mathematics

MDPI

Article

Latent-PER: ICA-Latent Code Editing Framework for Portrait Emotion Recognition

Isack Lee and Seok Bong Yoo *

Department of Artificial Intelligence Convergence, Chonnam National University, Gwangju 61186, Republic of Korea

* Correspondence: sbyoo@jnu.ac.kr; Tel.: +82-625303437

Abstract: Although real-image emotion recognition has been developed in several studies, an acceptable accuracy level has not been achieved in portrait drawings. This paper proposes a portrait emotion recognition framework based on independent component analysis (ICA) and latent codes to overcome the performance degradation problem in drawings. This framework employs latent code extracted through a generative adversarial network (GAN)-based encoder. It learns independently from factors that interfere with expression recognition, such as color, small occlusion, and various face angles. It is robust against environmental factors since it filters latent code by adding an emotion-relevant code extractor to extract only information related to facial expressions from the latent code. In addition, an image is generated by changing the latent code to the direction of the eigenvector for each emotion obtained through the ICA method. Since only the position of the latent code related to the facial expression is changed, there is little external change and the expression changes in the desired direction. This technique is helpful for qualitative and quantitative emotional recognition learning. The experimental results reveal that the proposed model performs better than the existing models, and the latent editing used in this process suggests a novel manipulation method through ICA. Moreover, the proposed framework can be applied for various portrait emotion applications from recognition to manipulation, such as automation of emotional subtitle production for the visually impaired, understanding the emotions of objects in famous classic artwork, and animation production assistance.

Keywords: generative adversarial network; latent code; portrait emotion recognition; independent component analysis

MSC: 68T45

Citation: Lee, I.; Yoo, S.B. Latent-PER: ICA-Latent Code Editing Framework for Portrait Emotion Recognition. *Mathematics* **2022**, *10*, 4260. https://doi.org/10.3390/math10224260

Academic Editors: Alvaro Figueira and Francesco Renna

Received: 7 September 2022
Accepted: 11 November 2022
Published: 14 November 2022

1. Introduction

In computer vision, facial expression analysis is an exciting research subject. Given that the input to the intelligent system is a facial image, facial expression recognition (FER) [1–5] is an essential visual recognition method to recognize emotions. In addition, FER has wide applications in the real world, such as autonomous vehicle driver monitoring, psychiatric treatments, education, and human-computer interaction. The deep neural network [6] has recently exhibited considerable performance in image recognition challenges. Furthermore, convolutional neural network (CNN) methods [7–17] are well-known deep learning techniques that automatically extract deep feature representations, as depicted in Figure 1.

The input space (i.e., a two-dimensional picture) is converted to a high-dimensional feature representation vector that captures the semantics of the input image for any visual recognition system with a defined set of classes. By combining features from lower to higher levels, deep CNN-based algorithms extract spatial characteristics that represent the abstract semantics of the input image. These extracted features reduce the amount of

information through a pooling layer. Then, the pooling layer is flattened into the form of a single vector through a fully connected layer. Therefore, a softmax function computes a probability distribution over all classes in the last stage. The softmax function expression for the i-th class is written as follows:

$$softmax(x_i) = \frac{e^{x_i}}{\sum_{j=1}^{k} e^{x_j}} \text{ for } i = 1, \ldots, k, \quad (1)$$

where k denotes the number of classes. Thereafter, similar to the mean squared error (MSE) loss, training proceeds to reduce the difference between the predicted class and ground truth through the loss function. The function expression is written as follows:

$$MSE = \frac{1}{n} \sum_{k=1}^{n} (y_k - \widetilde{y_k})^2 , \quad (2)$$

where n is the number of batch images, y_k denotes the ground truth, and $\widetilde{y_k}$ represents the predicted class. Figure 1 depicts this process graphically. Although CNN-based approaches have achieved promising accuracy in authentic facial photograph images, FER is still considered a challenging task in various portrait applications, such as the emotional recognition of famous drawings, paintings, cartoons, and animated films. In previous studies, the model has been designed to recognize facial expressions well when learning using actual photograph images. However, suppose that only the existing method of extracting feature maps through the CNN is used for portrait images that lack color information or have weak details compared to natural images, for example, cartoons, drawings, and paintings. In that case, the previous methods suffer severe emotion recognition performance degradation. Due to this problem, a model with acceptable expression recognition performance in various styles is needed. However, most of the current expression recognition research focuses on authentic-photograph images.

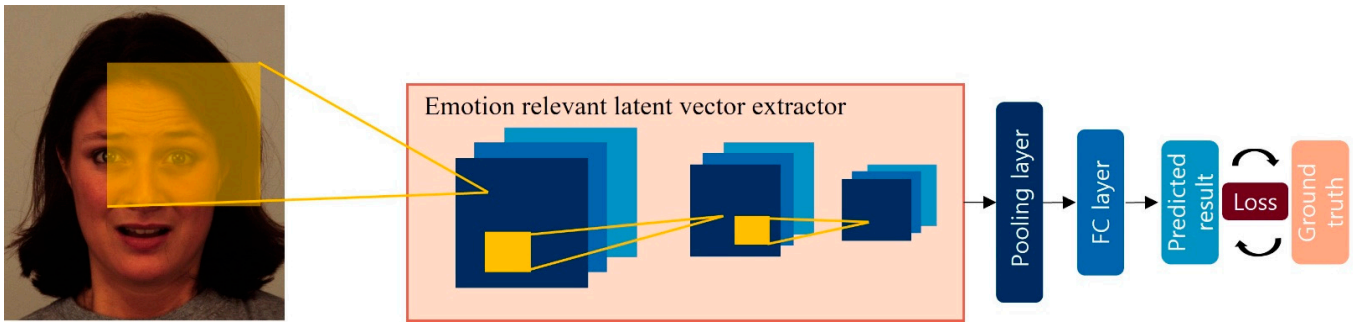

Figure 1. Typical emotion recognition model overview using a convolutional neural network (CNN).

To solve this problem, we propose a portrait emotion recognition (PER) framework using the latent code (Latent-PER), which is information extracted from the generated image. The latent code was edited using a new method based on independent component analysis (ICA) [18]. In this approach, we focus on the disentanglement and editability properties of generative adversarial network (GAN) [19] inversion. In the GAN, the disentanglement characteristic guarantees independence between styles without losing basic information when generating face images. Editability enables manipulation or editing concerning a specific attribute, allowing the latent code to be selectively used by excluding or retaining the information in emotion recognition models. The proposed method uses ICA [20] in the latent domain to determine elements that correlate highly with facial expression information. To facilitate analysis via ICA, we establish a statistically favorable situation.

For photographs of the same person, the information on the appearance is statistically similar. Furthermore, since facial expressions are made differently, information on facial expressions is statistically different. A dataset with the same object with different emotions is used to address this. Through an eigenvector that can classify the latent codes of these

images, we find elements related to emotions in the entire latent code. In addition, specific information in the latent code is used attentively to train the FER models. As illustrated in Figure 2a, the general emotion recognition method uses an image with entangled features and thus conditions, such as resolution and illumination of the input image are critical.

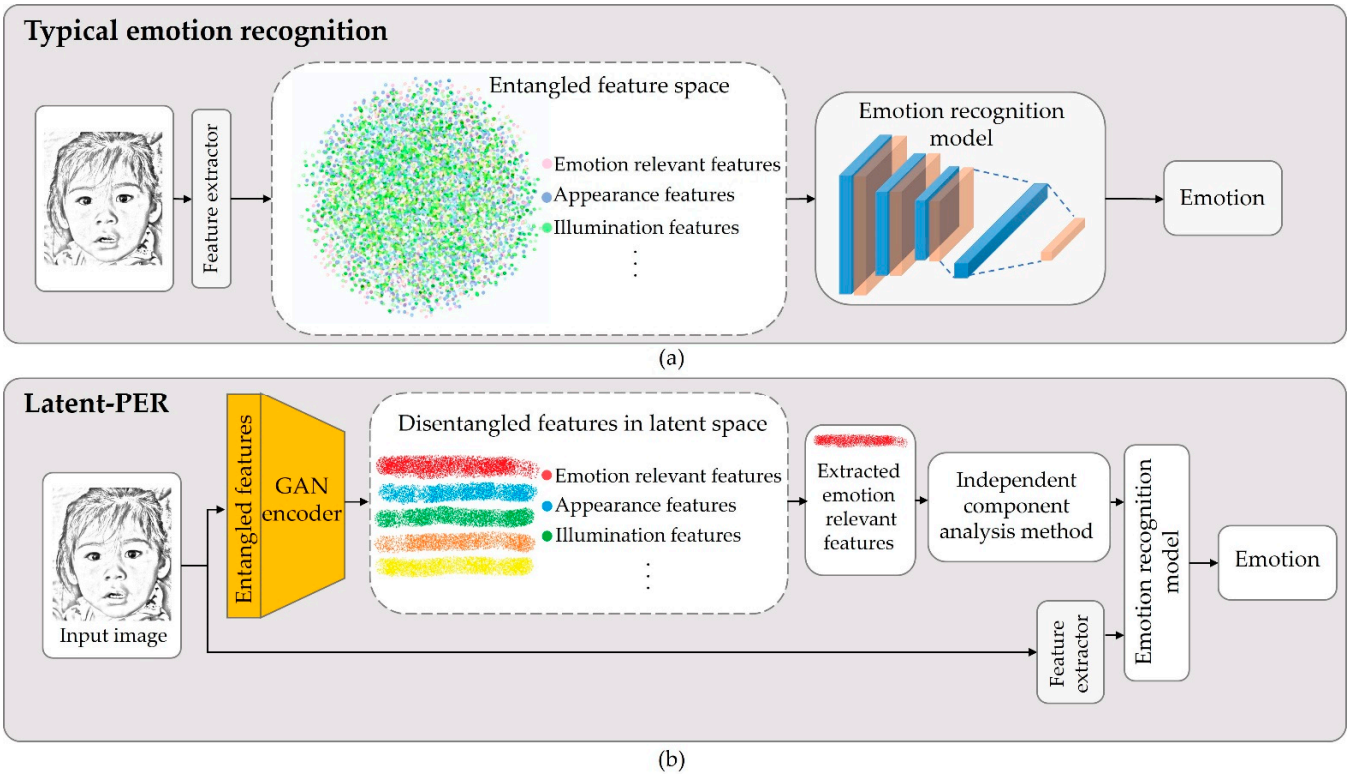

Figure 2. (**a**) Typical emotion recognition method using entangled features. (**b**) Latent-PER, which utilizes ICA-latent code with portrait image.

In contrast, as shown in Figure 2b, latent code-based approach disentangles information from the input image. The Latent-PER is a plug-and-play framework that employs disentangled features using a GAN inversion encoder on existing models. These two modules use images and latent codes to improve facial emotion recognition performance in portrait images with various styles and real photograph images.

Based on the above, the main contributions of this work can be summarized as follows:

1. We observe that existing authentic-photograph image-based models do not perform adequately in portrait images with various styles and propose a plug-and-play framework to prevent performance degradation in existing models.
2. We propose a latent code-based approach that disentangles the irrelevant emotional features of the image. Latent-PER provides robustness to various portrait image domains, such as drawings and paintings.
3. We propose a latent code editing method that rectifies the latent code to improve the PER performance by applying an eigenvector extracted using ICA. This eigenvector can also be employed for image manipulation while changing only relevant features.

2. Related Work

2.1. Facial Expression Recognition

Using FER, computers can better understand human behavior or communicate with people. The first FER system based on optical flow was introduced in 1991. Recent advances in deep learning have improved FER systems, and it is now possible to perform feature extraction and expression classification using a neural network. A FER system typically consists of three stages: Face detection, feature extraction, and expression recognition.

Although it varies from model to model, in face detection, several face detectors, such as MTCNN [21], FFHQ [22], and Dlib [23] are used to locate faces in complex scenes. Detected faces can be aligned further.

Various methods have been created to capture facial geometry and appearance features brought on by facial expressions. According to Fasel [24], a shallow CNN is robust to facial positions and scales. Using deep CNNs for feature extraction, Tang [9] and Kanou [25] and Ge et al. [26] pointed out that classification through SVM has lower performance than the CNN method. For this reason, they proposed a CNN-based facial expression recognition model. Kanou et al. [25] won in the FER challenges, respectively. An identity-aware CNN was created by Meng et al. [27] to discriminate between expression-related and identity-related information simultaneously. To determine the relative weights of several convolutional receptive fields in the network, Li et al. [28] proposed a multi scale CNN using an attention method. Wen et al. [29] proposed multiple cross-attention heads and ensured that they capture useful aspects of facial expressions without overlapping. Farzaneh et al. [30] proposed selecting a subset of significant feature components adaptively for improved discrimination. Moreover, Zhang et al. [31] used ResNet as the backbone to address the uncertainty problem in FER and proposed several uncertainty learning methods. Xu et al. [32] proposed a flexibly asymmetrical neural representation of facial expression recognition. Most studies [33–38] of facial expression recognition tasks developed with a focus on authentic-photograph images. Moetesum et al. [39] proposed an expression recognition method for sketch, but it is designed for images, such as emoticons. For this reason, when the existing model is applied to other style images, performance degradation occurs, and a new approach to solve this problem has not been proposed. Our proposed Latent-PER is a plug-and-play framework that employs disentangled features using a GAN inversion encoder on existing models.

2.2. Latent Space Embedding via GAN Inversion

Recent research has demonstrated that, as a result of picture production, GANs efficiently encode various semantic information in latent space [40]. Various manipulation techniques have been developed to extract and manipulate picture properties. Mirza et al. [41] trained early on creating conditional images, allowing for manipulating a specific picture property. Chrysos et al. [42] proposed a conditionally robust GAN network. Through latent spatial mapping, this network has significantly improved the image creation performance that meets the desired conditions. Abdal et al. [43] analyzed three semantic editing procedures that may be used on vectors in the latent space. Shen et al. [44] used principal component analysis (PCA) and a data-driven approach to determine the most important directions. In addition, Park et al. [38] provided a straightforward yet efficient method for conditional continuous normalizing flows in the GAN latent space conditioned by attribute features. However, these latent code manipulations are only relevant to pictures created by GANs that have already been trained, not to any actual image. We propose a novel manipulation method in the present study to discover the direction that correlates with the accuracy of face emotion identification.

Additionally, we demonstrate the value and necessity of latent code modification by demonstrating FER evaluations. The GAN inversion uses a pretrained generator to map an actual image into a latent space. Inversion must be semantically meaningful to perform editing and consider reconstruction performance. Zhu et al. [45] proposed a domain-guided encoder and domain-regularized optimizer to achieve semantically significant inversion. Furthermore, Tov et al. [46] studied the distortion, perception, and editability characteristics of high-quality inversion and demonstrated their inherent tradeoffs. Encoders generally learn to reduce distortion, which measures how close the input and target images are in the RGB and feature domains. He et al. [47] proposed an architecture that self-learns to separate and encode these unimportant variations. The advantage of the GAN-based encoder is disentanglement. Herein, we analyze this by applying ICA. As a result, it is effectively managed through an eigenvector, which expresses independent characteristics.

3. Method

This section describes the method proposed in this research for PER. The overall architecture of Latent-PER is presented in Figure 3. After facial detection and alignment, an image is inputted into a typical model and GAN-based encoder. The first module for the proposed method is the extraction of latent code through the GAN-based encoder to analyze and use it for learning. The second module is a typical model. Although the number of final features differs for each model, it is usually configured to output features through CNN. We propose a plug-and-play framework; therefore, any model the user desires can be used. Furthermore, in the third module, the concat layer combines the extracted latent code with the output value from the used model. The difference between prediction and ground truth is learned through MSE loss of Equation (2). The three modules are collaboratively learned.

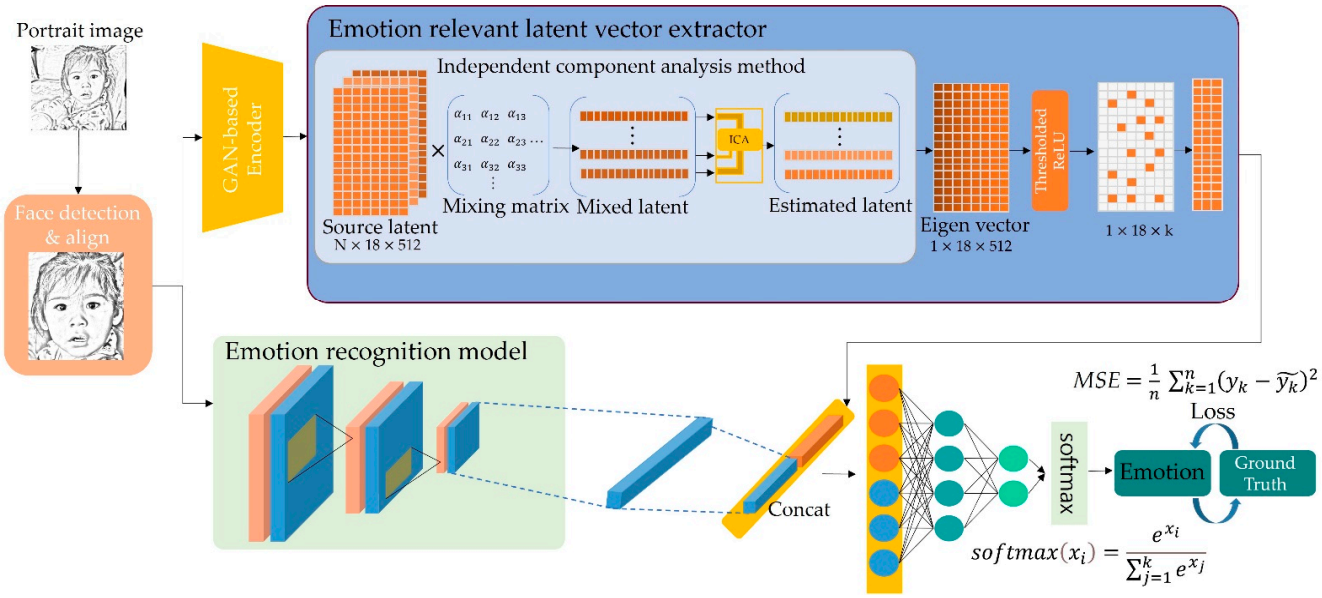

Figure 3. Overview of the proposed Latent-PER.

3.1. Conversion to Portrait-Style Images

We changed all the datasets we used to a drawing style following the aim of this work: PER. Figure 4 reveals that converting authentic-photograph images to a pencil drawing style is a five-step process. First, it converts the real photograph image to grayscale as follows:

$$imgGray = 0.2989 \times R + 0.5870 \times G + 0.1140 \times B, \quad (3)$$

where R, G, and B are the values for red, green, and blue in the same pixel, respectively.

Second, the converted grayscale image is inverted through a bitwise not operation as follows:

$$img\_Invert_{(2)} = 1 - imgGray_{(2)}, \quad (4)$$

where $imgGray_{(2)}$ represents the binary version of $imgGray$.

Third, $img\_Invert_{(2)}$ is converted to decimal and Gaussian blur is applied to this converted image using the Gaussian function:

$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}, \quad (5)$$

where x denotes the distance from the origin in the horizontal axis, y denotes the distance from the origin in the vertical axis, and σ represents the standard deviation of the Gaussian distribution.

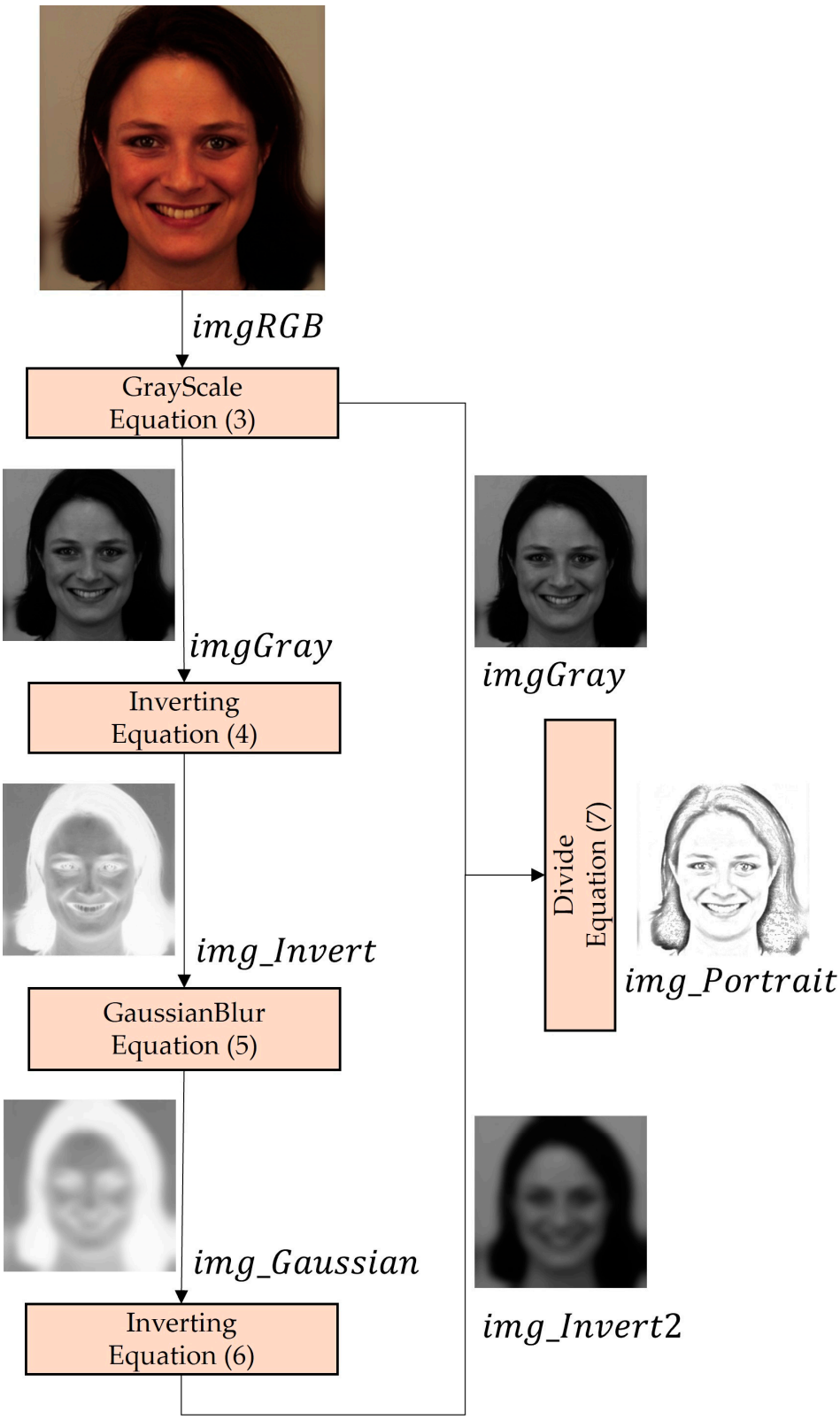

Figure 4. Conversion process of photographs to portrait drawing styles.

Fourth, the Gaussian blurred image is inverted again through a bitwise not operation as follows:

$$img\_Invert2_{(2)} = 1 - img\_Gaussian_{(2)}, \quad (6)$$

where $img\_Gaussian_{(2)}$ represents the binary version of *img_Gaussian* obtained from Gaussian blurring as in Equation (5).

Fifth, the grayscale image converted to decimal is divided with this inverted image as follows:

$$img\_Portrait = img\_Gray \; / \; img\_Invert2, \tag{7}$$

where *img_Invert*2 represents the decimal version of $img\_Invert2_{(2)}$.

To create drawing-style images, we set the kernel size in order that the edges are not blurred significantly. A larger kernel size of the Gaussian blur results in a more blurred image and the loss of detailed features. A kernel size of 3×3 or 5×5 is sufficient for small images but less effective for large images. Therefore, an appropriate kernel size is selected according to the size of the dataset. As a result, the photographs are converted into a portrait drawing-style, as shown in Figure 4.

3.2. Emotion Relevant Latent Code Extractor

3.2.1. Independent Component Analysis

As illustrated in Figure 5, ICA determines the same basis vector as the PCA. However, PCA finds the eigenvector in the direction with the most significant variance. Therefore, it is primarily used for dimension reduction. In contrast, ICA determines the basis vector that best represents each independent component. Therefore, it is suitable for the task since it is possible to determine an independent eigenvector for emotions using a data-driven approach from data with the same object with different emotions. In general, singular value decomposition and PCA are often used for manipulation in the GAN, which is suitable for diversifying the change in the generated image through the change in the latent code. However, we used the ICA method to locate the eigenvector representing the independent component rather than the basis vector with the most significant variance. The proposed Latent-PER uses eigenvectors. We only determined the index of the latent code that can change the facial expression.

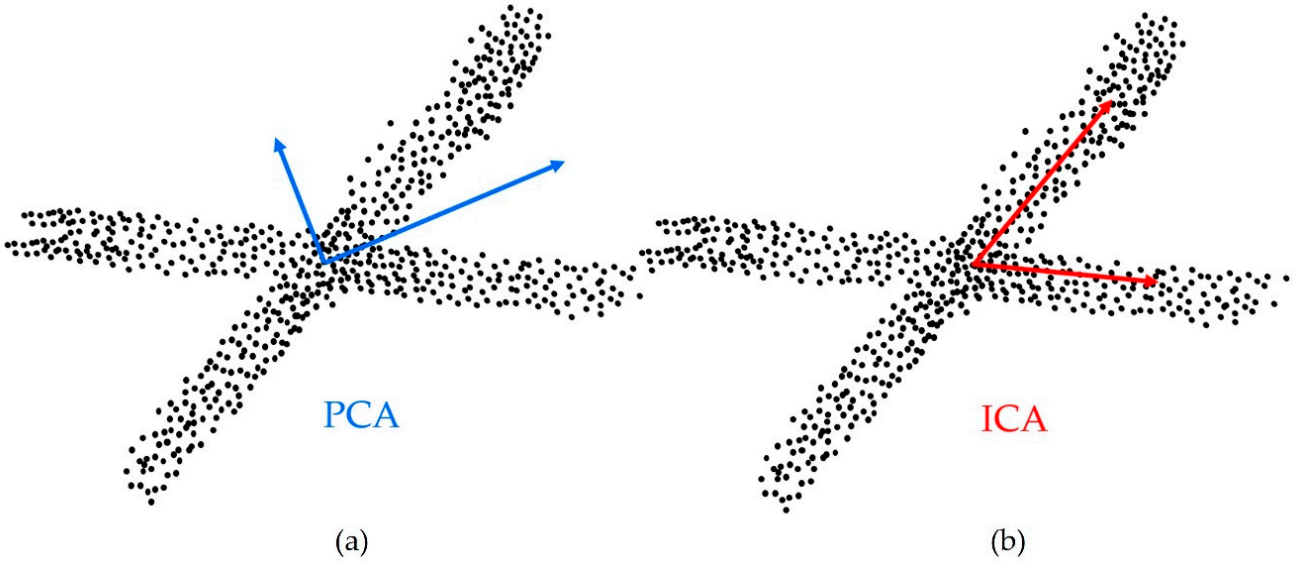

Figure 5. Visualization of eigenvectors after analyzing linear mixed signals using (**a**) PCA and (**b**) ICA.

Closely related to the problem of blind source separation, the goal of ICA is to decompose the observed signal into a linear combination of unknown independent signals, where s is the unknown source signal vector, and x is the vector of the observed mixture. If $\mathbf{A}$ is an unknown mixing matrix, the mixture model is written as follows:

$$x = \mathbf{A}s, \tag{8}$$

where **A** is an unknown non-square matrix that combines the components of the source s. Finding the mixing matrix **A** (more precisely, the inverse of **A**) is the aim of ICA to recover the original signal s from the observed data x. We can recover the underlying source $\hat{s}$ from the linearly converted data by building a new matrix **W** as follows:

$$\hat{s} = \mathbf{W}x. \tag{9}$$

The goal of ICA is to determine an unmixing matrix **W** that approximates $\mathbf{A}^{-1}$, resulting in $\hat{s} \approx s$. In addition, **A** can be divided into simpler pieces using a linear algebra singular value decomposition technique:

$$\mathbf{A} = \mathbf{U}\sum\mathbf{V}^{\mathrm{T}}, \tag{10}$$

where $\mathbf{U} \in \mathbb{R}^{\mathrm{N}\times\mathrm{M}}$ and $\mathbf{V} \in \mathbb{R}^{\mathrm{M}\times\mathrm{M}}$ are matrices with orthogonal columns and $\mathbf{\Sigma}$ is diagonal. A straightforward transformation of the probabilities reveals that $\mathbf{V}^{\mathrm{T}}$ is Gaussian with covariance.

3.2.2. Emotion-Relevant Latent Code Extractor

We propose a latent code extractor that can extract semantically meaningful information from the latent code. Figure 6 displays a method of extracting facial expression-related information using a dataset with the same object with different facial expressions. Therefore, the latent code for the external information of the same person is very similar. However, since the facial expressions are different, the location where the most significant difference appears in the latent code of the same person is facial expression-related information. For this reason, the Karolinska Directed Emotional Faces (KDEF) dataset is primarily used to analyze the latent code.

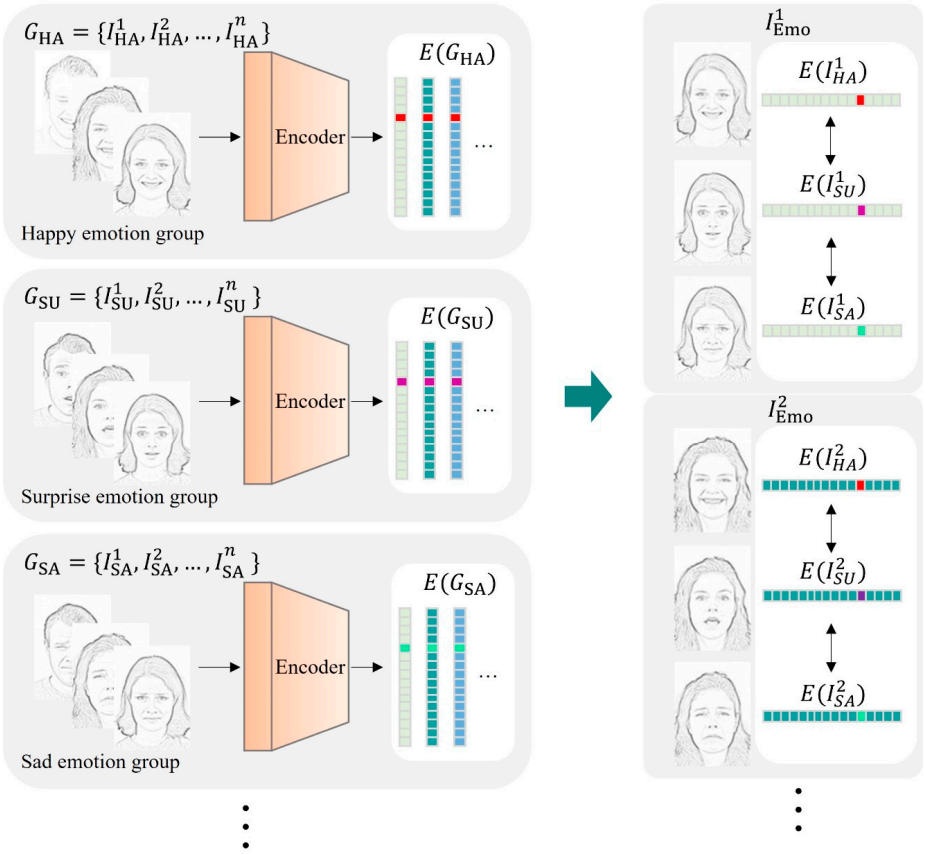

Figure 6. Explanatory diagram of latent code analysis through ICA.

The detailed process is described as follows. First, the image is inputted into the GAN-based encoder through facial detection and alignment. Through this process, the image is converted into latent code, which is an array of real numbers. The converted latent code contains information that can generate an input image. It can be estimated that the index of the latent code that can change the expression in the image as a result of creation contains information on the expression of the image. We used images of the same object but with different expressions to determine these indices. Since the images display the same face, the appearance information in the latent code is very similar, but the expression-related information differs since the expressions are different. As shown in Figure 6, we divide the groups by emotion. G_{HA} indicates a group whose label is happy, and G_{SU} indicates a group whose label is surprising. The upper subscript of I^1_{HA} distinguishes objects, and the lower subscript indicates emotion. Therefore, I^1_{HA} indicates the happiness emotion image of person 1. E indicates encoding, and $E(G_{HA})$ indicates the latent code of the encoded happiness group. The extracted latent codes for each object are compared. I^1_{EMO} indicates seven emotional latent codes for an object. Since the appearance information will be similar, the part with a large difference in the latent code is information related to expressions. This point uses ICA to determine the eigenvector that best distinguishes each independent component. The latent code corresponding to each expression can be distinguished through the ICA eigenvector. An index with a large value may be considered related to the expression. In contrast, an index with a small value is evaluated as external information irrelevant to the expression. This part can be verified qualitatively through the visual results.

Figure 7 presents the activation function for latent code editing, thresholded rectified linear unit (ReLU). The *Thresholded ReLU* [48] is designed to preserve the value when it exceeds the reference value, except when it is less than the reference value:

$$\textit{Thresholded ReLU}(x) = \begin{cases} x & x \geq h \\ 0 & x < h' \end{cases} \tag{11}$$

where h denotes the threshold value the user correctly specifies after analyzing the dataset.

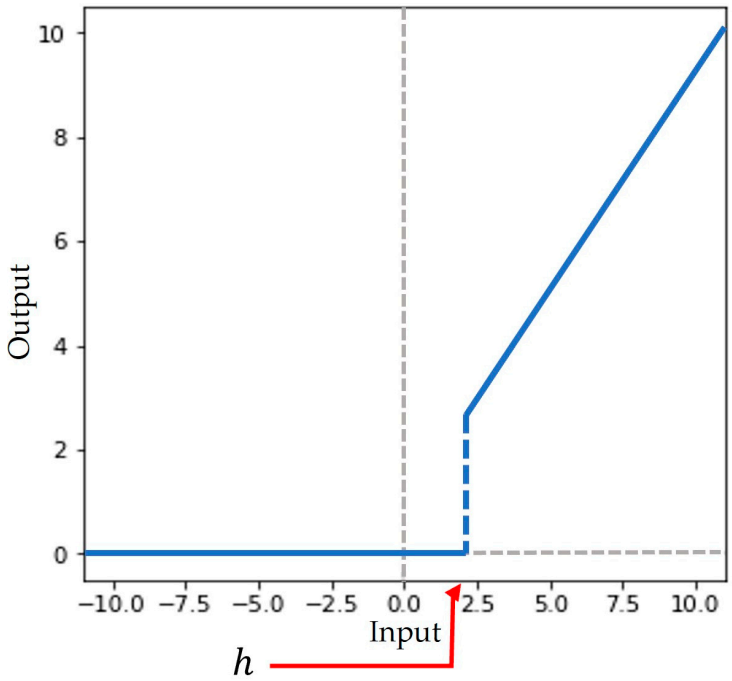

Figure 7. Transformation activation function for latent code editing with the *Thresholded ReLU*.

It is not necessary to use the entire eigenvector. Therefore, eigenvectors are filtered by the *Thresholded ReLU*. The reference value is the optimal value for each dataset, but

we used the average as the reference value. As a result, index values related to facial expressions are extracted from the latent code and used for learning.

Furthermore, as this result is important information related to facial expressions, it qualitatively proves that facial recognition performance increases when used for model training. Therefore, the emotion-relevant latent code extractor extracts expression-related information from the entire latent code and edits and manipulates it. First, it can improve expression recognition performance. Second, a novel manipulation method through ICA is suggested.

3.3. Combination Module of Existing Model and Emotion Relevant Features

Existing conventional models typically have a similar structure. Most CNN-based models are converted into high-dimensional feature representation vectors using the CNN. By integrating features from lower to higher levels, CNN-based algorithms extract spatial characteristics that represent the abstract semantics of the input image. Through a pooling layer, these extracted features lower the amount of information. Then, the layer is flattened into a single vector using a fully connected layer. In the final step, the softmax function computes the probability distribution for all classes.

However, unlike the typical method, we propose to additionally use latent code containing information that can generate images for training. Moreover, the advantage of the proposed plug-and-play framework is that it can be used regardless of any existing model. Therefore, it is possible to use the existing emotion recognition model, or another model suggested by the user in the module. The critical point is that the performance is best when using the plug-and-play framework we propose in combination rather than when only the corresponding model is used.

The number of features extracted through the existing model varies slightly from model to model, but the proposed framework combines any number of features with the modified latent code. The tensors combine the features extracted through the existing CNN-based model with the expression-related latent code extracted only from the expression-related information. In the method in Section 3.2, the tensors output one tensor through two fully connected layers. Emotions are inferred through these processes.

4. Experimental Results

4.1. Datasets and Setting

4.1.1. Portrait Emotion Recognition

The Real-World Affective Faces Database (RAF-DB) [49] contains 29,670 facial images acquired from the Internet using crowd-sourcing techniques. About 40 skilled people annotated this database with simple or complicated expressions. In this study, 12,271 images were used for training and 3068 for testing, each containing one of the seven basic facial expressions (i.e., neutral, happy, surprised, sad, angry, disgust, and fear).

The Extended Cohn-Kanade (CK+) [50] dataset contains 593 video sequences from 123 participants ranging in age from 18 to 50 years old and of various genders and nationalities. Each film presents a facial transition from neutral to a targeted peak emotion, captured at 30 frames per second with a resolution of 640 × 490 or 640 × 480 pixels. Three hundred and twenty-seven of these movies have been labeled with one of seven expression classes: Anger, contempt, disgust, fear, happiness, sorrow, and surprise. The CK+ database, which is used in the majority of facial expression classification methods, is largely recognized as the most frequently used laboratory-controlled facial expression classification database available.

The KDEF [51] is a set of 4900 images of human facial expressions. The photograph images represent 70 people with seven emotional expressions. Each expression is viewed from five viewpoints.

4.1.2. Face Image Generation

The Flickr-Faces-HQ (FFHQ) [22] only provides face images without labels for facial expressions. This database consists of 70,000 high-quality PNG images at 1024 × 1024 resolution and contains considerable variation in age, ethnicity, and image background. This dataset is not intended for facial recognition but is widely used in generative learning research.

4.1.3. Manipulation

Most datasets for FER consist of different objects. However, a dataset with the same object but different facial expressions is required to extract information related only to facial expressions in the latent code. Therefore, the KEDF dataset comprising the same object with different facial expressions is used for ICA.

4.2. Evaluation of Latent-PER on General Models

We quantitatively evaluated performance improvement through Latent-PER. As shown in Table 1, we compared the emotion recognition performance of the portrait drawing image using several state-of-the-art models and the emotion recognition performance of the proposed plug-and-play framework applied to the existing model. The performance of the proposed Latent-PER has the highest accuracy compared to most others, indicating the superiority of the framework. The function expression of the *PercentagePoint* is written as follows:

$$PercentagePoint = F - O, \tag{12}$$

where F denotes final percentage value, O denotes original percentage value. We use the percentage point as an indicator of performance improvement and the unit of *PercentagePoint* is %p. To evaluate the recognition accuracy of individual classes, we present the confusion matrices obtained with the Latent-PER framework for the RAF-DB dataset in Figure 8.

Table 1. Portrait emotion recognition performance according to the presence or absence of Latent-PER using latent codes. The dataset uses RAF-DB, CK+, and KDEF. The performance is compared through the existing state-of-the-art models, DAN, DACL, and RUL.

Dataset	Method	DAN [21]	DACL [22]	RUL [23]
RAF-DB	Without Latent-PER	74.0%	70.4%	67.7%
	With Latent-PER	84.3% (10.3%p ↑)	82.8% (12.4%p ↑)	78.2% (10.5%p ↑)
CK+	Without Latent-PER	63.7%	67.1%	42.3%
	With Latent-PER	73.4% (9.7%p ↑)	70.1% (3%p ↑)	70.3% (28%p ↑)
KDEF	Without Latent-PER	81.3%	84.6%	72.8%
	With Latent-PER	84.2% (2.9%p ↑)	85.8% (1.2%p ↑)	83.0% (10.2%p ↑)

Latent-PER improves the recognition accuracy of all classes of the three FER datasets compared to the existing methods. The detailed analysis results for each dataset for each model are as follows. First, the RAF-DB dataset for the DAN model and the Latent-PER are compared. As shown in Figure 8, the precision for happiness tends to be the highest among all emotions. When comparing the accuracy between models, an overall performance improvement occurs in all emotions. In particular, the precision for neutral and fear increased by 21%p and 19%p, respectively, exhibiting the most significant performance improvement. In addition, DAN mistakenly recognized fear as a surprise due to the enlarged mouth. Furthermore, the accuracy of disgust, anger, and sad emotions, which had poor performance, increased by 13%p, 14%p, and 17%p, respectively.

Figure 8. Performance comparison on RAF-DB dataset using confusion matrices without and with Latent-PER.

Similarly, this trend is also observed in DACL, and the precision for fear increased by 29%p with Latent-PER compared to DAN. In addition, the precision of the disgust emotion is 22%p lower in DACL compared to Latent-PER since the disgust emotion was mistakenly recognized as anger. Furthermore, the accuracy of neutral and sad, which had poor performance, increased by 15%p and 29%p, respectively. Happiness is rarely perceived as sadness, but errors concerning happiness and neutrality and regarding fear and surprise are common in existing models. Therefore, most of the recognition performance for facial expressions with ambiguous elements between facial expressions is lowered. In the case of RUL, there was the most significant difference between the case where Latent-PER was used and the case where it was not. In particular, in the previous two models, happiness was recognized well, whereas, in RUL, the emotion recognition performance of fear was not good. Moreover, the misjudgment rate for sadness and fear was high, and it can be seen that this part was significantly improved when Latent-PER was applied. In the case of fear, the precision of RUL is only 18%. In contrast, the precision of RUL with Latent-PER shows 38% performance. The precision of RUL with Latent-PER shows performance improvements of 13%p and 20%p in disgust and sad emotions, respectively. Furthermore, the accuracy of neutral, fear, anger, and surprise which had poor performance, increased by 16%p, 20%p, 13%p, and 10%p, respectively. Since Latent-PER learns through independent components between features, disentangled features learned from entangled features space can be used. Therefore, it performs better on ambiguous expression elements, as shown in Figures 9 and 10. As shown in Figure 9, the precisions of fear and sad emotions are very low in the case of DAN. Fear and sad emotions are often misjudged as a surprise. In contrast, DAN with Latent-PER presents 44%p and 22%p higher percentage points than basic DAN in fear and sad emotions, respectively. DACL with Latent-PER shows 25%p and 9%p higher percentage points than the base DACL in fear and sad emotions, respectively. In addition, RUL provides lower precision of all emotion on CK+ dataset. However, RUL with Latent-PER presents about 46%p and 47%p higher percentage points than basic RUL in fear and sad emotions, respectively. However, failure examples are observed for neutral emotion in the CK+ dataset, as shown in the results of most models with Latent-PER. The CK+ dataset has various emotional intensities. When the happy emotion with very weak intensity is included in the dataset, neutral can be recognized as happy. To solve this problem, if data preprocessing is performed on these weak emotions, the emotion recognition accuracy for neutral can be improved. Overall, our framework shows performance gains of 9.7%p, 3%p, and 28%p, respectively, for DAN, DACL, and RUL models using CK+ dataset.

4.3. Ablations Study

4.3.1. Latent Code Editing

In this paper, the parts related to facial expressions are extracted from the entire latent code using the ICA method. Seven emotional images are converted for one object into latent code using the KDEF dataset. The ICA algorithm extracts an eigenvector that can classify the latent code for each emotion. As depicted in Figure 11, a single latent code is created by mixing the latent code for all emotions through the mixing matrix of the source latent code. From this, one latent code is classified into seven original latent codes. If the value of the eigenvector is large, there is a significant correlation between the index and the expression, and when it is small, the correlation with the expression is also small. As illustrated in Figures 6 and 12, we edited the latent code based on certain thresholds to avoid using index information that is irrelevant to the expression for training.

Figure 9. Performance comparison on CK+ dataset using confusion matrices without and with Latent-PER.

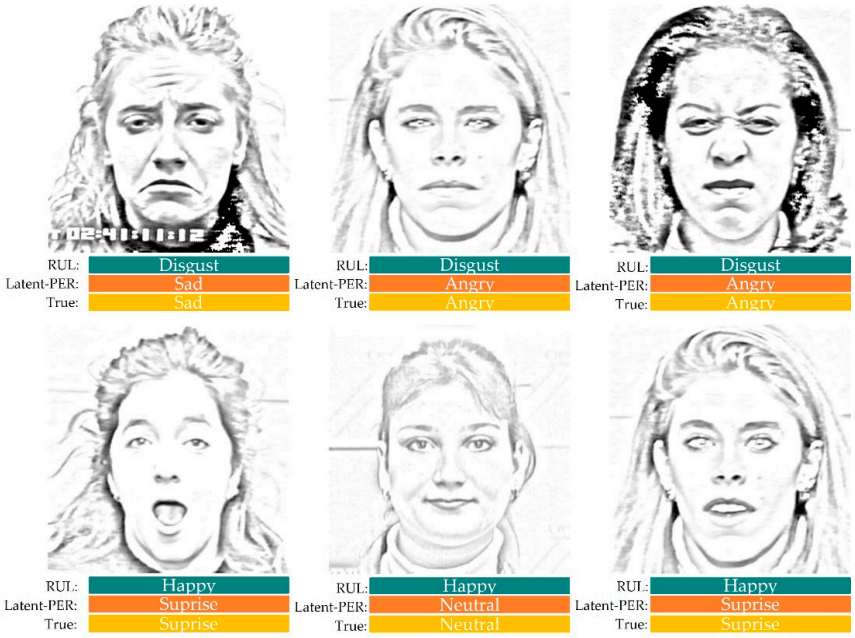

Figure 10. Visual prediction results of examples on CK+ dataset without and with Latent-PER.

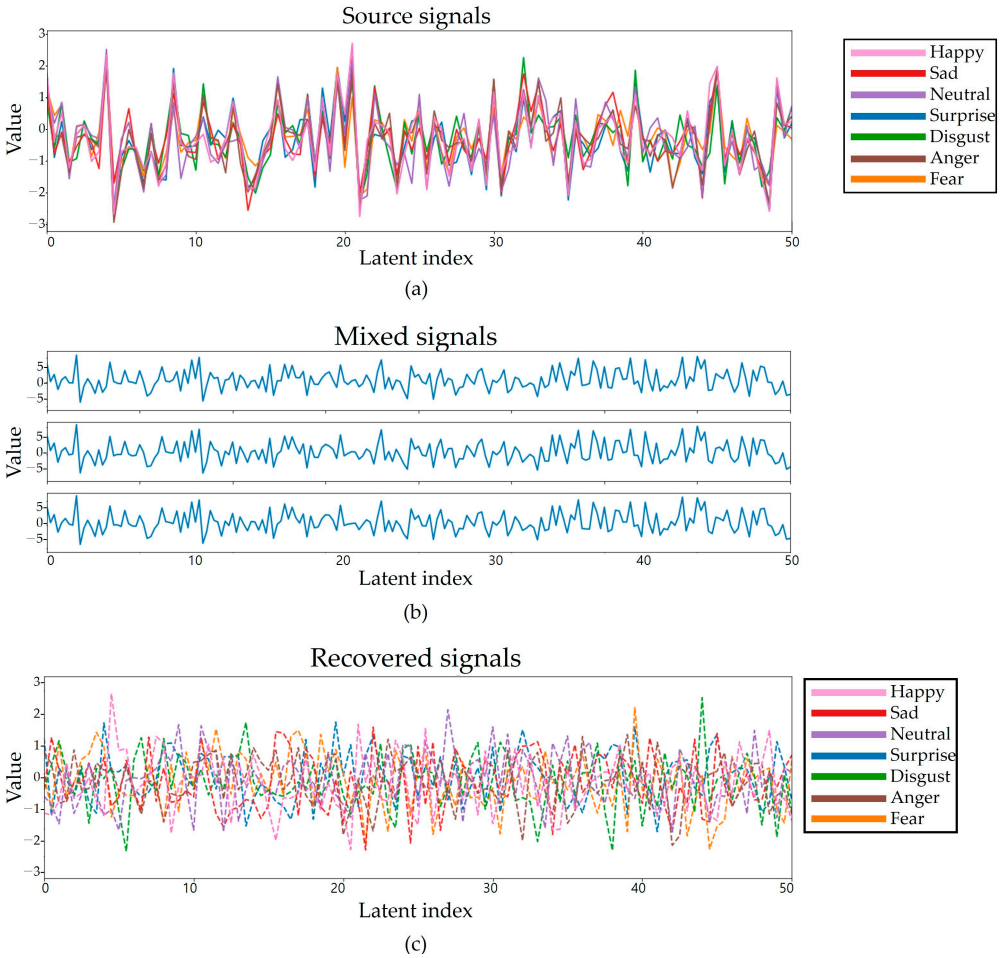

Figure 11. Example of an ICA solution that restores independent components. (**a**) Source information entered as input. It is mixed as in (**b**) through the mixing matrix and divided into independent signals as in (**c**) through ICA.

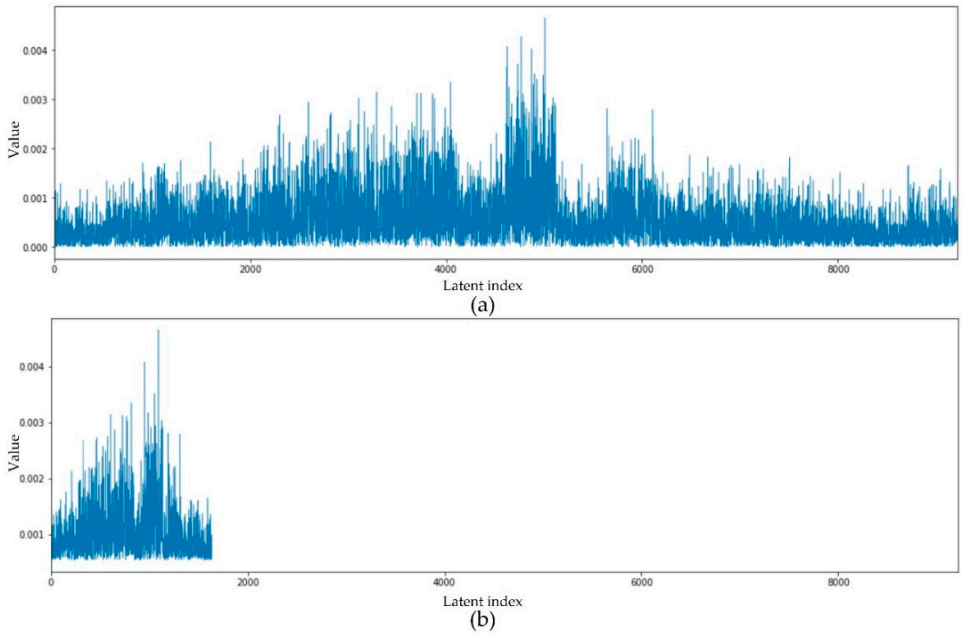

Figure 12. Visualization of the rectified eigenvector excluding information unrelated to the expression: (**a**) Entire eigenvector and (**b**) rectified eigenvector.

We quantitatively and qualitatively prove the correlation between these eigenvectors and expressions. For the quantitative proof, as shown in Tables 2–4, the editing hidden code for learning provides higher accuracy than the emotion recognition performance when the entire latent code is used. In particular, when the mean is filtered by the threshold, only about 17% of the information is used. However, the accuracy is higher than using the entire latent code.

Table 2. Portrait emotion recognition performance of latent code editing on the RAF-DB validation set in terms of accuracy.

Threshold (*h*)	Number of Tensors	DAN	RUL	DACL
0	Full latent code (9126 tensor)	82.7%	84.7%	78.2%
Median	Edited latent code (2194 tensor)	83.5% (0.8%p ↑)	85.2% (0.5%p ↑)	80.8% (2.6%p ↑)
Mean	Edited latent code (1638 tensor)	84.2% (1.5%p ↑)	85.8% (1.1%p ↑)	83.0% (4.8%p ↑)

Table 3. Portrait emotion recognition performance of latent code editing on the CK+ validation set in terms of accuracy.

Threshold (*h*)	Number of Tensors	DAN	RUL	DACL
0	Full latent code (9126 tensor)	72.1%	63.0%	68.0%
Median	Edited latent code (2194 tensor)	72.6% (0.5%p ↑)	68.2% (5.2%p ↑)	69.2% (1.2%p ↑)
Mean	Edited latent code (1638 tensor)	73.4% (1.3%p ↑)	70.3% (7.3%p ↑)	70.1% (2.1%p ↑)

Table 4. Portrait emotion recognition performance of latent code editing on the KDEF validation set in terms of accuracy.

Threshold (*h*)	Number of Tensors	DAN	RUL	DACL
0	Full latent code (9126 tensor)	83.2%	77.0%	81.2%
Median	Edited latent code (2194 tensor)	84.0% (0.8%p ↑)	78.2% (1.2%p ↑)	82.0% (0.8%p ↑)
Mean	Edited latent code (1638 tensor)	84.3% (1.1%p ↑)	78.4% (1.4%p ↑)	82.8% (1.6%p ↑)

For qualitative proof, the original latent code is changed to the greatest extent possible as the rectified eigenvector obtained through the above method, and an image is generated through a generator. If the information in the latent code is unrelated to the expression, such as hair length and skin color, an image with a changed appearance is created. However, if it is related to the expression, only the expression changes are disentangled while maintaining the existing appearance. For this proof, the degree of emotion was controlled by calculating the eigenvector in the latent code for a specific emotion. Figure 13 confirms that the proposed method changes the facial expression while disentangling the expression data and maintaining the appearance information. Our proposed manipulation method can be applied to various image styles, such as painting, drawing, and cartoon. The task of facial expression recognition using the proposed latent code and facial expression manipulation via eigenvectors can help in automating animation emotion caption generation tasks and in supporting existing expensive animation production systems.

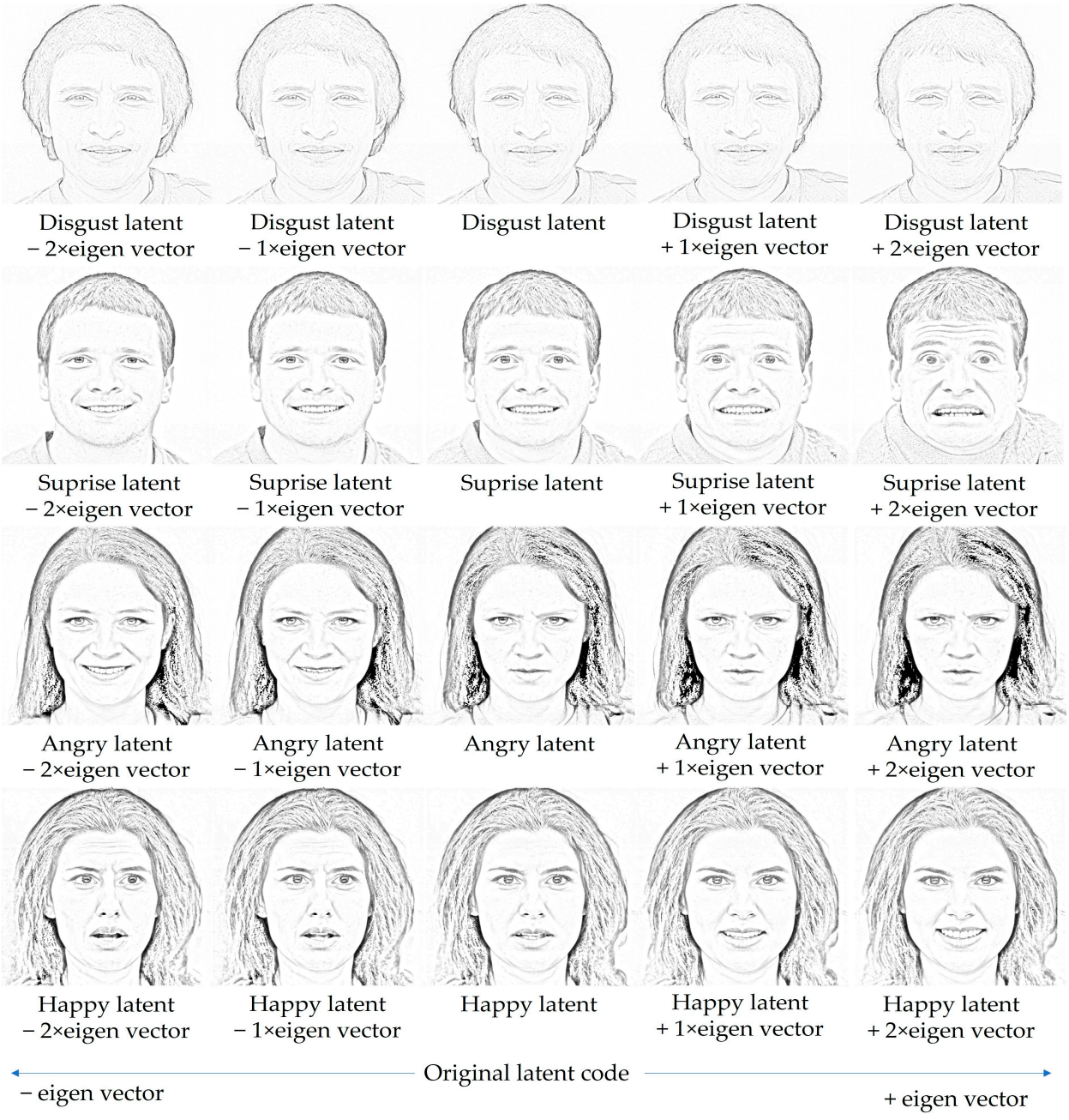

Figure 13. Emotional intensity manipulation based on latent code change.

4.3.2. Complexity

The model weight sizes for DAN, DACL, and RUL are approximately 150, 400, and 380 MB, respectively. Our proposed Latent-PER model is a plug-and-play framework for the existing facial expression recognition model. Therefore, as shown in Table 5, it requires more computing resources as additional encoders. For this reason, additional inference time is required the average 6 ms than the existing model.

Table 5. Comparison of computational complexity of various methods.

Method	Inference Time (ms)	Model Weight Size (MB)
DAN	6	150
DAN + Latent-PER	12	270
DACL	43	400
DACL + Latent-PER	49	520
RUL	42	380
RUL + Latent-PER	48	500

4.3.3. Limitation

Our method can have limitations in terms of computational complexity. Since it is a GAN-based method, the inference time is larger than a typical CNN-based method. Therefore, in the future work, additional network compression is needed to reduce the inference time to reduce the weights of the GAN-based method. Additionally, since a GAN-based encoder is used, it must be aligned to the input image to show the complete estimation performance. If face-occluded images are inputted into facial expression recognition models, the model suffers severe performance degradation. To address this problem, a GAN-based image recovery method can be applied. Moreover, if we develop this research, it becomes possible to recognize emotions in all existing image styles from the viewpoint of emotion recognition. Furthermore, from the viewpoint of manipulation, many computer vision researchers are currently interested in generative models. If manipulation through the proposed ICA is further developed, an actor's acting can be assisted through manipulation in making a movie, and for animation, the existing production work can be simplified.

5. Conclusions

Applying conventional emotion recognition models to portrait images presents unacceptable accuracy. In this case, it is significantly inaccurate to recognize the emotion of a style face different from the actual image, such as a famous drawing, painting, or animation. Emotion recognition in portraits is clearly necessary, but most of the papers focus on authentic-photograph images. Therefore, we propose a specialized model.

We propose a plug-and-play-style portrait emotion recognition framework to solve this problem by adding the existing model to the proposed method that transforms the input image into latent code via a GAN-based encoder. This approach is a first in the FER. In this latent code, only information related to portrait emotion is edited in the ICA method, and only the extracted information is used for training along with the existing model. The manipulation method we proposed is the first approach that applied ICA to latent code editing. We observe considerable performance improvement with the proposed method. Moreover, we quantitatively and qualitatively prove that we extracted only expression-related information in the latent code. Through this, we propose a novel method to manipulate visually independent components from the generative learning of the GAN.

Author Contributions: Conceptualization, S.B.Y.; methodology, I.L. and S.B.Y.; software, I.L.; validation, I.L.; formal analysis, I.L. and S.B.Y.; investigation, I.L. and S.B.Y.; resources, I.L. and S.B.Y.; data curation, I.L. and S.B.Y.; writing—original draft preparation, I.L. and S.B.Y.; writing—review and editing, I.L. and S.B.Y.; visualization, S.B.Y.; supervision, S.B.Y.; project administration, S.B.Y.; funding acquisition, S.B.Y. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (NRF-2020R1A4A1019191) and the Industrial Fundamental Technology Development Program (No. 20018699) funded by the Ministry of Trade, Industry & Energy (MOTIE) of Korea.

Data Availability Statement: Not applicable.

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

References

1. Tian, Y.; Kanade, T.; Cohn, J.F. Facial Expression Recognition. In *Handbook of Face Recognition*; Springer London: London, UK, 2011; pp. 487–519.
2. Shan, C.; Gong, S.; McOwan, P.W. Facial Expression Recognition Based on Local Binary Patterns: A Comprehensive Study. *Image Vis. Comput.* **2009**, *27*, 803–816. [CrossRef]
3. Zhao, G.; Pietikainen, M. Dynamic Texture Recognition Using Local Binary Patterns with an Application to Facial Expressions. *IEEE Trans. Pattern Anal. Mach. Intell.* **2007**, *29*, 915–928. [CrossRef] [PubMed]
4. Zhi, R.; Flierl, M.; Ruan, Q.; Kleijn, W.B. Graph-Preserving Sparse Nonnegative Matrix Factorization with Application to Facial Expression Recognition. *IEEE Trans. Syst. Man Cybern. Part B* **2011**, *41*, 38–52. [CrossRef]
5. Zhong, L.; Liu, Q.; Yang, P.; Liu, B.; Huang, J.; Metaxas, D.N. Learning Active Facial Patches for Expression Analysis. In Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition, Providence, RI, USA, 16–21 June 2012; pp. 2562–2569.
6. Szegedy, C.; Toshev, A.; Erhan, D. Deep Neural Networks for Object Detection. *Adv. Neural Inf. Process. Syst.* **2013**, *26*, 2553–2561.
7. LeCun, Y.; Boser, B.; Denker, J.S.; Henderson, D.; Howard, R.E.; Hubbard, W.; Jackel, L.D. Backpropagation Applied to Handwritten Zip Code Recognition. *Neural Comput.* **1989**, *1*, 541–551. [CrossRef]
8. Simonyan, K.; Zisserman, A. Very Deep Convolutional Networks for Large-Scale Image Recognition. In Proceedings of the 3rd International Conference on Learning Representations, Guilin, China, 19–20 May 2018.
9. Tang, Y. Deep Learning Using Linear Support Vector Machines. *arXiv* **2013**, arXiv:1306.0239.
10. Hong, Y.; Lee, S.; Yoo, S.B. AugMoCrack: Augmented Morphological Attention Network for Weakly Supervised Crack Detection. *Electron. Lett.* **2022**, *58*, 651–653. [CrossRef]
11. Lee, S.-J.; Yun, J.-S.; Lee, E.J.; Yoo, S.B. HIFA-LPR: High-Frequency Augmented License Plate Recognition in Low-Quality Legacy Conditions via Gradual End-to-End Learning. *Mathematics* **2022**, *10*, 1569. [CrossRef]
12. Yun, J.-S.; Yoo, S.-B. Single Image Super-Resolution with Arbitrary Magnification Based on High-Frequency Attention Network. *Mathematics* **2022**, *10*, 275. [CrossRef]
13. Lee, S.; Yun, J.S.; Yoo, S.B. Alternative Collaborative Learning for Character Recognition in Low-Resolution Images. *IEEE Access* **2022**, *10*, 22003–22017. [CrossRef]
14. Lee, S.-J.; Yoo, S.B. Super-Resolved Recognition of License Plate Characters. *Mathematics* **2021**, *9*, 2494. [CrossRef]
15. Yun, J.-S.; Na, Y.; Kim, H.H.; Kim, H.-I.; Yoo, S.B. HAZE-Net: High-Frequency Attentive Super-Resolved Gaze Estimation in Low-Resolution Face Images. *arXiv* **2022**, arXiv:2209.10167.
16. Hong, Y.; Yoo, S.B. OASIS-Net: Morphological Attention Ensemble Learning for Surface Defect Detection. *Mathematics* **2022**, *10*, 4114. [CrossRef]
17. Yun, J.-S.; Yoo, S.B. Infusion-Net: Inter- and Intra-Weighted Cross-Fusion Network for Multispectral Object Detection. *Mathematics* **2022**, *10*, 3966. [CrossRef]
18. Hyvärinen, A.; Jarmo, H.; Patrik, O. *Hoyer Independent Component Analysis. Natural Image Statistics*; Springer: London, UK, 2009; Volume 529.
19. Goodfellow, I.; Pouget, A.J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; Bengio, Y. Generative Adversarial Networks. *Commun. ACM* **2020**, *63*, 139–144. [CrossRef]
20. Mase, K. Recognition of Facial Expression from Optical Flow. *IEICE Trans. Inf. Syst.* **1991**, *74*, 3474–3483.
21. Xiang, J.; Zhu, G. Joint Face Detection and Facial Expression Recognition with MTCNN. In Proceedings of the 2017 4th International Conference on Information Science and Control Engineering, Changsha, China, 21–23 July 2017; pp. 424–427.
22. Karras, T.; Laine, S.; Aila, T. A Style-Based Generator Architecture for Generative Adversarial Networks. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 4401–4410.
23. King, D.E. Dlib-Ml: A Machine Learning Toolkit. *J. Mach. Learn. Res.* **2009**, *10*, 1755–1758.
24. Fasel, B. Robust Face Analysis Using Convolutional Neural Networks. In Proceedings of the Object Recognition Supported by User Interaction for Service Robots, Quebec City, QC, Canada, 11–15 August 2002; pp. 40–43.

25. Kanou, S.E.; Ferrari, R.C.; Mirza, M.; Jean, S.; Carrier, P.-L.; Dauphin, Y.; Boulanger-Lewandowski, N.; Aggarwal, A.; Zumer, J.; Lamblin, P. Combining Modality Specific Deep Neural Networks for Emotion Recognition in Video. In Proceedings of the 15th ACM on International Conference on Multimodal Interaction, Sydney, Australia, 9–13 December 2013; pp. 543–550.
26. Ge, H.; Zhu, Z.; Dai, Y.; Wang, B.; Wu, X. Facial Expression Recognition Based on Deep Learning. *Comput. Methods Programs Biomed.* **2022**, *215*, 106621. [CrossRef]
27. Meng, Z.; Liu, P.; Cai, J.; Han, S.; Tong, Y. Identity-Aware Convolutional Neural Network for Facial Expression Recognition. In Proceedings of the 2017 12th IEEE International Conference on Automatic Face & Gesture Recognition, Washington, DC, USA, 30 May–3 June 2017; pp. 558–565.
28. Li, Z.; Wu, S.; Xiao, G. Facial Expression Recognition by Multi-Scale CNN with Regularized Center Loss. In Proceedings of the 2018 24th International Conference on Pattern Recognition, Beijing, China, 20–24 August 2018; pp. 3384–3389.
29. Wen, Z.; Lin, W.; Wang, T.; Xu, G. Distract Your Attention: Multi-Head Cross Attention Network for Facial Expression Recognition. *arXiv* **2021**, arXiv:2109.07270.
30. Farzaneh, A.H.; Qi, X. Facial Expression Recognition in the Wild via Deep Attentive Center Loss. In Proceedings of the 2021 IEEE Winter Conference on Applications of Computer Vision, Waikoloa, HI, USA, 3–8 January 2021; pp. 2401–2410.
31. Zhang, Y.; Wang, C.; Deng, W. Relative Uncertainty Learning for Facial Expression Recognition. *Adv. Neural Inf. Process. Syst.* **2021**, *34*, 17616–17627.
32. Xu, P.; Peng, S.; Luo, Y.; Gong, G. Facial Expression Recognition: A Meta-Analytic Review of Theoretical Models and Neuroimaging Evidence. *Neurosci. Biobehav. Rev.* **2021**, *127*, 820–836. [CrossRef]
33. Ruan, D.; Yan, Y.; Lai, S.; Chai, Z.; Shen, C.; Wang, H. Feature Decomposition and Reconstruction Learning for Effective Facial Expression Recognition. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 7660–7669.
34. Li, B.; Lima, D. Facial Expression Recognition via ResNet-50. *Int. J. Cogn. Comput. Eng.* **2021**, *2*, 57–64. [CrossRef]
35. Zhao, Z.; Liu, Q.; Zhou, F. Robust Lightweight Facial Expression Recognition Network with Label Distribution Training. *Proc. AAAI Conf. Artif. Intell.* **2021**, *35*, 3510–3519. [CrossRef]
36. Pham, L.; Vu, T.H.; Tran, T.A. Facial Expression Recognition Using Residual Masking Network. In Proceedings of the 2020 25th International Conference on Pattern Recognition, Milan, Italy, 10–15 January 2021; pp. 4513–4519.
37. Liu, C.; Hirota, K.; Ma, J.; Jia, Z.; Dai, Y. Facial Expression Recognition Using Hybrid Features of Pixel and Geometry. *IEEE Access* **2021**, *9*, 18876–18889. [CrossRef]
38. Park, S.-J.; Kim, B.-G.; Chilamkurti, N. A Robust Facial Expression Recognition Algorithm Based on Multi-Rate Feature Fusion Scheme. *Sensors* **2021**, *21*, 6954. [CrossRef] [PubMed]
39. Moetesum, M.; Aslam, T.; Saeed, H.; Siddiqi, I.; Masroor, U. Sketch-Based Facial Expression Recognition for Human Figure Drawing Psychological Test. In Proceedings of the 2017 International Conference on Frontiers of Information Technology, Islamabad, Pakistan, 18–20 December 2017; pp. 258–263.
40. Lee, I.; Yun, J.-S.; Kim, H.H.; Na, Y.; Yoo, S.B. LatentGaze: Cross-Domain Gaze Estimation through Gaze-Aware Analytic Latent Code Manipulation. *arXiv* **2022**, arXiv:2209.10171.
41. Mirza, M.; Osindero, S. Conditional Generative Adversarial Nets. *arXiv* **2014**, arXiv:1411.1784.
42. Chrysos, G.G.; Kossaifi, J.; Zafeiriou, S. Robust Conditional Generative Adversarial Networks. International Conference on Learning Representations. *arXiv* **2018**, arXiv:1805.08657.
43. Abdal, R.; Qin, Y.; Wonka, P. Image2StyleGAN: How to Embed Images into the StyleGAN Latent Space? In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 4432–4441.
44. Shen, Y.; Gu, J.; Tang, X.; Zhou, B. Interpreting the Latent Space of GANs for Semantic Face Editing. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 9243–9252.
45. Zhu, J.; Shen, Y.; Zhao, D.; Zhou, B. In-Domain GAN Inversion for Real Image Editing. In Proceedings of the European Conference on Computer Vision (ECCV), Glasgow, UK, 23–28 August 2020; pp. 592–608.
46. Tov, O.; Alaluf, Y.; Nitzan, Y.; Patashnik, O.; Cohen-Or, D. Designing an Encoder for StyleGAN Image Manipulation. *ACM Trans. Graph.* **2021**, *40*, 1–14. [CrossRef]
47. He, Z.; Spurr, A.; Zhang, X.; Hilliges, O. Photo-Realistic Monocular Gaze Redirection Using Generative Adversarial Networks. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 6932–6941.
48. Konda, K.; Memisevic, R.; Krueger, D. Zero-Bias Autoencoders and the Benefits of Co-Adapting Features. *arXiv* **2014**, arXiv:1402.3337.
49. Li, S.; Deng, W.; Du, J. Reliable Crowdsourcing and Deep Locality-Preserving Learning for Expression Recognition in the Wild. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 2584–2593.
50. Lucey, P.; Cohn, J.F.; Kanade, T.; Saragih, J.; Ambadar, Z.; Matthews, I. The Extended Cohn-Kanade Dataset (CK+): A Complete Dataset for Action Unit and Emotion-Specified Expression. In Proceedings of the 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition—Workshops, San Francisco, CA, USA, 13–18 June 2010; pp. 94–101.
51. Calvo, M.G.; Lundqvist, D. Facial Expressions of Emotion (KDEF): Identification under Different Display-Duration Conditions. *Behav. Res. Methods* **2008**, *40*, 109–115. [CrossRef] [PubMed]