extract features from them.

In addition, we replaced softmax loss with large-margin loss for supervising the feature embedding process to increase similarity within the same class and discrepancy between different classes. We discuss in detail the construction of the proposed data integration network and the large-margin loss function in Sections 3.1 and 3.2, respectively.

3.1. SqueezeSegv3. Most face recognition models are based on deep convolutional neural networks (DCNNs) to have discriminatory power for classification. Facial feature representations can be extracted with standard convolution as

$$Y[m, p, q] = \sigma\left(\sum_{i,j,n} W[m, n, i, j] \times X\left[n, p + \hat{i}, q + \hat{j}\right]\right), \quad (1)$$

where $Y \in R^{O\times S\times S}$ and $X \in R^{I\times S\times S}$ are the output and input tensors; $W \in R^{O\times I\times K\times K}$ is the convolutional weight matrix, in which K is the convolutional kernel size; O and I are the output and input channel sizes; S represents the image size; and $\sigma(\cdot)$ is a nonlinear activation function such as ReLU [31]. In this method, $\hat{i}$ and $\hat{j}$ are defined as $\hat{i} = i - \lfloor K/2 \rfloor$ and $\hat{j} = j - \lfloor K/2 \rfloor$. As mentioned with regard to the SqueezeSegv3 model [3], standard convolution is based on the assumption that the distribution of visual features is invariant to the spatial location of the image. This assumption is largely true in the case of RGB images; thus, a convolution uses the same weight for all input locations. However, this assumption cannot be applied to point cloud data as X-coordinate point cloud data present very strong spatial priors, and the feature distribution of the point cloud varies substantially at different locations. In consideration of this fact, the SAC block, which is designed to be spatially adaptive and content aware using 3D coordinates of a point cloud, is proposed to apply different weights for different image locations as follows:

$$Y[m, p, q] = \sigma\left(\sum_{i,j,n} W(X_0)[m, n, p, q, i, \mathrm{j}] \times X\left[n, p + \hat{i}, q + \hat{j}\right]\right). \quad (2)$$

In SqueezeSegv3 [3], $W(\cdot) \in R^{O\times I\times S\times S\times K\times K}$ is a spatially adaptive function of the raw input X_0, which depends on the location (p,q). In this method, X_0 is only the raw input point cloud. $W(\cdot)$, the spatially adaptive function of SqueezeFace, is shown in detail in the lower part of Figure 2.

To process our data, which are gathered from different sources, an appropriate data fusion model is required. Seven channels are constructed for the input data by stacking RGB, depth, and point cloud data, which are collected from different sensors and possess different characteristics. To obtain attention map A, the point cloud data are fed into a 7×7 convolution followed by a sigmoid function. Next, this attention map A is combined with the input tensor X. Then, a standard convolution with weight W is applied to the adapted input. For the embedding network, we employ the well-known ResNet34 architecture [32]. The ResNet model reduces the image size as it passes through each layer. The downsampling process for the point cloud has difficulty in properly utilizing spatial coordinate information because of the small size of our dataset. Therefore, the SAC block is used at the initial layer, as shown in Figure 2. The network successfully maps the face input to face representation embedding features, combining the three types of data.

3.2. Large-Margin Loss. The face recognition task is a multiclass classification, defined as the problem of classifying images into one of certain classes. The most commonly used loss for multiclass classification is softmax loss, which is a softmax activation function followed by cross-entropy loss [33]. The softmax activation function outputs the probability for each class, whose sum is one, and the cross-entropy loss is the sum of the negative logarithms of these probabilities, defined as

$$L_1 = -\frac{1}{N}\sum_{i=1}^{N} \log \frac{e^{W_{y_i}^T x_i + b_{y_i}}}{\sum_{j=1}^{n} e^{W_j^T x_i + b_j}}, \quad (3)$$

where x_i is the feature vector of sample data, y_i represents the truth class corresponding to x_i, and W and b are weight and bias terms, respectively. Despite being widely used, softmax loss has some limitations as it does not strictly enforce higher similarity within the same class and discrepancy between different classes. Thus, traditional softmax loss may create a performance gap for face recognition when intraclass variation is high because of factors such as age gaps, differences in facial expression, and variations in pose (left, right, or frontal). To enable the model to circumvent this problem, A-softmax loss was proposed as a reformulation of the traditional softmax loss in SphereFace [5] as follows:

$$L_2 = -\frac{1}{N}\sum_{i=1}^{N} \log \frac{e^{\|x_i\| \cos\left(m\theta_{y_i,i}\right)}}{e^{\|x_i\| \cos\left(m\theta_{y_i,i}\right)} + \sum_{j\neq y_i} e^{\|x_i\| \cos\left(\theta_{j,i}\right)}}, \quad (4)$$

where m is the angular margin and $\theta_{y_i,i}$ is the angle between the vectors W_{y_i} and x_i. A-softmax loss adopts $W_{y_i}^T x_i$ as the linear form, which is expressed as $\|W_{y_i}\| \|x_i\| \cos(\theta_{y_i,i})$. This loss enables metric learning by constraining the classification weight's norm to 1 through normalization, setting the bias to 0 and incorporating the angular margin adjusted via parameter m to capture discriminative features with clear geometric interpretation.

Then, the CosFace model [8] was proposed, which includes a large-margin cosine loss function that normalizes both weights and features by L2 normalization to eliminate radial variations and adds a quantitative m value, a fixed parameter used to control the magnitude of the cosine

Table 1: Comparison of models' performance on the test set.

Model	Accuracy	$F1$ score
RGB-only, ResNet34	0.9979	0.8995
Our data, ResNet34	0.9980	0.9056
Our data+SqueezeFace	0.9988	0.9345

Table 2: Face verification performance comparison on three-shot learning between the RGB-only model and proposed three-sensor-data-type model.

Statistic	RGB	RGB + depth + point cloud
Number of output classes	83	83
Number of training images	248	248
Number of testing images	536	536
Testing accuracy	0.9973	0.9977
Testing $F1$ score	0.8884	0.9036
Best threshold	0.7255	0.8026

margin. The overall loss function can be expressed as

$$L_3 = -\frac{1}{N}\sum_{i=1}^{N} \log \frac{e^{s\left(\cos\left(\theta_{y_i,i}\right)-m\right)}}{e^{s\left(\cos\left(\theta_{y_i,i}\right)-m\right)} + \sum_{j\neq y_i} e^{s\left(\cos\left(\theta_{j,i}\right)\right)}}, \quad (5)$$

where s is a rescale parameter, used by the loss function to rescale the weights and features after normalizing them.

ArcFace [9] adds an additive angular margin penalty m between weights and features. This penalty is equal to the geodesic distance margin penalty in the normalized hypersphere and thus is named ArcFace. The loss function is formulated as follows:

$$L_4 = -\frac{1}{N}\sum_{i=1}^{N} \log \frac{e^{s\left(\cos\left(\theta_{y_i,i}+m\right)\right)}}{e^{s\left(\cos\left(\theta_{y_i,i}+m\right)\right)} + \sum_{j\neq y_i} e^{s\left(\cos\left(\theta_{j,i}\right)\right)}}. \quad (6)$$

Thus, we can supervise our model using additive angular margin loss that combines the margin penalties of SphereFace [5], CosFace [8], and ArcFace [9], which demonstrates the best performance, as follows:

$$L_5 = -\frac{1}{N}\sum_{i=1}^{N} \log \frac{e^{s\left(\cos\left(m_1\theta_{y_i,i}+m_2\right)-m_3\right)}}{e^{s\left(\cos\left(m_1\theta_{y_i,i}+m_2\right)-m_3\right)} + \sum_{j\neq y_i} e^{s\left(\cos\left(\theta_{j,i}\right)\right)}}, \quad (7)$$

where m_1, m_2, and m_3 are the angular margin parameters, each represented as m in the loss functions described above. Our main task is to identify a class for each input identity. By adopting the proposed additive angular margin loss, the proposed model can increase the similarity of positive classes and enforce a wide diversity of negative classes in metric learning. The proposed large-margin loss can generate high-quality embedding features from our data, enabling high-accuracy classification with both the training dataset and the unseen test dataset.

4. Numerical Experiments

4.1. Datasets. The face dataset consisted of 784 face scans from 83 Korean individuals. The face data were captured using Apple's latest device equipped with a LiDAR scanner. Specifically, the device was equipped with three cameras (main, wide, and telephoto) and a LiDAR scanner for capturing both RGB image and depth information. ARKit can be used to connect with the scanner on the Apple device and process the depth and point cloud (3D coordinate) data. ARKit recently introduced a new depth API available only for devices equipped with a LiDAR scanner and provides several methods to access depth information collected from LiDAR scanners. The LiDAR scanner allows this API to obtain per-pixel depth information of a person's face and generate 3D coordinates of the point cloud by setting the parameters for the device. We modified ARKit's sample code and set up the application to simultaneously store RGB and point cloud data within one scene. We installed this modified app on the device and collected data through the app.

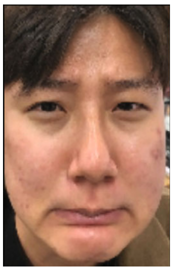
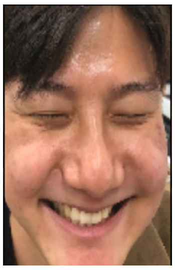
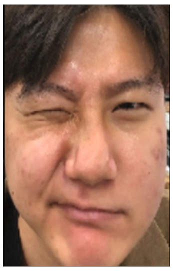
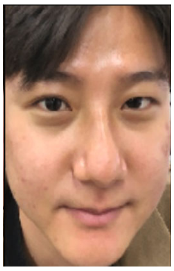
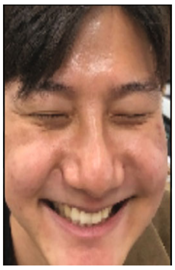
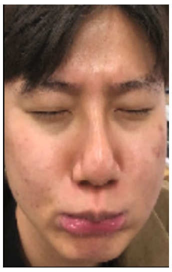
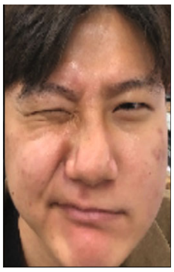
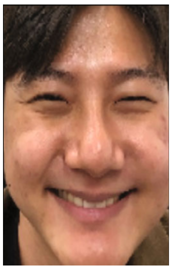
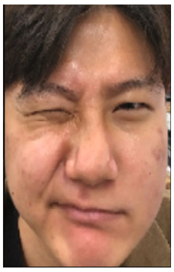
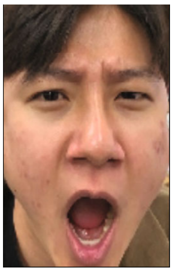
4.2. Experiment Setup. We trained three different models to compare their performance. The first model used only RGB data. The second model used three types of sensor data (RGB, depth, and point cloud) with three different characteristics, and the third model was the SqueezeFace model that uses the SAC block on the three types of sensor data. All three models used the ResNet34 architecture [32] and large-margin loss [6]. The ResNet34 model is pretrained using a facial image dataset of 400 Korean individuals, provided by AI Hub (https://aihub.or.kr/). For the three sensor data models, pretrained weights from ResNet34 were used as the weights of the RGB data, and the weights for the point cloud and depth data were initialized using the Xavier initializer.

4.3. Experiment Results. We split our face dataset into a training set and a test set, and the sensor data were configured as three types (RGB, depth, and point cloud). In addition, to evaluate the face verification performance, we constructed a face verification dataset with pairs of face images from the test set. Accuracy, precision, and recall were used as metrics to measure the model's performance for face verification. Accuracy is the ratio of the number of correct predictions to the total number of inputs. Precision is the ratio of the number of true positive predictions to the total number of the model's predicted positive values, and recall is the ratio of the number of true positive predictions to the number of all positive samples. These three definitions are represented as

$$\begin{aligned} \text{Accuracy} &= \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}}, \\ \text{Precision} &= \frac{\text{TP}}{\text{TP} + \text{FP}}, \\ \text{Recall} &= \frac{\text{TP}}{\text{TP} + \text{FN}}, \end{aligned} \quad (8)$$

where TP, TN, FP, and FN denote true positive, true

Table 3: Cosine similarity for various facial expressions.

Description	Pair_1	Pair_2	Pair_3	Pair_4	Pair_5
					
					
RGB	0.9118	0.7450	0.7734	0.5164	0.5558
RGB + depth + point cloud	0.9664	0.9050	0.8781	0.8258	0.8484

negative, false positive, and false negative, respectively. For the face verification dataset, the number of interclass combinations was much greater than the number of intraclass combinations. Because the intraclass and interclass counts were considerably imbalanced, the $F1$ score—the harmonic mean of precision and recall—was used as the evaluation metric for face verification:

$$F1\text{ score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}. \tag{9}$$

4.3.1. Analysis of Face Verification Results of the Proposed Method. According to the experimental results, shown in Table 1, the model using the three types of sensor data outperformed the model using only RGB data, demonstrating that employing depth information can enhance rich facial representation. More importantly, the proposed SqueezeFace model, with the added SAC attention block, achieved the best accuracy and $F1$ score. This result shows that the proposed model learned well the face points with high importance by actively utilizing the point cloud data with different distributions according to the spatial location. The intraclass variance due to pose variations and age gaps significantly increases the angle between positive pairs and therefore can increase the best threshold for face verification on test data. However, if the train data for each identity are limited, making the intraclass variance small, it is difficult to increase the best threshold for face verification on test data. A low threshold used in the evaluation of face verification indicates a low reliability of the model. The proposed model addresses this problem by adding point cloud and depth data to the RGB data.

The results for face verification performance on three-shot learning are compared in Table 2. Three-shot learning is learning that takes place using only three training samples. The best threshold is the threshold with the maximum $F1$ score. The model using the three types of sensor data shows higher accuracy, a higher $F1$ score, and an increase in the threshold than the RGB-images-only model. This demonstrates that by making use of supplementary information such as point cloud and depth data, the proposed model can increase intraclass variance and, as a result, increase the best threshold for face verification.

4.3.2. Analysis of Cosine Similarity on Three-Shot Learning of the Proposed Method. We examined the cosine similarity for various facial expressions on three-shot learning, with results as shown in Table 3. The proposed model produced better similarity values between positive pairs than the RGB-images-only model, even with a variety of facial expressions. Because the proposed method uses more information of face by adding depth and point cloud, the intraclass variance of the model can increase the angle between positive pairs. Therefore, the model can increase the cosine similarity, and the higher cosine similarity can increase the best threshold on face verification. This result demonstrates that adding depth and point cloud data enables the model to learn important facial features for face verification more effectively than the model with only RGB data. In addition, despite the difference between the same identities according to pose variations, the proposed method can distinguish the identity well in the test data by adding depth and point cloud data.

5. Conclusion

This paper has proposed a face recognition approach that considers depth information using point cloud data. By using depth information, false facial verification using a face photo or video of an authorized person can be avoided, thereby increasing the reliability of the face recognition system. The method incorporates the SAC block based on the attention mechanism to capture important features and weight them to enhance model performance. In addition, we used a modified loss function constructed by adding a large margin to reinforce high discriminatory power for face recognition applications [34]. The proposed method delivers a considerable performance improvement over the baseline models and uses a higher threshold for face verification when subjected to an increase in intraclass variance.

Data Availability

All source codes are available online at https://github.com/kyoungmingo/Fusion_face (author's webpage)

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This research was supported by the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (NRF-2020R1C1C1A01005229 and NRF-2021R1A4A5032622).

References

[1] S. Kumar, S. Singh, and J. Kumar, "A comparative study on face spoofing attacks," in *2017 International Conference on Computing, Communication and Automation (ICCCA)*, pp. 1104–1108, Greater Noida, India, 2017.

[2] T. Girdler and V. G. Vassilakis, "Implementing an intrusion detection and prevention system using software-defined networking: defending against ARP spoofing attacks and blacklisted MAC addresses," *Computers & Electrical Engineering*, vol. 90, p. 106990, 2021.

[3] C. Xu, B. Wu, Z. Wang et al., "Squeezesegv3: spatially adaptive convolution for efficient point-cloud segmentation," in *European Conference on Computer Vision*, pp. 1–19, Springer, 2020.

[4] J. Deng, Y. Zhou, and S. Zafeiriou, "Marginal loss for deep face recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pp. 60–68, Honolulu, HI, USA, 2017.

[5] W. Liu, Y. Wen, Z. Yu, M. Li, B. Raj, and L. Song, "Sphereface: deep hypersphere embedding for face recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 212–220, Honolulu, HI, USA, 2017.

[6] W. Liu, Y. Wen, Z. Yu, and M. Yang, "Large-margin softmax loss for convolutional neural networks," *ICML*, vol. 2, p. 7, 2016.

[7] F. Wang, J. Cheng, W. Liu, and H. Liu, "Additive margin softmax for face verification," *IEEE Signal Processing Letters*, vol. 25, no. 7, pp. 926–930, 2018.

[8] H. Wang, Y. Wang, Z. Zhou et al., "CosFace: large margin cosine loss for deep face recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 5265–5274, Salt Lake City, UT, USA, 2018.

[9] J. Deng, J. Guo, N. Xue, and S. Zafeiriou, "ArcFace: additive angular margin loss for deep face recognition," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4690–4699, Long Beach, CA, USA, 2019.

[10] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: towards real-time object detection with region proposal networks," 2015, https://arxiv.org/abs/1506.01497.

[11] G. Algan and I. Ulusoy, "Image classification with deep learning in the presence of noisy labels: a survey," *Knowledge-Based Systems*, vol. 215, article 106771, 2021.

[12] K. He, G. Gkioxari, P. Dollar, and R. Girshick, "Mask r-cnn," in *Proceedings of the IEEE international conference on computer vision*, pp. 2961–2969, 2017.

[13] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder-decoder with atrous separable convolution for semantic image segmentation," in *Proceedings of the European conference on computer vision (ECCV)*, pp. 801–818, 2018.

[14] W. Wang, Q. Lai, H. Fu, J. Shen, H. Ling, and R. Yang, "Salient object detection in the deep learning era: an in-depth survey," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021.

[15] O. Ronneberger, P. Fischer, and T. Brox, "U-net: convolutional networks for biomedical image segmentation," in *International Conference on Medical image computing and computer-assisted intervention*, pp. 234–241, Cham, 2015.

[16] S. Asgari Taghanaki, K. Abhishek, J. P. Cohen, J. Cohen-Adad, and G. Hamarneh, "Deep semantic segmentation of natural and medical images: a review," *Artificial Intelligence Review*, vol. 54, no. 1, pp. 137–178, 2021.

[17] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 7132–7141, 2018.

[18] J. Hu, L. Shen, S. Albanie, G. Sun, and A. Vedaldi, "Gather-excite: exploiting feature context in convolutional neural networks," 2018, https://arxiv.org/abs/1810.12348.

[19] J. Park, S. Woo, J.-Y. Lee, and I. S. Kweon, "BAM: bottleneck attention module," 2018, https://arxiv.org/abs/1807.06514.

[20] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, "CBAM: convolutional block attention module," in *Proceedings of the European conference on computer vision (ECCV)*, pp. 3–19, 2018.

[21] Y. Sun, X. Wang, and X. Tang, "Deep learning face representation by joint identification-verification," 2014, https://arxiv.org/abs/1406.4773.

[22] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf, "Deepface: closing the gap to human-level performance in face verification," in *Proceedingsof the IEEE conference on computer vision and pattern recognition*, pp. 1701–1708, 2014.

[23] F. Schroff, D. Kalenichenko, and J. Philbin, "Facenet: a unified embedding for face recognition and clustering," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 815–823, 2015.

[24] O. M. Parkhi, A. Vedaldi, and A. Zisserman, *Deep Face Recognition*, British Machine Vision Association, 2015.

[25] K. Zhang, Z. Zhang, Z. Li, and Y. Qiao, "Joint face detection and alignment using multitask cascaded convolutional networks," *IEEE Signal Processing Letters*, vol. 23, no. 10, pp. 1499–1503, 2016.

[26] X. Wang, S. Zhang, S. Wang, T. Fu, H. Shi, and T. Mei, "Misclassified vector guided softmax loss for face recognition," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 7, pp. 12241–12248, 2020.

[27] Y. Sun, D. Liang, X. Wang, and X. Tang, "Deepid3: face recognition with very deep neural networks," 2015, https://arxiv.org/abs/1502.00873.

[28] G. Krispel, M. Opitz, G. Waltner, H. Possegger, and H. Bischof, "Fuseseg: Lidar point cloud segmentation fusing multi-modal data," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 1874–1883, 2020.

[29] C. Hazirbas, L. Ma, C. Domokos, and D. Cremers, "Fusenet: incorporating depth into semantic segmentation via fusion-based CNN architecture," in *Asian Conference on Computer Vision*, pp. 213–228, Springer, 2016.

[30] V. John, M. Nithilan, S. Mita et al., "Sensor fusion of intensity and depth cues using the chinet for semantic segmentation of road scenes," in *2018 IEEE Intelligent Vehicles Symposium (IV)*, pp. 585–590, Changshu, China, 2018.

[31] A. F. Agarap, "Deep learning using rectified linear units (ReLU)," 2018, https://arxiv.org/abs/1803.08375.

[32] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedingsof the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

[33] Z. Zhang and M. R. Sabuncu, "Generalized cross entropy loss for training deep neural networks with noisy labels," 2018, https://arxiv.org/abs/1805.07836.

[34] W. Ali, W. Tian, S. U. Din, D. Iradukunda, and A. A. Khan, "Classical and modern face recognition approaches: a complete review," *Multimedia Tools and Applications*, vol. 80, no. 3, pp. 4825–4880, 2021.