Article

Thermal Image Generation for Robust Face Recognition

Vicente Pavez, Gabriel Hermosilla *, Francisco Pizarro , Sebastián Fingerhuth and Daniel Yunge

Escuela de Ingeniería Eléctrica, Pontificia Universidad Católica de Valparaíso, Av. Brasil 2147, Valparaíso 2362804, Chile; vicente.pavez.o@mail.pucv.cl (V.P.); francisco.pizarro.t@pucv.cl (F.P.); sebastian.fingerhuth@pucv.cl (S.F.); daniel.yunge@pucv.cl (D.Y.)

* Correspondence: gabriel.hermosilla@pucv.cl

Abstract: This article shows how to create a robust thermal face recognition system based on the FaceNet architecture. We propose a method for generating thermal images to create a thermal face database with six different attributes (frown, glasses, rotation, normal, vocal, and smile) based on various deep learning models. First, we use StyleCLIP, which oversees manipulating the latent space of the input visible image to add the desired attributes to the visible face. Second, we use the GANs N' Roses (GNR) model, a multimodal image-to-image framework. It uses maps of style and content to generate thermal imaging from visible images, using generative adversarial approaches. Using the proposed generator system, we create a database of synthetic thermal faces composed of more than 100k images corresponding to 3227 individuals. When trained and tested using the synthetic database, the Thermal-FaceNet model obtained a 99.98% accuracy. Furthermore, when tested with a real database, the accuracy was more than 98%, validating the proposed thermal images generator system.

Keywords: generative models; data generation; images generation; real-world applications

Citation: Pavez, V.; Hermosilla, G.; Pizarro, F.; Fingerhuth, S.; Yunge, D. Thermal Image Generation for Robust Face Recognition. *Appl. Sci.* **2022**, *12*, 497. <https://doi.org/10.3390/app12010497>

Academic Editors: Mauro Castelli and Luca Manzoni

Received: 1 December 2021

Accepted: 25 December 2021

Published: 5 January 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

During the last decade, the field of Artificial Intelligence has grown considerably. Through new algorithms, it helped to create new applications such as autonomous driving, the prediction of 3D structures in proteins for amino acid sequences [1], mastering games with limited training data using deep reinforcement learning [2,3], learning to create visual concepts and draw artworks from natural language supervision [4], or the effective use of machine learning algorithms for complicated tasks such as the diagnosis and modeling of aerospace structural defects [5]. All these would not have been possible without neural networks and deep learning. Deep learning is a subfield within machine learning that uses different architectures based on neural networks that allow learning various tasks using deep representations of neural layers. The models can learn to conduct these tasks automatically when trained with hundreds or thousands of input data.

Face recognition (FR) is a natural way of recognizing people. It has also been used with great success in deep learning through different architectures, mainly based on convolutional neural networks (CNN). Currently, architectures such as Inception-ResNet [6], vision transformers [7], and algorithms designed for efficient and robust face comparison [8] are used for FR. These architectures have achieved fascinating results in controlled conditions. However, in real applications, different factors, such as changes in lighting conditions and pose variations, significantly affect the performance of such systems [9]. Accordingly, to improve the performance of such face recognition systems, developers should use many face images with different poses and lighting conditions for optimal performance.

A solution to improve the capabilities and overcome the limitations of the visible images in facial recognition applications is the use of thermal images since they have properties such as invariance to lighting conditions and robustness in pose variations [10,11]. Due to the properties of infrared radiation, thermal cameras make it possible to obtain

images captured in the longwave infrared spectrum (8–12 μm) so that they can operate in complete darkness. e.g., from Figure 1, it is possible to observe how the thermal image remains unchanged in the scene, even when it has extreme variations in illumination (low and high light intensity, darkness) in the visible image. The use of thermal images could help to improve the performance of face recognition systems in deep learning applications. However, massive training data is necessary to build efficient models. Unfortunately, there are no extensive databases in the literature having thermal images available to train FR models with deep learning.

Figure 1. Subject captured under different lighting conditions in the visible and thermal spectrum.

On the other hand, within deep learning, some architectures allow the generation of synthetic data indistinguishable from the real ones, mainly based on three strategies: Variational Autoencoders (VAEs) [12–14], Generative Adversarial Networks (GANs) [15–18], Denoising Diffusion Probabilistic Models (DDPM) [19–22]. From the three types of architectures, GANs can generate more realistic images. VAEs, on the other hand, are good at learning latent, even though their resulting images are blurry. Finally, DDPMs are much slower due to their Markov process.

In this work, we propose using thermal images to improve the performance of facial recognition systems based on deep learning approaches. The objective of the work is to enhance the performance of facial recognition systems by generating synthetic thermal images to build an extensive database of thermal images of faces and thus train face recognition systems. As an input to the system, we use visible images manipulated to obtain different characteristics. The visible image is manipulated using StyleCLIP [23] to get various characteristics in the image, such as frowns, smiles, etc. Then, we use the GANs N' Roses (GNR) [24] model to generate thermal images. This model can generate thermal images from visible images by learning a map of a content code from visible images and adding a randomly chosen style (e.g., temperature, beard, glasses, hairstyles, etc.). Thus, the model can generate very diverse outputs. To control the different styles in the thermal images, we built classifiers to obtain desired features used to create the synthetic thermal face database. We used the database to train a robust Face Recognition model called FaceNet [8]. FaceNet is a neural network to efficiently recognize faces using image embedding by optimizing the distance between embeddings. FaceNet minimizes the distance between images of the same person and maximizes it if the images are of different people, thus obtaining a robust comparator model.

The contributions that we highlight from the manuscript are as follows. First, from the work presented here, we created an extensive database of thermal images, which

developers can use to train deep learning models for face detection and face recognition. Second, we present a generation system for the automatic creation of thermal images without the need for human supervision. Third, we describe a robust facial recognition system (Thermal-FaceNet) that operates automatically in the thermal range (8–12 μm). Accordingly, we use our synthetic thermal database to perform face recognition tests using FaceNet to analyze the fidelity of artificially created thermal faces.

2. Related Work

This section presents the state of the art of image generation, latent space manipulation, and face recognition, which allow us to explain the Deep learning models used in this article.

2.1. Image Generation

Generative Adversarial Networks (GAN) were introduced in 2014 by Goodfellow et al. [14]. GANs have achieved impressive results in computer vision [17,18,25], image-to-image translation [26,27], inpainting [28,29], and image segmentation [30,31]. Facial manipulation tasks have continuously gained attention in recent years due to the high demand for facial editing applications. Facial manipulation classifies as latent space manipulation, which allows working with unique features of the face.

GAN works using a probability distribution for the noise input $p_z(z)$ that will allow us to learn the distribution of generator p_g of data x , which represents a mapping of the data space $G(z; \theta_g)$. G is a differentiable function obtained from a multilayer perceptron with parameter θ_g . In addition, another neural network $D(x; \theta_d)$ is defined: a discriminator that allows classification between two classes, obtaining a scalar output. $D(x)$ represents the probability that x is derived from real data and is not obtained from synthetic data p_g . We want to train the discriminator D to maximize the probability of assigning a correct label for real training examples and examples provided by generator G . Simultaneously, G is trained to minimize $\log(1 - D(G(z)))$ and mislead the discriminator. Therefore, both generator G and discriminator D compete in a game referred to as a “two-player minimax game”, that is, G must be minimized, and D must be maximized in the following function (Equation (1)):

$$\min_G \max_D V(D, G) = \mathbb{E}_{x \sim p_{\text{data}}(x)} [\log D(x)] + \mathbb{E}_{z \sim p_z(z)} [\log(1 - D(G(z)))] \quad (1)$$

Unlike GAN, Variational Autoencoder (VAE) and Denoising Diffusion Probabilistic Models (DDPM) offer diversity rather than quality. New variants have shown promising results [32]; the very deep VAES (VDVAEs) architecture shows to be the first VAE that works in tandem with autoregressive image models in which its sampling rate matches that of a GAN. VQGAN [33], which combines VQVAE using transformer models and a GAN discriminator, was proposed to outperform BigGAN [25] in terms of quality. Researchers at OpenAI have presented incredible results [4], by using the transformer architecture, the text and image inputs can be combined using a discrete VAE. The authors trained the algorithm with 250 million pairs of text and images to achieve unprecedented diversity and rendering quality. Furthermore, new works with probabilistic diffusion models have shown comparable results to GANs. In [22], the authors show how DDPMs are improved by simple modifications in the algorithm, obtaining high-quality results.

2.2. Latent Space Manipulation

There are several algorithms for the manipulation of the latent space. In general terms, GANs have inputs called latent space. By modifying the latent space, it is possible to identify different directions that generate specific images, such as the exact content of a scene but with changes in texture, color, or other attributes.

There are several methods for supervised control to untangle the space from the GAN, e.g., ref. [34] uses labels to train for different features, and [35] codes the semantics in the latent space. On the other hand, some unsupervised methods try to determine what the

GAN space has learned by asking the same model to show what it has learned. In addition, there are new important advances with StyleCLIP [23] which allows modifying the latent space of the GAN, using text as the input. The model uses a GAN called StyleGAN2 [17], which is guided using a text prompt that is processed using the CLIP model (Contrastive Language–Image Pre-training) [36]. CLIP has a loss function that uses a constructive measure between the visual characteristics of the image and the characteristics obtained from natural language. Other models propose similar ideas, such as DALL-E [4], a neural network that generates images from text and is the natural extension of GPT-3 [37], a neural network capable of generating text similar to what humans do. DALL-E is a 12 billion parameter neural network trained using a paired text and image data set. The authors have processed images at 256×256 resolution during training, after which they used a discrete VAE [32] to compress the images to 32×32 grids (discrete latent codes), such as in VQVAE [33]. The text is encoded by transformers resulting in a resolution of 256 tokens that are concatenated with the 1024 (32×32 grid) tokens from images.

2.3. Face Recognition

On the other hand, the automatic recognition of human faces in unconstrained environments has attracted increasing interest in the research community in recent years. Unconstrained environments mean real-world conditions that include natural variations in illumination, different facial expressions, poses, accessories, and occlusions, i.e., no restrictions on environmental conditions. Today, most systems use face recognition algorithms based on the visible spectrum, which is not suitable for proper recognition in an uncontrolled environment. The visible light systems use reflected visible light for capturing images. When lighting conditions are poor, the system performance decreases, mainly due to the dependence on the ambient lighting conditions and variations in the face pose and facial expression changes.

Neural networks have been used for face recognition using Deep Learning approaches based mainly on Convolutional Neural Networks (CNN). Deep learning pre-trained models are mainly employed for prediction, feature extraction, and fine-tuning in face recognition. The following pre-trained models stand out: the VGG16 [38], InceptionResNetV2 [6], InceptionV3 [39], MobileNetV2 [40], DenseNet121 [41], and Xception [42] models, which are applied for feature extraction. Currently, Deep Learning-based face recognition algorithms, such as FaceNet [8], SpheroFace [43], VGGface2 [44], and ArcFace [9], use complex convolutional network architectures based mainly on ResNet architecture. They obtain high recognition rates (>99% accuracy) for standard databases, e.g., Labeled Faces in the Wild database (LFW), commonly used in state-of-the-art methods. Additionally, the recent use of vision transformers-based architectures [6] gets results comparable to CNN [45].

3. Visible and Thermal Face Database

In the literature, there are few thermal databases available that are public and also show variability in thermal images. Thermal databases generally present images acquired in a single session, contain few individuals, and have low resolution. We need a database with visible and thermal images with different characteristics and good resolution for our system. Therefore, we have selected the PUCV-VisibleThermal-Face (PUCV-VTF) database [46]. The PUCV-VTF database contains thermal and visible images acquired with two different sensors and allows us to study the fusion of these images. We acquired visible images with a 640×480 pixels PS3Eye camera, while we acquired the thermal images using a FLIR Tau 2640 camera with 640×512 14-bit resolution. The database includes 76 individuals with five subsets, in which each subset has several images with different attributes. This gives a total of 12,160 images in both visible and thermal spectra. In addition, we added thermal images of 46 individuals to the database, obtained with the same thermal and visible camera, corresponding to images not included in the PUCV-VTF database. Therefore, the total number of images in this database is 15,668 images. Figure 2 shows some examples.

Figure 2. Some examples of the PUCV-VTF database. (**top**) visible images, (**bottom**) thermal images.

All the database's images, visible and thermal, are aligned to the eyes' position and cropped to 256×256 pixels, obtaining only the face. A visible face detector called multi-task cascaded convolutional networks, MTCNN [47], was used to get the bounding box coordinates. When observing the visible images of the PUCV-VTF database, their noise can be noticed, which can be detrimental to the training of our proposed generation model. The Deep Face Dictionary Network, DFDNet [48], is used to eliminate the noise, which corresponds to a deep face model that operates with a network of dictionaries to perform face restoration.

DFDNet consists of two stages. Initially, it generates the deep components of the dictionaries from the high-definition image database using k-means. The model uses these generated dictionaries as candidate reference components. In the second stage, the model selects the dictionary with the most similar structure to the input for each degraded component. Each dictionary is then normalized using component AdaIN [49], based on the input components to eliminate style diversity in the distribution. The restoration process uses the selected dictionaries to guide the dictionary transform method. Then a confidence score is used in the chosen dictionary to generalize different levels of degradation through the fusion of the weighted value. Our experiments used 15,668 visible images to train the model and remove noise. Figure 3 shows some examples of the application of the DFDNet algorithm.

Figure 3. Results of the application of the DFDNet to the PUCV-VTF database. **(top)** original visible images; **(bottom)** restoration results of DFDNet.

4. Thermal Face Generation System

We describe below the proposed system for generating thermal images from visible ones. It is employed to create an extensive database for training the FaceNet model. In general terms, the system uses a visible image as input and a text prompt that feeds the attribute to assign to the thermal image at the output. For example, let's consider a visible image of a person and a text with the word "smile". In the first stage, the StyleCLIP algorithm oversees manipulating the visible image using the text, turning the person in the image into a smiling person. The second stage uses the GANs N' Roses (GNR) model to perform the visible-to-thermal image transformation. Since the GNR model allows generating diverse thermal images, we use classifiers trained with six different styles (frown, glasses, rotation, normal, vocal, and smile) to select the seeds for generating such thermal images. This workflow is presented in Figure 4. The proposed system explanation is presented in the following subsections.

Figure 4. Workflow of the proposed Thermal Face Generation System.

4.1. StyleCLIP

StyleCLIP [23] can modify visible images through a scheme that uses the CLIP loss function as a base to optimize a latent input space of the StyleGAN2 [17] generator. CLIP is a powerful algorithm to connect the text with images in a contrastive way. The general idea of StyleCLIP is to guide the source latent space w_s of a GAN generator from a text prompt, which is done by three methods: latent optimizer, latent mapper, global addresses. The loss function (Equation (2)) used for StyleCLIP uses three components: (i) the loss of CLIP, D_{CLIP} , which is calculated by measuring the cosine similarity between encoded text “ t ” and the encoded image; (ii) the loss of similarity, $\lambda_{L2} \|w - w_s\|_2$, between the output and the input, to maintain a percentage of similarity with the input image; (iii) the identity loss, $\lambda_{ID} \mathcal{L}_{ID}(w)$, based on a pre-trained ArcFace network [9], $R(G(w))$, for facial recognition, which is computed by using the cosine similarity $\langle \cdot, \cdot \rangle$. The values λ_{L2} and λ_{ID} can be chosen in dependence on the desired output. By solving the optimization problem by backpropagation using the fixed StyleGAN2 generator $G(w)$ and the CLIP image encoder it is possible to obtain the desired changes. Figure 5 shows examples where visible images have been modified.

$$\begin{aligned} \mathcal{L}(w) &= D_{\text{CLIP}}(G(w), t) + \lambda_{L2} \|w - w_s\|_2 + \lambda_{ID} \mathcal{L}_{ID}(w) \\ \mathcal{L}_{ID} &= 1 - \langle R(G(w_s)), R(G(w)) \rangle \end{aligned} \quad (2)$$

Figure 5. Some examples of StyleCLIP manipulations. The left image shows the input image, the other images represent de manipulation by the text prompt.

4.2. GANsN’Roses (GNR)

GNR [24] is a model for converting one style to another (image-to-image translation) using GAN. This model is characterized by separating style and content from the generated image. GNR can produce a wide range of output images, thermal images in our case, with a single content code. The result is diverse and represents a thermal image made from a single visible image. GNR is a multimodal image-to-image translation framework that uses a simple formalization of the maps using style and content. The content is the set of elements that change when the facial images are subjected to a family of data augmentation transformations. In contrast, the style corresponds to the elements that do not change. Therefore, the content affects the face parts’ locations in the image, and the style affects how the face parts are rendered. Accordingly, it is possible to learn a mapping from face images to content codes with this definition. Figure 6 shows a diagram of the generator and discriminator used by GNR.

Figure 6. GANs N' Roses framework applied to thermal face image generation. Based on a figure from [24].

Figure 6 shows how the GNR algorithm works. Let's consider a visible image to apply data augmentation, obtaining multiple images with the same style. These images are encoded using the E encoder, making it possible to get encoded vectors with the content information c_A and style s_A of the input visible image. Subsequently, the content vector c_A is decoded together with a random style s_B , obtaining multiple thermal images with variations in content. These thermal images are then re-encoded using only the c_B content. Combining the thermal content c_B with the original style generates the reconstruction of the visible image using another decoder. The idea of GNR is to keep the content vector throughout the training, and for the reconstructed image, to be as close to the initial visible image as possible, giving different styles to the output thermal image. The discriminator in its penultimate layer calculates the standard deviation of the minibatch and passes it to another network of the fully connected type. This means that the discriminator can find the diversity differences without batch variations, assuring content diversity for the samples generated.

The GRN training employs the loss function shown in Equation (3). This loss is composed of three-loss functions. The first corresponds to the style-consistency loss function ($\mathcal{L}_{scon}$), which keeps the style invariant. The second is the cycle-consistency loss function ($\mathcal{L}_{cyc}$), which aims at maintaining the initial content code by calculating the distance measured between the reconstructed image and the initial image. The third component corresponds to the adversary loss function ($\mathcal{L}_{adv}$), which uses $R1$ regularization [50] in the discriminators. The values λ_{scon} , λ_{cyc} , and λ_{adv} correspond to parameters used to improve the model optimization. We used the following parameters to train the model: $\lambda_{scon} = 10$, $\lambda_{cyc} = 20$, $\lambda_{adv} = 1$.

$$\mathcal{L} = \lambda_{scon}\mathcal{L}_{scon} + \lambda_{cyc}\mathcal{L}_{cyc} + \lambda_{adv}\mathcal{L}_{adv} \quad (3)$$

$$\mathcal{L}_{cyc} = \mathbb{E}_x[||\hat{x}_i - x_i||_2]$$

$$\mathcal{L}_{\text{scon}} = \text{Var}(s)$$

4.3. Thermal Classifier and Comparator Module

After obtaining the thermal image generated by the GNR model, we must select which characteristics we want for our synthetic database. Therefore, we use three deep neural networks models called Inception ResnetV2, VGG16, and InceptionV3. Inception ResnetV2 is a model characterized by its residual connections, which allows it to solve the fading of the gradient. The main idea is the jump of connections, which allows the activation from a layer to a future or deeper layer. VGG16 is a convolutional neural network model that reaches 92.7% accuracy having the fifth-best test accuracy score in ImageNet, a database of more than 14 million images with 1000 classes. InceptionV3 is characterized by having multiple filters of different sizes in the same layer or level. This neural network solves the bottleneck action in some convolutional neural networks. In addition, it incorporates new factoring methods to make training computationally more efficient. In our case, we have trained all models to classify six attributes: frown, glasses, rotation, normal, vocal, and smile.

The seed to generate the synthetic thermal images comes from the comparator module. With the information from the thermal classifier, it is possible to know the class of the thermal image obtained from the GNR model. This image is contrasted with the text prompt used in StyleCLIP. If the label received from the classifier and the text prompt is identical, that seed is chosen to generate different thermal images of the same person obtained with the seed. Then, we selected several persons using the six chosen attributes to generate the synthetic database using this principle.

4.4. Results of the Proposed System

Regarding the proposed system, the StyleCLIP model cannot be used directly with thermal images because the CLIP algorithm was trained with visible images and text. On the other hand, the GNR model adds thermal styles randomly, so it is necessary to use classifiers to select the images we want in our database. Thus, to create our proposal it is necessary to train the GNR model with visible and thermal images and train the thermal classifiers.

The image generation model, GNR, was trained with the PUCV-VTF database, using 15,668 visible and thermal images. To evaluate the result obtained by GNR, two metrics, *FID* [51] and LPIPS [52], are used. Fréchet Inception Distance (*FID*) is a metric of the differences in the density of two distributions in the high-dimensional feature space of an InceptionV3 classifier, comparing activations of a previously trained classification network on real and generated images. To calculate the *FID*, Equation (4) is used:

$$FID = \|m - m_w\|_2^2 + \text{Tr} \left(C + C_w - 2 (CC_w)^{1/2} \right) \quad (4)$$

The Wasserstein-2 distance is combined with the low-dimensional embedded vector from the network Inception V3. The m and C parameters are the mean vectors and covariance matrices in the embedding space. The subscript w corresponds to the generated image, while the terms without subscripts indicate the real image. A low *FID* value shows a better generation of synthetic images.

Learned Perceptual Image Patch Similarity, LPIPS is a metric that calculates the perceptual similarity between two images using deep network activations. A small LPIPS score indicates mode collapse and lack of diversity when varying style codes. In our case, we generated ten output images for each test image, from which we calculated their LPIPS distance. Then, we averaged all distances for all test images.

To compare the performance of our thermal generation imaging system, we have selected the Pix2pix [53] model. Pix2pix is a conditional GAN designed for image-to-image translation; therefore, thermal images are generated from visible images (as our proposed system does). The Pix2pix model is characterized by a U-Net type generator [54] and a

PatchGAN type discriminator [55]. We used the same database (PUCV-VTF) as the GNR model to train this model. In addition, we used 100 epochs and the parameters of the original Pix2pix article [53] for the training process.

This way, Table 1 presents the results obtained for the GNR model and Pix2pix. Note that the *FID* is lower for the GNR model while for Pix2pix, the *FID* is higher; this implies that the translation from visible images to thermal images is inadequate for the Pix2pix model. The LPIPS value is not so low for both models, which indicates that the generator has diversity when generating thermal images.

Table 1. GNR and Pix2pix training results.

Model	<i>FID</i>	LPIPS
GNR	65.71	0.34
Pix2pix [53]	195.88	0.38

We trained the thermal classifiers to select six attributes (frown, glasses, normal, rotation, smile, and vocal) to generate the synthetic database. We used 1644 images for each class for training these models, obtained from the PUCV-VTF database. Table 2 shows the result of classifying a test set, which contains 1567 thermal images never seen before by the models. We can highlight the InceptionResnetV2 method with a performance of 0.9827 of accuracy. This classifier will be used to build our automatic thermal imaging system.

Table 2. Thermal classifier results.

Models	Test Accuracy (%)
InceptionResnetV2	98.27
VGG16	97.76
InceptionV3	97.16

Finally, we conducted a thermal imaging generation experiment using the proposed model. As a result of the iterative process described in Figure 4, a database composed of 103,137 thermal images was obtained, corresponding to 3327 different subjects, with 31 images for each person in the synthetic database. The processing time used to generate the thermal database was 7 h and 30 min. Figure 7 presents the results of some images generated with the proposed system. Note that the results obtained are diverse; they show different subjects with the six selected attributes. We implemented our system in a computer running the Linux operating system, an Intel Xeon CPU at 2.30 GHz, 13 GB of RAM, and a GPU Tesla K80 with 24 GB of memory. We implemented the generation system in Python 3.8 from the Python Software Foundation (<http://www.python.org>) and the Open Source Machine Learning framework PyTorch 1.8.0 (<http://pytorch.org>).

Figure 7. Examples of synthetic thermal face images obtained with the proposed system.

5. Thermal-FaceNet

This section implements the FaceNet model to perform face recognition with the synthetic thermal images obtained with our generation model. However, as FaceNet works with visible images, it must be trained using thermal images to create Thermal-FaceNet. We have more than 100k thermal images to carry out this task. FaceNet is a deep neural network system capable of performing facial recognition tasks using image embedding. This embedding allows representing the face to be classified using a distance comparison or classifiers. FaceNet training is based on three image L2 distance optimization: the anchor, positive and negative. The FaceNet loss function (Equation (5)) is called triple loss, and it is composed of a function $f(x)$ that refers to the embedding of the image, α which represents the margin between positive and negative images. The prefixes “a, p, n” refer to “anchor, positive and negative”. Given an image x , we want to minimize the loss function in relation to the distance between the anchor and the positive image. We also want to maximize the distance between the anchor and the negative image. FaceNet ensures that the images of different people are kept further apart while the images of the same person are closer in the embedding space.

$$\sum_i^N \left[\|f(x_i^a) - f(x_i^p)\|_2^2 - \|f(x_i^a) - f(x_i^n)\|_2^2 + \alpha \right] \quad (5)$$

FaceNet uses two types of CNNs, one called the Zeiler and Fergus architecture [56] and the other based on Google’s Inception models [57]. Zeiler and Fergus can see the internal CNN process, which allows the visualization of layers and operations. The inception model uses multiple convolutions simultaneously in parallel and concatenated. This work uses

the implementation proposed in the original article [8] in TensorFlow but retraining the system using thermal imaging.

The face recognition experiment consists of training a FaceNet system with thermal images to correctly recognize the faces of every subject in the database. The thermal database was divided into two sets, one for the training and the other to perform a test, with data not used in the training phase. The size for the training database was 97,991 images, while for the test, it was 5146 thermal images. All the images are squares of size 256×256 pixels, and the number of classes was 3327. Table 3 shows the parameters used to train the Thermal-FaceNet system.

Table 3. Face recognition experiment parameters.

Parameter	Value
Learning rate decay factor	1.0
Epochs	20
Steps	1000
Learning rate decay epochs	90
Embedding size	512
Optimizer	Adam
Loss	Categorical Crossentropy

We use the validation rate to evaluate the model, such as in FaceNet [8]. For the classification of the images, a squared L2 distance threshold $D(x_i, x_j)$ is used with a pair of two faces. All face pairs (i, j) of the same identity are denoted with P_{same} . We can define $TA(d)$ as all the face pairs that were correctly classified as the same as threshold d . Thus, the validation rate $VAL(d)$ is defined in Equation (6) as:

$$VAL(d) = \frac{|TA(d)|}{|P_{same}|} \quad (6)$$

Figure 8 shows the model's training results, where the convergence of the model for the accuracy and loss is appreciated. We use the $VAL(d)$ as the system accuracy. Note that the model is capable of learning, and, after about 5000 steps, the model has already reached its convergence, obtaining a high value of accuracy, almost 100% accuracy.

Figure 8. Thermal-FaceNet model training charts for accuracy and loss.

We tested the already-trained Thermal-FaceNet model with a set of 5146 thermal images of 256×256 pixels. The results are presented in Table 4, showing the excellent results obtained by Thermal-FaceNet, allowing robust recognition of people's faces from the created database. The accuracy of the model is almost 100%.

Table 4. Thermal-FaceNet accuracy for synthetic thermal faces.

Face Recognition System	Accuracy
Thermal-FaceNet	99.86%

The following experiment aimed to evaluate the effectiveness of the ThermalFaceNet with real images (non-synthetic). For this experiment, we used the images from the PUCV-VTF database, 59 subjects randomly selected, and the UCHThermalFace database [58]. The UCHThermalFace database has 102 subjects and six images per person, showing variations in gestures and facial expressions. We use the Yolov3 implementation to detect thermal faces [59] directly. We cropped the thermal faces to 256×256 . Some examples are presented in Figure 9.

Figure 9. Real thermal images. (top) PUCV-VTF samples; (bottom) UCHThermalFace samples.

We used two images per person in the gallery to compare the FaceNet embeddings with the test images for the experiment. Table 5 shows the results from Thermal-FaceNet, which obtained excellent performance for both the PUCV-VTF and the UCHThermalFace, with 98% and 99% accuracy, respectively. This result indicates that it is possible to use generative models for training facial recognition models and then apply them effectively with real thermal images.

Table 5. Thermal-FaceNet results for real thermal images (non-synthetic).

Database	Subjects	Accuracy
PUCV-VTF	59	97.99%
UCHThermalFace [58]	102	99.33%

6. Conclusions

This article proposes using generative models to create thermal images and build a massive image database to train face recognition models. As a result of the proposed generator system, a database of 103,137 synthetic thermal images were obtained, with 3327 different subjects. When training the Thermal-FaceNet model with synthetic images, we got a 99.98% accuracy over the test set, which shows the relevance of using generative models to train Deep Learning models.

We built the proposed thermal image generation system using several deep learning modules for successfully creating thermal images. It is worth noting that the input to our system consists only of a visible image and a prompt text; through the StyleCLIP module,

it was possible to modify the visible input images through the CLIP algorithm. Once we modified the image, the GNR algorithm transferred the content of the visible spectrum to the thermal one. Accordingly, a trained GNR model with a visible and thermal images database called PUCV-VTF was required to carry out this transformation. The output of the proposed system was the abovementioned database of 100k images after 7 h and 30 min of processing. We used the synthetic database to train FaceNet, resulting in a new model called Thermal-FaceNet, achieving high performance in the evaluation.

Deep generative models allowed us to train new Deep Learning models reliably. Then, we obtained a robust system, Thermal FaceNet, that can recognize more than 3000 different subjects with an accuracy of almost 100%. Furthermore, we obtained high accuracy when testing our method with real thermal images, showing the effectiveness of the proposed system.

Although the results obtained by our thermal generation imaging model are suitable for training thermal face recognition models, we believe that the latent space manipulation process should be done directly with a thermal generator and the CLIP model as a guide. Therefore, as future work, we want to implement a version of Thermal-CLIP, trained with thermal images and text, to guide the latent space in a more efficient way and without using visible images, only thermal generators.

Author Contributions: Investigation, V.P. and G.H.; Software, V.P.; Supervision, G.H.; Writing—original draft, V.P.; Writing—review & editing, G.H., F.P., S.F. and D.Y. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported in part by PUCV under Grant COD. PROYECTO: 039.381/2021.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

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

References

1. Jumper, J.; Evans, R.; Pritzel, A.; Green, T.; Figurnov, M.; Ronneberger, O.; Tunyasuvunakool, K.; Bates, R.; Žídek, A.; Potapenko, A.; et al. Highly accurate protein structure prediction with AlphaFold. *Nature* **2021**, *596*, 583–589. [\[CrossRef\]](#)
2. Ye, W.; Liu, S.; Kurutach, T.; Abbeel, P.; Gao, Y. Mastering Atari Games with Limited Data. *Adv. Neural Inf. Processing Syst.* **2021**, *34*, 1–13.
3. Schrittwieser, J.; Antonoglou, I.; Hubert, T.; Simonyan, K.; Sifre, L.; Schmitt, S.; Guez, A.; Lockhart, E.; Hassabis, D.; Graepel, T.; et al. Mastering atari, go, chess and shogi by planning with a learned model. *arXiv* **2019**, arXiv:1911.08265. [\[CrossRef\]](#)
4. Ramesh, A.; Pavlov, M.; Goh, G.; Gray, S.; Voss, C.; Radford, A.; Sutskever, I. Zero-Shot Text-to-Image Generation. *arXiv* **2021**, arXiv:2102.12092.
5. D’Angelo, G.; Palmieri, F. Knowledge elicitation based on genetic programming for non destructive testing of critical aerospace systems. *Future Gener. Comput. Syst.* **2020**, *102*, 633–642. [\[CrossRef\]](#)
6. Szegedy, C.; Ioffe, S.; Vanhoucke, V. Inception-v4, Inception-Resnet and the Impact of Residual Connections on Learning. In Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, San Francisco, CA, USA, 4–9 February 2017; Volume 7, pp. 4278–4284.
7. Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. An image is worth 16 × 16 words: Transformers for image recognition at scale. *arXiv* **2020**, arXiv:2010.11929.
8. Schroff, F.; Kalenichenko, D.; Philbin, J. Facenet: A unified embedding for face recognition and clustering. *arXiv* **2015**, arXiv:1503.03832.
9. Deng, J.; Guo, J.; Zafeiriou, S. ArcFace: Additive angular margin loss for deep face recognition. *arXiv* **2018**, arXiv:1801.07698.
10. Socolinsky, D.; Selinger, A. A Comparative Analysis of Face Recognition Performance with Visible and Thermal Infrared Imagery. In Proceedings of the International Conference on Pattern Recognition (ICPR), Quebec City, QC, Canada, 11–15 August 2002.
11. Selinger, A.; Socolinsky, D.A. *Appearance-Based Facial Recognition Using Visible and Thermal Imagery: A Comparative Study*; Equinox Corporation: New York, NY, USA, 2001.
12. Kingma, D.P.; Welling, M. Auto-encoding variational bayes. *arXiv* **2013**, arXiv:1312.6114.
13. Doersch, C. Tutorial on variational autoencoders. *arXiv* **2016**, arXiv:1606.05908.
14. Diederik, P.K.; Max, W. An Introduction to Variational Autoencoders. *arXiv* **2019**, arXiv:1906.02691.

15. Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; Bengio, Y. Generative Adversarial Networks (PDF). In Proceedings of the International Conference on Neural Information Processing Systems, Montreal, QC, Canada, 8–13 December 2014; pp. 2672–2680.
16. 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.
17. Karras, T.; Laine, S.; Aittala, M.; Hellsten, J.; Lehtinen, J.; Aila, T. Analyzing and improving the image quality of stylegan. *arXiv* **2019**, arXiv:1912.04958.
18. Karras, T.; Aittala, M.; Hellsten, J.; Laine, S.; Lehtinen, J.; Aila, T. Training generative adversarial networks with limited data. *arXiv* **2020**, arXiv:2006.06676v1.
19. Sohl-Dickstein, J.; Weiss, E.; Maheswaranathan, N.; Ganguli, S. Deep Unsupervised Learning using Nonequilibrium Thermodynamics. In Proceedings of the 32nd International Conference on Machine Learning (ICML), Lille, France, 6–11 July 2015.
20. HO, J.; Jain, A.; Abbeel, P. Denoising diffusion probabilistic models. *arXiv* **2020**, arXiv:2006.11239.
21. Nichol, A.; Dhariwal, P. Improved denoising diffusion probabilistic models. *arXiv* **2021**, arXiv:2102.09672.
22. Dhariwal, P.; Nichol, A. Diffusion Models Beat GANs on Image Synthesis. *arXiv* **2021**, arXiv:2105.05233.
23. Patashnik, O.; Wu, Z.; Shechtman, E.; Cohen-Or, D.; Lischinski, D. Styleclip: Text-driven manipulation of stylegan imagery. *arXiv* **2021**, arXiv:2103.17249.
24. Chong, M.J.; Forsyth, D. GANs N’Roses: Stable, Controllable, Diverse Image to Image Translation (works for videos too). *arXiv* **2021**, arXiv:2106.06561.
25. Brock, A.; Donahue, J.; Simonyan, K. Large scale gan training for high fidelity natural image synthesis. *arXiv* **2018**, arXiv:1809.11096.
26. Durall, R.; Pfrendt, F.-J.; Keuper, J. Local facial attribute transfer through inpainting. *arXiv* **2020**, arXiv:2002.03040.
27. Laxman, K.; Dubey, S.R.; Kalyan, B.; Kojjarapu, S.R.V. Efficient High-Resolution Image-to-Image Translation using Multi-Scale Gradient U-Net. *arXiv* **2021**, arXiv:2105.13067.
28. Jam, J.; Kendrick, C.; Drouard, V.; Walker, K.; Hsu, G.-S.; Yap, M.H. R-mnet: A Perceptual Adversarial Network for Image Inpainting. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, Waikola, HI, USA, 5–9 January 2021; pp. 2714–2723.
29. Yu, J.; Lin, Z.; Yang, J.; Shen, X.; Lu, X.; Huang, T.S. Free-form image inpainting with gated convolution. *arXiv* **2018**, arXiv:1806.03589.
30. Khan, K.; Mauro, M.; Leonardi, R. Multi-Class Semantic Segmentation of Faces. In Proceedings of the 2015 IEEE International Conference on Image Processing (ICIP), Quebec City, QC, Canada, 27–30 September 2015.
31. Kalayeh, M.M.; Gong, B.; Shah, M. Improving Facial Attribute Prediction using Semantic Segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 6942–6950.
32. Child, R. Very Deep VAEs Generalize Autoregressive Models and Can Outperform Them on Images. *arXiv* **2020**, arXiv:2011.10650.
33. Esser, P.; Rombach, R.; Ommer, B. Taming Transformers for High-Resolution Image Synthesis. *arXiv* **2020**, arXiv:2012.09841.
34. Nie, W.; Karras, T.; Garg, A.; Debhath, S.; Patney, A.; Patel, A.B.; Anandkumar, A. Semisupervised Stylegan for Disentanglement Learning. In Proceedings of the International Conference on Machine Learning (ICML), Vienna, Austria, 12–18 July 2020.
35. Shen, Y.; Yang, C.; Tang, X.; Zhou, B. Interfacegan: Interpreting the disentangled face representation learned by gans. *arXiv* **2020**, arXiv:2005.09635. [[CrossRef](#)] [[PubMed](#)]
36. Radford, A.; Kim, J.W.; Hallacy, C.; Ramesh, A.; Goh, G.; Agarwal, S.; Sastry, G.; Askell, A.; Mishkin, P.; Clark, J.; et al. Learning transferable visual models from natural language supervision. *arXiv* **2021**, arXiv:2103.00020.
37. Brown, T.B.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; et al. Language models are few-shot learners. *arXiv* **2020**, arXiv:2005.14165.
38. Simonyan, K.; Zisserman, A. Very Deep Convolutional Networks for Large-Scale Image Recognition. In Proceedings of the International Conference on Learning Representations (ICLR), San Diego, CA, USA, 7–9 May 2015.
39. Szegedy, C.; Vanhoucke, V.; Ioffe, S.; Shlens, J.; Wojna, Z. Rethinking the inception architecture for computer vision. *arXiv* **2015**, arXiv:1512.00567.
40. Sandler, M.; Howard, A.; Zhu, M.; Zhmoginov, A.; Chen, L.-C. Mobilenetv2: Inverted Residuals and Linear Bottlenecks. In Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–23 June 2018.
41. Huang, G.; Liu, Z.; Weinberger, K.Q.; Maaten, L. Densely Connected Convolutional Networks. In Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017.
42. Chollet, F. Xception: Deep learning with depthwise separable convolutions. *arXiv* **2016**, arXiv:1610.02357.
43. Liu, W.; Wen, Y.; Yu, Z.; Li, M.; Raj, B.; Song, L. Sphereface: Deep Hypersphere Embedding for Face Recognition. In Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017.
44. Cao, Q.; Shen, L.; Xie, W.; Parkhi, O.M.; Zisserman, A. Vggface2: A Dataset for Recognising Faces across Pose and Age. In Proceedings of the 13th IEEE International Conference on Automatic Face & Gesture Recognition (FG), Xi’an, China, 15–19 May 2018; pp. 67–74.
45. Zhong, Y.; Deng, W. Face Transformer for Recognition. *arXiv* **2021**, arXiv:2103.14803.

46. Hermosilla, G.; Gallardo, F.; Farias, G.; Martin, C.S. Fusion of visible and thermal descriptors using genetic algorithms for face recognition systems. *Sensors* **2015**, *15*, 17944–17962. [[CrossRef](#)] [[PubMed](#)]
47. Zhang, K.; Zhang, Z.; Li, Z.; Qiao, Y. Joint face detection and alignment using multi-task cascaded convolutional networks. *IEEE Signal Processing Lett.* **2016**, *23*, 1499–1503. [[CrossRef](#)]
48. Li, X.; Chen, C.; Zhou, S.; Lin, X.; Zuo, W.; Zhang, L. Blind Face Restoration via Deep Multi-Scale Component Dictionaries. In Proceedings of the European Conference on Computer Vision (ECCV), Glasgow, UK, 23–28 August 2020; Springer: Cham, Switzerland, 2020; pp. 399–415.
49. Huang, X.; Belongie, S. Arbitrary Style Transfer in Real-Time with Adaptive Instance Normalization. In Proceedings of the International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017.
50. Mescheder, L.; Geiger, A.; Nowozin, S. Which training methods for gans do actually converge? *arXiv* **2018**, arXiv:1801.04406.
51. Salimans, T.; Goodfellow, I.; Zaremba, W.; Cheung, V.; Radford, A.; Chen, X. Improved Techniques for Training Gans. In Proceedings of the Advances in Neural Information Processing Systems, Barcelona, Spain, 5–10 December 2016; pp. 2234–2242.
52. Zhang, R.; Isola, P.; Efros, A.A.; Shechtman, E.; Wang, O. The Unreasonable Effectiveness of Deep Features as a Perceptual Metric. In Proceedings of the Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–23 June 2018.
53. Isola, P.; Zhu, P.J.-Y.; Zhou, T.; Efros, A.A. Image-To-image Translation with Conditional Adversarial Networks. In Proceedings of the Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017.
54. Ronneberger, O.; Fischer, P.; Brox, T. U-net: Convolutional Networks for Biomedical Image Segmentation. In Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI), Munich, Germany, 5–9 October 2015; Springer: Cham, Switzerland, 2015; pp. 234–241.
55. Li, C.; Wand, M. Precomputed Real-Time Texture Synthesis with Markovian Generative Adversarial Networks. In Proceedings of the European Conference on Computer Vision (ECCV), Amsterdam, The Netherlands, 11–14 October 2016.
56. Zeiler, M.D.; Fergus, R. Visualizing and Understanding Convolutional Networks. In Proceedings of the European Conference on Computer Vision, Zurich, Switzerland, 6–12 September 2014; Springer: Cham, Switzerland, 2014; pp. 818–833.
57. Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; Rabinovich, A. Going Deeper with Convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 1–9.
58. Hermosilla, G.; Ruiz-del-Solar, J.; Verschae, R.; Correa, M. A comparative study of thermal face recognition methods in unconstrained environments. *Pattern Recognit* **2012**, *45*, 2445–2459. [[CrossRef](#)]
59. Hermosilla, G.; Tapia, D.-I.H.; Allende-Cid, H.; Castro, G.F.; Vera, E. Thermal Face Generation Using StyleGAN. *IEEE Access* **2021**, *9*, 80511–80523. [[CrossRef](#)]