applied sciences

MDPI

Article

Change and Detection of Emotions Expressed on People's Faces in Photos

Zbigniew Piotrowski *, Maciej Kaczyński and Tomasz Walczyna

Faculty of Electronics, Military University of Technology, 00-908 Warsaw, Poland; maciej.kaczynski@wat.edu.pl (M.K.); tomasz.walczyna@wat.edu.pl (T.W.)
* Correspondence: zbigniew.piotrowski@wat.edu.pl

Abstract: Human emotions are an element of attention in various areas of interest such as psychology, marketing, medicine, and public safety. Correctly detecting human emotions is a complex matter. The more complex and visually similar emotions are, the more difficult they become to distinguish. Making visual modifications to the faces of people in photos in a way that changes the perceived emotion while preserving the characteristic features of the original face is one of the areas of research in deepfake technologies. The aim of this article is to showcase the outcomes of computer simulation experiments that utilize artificial intelligence algorithms to change the emotions on people's faces. In order to detect and change emotions, deep neural networks discussed further in this article were used.

Keywords: detection; emotion; face; facial expression; human; people; classification; deepfake; change; neural network

check for updates

Citation: Piotrowski, Z.; Kaczyński, M.; Walczyna, T. Change and Detection of Emotions Expressed on People's Faces in Photos. *Appl. Sci.* **2024**, *14*, 10681. https://doi.org/10.3390/app142210681

Academic Editor: Thomas Lindner

Received: 22 October 2024
Revised: 7 November 2024
Accepted: 16 November 2024
Published: 19 November 2024

1. Introduction

In the era of dynamic development of digital technologies and artificial intelligence, the analysis of emotions expressed on people's faces is becoming increasingly important. Emotions are a key element of interpersonal communication, and their detection and interpretation are widely used in various fields, such as psychology, marketing, medicine, and public safety. Understanding, simulating, and properly detecting emotions can lead to innovative solutions that enable more effective interaction between people and computer systems. The aim of this article is to present the results of computer simulation experiments on changing emotions expressed on people's faces using artificial intelligence algorithms. In this study, we introduce a novel approach to training deep neural networks (DNNs) for detecting and modifying emotions in images, focusing on enhanced realism and classification accuracy. The method presented in this manuscript departs from traditional models by omitting constraints like cycle-consistency loss, commonly seen in algorithms such as StarGAN, which can overly restrict feature transformation. Instead, we utilize a tailored reconstruction loss applied only when target and ground-truth emotions align, allowing for more natural emotion modification while controlling for some feature reduction. Leveraging the AffectNet dataset, typically used for emotion recognition, our study further investigates its suitability for emotion modification tasks, thus expanding its applications. Additionally, the presented model incorporates multiple classifiers trained in parallel to improve emotion classification by comparing generated data to real-world benchmarks, establishing a robust framework for future emotion analysis studies.

This article discusses the methodology of conducting the simulation, starting from collecting and preparing data, through selecting and training the models, to evaluating the results and analyzing errors. The research results are also presented, including examples of modified photos that illustrate the effectiveness of the applied algorithms.

This paper is organized as follows. Section 2 presents a literature review relating to the discussed issue. Section 3 presents the methodology of the conducted research. Section 4 presents the results and compares them with other state-of-the-art methods.

Section 5 concludes the paper. Appendix A shows a preview of sample generated images for individual emotions.

2. Related Work

In recent years, more and more information about deepfake technologies and their individual issues has appeared in the form of publications. One of the aspects of deepfake technology is the proper reproduction of human emotions in a way that ensures the integrity of the modifications made with the whole image. The changes should reflect the new emotion while maintaining the identity of the person (the ability to recognize the person without visible and disturbing graphic artifacts). This section covers human emotion profile classification and the artificial intelligence algorithms used to analyze and change expressed emotions.

2.1. Human Emotion Profile Classification

Classifying facial emotions is a complex task that involves identifying and distinguishing subtle emotional signals. Paul Ekman and Wallace V. Friesen [1] in their classic studies identified six basic emotions: joy, sadness, fear, anger, surprise, and disgust. These emotions are widely recognized and form the foundation of many facial expression analysis systems. However, modern research goes further, trying to capture a wider range of emotions and their mixtures. Emotion recognition now includes more complex emotional states, such as guilt, jealousy, embarrassment, and pride. These technologies are based on more advanced algorithms that analyze not only static images but also dynamic changes in facial expressions. In order to present the relationships between individual emotions, an emotion wheel was proposed (Figure 1) [2]. It is a theoretical model that presents eight basic emotions and their different intensities and combinations. The wheel is often used to understand and classify emotions in a psychological and emotional context.

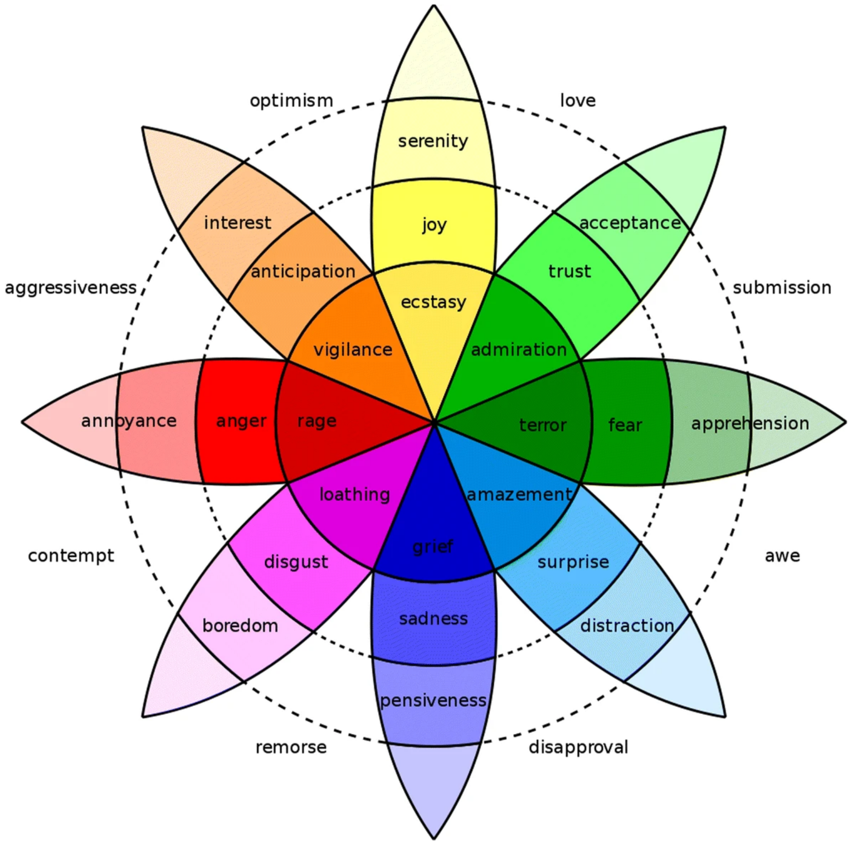


Figure 1. Wheel of emotion [3].

Another representation of emotion is the circumplex theory of affect presented by David Watson and Auke Tellegen (Figure 2) [4]. The circumplex theory of affect includes four dimensions corresponding to the following affects: negative, positive, engaged, and pleasant. Each has two directions: high and low.

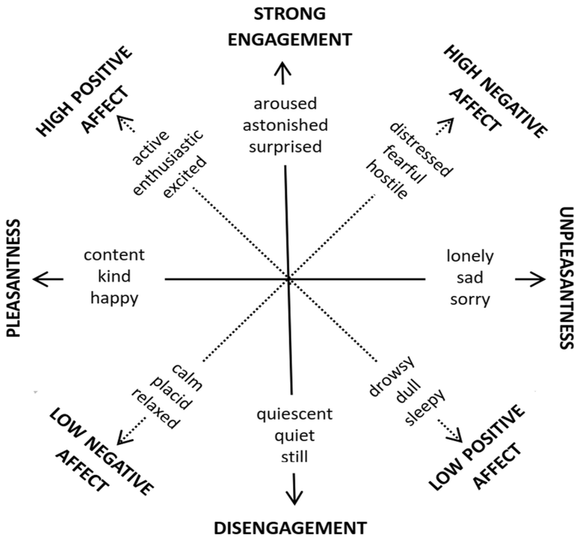


Figure 2. Circumplex theory of affect [3].

2.2. *Artificial Intelligence Algorithms Used to Analyze Emotions*

The process of training artificial neural networks to classify emotions is challenging due to several important factors. Emotions are subjective and context-dependent, making generalization difficult. The same facial expression can be interpreted as different emotions by different people. Emotion datasets are unbalanced (overrepresentation of training patterns for some emotions and underrepresentation of training patterns for others). Datasets often differ in terms of emotion categories, making it difficult to compare studies. Creating databases about people's emotions is an additionally complex process due to ethical and legal issues. Moreover, contextual factors such as lighting and background noise can affect emotion recognition.

In recent years, there has been rapid development in the field of artificial intelligence, especially in the context of image analysis. Deep learning techniques, including convolutional neural networks (CNNs) and generative adversarial networks (GANs), have become a major tool in this field.

Convolutional neural networks (CNNs) are widely used for image classification and pattern recognition [5–8]. The research of Zhang et al. [9] showed that multi-layer CNNs can effectively classify emotions based on features extracted from facial images. The introduction of multi-task networks allowed for simultaneous recognition of emotions and their intensity, which increases the classification accuracy. While CNNs are highly effective at accurately identifying basic emotions, detecting subtle emotions remains a difficult task. This requires advanced network architectures, high-quality datasets, and an appropriate training process to enhance their performance.

Generative adversarial networks (GANs) have revolutionized the approach of changing emotions in images. A GAN consists of two networks: a generator and a discriminator, which cooperate to create realistic images. The work of Karras et al. [10] introduced new methods for generating high-quality facial images, enabling more realistic manipulation of emotions.

Manipulating facial emotions in images has become more effective thanks to advances in generative neural networks. In 2021, Ning et al. [11] presented the FEGAN method, which allows precise modification of the expressed emotions while preserving the identity of the person. This algorithm uses advanced style transfer techniques to modify the specific facial features responsible for expressing emotions. FEGAN has region-specific editing capability enabling nuanced modifications that can convey subtle emotional changes rather than merely altering broad facial features. Many GAN-based models also allow for emotion modification, but they generally modify the entire face in a single process. While these models can generate high-quality images, they often face challenges in preserving realistic textures when making intense emotional changes. Furthermore, they may occasionally introduce artifacts when aiming for subtle adjustments, such as a slight smile or a small shift in eyebrow positioning.

Another important achievement is the research of Zhu et al. [12], which introduced a system capable of learning to translate an image from a source domain to a target domain in the absence of paired examples. This system is particularly useful in creating training datasets.

In recent years, there have been numerous innovations in the field of detecting [13–25] and changing [26–32] emotions in images.

In the field of emotion detection, Wang et al. proposed The Region Attention Network to addresses pose and occlusion challenges in facial expression recognition (FER) by applying attention to specific facial regions [22]. Vo et al. introduced a pyramid architecture with super-resolution for FER [20]. Zhang et al. introduced a dual-direction attention mixed feature network aimed at improving FER. The model uses dual-directional attention mechanisms to capture both local and global features in facial images, enhancing the accuracy of emotion classification [13]. Ning et al. proposed an approach combining representation learning with identity adversarial training to better understand facial behaviors. The technique aims to disentangle identity-related features from expressive ones, improving FER by focusing on emotion-specific features [14].

In the area of facial attribute change, in general, there are two kinds of the methods: target-based and source-based methods. Source-based methods focus on adapting and changing existing materials while target-based methods allow editing in the creator's environment, allowing to influence the modeling process directly.

In the field of emotion changing, Chen et al. proposed a face-swapping framework that prioritizes both realism and identity preservation. The approach enables high-quality swaps by maintaining facial details while matching the target's head pose and expressions [28]. Le et al. introduced a model that accounts for occlusions, such as hands or hair covering parts of the face. The model enhances realism by generating convincing occlusion-aware face swaps [29]. Wang et al. presented HifiFace, which employed 3D shape and semantic priors to guide face-swapping, aiming for a high-fidelity output. This model enhances realism by using 3D facial structure information, which helps retain identity details and ensures that swaps appear highly realistic, even in complex poses or expressions [32].

It is worth noting that along with the emergence of deepfake technology, methods for detecting its use have also been developed [33–36].

3. Methodology

The research was conducted on a Linux operating system (Ubuntu 24.04) using the PyTorch version 2.3 and TensorFlow version 2.16.1 machine learning libraries. The source codes were written in Python version 3.12.3. The AffectNet [23] dataset was used to train the DNNs. In this section, the concept of the proposed method as well as the architecture of DNNs will be presented.

3.1. Concept of the Proposed Method

In order to conduct the research, four DNNs were trained. The first one changed emotions in photos of people, the second one constituted a discriminator used in the learning process, and the other two constituted externally trained separate classifiers. In the further part of this article, a DNN that changes the emotions of people in photos is referred to as EmoDNN to distinguish it from the discriminator and classifiers, which are also DNNs. EmoDNN, the discriminator, and one of the two classifiers were trained using the PyTorch machine learning library. An additional validation discriminator was trained using the TensorFlow machine learning library.

At the beginning, the process of training EmoDNN alongside with the discriminator will be discussed, and then the classifier will be discussed later in this section. The input of EmoDNN is fed with an image that undergoes the modification process to one of eight target emotions, which are also fed to the input of the network in the form of a vector, where the target emotion is represented by value 1, the remaining by value 0. The modified image constituting the result of EmoDNN processing is fed to the input of the discriminator. At its output, the discriminator returns a value whether the given image is original or modified.

3.2. Learning Process

The classifier used in the generation process was trained using the PyTorch library. Its goal is to correctly classify emotions in images, which is essential for validating and assessing the quality of generated images. The classifier's objective function is based on cross-entropy loss, which is commonly used in classification problems.

Class weights were calculated based on the frequency of occurrence of individual emotions in the dataset. These weights are inversely proportional to the number of occurrences of each emotion, which helps balance the impact of less frequent emotions on the learning process. The cross-entropy loss for the classifier is given by

$$\mathcal{L}_{\text{cls}} = -\textstyle\sum_{i=1}^{N} w_i \cdot y_i \cdot \log(\hat{y}_i) \tag{1}$$

where $\mathcal{L}_{\text{cls}}$ is the classifier loss; N is the number of classes; w_i is the weight for class i; y_i is the actual label for class i (value 1 for the true class, 0 for others); $\hat{y}_i$ is the predicted probability for class i.

In the training process of the EmoDNN model, a discriminator was used to distinguish original images from modified ones. The discriminator was trained using a cost function based on hinge loss [37]. The hinge loss for the discriminator consists of two main components: the loss for real images and the loss for generated images. Additionally, a gradient penalty [38] is included to improve training stability.

The discriminator analyzes real images x_{real} and its output out_{real} should be as high as possible to classify them as real. The hinge loss for real images is calculated as

$$\mathcal{L}_{\text{real}} = \frac{1}{m}\sum_{i=1}^{m} \max\left(0.1 - \text{out}_{\text{real}}^{(i)}\right) \tag{2}$$

where m is the number of samples.

The generated images x_{fake} are created by the generator based on real images x_{real} and desired emotion labels $x_{\text{fake\_cls}}$. The discriminator analyzes the generated images x_{fake} and its output out_{fake} should be as low as possible to classify them as fake. The hinge loss for generated images is calculated as

$$\mathcal{L}_{\text{fake}} = \frac{1}{m}\sum_{i=1}^{m} \max\left(0.1 + \text{out}_{\text{fake}}^{(i)}\right) \tag{3}$$

To improve training stability and ensure that gradients are well-conditioned, an additional component called gradient penalty (GP) is used. GP is calculated as the norm of the discriminator's gradient with respect to interpolated samples between real and generated images. The gradient penalty is given by

$$\text{GP} = (\| \nabla_{\hat{x}} D(\hat{x}) \|_2 - 1)^2 \tag{4}$$

The final cost function for the discriminator is the sum of the losses for real and generated images, as well as the gradient penalty:

$$\mathcal{L}_D = \mathcal{L}_{\text{real}} + \mathcal{L}_{\text{fake}} + \text{GP} \tag{5}$$

The generator cost function is crucial for training the generator network, which aims to produce realistic images with desired emotions that are indistinguishable from real images. The cost function for the generator combines several components to achieve this goal.

The adversarial loss encourages the generator to produce images that the discriminator classifies as real. This is achieved by minimizing the negative output of the discriminator for the generated images x_{fake}.

$$\mathcal{L}_{\text{adv}} = -\frac{1}{m}\sum_{i=1}^{m} \text{out}_{\text{fake}}^{(i)} \tag{6}$$

where m is the batch size and out_{fake} is the discriminator's output for the generated images.

The classification loss ensures that the generated images are classified with the desired emotion labels $x_{\text{fake\_cls}}$. This is implemented using the cross-entropy loss between the predicted labels and the target labels:

$$\mathcal{L}_{\text{cls}} = \frac{1}{m}\sum_{i=1}^{m} \text{CrossEntropy}\left(\text{out}_{\text{class}}^{(i)}, x_{\text{fake\_cls}}^{(i)}\right) \tag{7}$$

where $\text{out}_{\text{class}}$ is the classifier's output for the generated images.

The reconstruction loss encourages the generated images to resemble the real images when the target emotion matches the original emotion. This loss is masked to only include samples where the target emotion is the same as the original emotion, and is computed as the mean squared error between the real and generated images:

$$\mathcal{L}_{\text{rec}} = \frac{\sum_{i=1}^{m} \text{MSE}\left(x_{\text{real}}^{(i)}, x_{\text{fake}}^{(i)}\right) \cdot \text{mask}^{(i)}}{\sum_{i=1}^{m} \text{mask}^{(i)} + \epsilon} \tag{8}$$

where mask is a binary mask indicating samples where the target and original emotions match, and ϵ is a small constant to prevent division by zero.

The total generator loss is a weighted sum of the adversarial loss, classification loss, and reconstruction loss:

$$\mathcal{L}_G = \mathcal{L}_{\text{adv}} + \mathcal{L}_{\text{cls}} + \mathcal{L}_{\text{rec}} \tag{9}$$

The learning process of the generator consisted of 30 epochs of the learning algorithm. In this process, the Adam Optimizer [39] algorithm was used. The batch size was 64.

The training set consisted of automatically and manually annotated facial images from the AffectNet [23] database, which were transformed to a resolution of 128 × 128. In the training process, a series of data augmentation techniques are applied to the images to enhance the robustness and generalization capabilities of the model. These transformations include resizing the images to a specified size, ensuring they fit within the model's input dimensions. Additionally, the images undergo the longest max size transformation to maintain aspect ratio while fitting within a maximum size constraint. Fancy PCA [40] is applied to adjust the principal components of the image colors, followed by random gamma adjustments to vary the brightness non-linearly. Sharpening is used to enhance the clarity of the image features. Horizontal flipping is randomly applied to introduce variability and improve the model's ability to generalize to different orientations. Finally, a random crop is performed to extract fixed-size patches from the images, which helps the model learn from different parts of the image and reduces overfitting.

In order to properly evaluate the effectiveness of EmoDNN, a separate additional classifier using TensorFlow machine learning library was trained. The input and output of the classifier are identical to those of the discriminator. The input of the classifier is an image, which returns a vector with the detected probability for emotions as the output. In order to present the learning process of the classifier, the following loss function relationships will be discussed:

$$\mathcal{L}_{\text{clsTF}} = \frac{1}{m}\sum_{i=1}^{m} 1.14 \cdot \left(\text{CrossEntropy}\left(\text{v}_{Ref}\ ,\ \text{v}_{Out}\right) + \left(\frac{1 - \text{v}_{Out}[i]}{4}\right)\right) \tag{10}$$

where v_{Ref} is a reference vector containing the value 1 for the correct emotion and 0 for others; v_{Out} is the result vector of the classifier containing the probability of individual emotions; $[i]$ is the value of the i-th element of the vector representing the emotion currently presented at the network input.

The learning process of the classifier consisted of three epochs of the learning algorithm. In this process, the Adam Optimizer algorithm was used. The learning rate value was set to 10^{-4}. The batch size used was 16.

The training set consisted only of manually annotated facial images from the AffectNet [23] database, which were transformed to a resolution of 128 × 128. In the training process, a series of data augmentation techniques are applied to the images to enhance the robustness and generalization capabilities of the model as for the classifier written using the PyTorch machine learning library described earlier in this section.

3.3. Neural Network Architecture

The generator is designed using several sub-models to perform image-to-image translation conditioned on emotion vectors. It employs a U-Net-like architecture [41]. Table 1 outlines the structure of the generator, including its sub-models and their components.

Table 1. Generator structure.

Component	**Description**
Generator	The main model for generating images. It uses a U-Net architecture conditioned on emotion vectors.
Classification MLP	A multi-layer perceptron that processes the one-hot emotion vector into feature vectors.
- Linear Layer 1	Transforms the input emotion vector to a higher-dimensional feature space.
- Activation (GELU)	Applies GELU activation function for non-linearity.
- Linear Layer 2	Further transforms the features to match the required input for the U-Net.
- Activation (GELU)	Applies GELU activation function for non-linearity.
U-Net	The backbone of the generator, consisting of an encoder and a decoder for image generation.
Encoder	Encodes the input image into a lower-dimensional latent space.
- Initial Convolution	A convolutional layer to process the input image.
- ResDown Blocks	Residual blocks with downsampling, including conditional normalization layers.
- ResBlock	A residual block that processes features before passing to the decoder.
Decoder	Decodes the latent representation back into an image.
- ResBlock	A residual block that processes features before upsampling.
- ResUp Blocks	Residual blocks with upsampling, including conditional normalization layers.
- Final Convolution	A convolutional layer to produce the final output image.

Table 1. *Cont.*

Component	Description
ConditionalNorm2d	Applies conditional normalization based on the emotion vector.
ResDown	Residual downsampling block with conditional normalization and spectral normalization.
ResUp	Residual upsampling block with conditional normalization and spectral normalization.
ResBlock	Standard residual block with conditional normalization and spectral normalization.

The discriminator is designed to differentiate between real and generated images. Table 2 outlines the structure of the discriminator and its components.

Table 2. Discriminator structure.

Component	Description
Discriminator	The main model for distinguishing real images from generated ones. It uses an encoder to process images.
Encoder	Encodes the input image into a lower-dimensional feature representation.
- Initial Convolution	A convolutional layer that processes the input image.
- ResDown Blocks	Residual blocks with downsampling. These blocks consist of convolutional layers and normalization.
- ResBlock	A residual block that processes the features before passing them to the final layers.
Output Layer	A convolutional layer that reduces the feature map to a single-channel output for real/fake classification.

The classifier is designed to classify the emotion of input images. Table 3 outlines the structure of the classifier and its components.

Table 3. PyTorch based classifier structure.

Component	Description
Classifier	The main model for classifying the emotion of input images. It uses an encoder to process images.
Encoder	Encodes the input image into lower-dimensional feature representation.
- Initial Convolution	A convolutional layer that processes the input image.
- ResDown Blocks	Residual blocks with downsampling. These blocks consist of convolutional layers and normalization.
- ResBlock	A residual block that processes the features before passing them to the final layers.
Output Layer	A convolutional layer that reduces the feature map to a multi-channel output for emotion classification.

The model employs several advanced techniques and methods to enhance its performance and stability. Spectral Normalization [42] is used in convolutional layers within both the generator and discriminator to stabilize training by controlling the Lipschitz constant, which helps prevent exploding gradients and improves the robustness of the model. Conditional Normalization (ConditionalNorm2d) [43], which includes InstanceNorm and

BatchNorm variants, is applied in the generator to condition the normalization process on the emotion vectors, allowing the model to effectively incorporate emotion-specific features into the generated images. Exponential Linear Unit (ELU) [44] activation functions are used throughout the network to introduce non-linearity, which helps the model learn complex patterns and improves convergence by mitigating the vanishing gradient problem. Additionally, Residual Blocks (ResBlock) and Residual Down/Upsampling Blocks (ResDown, ResUp) [45] are used to facilitate the flow of gradients through the network, promoting efficient training and better feature learning. These residual connections ensure that the model can learn both low-level and high-level features effectively, contributing to the overall performance and stability of the models in generating and classifying images.

The structure of a classifier using TensorFlow presented in this manuscript uses standard layers from the TensorFlow machine library. The LeakyReLu [46] activation function was used for all layers except the last layer for which the Softmax [47] activation function was applied. For the output layer, the value of units is equal to the number of recognized emotion types which is eight. Table 4 shows the classifier structure.

Table 4. TensorFlow-based classifier structure.

#	Layer Type	#	Layer Type
1	Conv2D(filters=32, kernel_size=(3, 3))(Input)	17	Dense(units=256)(16)
2	MaxPooling2D(pool_size=(2, 2))(1)	18	Dense(units=256)(17)
3	BatchNormalization(2)	19	Dense(units=256)(15)
4	Conv2D(filters=32, kernel_size=(3, 3))(3)	20	Dense(units=256)(19)
5	Conv2D(filters=64, kernel_size=(5, 5))(4)	21	Concatenate(axis=1)(18, 20)
6	MaxPooling2D(pool_size=(2, 2))(5)	22	BatchNormalization(21)
7	BatchNormalization(6)	23	Dropout(rate=0.31, seed=321)(22)
8	Conv2D(filters=64, kernel_size=(5, 5))(7)	24	BatchNormalization(23)
9	Conv2D(filters=128, kernel_size=(7, 7))(8)	25	Dense(units=512)(24)
10	Conv2D(filters=128, kernel_size=(7, 7))(9)	26	BatchNormalization(25)
11	MaxPooling2D(pool_size=(2, 2))(10)	27	Dense(units=512)(26)
12	BatchNormalization(11)	28	BatchNormalization(27)
13	Flatten(12)	29	Dense(units=1024)(28)
14	BatchNormalization(13)	30	BatchNormalization(29)
15	Dense(units=256)(14)	31	Dense(units=8)(30)
16	Dropout(rate=0.37, seed=274)(15)		

4. Results

The results of changing emotions in photos and the performance of classifiers will be discussed and presented in this section. First, the effects of changing emotions in photos will be presented, then the results of the classifiers will be discussed in comparison with other state-of-the-art methods.

4.1. Applied Dataset

The research presented in this manuscript used the AffectNet dataset under the Academic Use License for scientific research purposes. The dataset can be obtained by making a prior request on the AffectNet website [48].

The AffectNet dataset provides 11 annotated emotions for images and indexed as follows: 0—Neutral, 1—Happiness, 2—Sadness, 3—Surprise, 4—Fear, 5—Disgust, 6—Anger, 7—Contempt, 8—None, 9—Uncertain, 10—No-Face. In the learning process, the first eight categories defining a specific emotional state were used, the number of which is presented in Table 5. For research purposes, these images were resized to 128 × 128 resolution, which is the image resolution for the DNNs discussed below. Additionally, the images were normalized for processing by the networks in the range of values $\langle 0; 1 \rangle$.

Traditionally, AffectNet has been widely used for emotion recognition. In this study, we explore its potential for emotion modification, using it to evaluate model effectiveness in generating realistic emotional changes. By employing AffectNet in this unique capacity, we gain insights into its suitability for generative tasks.

Table 5. Number of images for each emotion type from the AffectNet database.

Annotated Training Set			**Validation Set**	
Automatically	**Manually**	**Total**	**Emotion**	
143,142	74,874	218,016	Neutral	500
246,235	134,415	380,650	Happy	500
20,854	25,459	46,313	Sad	500
17,462	14,090	31,552	Surprise	500
3799	6378	10,177	Fear	500
890	3803	4693	Disgust	500
28,000	24,882	52,882	Anger	500
2	3750	3752	Contempt	500
460,384	287,651	748,035		4000

4.2. Emotions Change

Due to the lack of clearly determining performance metrics of the correctness of emotion change for deepfake technology, the emotion classifiers discussed later in this section were trained. Examples of emotion changes in photos by EmoDNN are presented in Figure 3. More results on emotion changes by EmoDNN are provided in Appendix A.

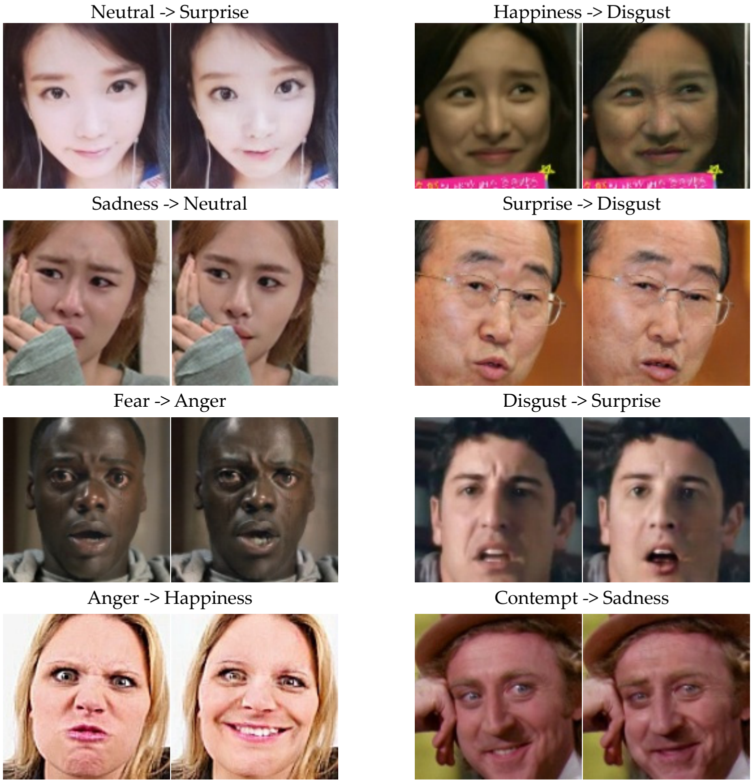


Figure 3. EmoDNN emotion change preview.

By analyzing the results obtained for individual cases in detail, small graphic artifacts are visible to the human eye. An example that may raise suspicion of the use of deepfake technology in Figure 3 is the change in emotion from Disgust to Surprise, for which the lower lips may or may not be a cause for suspicion of the use of deepfake technology. Moreover, the appearance of a smile from the change of emotion from Angry to Happiness may or may not be a cause for suspicion of the use of deepfake technology.

The backbone of the generator, consisting of an encoder and a decoder for image generation is U-Net [41]. Applying a U-Net-like network enhanced facial expression recognition by segmenting key regions on faces such as the eyes, mouth, and brows. These regions are critical for understanding subtle emotional cues. By focusing on these regions, the model can identify micro-expressions and subtle emotion indicators with improved accuracy.

The presented approach omits the cycle-consistency loss employed in models such as StarGAN [49], instead using a direct reconstruction loss limited to cases where the target and ground-truth emotions match. This allows for a higher level of realism in emotion modifications, reducing artifacts commonly associated with unintended source features. Although some facial features may diminish in this process, the resulting images retain a more authentic appearance of the intended emotion.

Furthermore, compared to FEGAN, which uses region-specific editing to alter emotions precisely while preserving identity, the presented approach achieves a balance by modifying the entire facial structure. FEGAN's region-specific editing can yield nuanced expressions, but it sometimes struggles with texture consistency during subtle adjustments. In contrast, the presented method performs broader modifications, achieving smoother transitions without introducing artifacts, especially during more intense emotional shifts.

4.3. Emotions Detection

Due to the difficulty of objectively assessing the success of changing emotions in a photo, two emotion classifiers based on a neural network were trained. These classifiers have different overall accuracy; however, for individual cases, the classifier with lower overall accuracy can return the correct emotion as a result, while the theoretically more accurate classifier can classify emotions incorrectly in individual cases. Training more than one classifier increases the probability of correct emotion detection in the case of two different classifiers detecting the same emotion. The confusion matrices of the trained classifiers are presented in Figure 4.

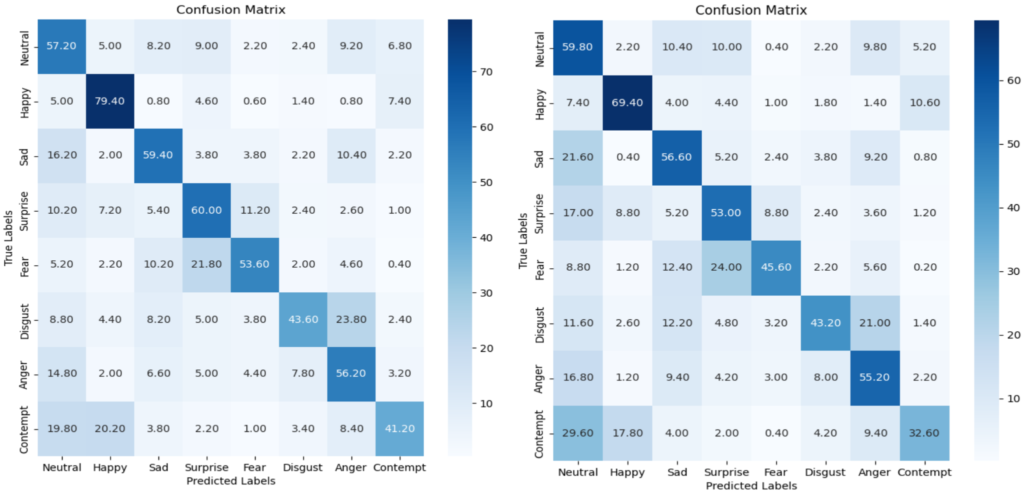


Figure 4. Confusion matrices of trained classifiers (from left based on PyTorch; from right based on TensorFlow).

The performance of the classifiers in tabular form is presented in Table 6.

The comparison of the trained classifiers with other state-of-the-art classifiers trained for eight emotions on the AffectNet database is given in Table 7.

A comparison of the accuracy of the classifiers trained for this article with other state-of-the-art classifiers shows that there are more effective solutions in available publications. These methods may be more accurate in general, but they may detect incorrect emotions for individual cases for which a less accurate neural network may detect these emotions correctly. Therefore, for emotion detection, it is proposed that two different neural networks are trained in order to ensure an increased probability of correct detection by confirming the detection of a given emotion from more than one source. The difference between the accuracy of a classifier based on PyTorch and TensorFlow results not only from the applied neural network architecture, but also from the reduced training set for the classifier based on TensorFlow.

Table 6. Performance of classifiers for individual emotion types with respect to Recall and F1 score machine learning evaluation metrics.

PyTorch-Based			TensorFlow-Based	
Recall	**F1 Score**	**Emotion**	**Recall**	**F1 Score**
0.57	0.48	Neutral	0.60	0.33
0.79	0.71	Happy	0.69	0.68
0.59	0.59	Sad	0.57	0.53
0.60	0.57	Surprise	0.53	0.51
0.54	0.59	Fear	0.46	0.55
0.44	0.53	Disgust	0.43	0.51
0.56	0.52	Anger	0.55	0.51
0.41	0.50	Contempt	0.33	0.42

Table 7. Classifier comparison.

Method	Accuracy (%)
DDAMFN++ [13]	65.04
FMAE [14]	65.00
BTN [15]	64.29
DDAMFN [13]	64.25
POSTER++ [16]	63.77
S2D [17]	63.06
Multi-task EfficientNet-B2 [18]	63.03
DAN [19]	62.09
PSR [20]	60.68
EfficientFace [21]	59.89
RAN [22]	59.50
ViT-tiny [24]	58.28
Weighted-Loss [23]	58.00
ViT-base [24]	57.99
LResNet50E-IR [25]	53.93
PyTorch-based classifier	56.33
TensorFlow-based classifier	51.93

In order to check the efficiency of the presented method of changing emotions in photos, a dataset consisting of 8000 images was generated. Each emotion is represented by 1000 images generated by EmoDNN from the AffectNet database. The confusion matrices of the trained classifiers of generated faces with changed emotion are presented in Figure 5.

The performance of the classifiers in tabular form on the dataset generated by EmoDNN is presented in Table 8.

The accuracy of the PyTorch-based classifier is 0.99. This accuracy value is due to the fact that the generator was trained with guidance from the classifier, resulting in generated images that align closely with the classifier's learned representations. Consequently, the classifier achieves high accuracy when evaluating these generated images. Nearly 100% classification accuracy achieved by generated samples when evaluated by the classifier used in the generation process uncovers the use of classifiers as a potential research objective, suggesting that further studies could explore the effects of incorporating diverse datasets or multiple classifiers within the generation and classification pipelines. Such research could promote broader algorithmic generalization across various emotional contexts, enhancing the robustness and reliability of emotion modification models.

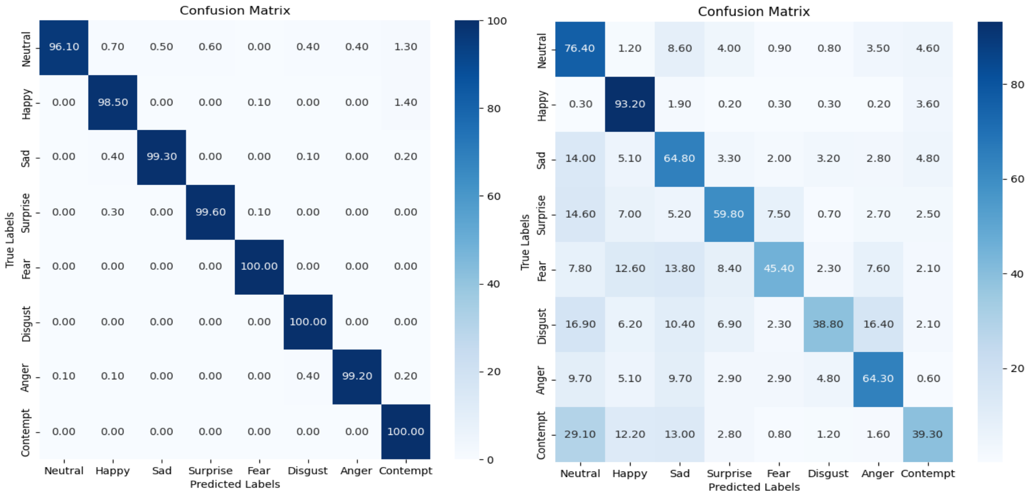


Figure 5. Confusion matrices of trained classifiers of generated faces with changed emotion (from left based on PyTorch; from right based on TensorFlow).

Table 8. Performance of classifiers for individual emotion types with respect to Recall and F1 score machine learning evaluation metrics.

PyTorch-Based			TensorFlow-Based	
Recall	**F1 Score**	**Emotion**	**Recall**	**F1 Score**
0.96	0.98	Neutral	0.76	0.57
0.98	0.98	Happy	0.93	0.77
0.99	0.99	Sad	0.65	0.57
1.00	1.00	Surprise	0.60	0.64
1.00	1.00	Fear	0.45	0.56
1.00	1.00	Disgust	0.39	0.51
0.99	0.99	Anger	0.64	0.65
1.00	0.98	Contempt	0.39	0.49

The accuracy of the TensorFlow-based classifier is smaller than for Pytorch-based classifier and has a value of 0.60. The TensorFlow-based classifier was trained exclusively on the training set consisting only of manually annotated facial images from the AffectNet database. Comparing the accuracy of this classifier on the set of images generated by EmoDNN (0.60) with the accuracy obtained on the validation set from the AffectNet database (0.51), it should be noted that the efficiency of the classifier is different. The classifier performed more efficiently on images generated by EmoDNN; this may be due to the generalization ability of DNN and the ambiguity of emotions from images selected for the validation set of the AffectNet dataset.

5. Conclusions

Changing facial expressions to convey emotion is a complex task. Changing people's emotions leads to a change in facial expression. Changing emotions through a DNN can cause the appearance of more or less visible graphic artifacts. Adequate change in expression without introducing visible graphic artifacts to the human eye or specialized DNNs for deepfake detection poses a challenge. It should be noted that human emotions can have very similar facial expressions, which leads to the recognition of different emotions in the same photo not only by people but also by artificial intelligence algorithms.

To further enhance the training process, stability algorithms such as Gradient Penalty and hinge loss have been implemented, which have been shown to stabilize the learning process and improve the quality of the generated images. Furthermore, to refine the classification of emotions, using multiple classifiers trained in parallel has been proposed. This multi-classifier approach increases the likelihood of obtaining accurate results on generated data, mirroring the performance seen on real data. The authors of this paper suggest that this metric, comparing classifier accuracy on real and generated data, could be valuable for other researchers as a benchmark for evaluating the efficacy of emotion generation models.

The classifiers trained for the purposes of this article constitute an element of emotion detection before its change as well as the success of its change. Taking into account the fact that classifiers are trained on different types of architectures and different sizes of the training set (only on manually annotated photos or additionally with automatically annotated photos), detection of the same emotion by two separate classifiers increases the probability of its correct detection.

The efficiency of the solution presented in this article was confirmed by dedicated classifiers. Changing the emotions of a person in a photo using the presented solution provides the possibility of changing emotions; however, it should be noted that this change is not always successful. Depending on the image input to the neural network and the target change in emotions, undesirable graphical effects may appear that reveal the use of photo manipulation. In order to eliminate possible graphic artifacts, an additional neural network should be used to detect unnatural defects in human faces. Moreover, an additional network should be added that modifies the image in an invisible way (imperceptible to the human eye) before it is fed to the input of the emotion change network, so as to maximize the similarity of characteristic facial features while ensuring success in changing emotions expressed as a lack of potentially visible graphic artifacts.

In conclusion, the contributions of this work extend beyond the application of existing emotion manipulation techniques by incorporating advanced methodologies that enhance both the precision and stability of emotion generation. The results demonstrate the effectiveness of the proposed approach in producing realistic emotion changes, while also highlighting areas for future exploration, such as improving the subtlety of emotion transitions and further reducing visible artifacts. Overall, this research lays the groundwork for advancing emotion detection and generation technologies, with potential applications in areas like human–computer interaction, virtual reality, and psychological analysis.

Author Contributions: Conceptualization, Z.P., T.W. and M.K.; funding acquisition, Z.P.; methodology, Z.P., T.W. and M.K.; project administration, Z.P.; software, T.W. and M.K.; supervision, Z.P.; visualization, T.W. and M.K.; writing—original draft, Z.P., T.W. and M.K. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Military University of Technology, Faculty of Electronics, grant number UGB/22-747 on the application of artificial intelligence methods to cognitive spectral analysis, satellite communications, and watermarking and technology deepfake.

Institutional Review Board Statement: Not applicable.

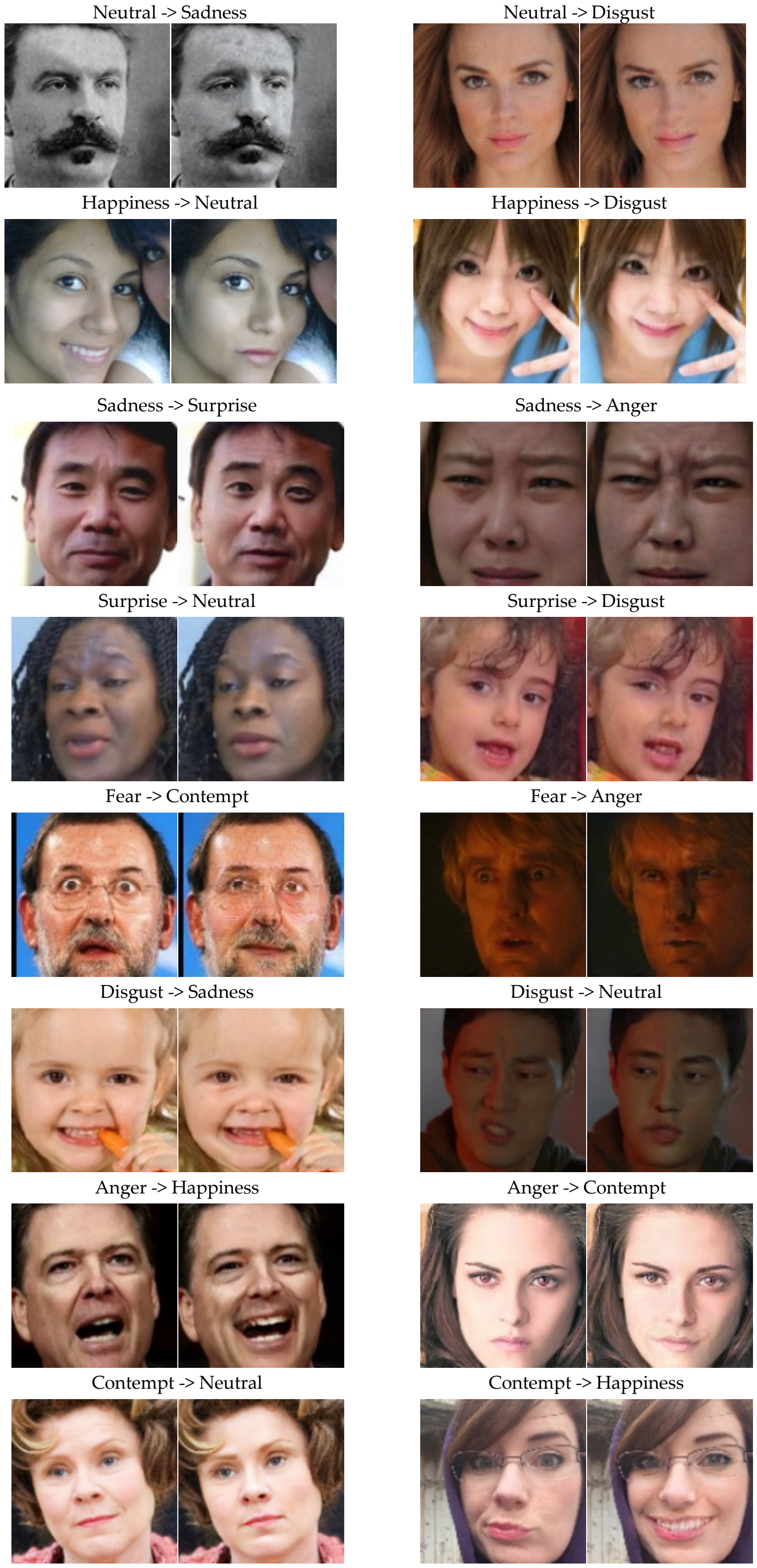
Informed Consent Statement: Not applicable.

Data Availability Statement: The research used the AffectNet^{TM} dataset under the Academic Use License for scientific research purposes. The dataset can be obtained by making a prior request at http://mohammadmahoor.com/affectnet/ (accessed on 6 November 2024).

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

Appendix A

The results of EmoDNN in the process of changing people's emotions in images based on the AffectNet dataset are presented in Figure A1.



Neutral -> Sadness

Neutral -> Disgust

Happiness -> Neutral

Happiness -> Disgust

Sadness -> Surprise

Sadness -> Anger

Surprise -> Neutral

Surprise -> Disgust

Fear -> Contempt

Fear -> Anger

Disgust -> Sadness

Disgust -> Neutral

Anger -> Happiness

Anger -> Contempt

Contempt -> Neutral

Contempt -> Happiness

Figure A1. Preview of sample generated images for individual emotions (viewed from the top, the rows represent different emotions; viewed from the left, the consecutive columns represent pairs of images: [original image, image with changed emotion generated by EmoDNN]).

References

1. Ekman, P.; Friesen, W.V. Constants across cultures in the face and emotion. *J. Personal. Soc. Psychol.* **1971**, *17*, 124–129. [CrossRef]
2. Plutchik, R. A General Psychoevolutionary Theory of Emotion. In *Theories of Emotion*; Plutchik, R., Kellerman, H., Eds.; Elsevier: Amsterdam, The Netherlands, 1980; pp. 3–33.
3. Williams, L.; Arribas-Ayllon, M.; Artemiou, A.; Spasić, I. Comparing the Utility of Different Classification Schemes for Emotive Language Analysis. *J. Classif.* **2019**, *36*, 619–648. [CrossRef]
4. Watson, D.; Tellegen, A. Toward a Consensual Structure of Mood. *Psychol. Bull.* **1985**, *98*, 219–235. [CrossRef] [PubMed]
5. Bistroń, M.; Piotrowski, Z. Comparison of Machine Learning Algorithms Used for Skin Cancer Diagnosis. *Appl. Sci.* **2022**, *12*, 9960. [CrossRef]
6. Walczyna, T.; Piotrowski, Z. Overview of Voice Conversion Methods Based on Deep Learning. *Appl. Sci.* **2023**, *13*, 3100. [CrossRef]
7. Kaczyński, M.; Piotrowski, Z. High-Quality Video Watermarking Based on Deep Neural Networks and Adjustable Subsquares Properties Algorithm. *Sensors* **2022**, *22*, 5376. [CrossRef]
8. Kaczyński, M.; Piotrowski, Z.; Pietrow, D. High-Quality Video Watermarking Based on Deep Neural Networks for Video with HEVC Compression. *Sensors* **2022**, *22*, 7552. [CrossRef]
9. Zhang, K.; Zhang, Z.; Li, Z.; Qiao, Y. Joint face detection and alignment using multitask cascaded convolutional networks. *IEEE Signal Process. Lett.* **2016**, *23*, 1499–1503. [CrossRef]
10. Karras, T.; Laine, S.; Aila, T. A style-based generator architecture for generative Adversarial Networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2021**, *43*, 4217–4228. [CrossRef]
11. Ning, X.; Xu, S.; Li, W.; Nie, S. Fegan: Flexible and efficient face editing with pre-trained generator. *IEEE Access* **2020**, *8*, 65340–65350. [CrossRef]
12. Zhu, J.-Y.; Park, T.; Isola, P.; Efros, A.A. Unpaired image-to-image translation using cycle-consistent adversarial networks. In Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017; pp. 2242–2251. [CrossRef]
13. Zhang, S.; Zhang, Y.; Zhang, Y.; Wang, Y.; Song, Z. A Dual-Direction Attention Mixed Feature Network for Facial Expression Recognition. *Electronics* **2023**, *12*, 3595. [CrossRef]
14. Ning, M.; Salah, A.A.; Ertugrul, I.O. Representation Learning and Identity Adversarial Training for Facial Behavior Understanding. *arXiv* **2024**, arXiv:2407.11243.
15. Her, M.B.; Jeong, J.; Song, H.; Han, J.-H. Batch Transformer: Look for Attention in Batch. *arXiv* **2024**, arXiv:2407.04218.
16. Mao, J.; Xu, R.; Yin, X.; Chang, Y.; Nie, B.; Huang, A. POSTER++: A Simpler and Stronger Facial Expression Recognition Network. *arXiv* **2023**, arXiv:2301.12149. [CrossRef]
17. Chen, Y.; Li, J.; Shan, S.; Wang, M.; Hong, R. From Static to Dynamic: Adapting Landmark-Aware Image Models for Facial Expression Recognition in Videos. *IEEE Trans. Affect. Comput.* **2024**. *early access*. [CrossRef]
18. Savchenko, A.V.; Savchenko, L.V.; Makarov, I. Classifying Emotions and Engagement in Online Learning Based on a Single Facial Expression Recognition Neural Network. *IEEE Trans. Affect. Comput.* **2022**, *13*, 2132–2143. [CrossRef]
19. Wen, Z.; Lin, W.; Wang, T.; Xu, G. Distract Your Attention: Multi-Head Cross Attention Network for Facial Expression Recognition. *Biomimetics* **2023**, *8*, 199. [CrossRef]
20. Vo, T.-H.; Lee, G.-S.; Yang, H.-J.; Kim, S.-H. Pyramid with Super Resolution for In-the-Wild Facial Expression Recognition. *IEEE Access* **2020**, *8*, 131988–132001. [CrossRef]
21. 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]
22. Wang, K.; Peng, X.; Yang, J.; Meng, D.; Qiao, Y. Region Attention Networks for Pose and Occlusion Robust Facial Expression Recognition. *arXiv* **2019**, arXiv:1905.04075. [CrossRef]
23. Mollahosseini, A.; Hasani, B.; Mahoor, M.H. AffectNet: A Database for Facial Expression, Valence, and Arousal Computing in the Wild. *IEEE Trans. Affect. Comput.* **2019**, *10*, 18–31. [CrossRef]
24. Li, J.; Nie, J.; Guo, D.; Hong, R.; Wang, M. Emotion Separation and Recognition from a Facial Expression by Generating the Poker Face with Vision Transformers. *arXiv* **2023**, arXiv:2207.11081. [CrossRef]
25. Zhou, H.; Meng, D.; Zhang, Y.; Peng, X.; Du, J.; Wang, K.; Qiao, Y. Exploring Emotion Features and Fusion Strategies for Audio-Video Emotion Recognition. In Proceedings of the ICMI'19: 2019 International Conference on Multimodal Interaction, Suzhou China, 14–18 October 2019; pp. 562–566. [CrossRef]
26. Walczyna, T.; Piotrowski, Z. Fast Fake: Easy-to-Train Face Swap Model. *Appl. Sci.* **2024**, *14*, 2149. [CrossRef]
27. Perov, I.; Gao, D.; Chervoniy, N.; Liu, K.; Marangonda, S.; Umé, C.; Dpfks, M.; Facenheim, C.S.; RP, L.; Jiang, J.; et al. DeepFaceLab: Integrated, flexible and extensible face-swapping framework. *arXiv* **2021**, arXiv:2005.05535.
28. Chen, R.; Chen, X.; Ni, B.; Ge, Y. SimSwap: An Efficient Framework for High Fidelity Face Swapping. In Proceedings of the 28th ACM International Conference on Multimedia, Seattle, WA, USA, 12–16 October 2020; pp. 2003–2011. [CrossRef]
29. Li, L.; Bao, J.; Yang, H.; Chen, D.; Wen, F. FaceShifter: Towards High Fidelity and Occlusion Aware Face Swapping. *arXiv* **2020**, arXiv:1912.13457.
30. Groshev, A.; Maltseva, A.; Chesakov, D.; Kuznetsov, A.; Dimitrov, D. GHOST—A New Face Swap Approach for Image and Video Domains. *IEEE Access* **2022**, *10*, 83452–83462. [CrossRef]

31. Kim, K.; Kim, Y.; Cho, S.; Seo, J.; Nam, J.; Lee, K.; Kim, S.; Lee, K. DiffFace: Diffusion-based Face Swapping with Facial Guidance. *arXiv* **2022**, arXiv:2212.13344.
32. Wang, Y.; Chen, X.; Zhu, J.; Chu, W.; Tai, Y.; Wang, C.; Li, J.; Wu, Y.; Huang, F.; Ji, R. HifiFace: 3D Shape and Semantic Prior Guided High Fidelity Face Swapping. *arXiv* **2021**, arXiv:2106.09965.
33. Tarchi, P.; Lanini, M.C.; Frassineti, L.; Lanatà, A. Real and Deepfake Face Recognition: An EEG Study on Cognitive and Emotive Implications. *Brain Sci.* **2023**, *13*, 1233. [CrossRef]
34. Gupta, G.; Raja, K.; Gupta, M.; Jan, T.; Whiteside, S.T.; Prasad, M. A Comprehensive Review of DeepFake Detection Using Advanced Machine Learning and Fusion Methods. *Electronics* **2024**, *13*, 95. [CrossRef]
35. Alhaji, H.S.; Celik, Y.; Goel, S. An Approach to Deepfake Video Detection Based on ACO-PSO Features and Deep Learning. *Electronics* **2024**, *13*, 2398. [CrossRef]
36. Javed, M.; Zhang, Z.; Dahri, F.H.; Laghari, A.A. Real-Time Deepfake Video Detection Using Eye Movement Analysis with a Hybrid Deep Learning Approach. *Electronics* **2024**, *13*, 2947. [CrossRef]
37. Lim, J.H.; Ye, J.C. Geometric GAN. *arXiv* **2017**, arXiv:1705.02894.
38. Arjovsky, M.; Chintala, S.; Bottou, L. Wasserstein Generative Adversarial Networks. In Proceedings of the 34th International Conference on Machine Learning, Sydney, NSW, Australia, 6–11 August 2017; Precup, D., Teh, Y.W., Eds.; PMLR: New York, NY, USA, 2017; Volume 70, pp. 214–223.
39. Zhang, Z. Improved adam optimizer for deep neural networks. In Proceedings of the 2018 IEEE/ACM 26th International Symposium on Quality of Service (IWQoS), Banff, AB, Canada, 4–6 June 2018; pp. 1–2.
40. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. ImageNet Classification with Deep Convolutional Neural Networks. In *Advances in Neural Information Processing Systems*; Pereira, F., Burges, C.J., Bottou, L., Weinberger, K.Q., Eds.; Curran Associates, Inc.: Red Hook, NY, USA, 2012; Volume 25.
41. Ronneberger, O.; Fischer, P.; Brox, T. U-Net: Convolutional Networks for Biomedical Image Segmentation. In *Medical Image Computing and Computer-Assisted Intervention—MICCAI 2015*; Springer: Berlin/Heidelberg, Germany, 2015; pp. 234–241. [CrossRef]
42. Miyato, T.; Kataoka, T.; Koyama, M.; Yoshida, Y. Spectral Normalization for Generative Adversarial Networks. *arXiv* **2018**, arXiv:1802.05957.
43. Dumoulin, V.; Shlens, J.; Kudlur, M. A Learned Representation For Artistic Style. *arXiv* **2017**, arXiv:1610.07629.
44. Clevert, D.-A.; Unterthiner, T.; Hochreiter, S. Fast and Accurate Deep Network Learning by Exponential Linear Units (ELUs). *arXiv* **2016**, arXiv:1511.07289.
45. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. *arXiv* **2015**, arXiv:1512.03385.
46. Maas, A.L.; Hannun, A.Y.; Ng, A.Y. Rectifier nonlinearities improve neural network acoustic models. In Proceedings of the 30th International Conference on Machine Learning, Atlanta, GA, USA, 16–21 June 2013; Volume 30.
47. Goodfellow, I.; Bengio, Y.; Courville, A. 6.2.2.3 Softmax Units for Multinoulli Output Distributions. In *Deep Learning*; MIT Press: Cambridge, MA, USA, 2016; pp. 180–184, ISBN 978-0-26203561-3.
48. Mahoor, M. *AffectNet*; Mohammad Mahoor: Denver, CO, USA, 2024. Available online: http://mohammadmahoor.com/affectnet/ (accessed on 6 November 2024).
49. Choi, Y.; Choi, M.; Kim, M.; Ha, J.-W.; Kim, S.; Choo, J. StarGAN: Unified Generative Adversarial Networks for Multi-Domain Image-to-Image Translation. *arXiv* **2018**, arXiv:1711.09020.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.