Generative Adversarial Networks for Data Augmentation in Person Re-Identification

Laura Álvarez-González[1], Víctor Uc-Cetina[1*†], Anabel Martin-González[1†]

[1*]Facultad de Matemáticas, Universidad Autónoma de Yucatán, Periférico Norte, Mérida, 97000, Yucatán, Mexico.

*Corresponding author(s). E-mail(s): uccetina@correo.uady.mx;
Contributing authors: laura.alvargonza@gmail.com;
amarting@correo.uady.mx;
†These authors contributed equally to this work.

Abstract

People re-identification systems based on deep neural networks require as many training examples as possible. The possibility of automatically identifying a person in different images is highly relevant in various security surveillance applications. However, getting enough data to train these networks is sometimes problematic or not possible at all, which limits the quality of the re-identification model. One strategy to cope with this scarcity of data is through the generation of synthetic examples that can increase the size of our training dataset. In this article we propose a data augmentation methodology for re-identification systems, using generative adversarial networks. We provide empirical evidence showing that a StyleGAN model can be used to generate fake but useful images, when they are used to further train a re-identification network.

Keywords: person re-identification, generative adversarial models, data augmentation

1 Introduction

People re-identification is a technique used in the field of artificial intelligence and machine learning to recognize a person in different images or videos, even if they are in different angles, lighting or clothing. This technique is used in various applications,

such as security surveillance, people identification in digital images, and behavior analysis in videos. It is based on the use of machine learning models that learn to recognize the characteristics that identify a person under different circumstances. These models are trained on a image and video data sets of people, along with information about the characteristics that identify each person. The re-identification of people can be challenging due to the variability of the characteristics that identify a person, such as clothing, hairstyle and other aspects that can change their appearance. It can also be challenging if you have a limited amount of training data. To overcome these challenges, image and video pre-processing techniques can be used, as well as deep learning techniques that allow the model to adapt to variations in a person's appearance and recognize relevant features in low-quality images or videos.

Currently, the most prominent training data sets for people re-identification are very limited because they do not contain a large number of images. For example, Market1501 includes only 1,501 people recorded on 6 different cameras, while DukeMTMC-reID has 702 people on 8 different cameras. There are various challenges for the re-identification of people in images, such as low image resolution, variations in lighting and contrast, as well as other factors that complicate the task such as changes in clothing, presence of objects such as backpacks or sweaters, etc. Moreover, the presence of obstacles or people in the background limit the visibility of the person of interest in an open space.

This study investigates the use of a generative adversarial network, together with data augmentation techniques, to train a person re-identification model. The generative adversarial network is a type of machine learning model that is used to generate synthetic images that can be used as additional training data. Various techniques for extending training data, such as image generation and feature expansion, are discussed, and their effectiveness in training a person re-identification model using an adversarial generative network is evaluated.

2 Related work

Generative adversarial networks, originally proposed in 2014 by Ian Goodfellow et al. [1], are capable of artificially generating images with great diversity. Over time, new architectures have been proposed that improve the quality of the data generated, such as the CycleGan architecture, proposed in 2017 by Zhu et al. [2]. Apart from improving the quality of the generated images, it manages to transfer the style or domain of a group of images to another group, using two generative adversarial networks.

StyleGAN is a Generative Adversarial Network (GAN) architecture developed by NVIDIA in the year 2018 [3]. This architecture has been trained to be able to generate high quality images of non-existent people's faces. In this case, it was trained with the FFHQ database. which consists of images of faces of people from the Flickr social network. It uses a generative network structure based on layers of styles that allows you to independently control different aspects of the generated image, such as pose, facial expression, gender, etc.

The generator network consists of an encoder module and a generator module, the encoder module converts the input image into a style tensor, which is a vector of

fixed dimensions that represents the different aspects of the image. The styles tensor is used as input to the generator module, which is a generator network that uses a layered structure of styles to generate a synthetic image that approximates the input image. Once trained, the generator network can be used to generate high-quality synthetic images that approximate the images in the training data set. In addition, the layered structure of styles allows you to independently control different aspects of the generated images, such as pose, facial expression, gender, etc. This ability to control the generated images allows StyleGAN to be used in applications such as data augmentation for re-identification models.

Once trained, the generator network can be used to generate high-quality synthetic images that approximate the images in the training data set. The Generator is a multidimensional latent space and is shown as a mathematical space in which the input and output vectors of an artificial neural network are represented. This space is called latent because it is not directly observed, but inferred from the input and output observations. In this case, it is made up of 512 dimensions, and each of the positions corresponds to an image. By introducing a random numerical vector of size 512 as input into the Generator, it will generate an image of an artificial face. This means that each of the images that can be generated is composed of a latent vector of 512 positions.

Algorithms for re-identification of people have been proposed since 1996 [4]. Currently, the most widespread method is the use of a neural network as feature extractor. In recent years, there has been a great deal of interest in the use of adversarial generative networks in people re-identification models. In 2019, Hamed Alqahtani et al. [5] provided a detailed introduction to the state of the art in this field, describing different types of adversarial generative networks and 11 different architectures used for style transfer, labeled LSRO and the globalization of the model. Furthermore, Zhiyuan Luo et al. [6] focused exclusively on architectures that generate artificial images by switching styles between different cameras or databases. Finally, Yiqi Jiang et al. [7] conducted a detailed study on the quality of images generated by different adversarial generative network architectures in re-identification models, analyzing the details of these artificial images.

An important increment in the study of GANs for data augmentation in the training of re-identification models can be noted since 2018. The most relevant approaches can been grouped into three categories, corresponding to different methods used to generate new artificial images.

1. Style transfer [2, 8–12]. New images are artificially generated from an input image, using different styles, known also as domains. The styles are imposed on the new images at the moment they are generated by neural networks previously trained for that purpose. In the new generated images, you can see modifications with respect to the input image, such as color, tone, and lighting.
2. Pose transfer [13–17]. In this approach the inputs are one image of a real person and the target posture that we want to impose on that person. The posture can be specified whether as a heat map or by the joints that correspond to the skeleton of the desired posture. The model is capable to generate the image of the input person with a determined posture.

3. Random generation [18–23]. Methods in this category are less constrained and they focus on randomly generation of synthetic images with the only condition that the generated images should have similar characteristics of those images in the dataset that we want to augment.

3 Methods

In general, the operation of the re-identification model is simple. First, a database with tagged images of people is obtained through different security cameras. Second, the model is trained in a supervised manner so that it can classify a certain finite number of people from the training set. For example, using the Market 1501 database, we can train a model that is capable of identifying 751 people. Once the model has been trained, the following steps are followed:

1. Each one of the images is introduced into the model in order to obtain a vector of size 751. This vector represents the proportion of each of the 751 people that the input person contains.
2. A vector representing the image of the person we want to search for is generated. We measure the distance with each of the other vectors that represent the other images of people. Within the literature, the use of the cosine distance or Euclidean distance is a common practice.
3. A classification by shortest distance is made among all the images in the dataset. The least distance means that the image is more similar to the original and it is probably the same person.

In this article we propose a methodology that consists of four phases. Each of these phases is described next.

a) Training of the adversarial generative network StyleGAN3 (see Fig. 1).
b) Generation of multiple images of artificial people in different postures. Using an image of a real person from the database, we generate artificial images of that same person in different postures. Image filtering through automatic elimination of generated images that present noise or have been generated incorrectly (see Fig. 1).
c) Re-training of the re-identification model, which involves designing the architecture and training of the re-identification model (see Fig. 2).
d) Testing of the re-identification model (see Fig. 2).

For the generation of artificial images, the architecture of the adversarial generative network we used StyleGAN3 [23]. It is a generative image model developed by the Nvidia research team in 2021. It is an improved version of the StyleGAN2 model, which is characterized by its ability to generate high-quality and realistic images in a wide variety of image categories. StyleGAN3 uses a deep learning approach based on generators and discriminators. The generator is a neural network that is trained to generate images that are as realistic as possible. To do this, you are shown a set of real images and asked to generate images that resemble them. As it trains, the generator learns to extract relevant features from actual images and use them to generate images that are as realistic as possible.

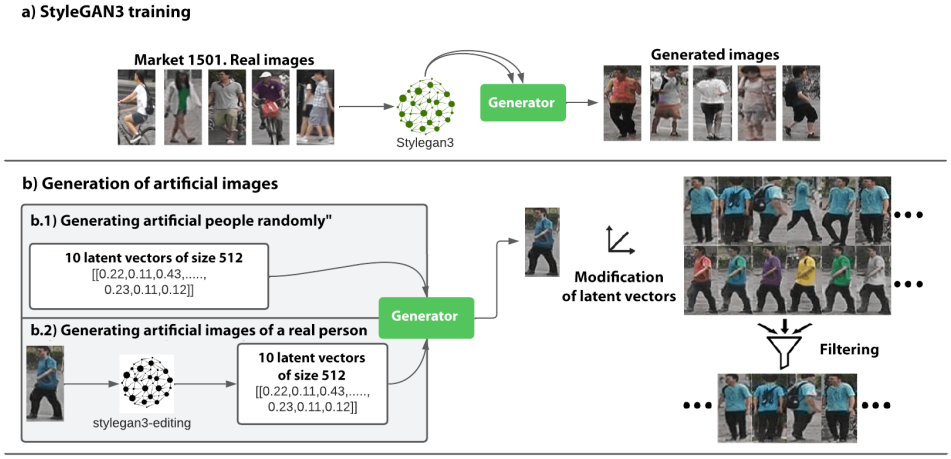

Fig. 1 Training of StyleGAN3 and generation of artificial images.

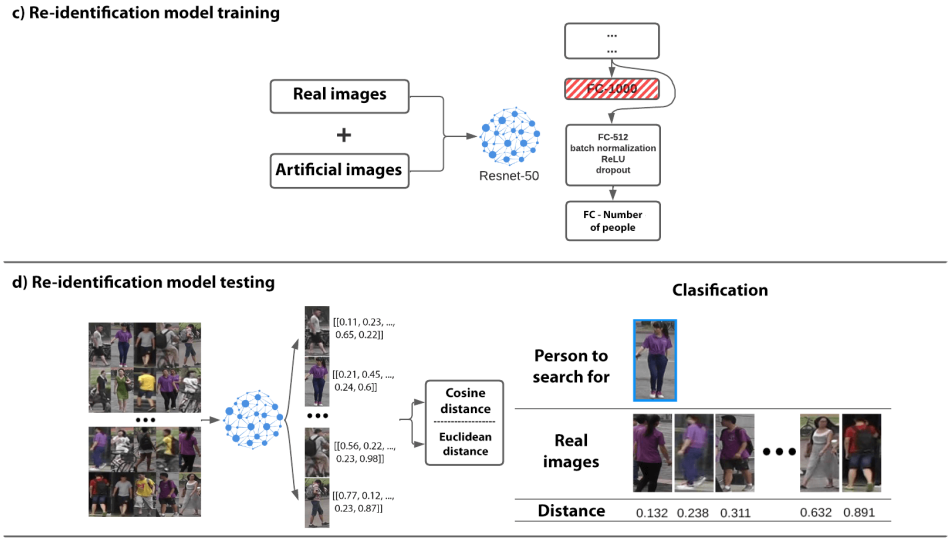

Fig. 2 Re-training and testing the re-identification model.

The discriminator is a neural network that is trained to distinguish between real and generated images. You are shown both real and generated images and are asked to determine which are real and which are generated. As it is trained, the discriminator learns to identify the features that differentiate real images from generated ones, and is used to guide training of the generator towards generating more realistic images.

StyleGAN is pre-trained with 25 million face images of which 70 thousand are real from the high quality FFHQ database of resolution 1024×1024 pixels and the rest were generated by the discriminator. Currently, StyleGAN3 works as a high-quality people face image generator. StyleGAN3 has been trained to generate faces and we will apply the transferred learning process to retrain it. For the re-training of StyleGAN3, we used the Market-1501 database, which consists of 51,247 images of 1,501 different people captured by six different cameras.

3.1 Evaluation of StyleGAN

To evaluate the performance of StyleGAN, we use the metric Fréchet inception distance (FID), proposed by P. Dimitrakopoulos et al. [24]. This distance metric is used to measure the similarity between two image distributions. The FID metric is based on the idea that the distance between two image distributions is the same as the distance between image features extracted from a deep neural network. To calculate the FID distance between two image distributions, features are first extracted from each distribution using a deep neural network, and then the distance between those features is calculated using the Fréchet distance. Mathematically, the FID distance between two image distributions can be calculated as follows:

$$\texttt{FID} = |\mu - \mu_w|^2 + \texttt{tr}(\sigma + \sigma_w - 2(\sigma\sigma_w)^{1/2}). \tag{1}$$

We compare the mean and the convariance matrix of the real and fictitious images, obtaining the data from one of the deepest layers of the neural network. It aims to mimic human perception to identify the similarity between two images using the discriminator as a feature extractor. If the returned value is zero, it indicates that the generated data and the actual data are identical, which means that the lower the returned value, the greater the similarity between the generated images and the actual images.

We implemented two ways to generate artificial images. The first one is totally random from artificial people. The second one involves taking a real image of a person and then we generate variations of it. By means of a random number, also known as a seed, the generator assigns a latent vector that corresponds to an image. To obtain variations of the original image, another random latent vector can be used, through another seed or through interpolations. In the different AdaIN layers of the model, also called a mixture of styles, the latent vector is modified and different variations of the original image are achieved. It is possible to achieve from a total change in the structure of the image to more subtle changes, such as changes in hue, lighting, colors, saturation, among others. Another way to generate variations of the original image is through the modification of the original latent vector through interpolations.

For the loss function, the Learned Perceptual Image Patch Similarity (LPIPS) metric is used, which will compare the real image with the one obtained in the generator. It is a measure of the distance or difference between two images in terms of their perceived similarity by a human observer. This metric is based on a pre-trained neural network called VGG-16, which has been designed to recognize patterns in images. The idea behind LPIPS is that if two images have a small LPIPS distance, then they

are perceived as similar by a human observer. Mathematically, the LPIPS metric is calculated as follows:

First, the VGG-16 network is used to extract a representation of each of the two images in question. This representation is called feature map and is a three-dimensional tensor that contains information about the visual features present in each image. We denote these feature maps as f_1 and f_2.

Next, the Euclidean distance between the two feature maps is calculated. This distance is interpreted as the perceived similarity between the two images. The smaller this distance, the more similar the images will be. The Euclidean distance between f_1 and f_2 is defined as:

$$\text{dist}(f_1, f_2) = |f_1 - f_2|_2 .$$

Once the latent vector of the real image is obtained within the latent space of StyleGAN3, new images are generated in the same way as in the previous case. Then, it is necessary to filter out noisy or distorted images, which may have been generated. To measure and discard the images generated by the adversary generative network, we use metrics based on the quality of the generated data. The first filter that is applied is YOLOv4 tiny, a model for the detection of pedestrians. Then, to measure the similarity of the generated images, we use the metric known as structural similarity index measure (SSIM).

3.2 YoloV4 tiny filtering

YOLOv4 tiny, a model proposed by Z. Jiang et al. [25], consists of a reduced version of the YOLOv4 model, whose purpose is to detect objects in images and videos. YOLO (You Only Look Once) is an object detection technique that stands out for its speed and precision. The tiny version of YOLOv4 is particularly useful for low-power devices, as it is less resource demanding and can run efficiently on mobile devices and low-performance computers. Generally speaking, YOLOv4 tiny uses a convolutional neural network to extract features from an image and then uses a combination of machine learning techniques to perform object detection.YOLOv4 tiny has been optimized to detect pedestrians with accuracy and speed comparable to that of other larger models, but with a lower load in terms of resources, making it an excellent choice for real-time applications on devices with limited capabilities.

3.3 SSIM filtering

The Structural SIMilarity (SSIM) metric is a measure of structural similarity between two images. The SSIM metric is often used to assess the quality of a processed image compared to an original image, and is calculated by comparing the structural features of both images. SSIM is based on the fact that the human perception of the quality of an image is based on its structural content, and not just on the pixel difference between two images. Therefore, the SSIM is used to measure the structural similarity between two images and give a score that reflects the quality perceived by a human observer.

To carry out the calculation of the SSIM metric, three structural characteristics of two images are compared: their mean intensity, their intensity variance and their

intensity covariance. The SSIM is obtained from the product of these three characteristics, and two images are considered to have high SSIM if they have similar mean intensity, similar intensity variance, and similar intensity covariance. The result is obtained from the product of these three characteristics, and is denoted as:

$$SSIM(x, y) = l(x, y) \cdot c(x, y) \cdot s(x, y), \tag{2}$$

where x and y are the two images being compared, $l(x, y)$ is the similarity in mean intensity, $c(x, y)$ is the similarity in intensity covariance and $s(x, y)$ is the similarity in intensity variance.

The similarity in mean intensity is calculated as:

$$l(x, y) = \frac{2 \cdot \mu_x \cdot \mu_y + C_1}{\mu_x^2 + \mu_y^2 + C_1}, \tag{3}$$

where μx and μy are the mean intensities of the images x and y, respectively, and C_1 is a constant which is used to avoid division by zero.

The similarity in intensity covariance is calculated as:

$$c(x, y) = \frac{2 \cdot \sigma xy + C_2}{\sigma x^2 + \sigma y^2 + C_2}, \tag{4}$$

where σxy is the intensity covariance between the images x and y, σx and σy are the intensity variances of the images x and y, respectively, and C_2 is a constant used to avoid division by zero.

The similarity in intensity variance is calculated as:

$$s(x, y) = \frac{2 \cdot \sigma x \cdot \sigma y + C_3}{\sigma x^2 + \sigma y^2 + C_3}, \tag{5}$$

where σ_{xy} is the intensity covariance between images x and y, σ_x and σ_y are the intensity variances of images x and y, respectively , and C_3 is a constant used to avoid division by zero.

In summary, the SSIM metric is calculated by comparing the mean intensities, intensity covariances, and intensity variances of two images. The SSIM is obtained as the product of the similarity in each of these characteristics, and is used to evaluate the quality of a processed image in comparison with an original image. It is based on the comparison of the structural characteristics of both images. The higher the result, the more variation there will be in the generated images.

$$\texttt{SSIM}(x, y) = \frac{(2\mu_x\mu_y + c_1)(2\sigma xy + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma x^2 + \sigma y^2 + c_2)}. \tag{6}$$

3.4 Re-identification model

A re-identification model is an algorithm used in image processing and artificial intelligence that makes it possible to identify and track objects or people in a sequence of images. These models are based on the comparison of visual characteristics between different images to determine if it is the same object or person. Mathematically, a re-identification model uses a similarity function to calculate the similarity between two

images. This function takes two visual feature vectors (one from the reference image and the other from the image to be compared) and returns a value indicating the similarity between the two images. If the value returned by the similarity function exceeds a certain threshold, it is determined that the images correspond to the same object or person. To compute the visual feature vectors, the model uses a neural network that has been pre-trained on a dataset of labeled images. The neural network extracts relevant features from the images and groups them into a feature vector. These vectors are then used in the similarity function to determine the similarity between the images.

We used Resnet50 convolutional neural network, in which the last layer is modified so that the output adapts to the number of people with whom the model will be trained. During training, the cross-entropy loss is used as the loss function. This function is defined as:

$$L(y, \hat{y}) = -\frac{1}{N} \sum_{i=1}^{N} y_i \log \hat{y}_i.$$

In this equation, y represents the actual label or desired value of the output, $\hat{y}$ represents the output predicted by the model, and N is the number of examples in the data set.

Cross entropy is a loss function commonly used in classification problems, where the model output is interpreted as the probability that an instance belongs to each class. The idea behind cross-entropy is that if the model output is a good approximation of the true probability distribution, then the cross-entropy loss function will have a low value. In contrast, if the model output is very different from the true probability distribution, then the cross-entropy loss function will have a high value. It is used to measure how well the model is making predictions about the actual probability distribution of the classes. During model training, the loss function is optimized to improve its prediction accuracy. Once the training is finished, the model works as a feature extractor and the images are classified.

We introduce all the images to be evaluated one by one into the model to obtain their respective feature vectors and to classify which images are of the same person, each of the image vectors is compared with the vector of the original image using the cosine distance. It is a measure of similarity between two vectors in a vector space. This measure is calculated using the cosine of the angle between the two vectors and can be interpreted as the projection of the shorter vector onto the longer vector. The cosine distance between two vectors $\mathbf{a}$ and $\mathbf{b}$ is calculated as follows:

$$d_c(\mathbf{a}, \mathbf{b}) = 1 - \frac{\mathbf{a} \cdot \mathbf{b}}{|\mathbf{a}||\mathbf{b}|}.$$

In this equation, $\mathbf{a} \cdot \mathbf{b}$ is the dot product of the vectors $\mathbf{a}$ and $\mathbf{b}$, and $|\mathbf{a}|$ and $|\mathbf{b}|$ are the norms of the vectors $\mathbf{a}$ and $\mathbf{b}$, respectively. The cosine distance has a value between 0 and 1, where a value closer to 1 indicates a greater similarity between the vectors $\mathbf{a}$ and $\mathbf{b}$, and a value closer to 0 indicates a greater similarity. less similarity between them. After having obtained the cosine distance of all the images, they are ordered and those with the smallest cosine distance will be the closest to the original image, that is, they will be those that have been detected as images of the same person.

4 Results

The adversarial generative network StyleGAN3 [23] has been used for the generation of artificial images. The model is pre-trained with 25 million face images of which 70 thousand are real from the Flickr-Faces-HQ Dataset (FFHQ) of high quality resolution 512×512 pixels and the rest were generated using the discriminator. Training StyleGAN3 consisted required transfer learning and subsequent re-training with 51,247 images from the Market-1501 database. To measure training performance, the Fréchet Inception Distance [26] (FID) metric was used, which was applied to both generated and real images. The closer the value of both is, the better the generation of images will have been. In general, the performance of StyleGAN3 is much higher than that of other generative adversarial networks trained with the same database. To generate the variations of the images of the same person, the mixture of styles was used on the model trained with Market-1501.

Characteristics	AdaIN Layers	Example
Smooth	(12,13,14,15)	
Medium	(5,6,7,8,9,10,11)	
Hard	(0,1,2,3,4,5)	

Table 1 Layers used to generate new images of the same person based on the modification of their soft, medium or hard characteristics.

Once the artificial images were generated, two filters were applied to discard images that may have been generated incorrectly or contain noise.

- Filtered YoloV4 tiny The trained model YoloV4 tiny [25] was used for the detection of pedestrians in the generated images. All images whose classification was below the threshold of 0.6 were discarded, which was determined by analyzing a histogram created with different threshold, generating different percentages of images classified as non-pedestrians, on the actual images from the Market-1501 database.
- SSIM Filtering The Structural Similarity Metric (SSIM) [27] was used to assess the similarity between two images. The methodology used to apply this metric consisted of selecting an image of a person and comparing it with the rest of the images of that same person in different postures. If the similarity value was equal to one, it

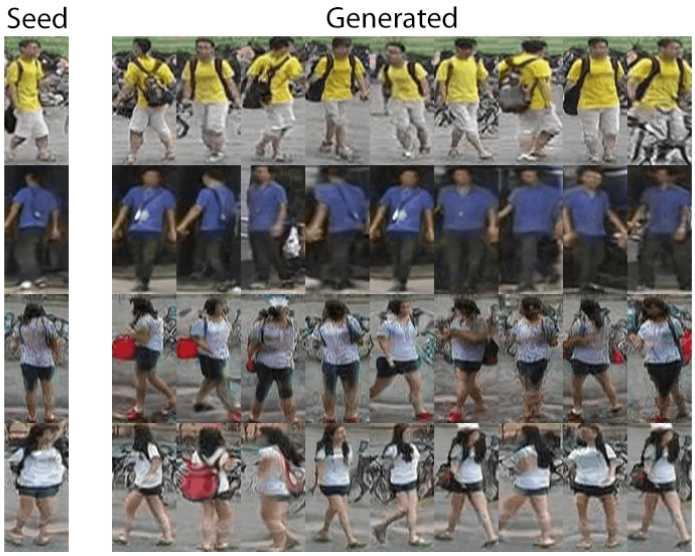

Fig. 3 The seed contains the randomly generated image to which its latent vectors will be modified to modify its average characteristics. The generated columns are the images that have been generated by modifying the latent vectors of the seed image.

was considered to be the same image. This metric was applied to the real images from the Market-1501 database and a histogram was generated. Through analysis of such histogram, it was determined that those images whose SSIM value was less than 0.75 would be discarded.

4.1 Generation of artificial people

During experimentation, images of 401 artificial people were generated completely randomly. By modifying their latent vectors, 51 images were generated per person in different postures, making a total of 20,451 images (see Fig. 3). The Yolo V4 filter was applied to the generated images to detect pedestrians, eliminating 3,419 images that represent 16.7% of the total. Afterwards, the SSIM filter was applied and 386 were discarded, 2.3% of the images. In Fig. 4 you can see some examples of images discarded with this method. Once the filters were applied, a total of 3,815 images were discarded.

In the context of generating images of real people, an encoder that has been retrained with the StyleGAN3 model pretrained with the Market-1501 database is used. This encoder is the same one used in the article presented by Yuval Alaluf et al. [28], which has proven to be highly effective in generating images with a visual quality very similar to those of human faces. The objective of using this encoder is to be able to encode images of real people and find their corresponding latent vectors within the StyleGAN3 generator, which will allow generating variations of the original image through manipulations on the latent vectors.

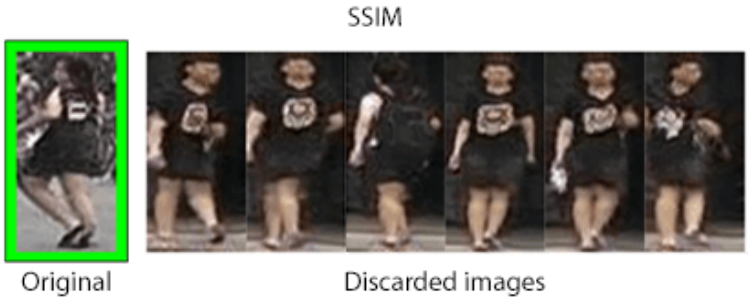

Fig. 4 Example of some discarded images using the SSIM [27] metric. Starting from one of the images of a person (original image), it is compared with the rest of the images generated of that same person.

To generate images of a real person, it was necessary to train the encoder for 240,000 epochs. To evaluate the performance of the model during training, three different loss functions were used, which allow measuring different aspects of the quality of the generated images. The first of these is the Perceptual Similarity Metric [29] (LPIPS), which measures the perceptual similarity between two images. The second is L2 [30], which measures the Euclidean difference between two images. Finally, the Momentum Contrast [31] (MOCO) loss function was used, which takes into account the correlation between the characteristics of different images.

Figure 5 shows how the three loss functions fluctuate at different times of training. It is important to note that model performance can vary depending on the complexity of the input data, and that StyleGAN3's architecture is originally designed to work with simpler data sets, such as faces. This is why the model's performance may suffer when more complex images are used, such as the full body of a person in different poses.

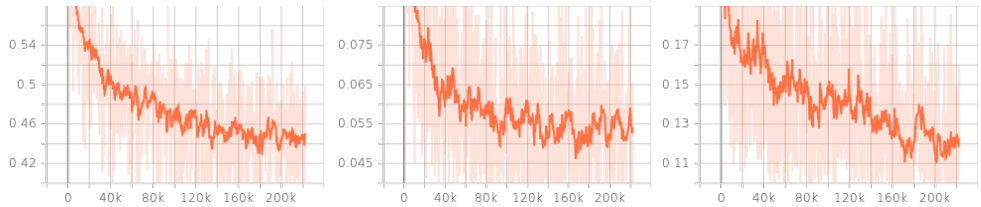

Fig. 5 LPIPS loss, L2 loss, and MOCO loss, during the training phase.

Next, it can be seen how in Figure 6 the learning is smoother and it can be seen how the LPIPS metric is the one that improves more stably.

Using the model generated in epoch 220,000, the latent vectors representing the input images were obtained. As can be seen in Fig. 8, the images obtained are similar to the original ones, although they do not reach the quality of the images generated in the previous point.

Once the latent vector has been generated, we proceed to generate the variants of that person in different postures by modifying the latent vectors. A total of 75,788

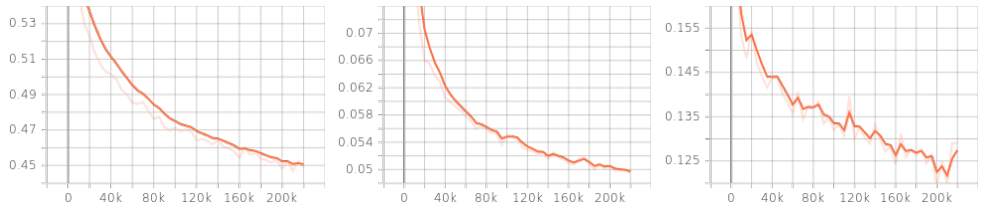

Fig. 6 LPIPS loss, L2 loss, and MOCO loss, during the testing phase.

images of 751 different people were generated. After applying the different filters, 63,659 images remained.

4.2 Re-identification using 320 artificially generated people.

During the experimentation, the model was tested with a different number of added persons, adding ten persons at a time until reaching three hundred and twenty. In Fig. 9, it can be seen that the performance of the base re-identification model remains stable or slightly decreases and then starts to increase. This may be due to the fact that increasing the number of persons also increases the number of classes, and some classes may not be as relevant as others due to different numbers of images. Adding more persons only generates small noise. The performance improved by 1% when adding 280 persons during training.

Pers.	Img.	Rank
0	**0**	**Rank@1:0.893112 Rank@5:0.964964 Rank@10:0.977732 mAP:0.742632**
10	473	Rank@1:0.888955 Rank@5:0.964074 Rank@10:0.980404 mAP:0.743795
20	874	Rank@1:0.892221 Rank@5:0.961105 Rank@10:0.974169 mAP:0.745116
30	1252	Rank@1:0.891627 Rank@5:0.965558 Rank@10:0.979513 mAP:0.741414
40	1.682	Rank@1:0.890143 Rank@5:0.964371 Rank@10:0.979513 mAP:0.748799
50	2.046	Rank@1:0.894299 Rank@5:0.962589 Rank@10:0.978028 mAP:0.746853
60	2.427	Rank@1:0.901128 Rank@5:0.967637 Rank@10:0.980404 mAP:0.751229
70	2.859	Rank@1:0.892815 Rank@5:0.965261 Rank@10:0.978325 mAP:0.748843
80	3.214	Rank@1:0.892518 Rank@5:0.965261 Rank@10:0.977732 mAP:0.748257
90	3647	Rank@1:0.899347 Rank@5:0.964964 Rank@10:0.979216 mAP:0.758927
100	4058	Rank@1:0.898159 Rank@5:0.964667 Rank@10:0.980998 mAP:0.755366
150	6.167	Rank@1:0.898753 Rank@5:0.965261 Rank@10:0.979513 mAP:0.761433
200	8.223	Rank@1:0.896378 Rank@5:0.967340 Rank@10:0.980701 mAP:0.763768
250	10.307	Rank@1:0.893705 Rank@5:0.964964 Rank@10:0.980107 mAP:0.762484
280	**11.244**	**Rank@1:0.903504 Rank@5:0.966746 Rank@10:0.982185 mAP:0.767955**
300	12.320	Rank@1:0.896081 Rank@5:0.963777 Rank@10:0.978919 mAP:0.768727
320	14.371	Rank@1:0.896675 Rank@5:0.965855 Rank@10:0.978622 mAP:0.769509

Table 2 Results of training with a different number of added people. The first row is the base, without adding any images.

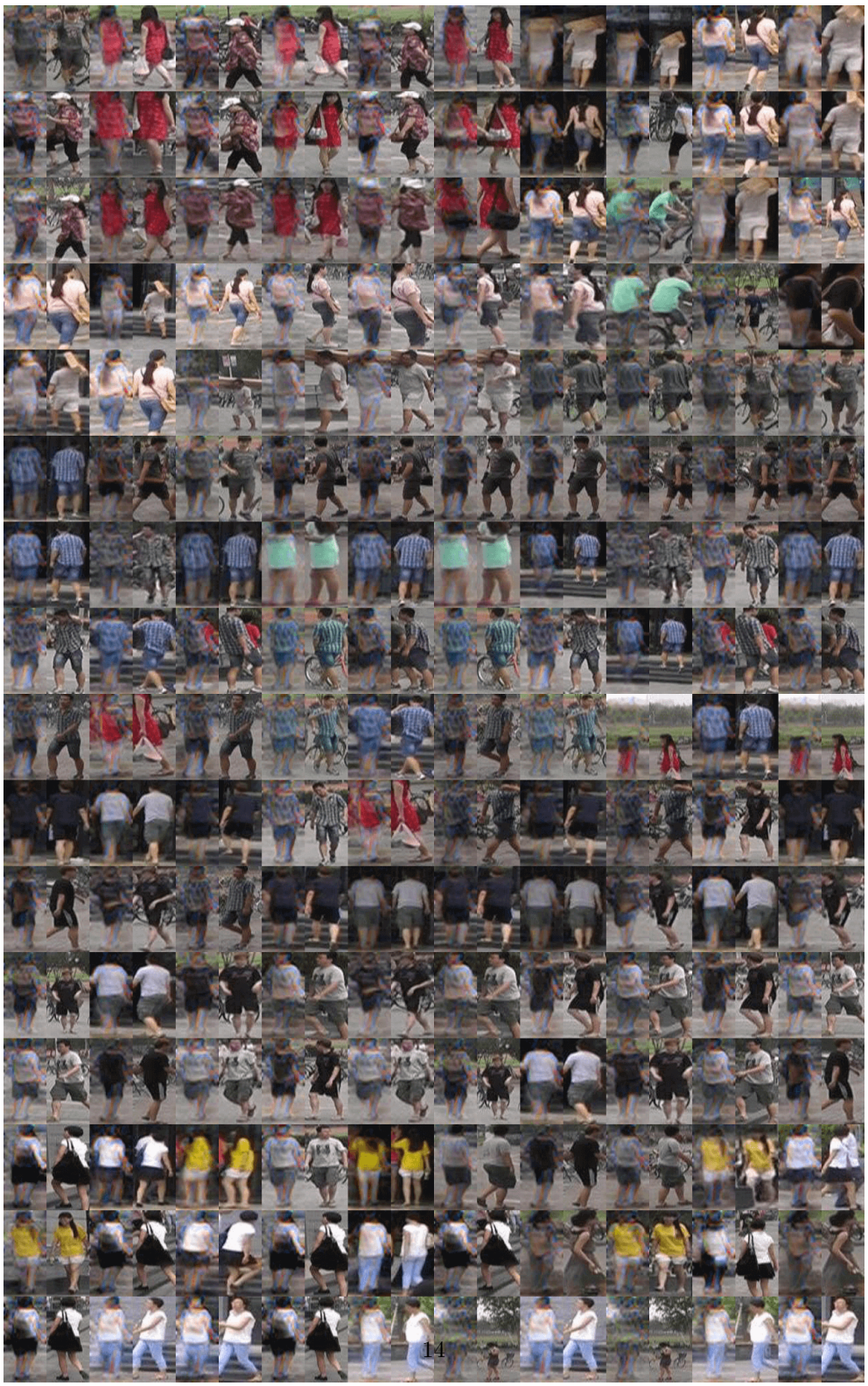

Fig. 7 Image pairs. On the right is the real image, and on the left its counterpart obtained through the encoder within the latent space of StyleGAN3. It can be observed that the model works better when the whole body is shown, although it does not reach the quality of the randomly generated images.

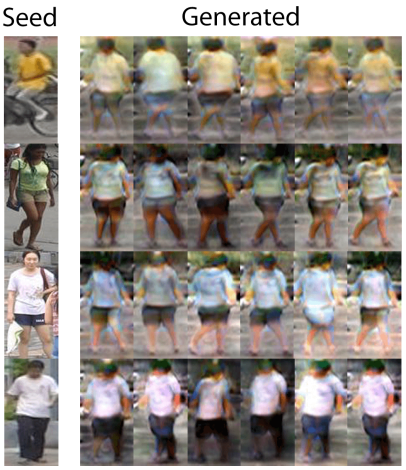

Fig. 8 Seed represent the real image. Generated represents the images that have been generated by modifying the latent vectors of the seed image.

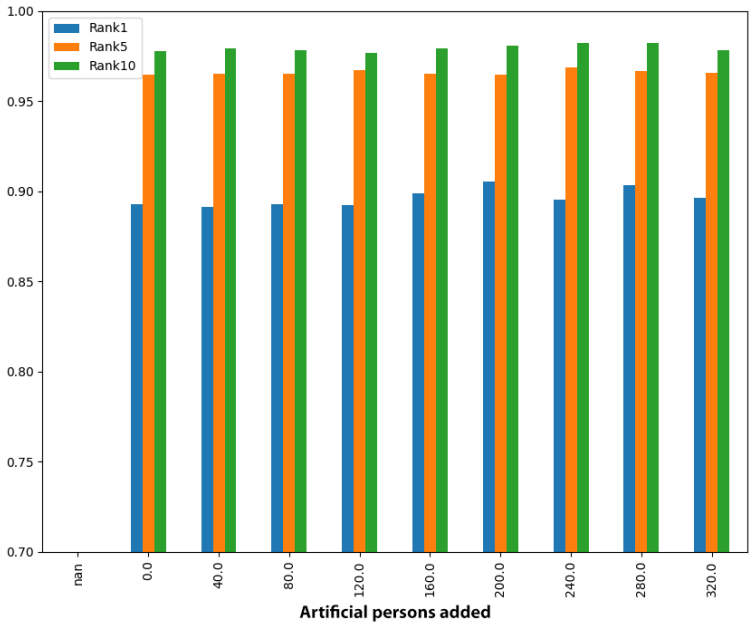

Fig. 9 Performance of some models trained with different numbers of artificially generated persons (see Table 2). Adding 0, 40, 80, 120, 160, 200, 240, 280, and 320 persons.

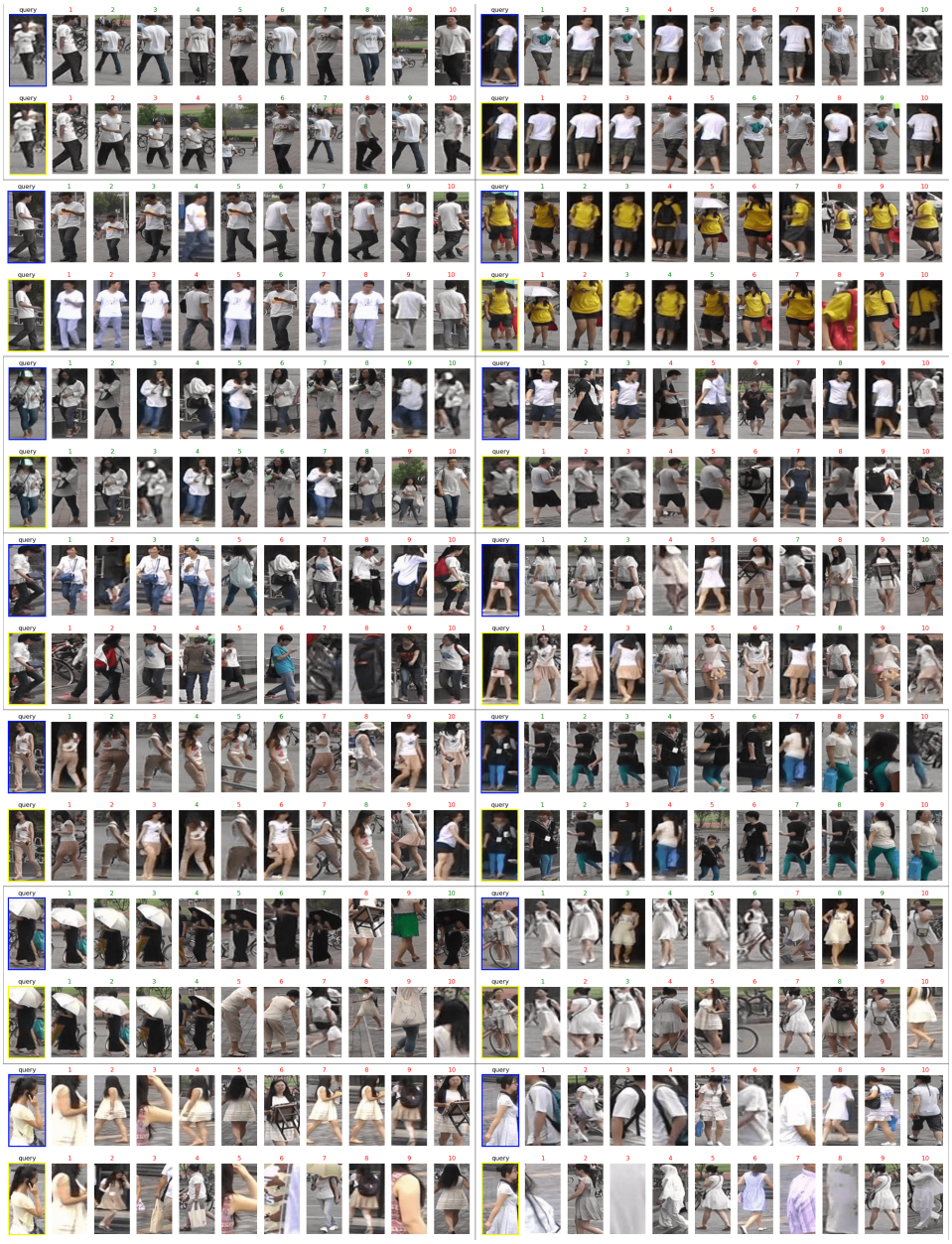

Fig. 10 Results of re-identification model. Comparison between the results of the base model and the model trained after adding 280 artificial persons. The query is the image of the person to be searched for, and the following images represent the model's output, with green indicating a correct match and red indicating an error.

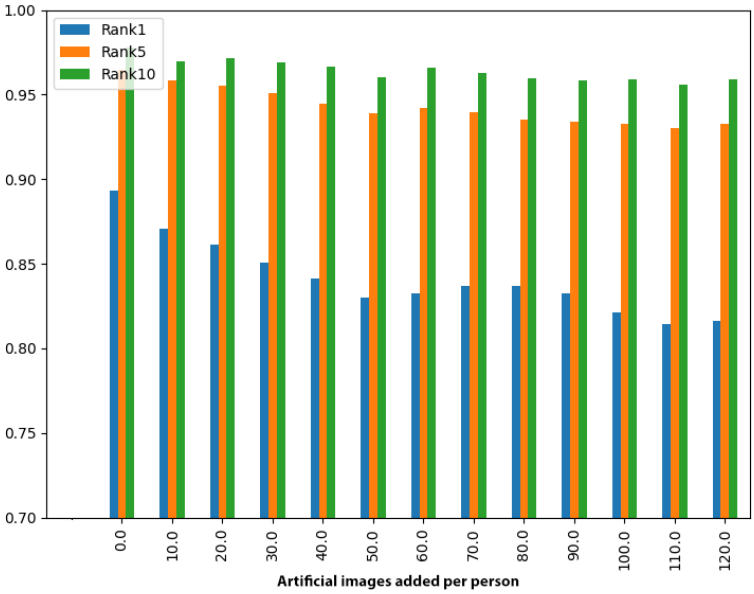

Fig. 11 Performance of some models trained with different number of artificial images per person (see Table 3). Adding 0, 10, 20, 30, 40, 50, 60, 70, 80, 90, 100, 110 and 120 images per person.

4.3 Re-identification adding artificial images in different poses to each real person

During the experimentation, we tested adding more images to each person in the training batch, adding them in increments of five. In Fig. 11, it can be observed that the performance of the base re-identification model worsens as we add more images, with performance decreasing by up to 8% when adding 100 images per person. This is due to the poor quality of the generated images, which appear very diffused compared to the real training images.

5 Conclusions

The use of adversarial generative networks to augment data is a promising technique to improve performance in re-identification models. The results obtained in our experiments demonstrate that this technique is effective in generating high-quality data and the versatility of generating modifications thereof.

As a result of the experimentation, it was possible to observe that adding totally artificial people to the re-identification model could improve its performance by 1%. The same could not be achieved with the augmentation of the images in existing people because the encoder was not able to obtain the latent vectors similar to the real images, this may be because these tools are prepared to work with faces not with images so complex as a whole body.

Pers.	Img.	Rank
0	**0**	**Rank@1:0.893112 Rank@5:0.964964 Rank@10:0.977732 mAP:0.742632**
751	5	Rank@1:0.886876 Rank@5:0.959620 Rank@10:0.975950 mAP:0.719692
751	10	Rank@1:0.878860 Rank@5:0.958432 Rank@10:0.977138 mAP:0.704901
751	15	Rank@1:0.870546 Rank@5:0.958729 Rank@10:0.969715 mAP:0.686172
751	20	Rank@1:0.865796 Rank@5:0.950119 Rank@10:0.972981 mAP:0.674047
751	25	Rank@1:0.861342 Rank@5:0.955166 Rank@10:0.971793 mAP:0.663773
751	30	Rank@1:0.850950 Rank@5:0.951010 Rank@10:0.969121 mAP:0.654492
751	35	Rank@1:0.845606 Rank@5:0.940915 Rank@10:0.964074 mAP:0.635790
751	40	Rank@1:0.841449 Rank@5:0.944477 Rank@10:0.966746 mAP:0.640035
751	45	Rank@1:0.826306 Rank@5:0.935273 Rank@10:0.961105 mAP:0.630469
751	60	Rank@1:0.829869 Rank@5:0.938836 Rank@10:0.960214 mAP:0.624382
751	70	Rank@1:0.832245 Rank@5:0.942102 Rank@10:0.965855 mAP:0.623470
751	85	Rank@1:0.825713 Rank@5:0.935570 Rank@10:0.959620 mAP:0.614014
751	105	Rank@1:0.817399 Rank@5:0.932304 Rank@10:0.958729 mAP:0.605653
751	135	Rank@1:0.813539 Rank@5:0.931710 Rank@10:0.958135 mAP:0.601084

Table 3 Results of training with different numbers of artificially generated people. The first row is the base, without adding any images.

The use of these generative adversarial networks allows the use of less original data in the training, which can reduce the resource and time requirements in the model training process. In summary, the use of adversarial generative networks in the realm of re-identification is a valuable technique that can bring a significant improvement in the performance of re-identification models. It allows the adaptation to different data sets and situations. This makes it a valuable tool not only for the field of re-identification, but also for other fields where data generation and improvement is required.

6 Competing Interests

The authors have no competing interests to declare that are relevant to the content of this article.

7 Authors contribution statement

All authors contributed to the study conception and design. Material preparation, data collection and analysis were performed by Laura Alvarez-González and Víctor Uc-Cetina. Edition of the text was performed by Anabel Martin-González and Víctor Uc-Cetina.

8 Ethical and informed consent for data used

None ethical and informed consent was needed in order to use the Market1501 dataset, which is a third party dataset publicly available at https://zheng-lab.cecs.anu.edu.au/Project/project_reid.html

9 Data availability and access

The codes and datasets generated during the current study are available in the person-reidentification GitHub repository at https://github.com/uselessai/person-reidentification

References

[1] Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial nets. In: Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N., Weinberger, K.Q. (eds.) Advances in Neural Information Processing Systems, vol. 27. Curran Associates, Inc., ??? (2014). https://proceedings.neurips.cc/paper/2014/file/5ca3e9b122f61f8f06494c97b1afccf3-Paper.pdf

[2] Zhu, J.Y., Park, T., Isola, P., Efros, A.A.: Unpaired Image-to-Image Translation Using Cycle-Consistent Adversarial Networks. Proceedings of the IEEE International Conference on Computer Vision **2017-Octob**, 2242–2251 (2017) https://doi.org/10.1109/ICCV.2017.244 arXiv:1703.10593

[3] Karras, T., Laine, S., Aila, T.: A style-based generator architecture for generative adversarial networks. Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition **2019-June**, 4396–4405 (2019) https://doi.org/10.1109/CVPR.2019.00453 arXiv:1812.04948

[4] Cai, Q., Aggarwal, J.K.: Tracking Human Motion Using Multiple Cameras (1996)

[5] Alqahtani, H., Kavakli-Thorne, M., Liu, C.Z.: An introduction to person re-identification with generative adversarial networks. arXiv, 1–15 (2019)

[6] Luo, Z.: Review of GAN-Based Person Re-Identification (2021) https://doi.org/10.32604/jnm.2021.018027

[7] Jiang, Y., Chen, W., Sun, X., Shi, X., Wang, F., Li, H.: Exploring the Quality of GAN Generated Images for Person Re-Identification vol. 1, pp. 4146–4155. Association for Computing Machinery, ??? (2021). https://doi.org/10.1145/3474085.3475547

[8] Zhong, Z., Zheng, L., Zheng, Z., Li, S., Yang, Y.: CamStyle: A Novel Data Augmentation Method for Person Re-Identification. IEEE Transactions on Image Processing **28**(3), 1176–1190 (2019) https://doi.org/10.1109/TIP.2018.2874313

[9] Dai, P., Ji, R., Wang, H., Wu, Q., Huang, Y.: Cross-modality person re-identification with generative adversarial training. IJCAI International Joint Conference on Artificial Intelligence **2018-July**, 677–683 (2018) https://doi.org/10.24963/ijcai.2018/94

[10] Liang, W., Wang, G., Lai, J., Zhu, J.: M2M-GAN: Many-to-Many Generative Adversarial Transfer Learning for Person Re-Identification (2018) arXiv:1811.03768

[11] Zheng, Z., Yang, X., Yu, Z., Zheng, L., Yang, Y., Kautz, J.: Joint discriminative and generative learning for person re-identification. Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition **2019-June**, 2133–2142 (2019) https://doi.org/10.1109/CVPR.2019.00224 arXiv:1904.07223

[12] Pang, Z., Guo, J., Sun, W., Xiao, Y., Yu, M.: Cross-domain person re-identification by hybrid supervised and unsupervised learning. Applied Intelligence (2021) https://doi.org/10.1007/s10489-021-02551-8

[13] Qian, X., Fu, Y., Xiang, T., Wang, W., Qiu, J., Wu, Y., Jiang, Y.-G., Xue, X.: Pose-Normalized Image Generation for Person Re-identification. The European Conference on Computer Vision (ECCV), 650–667 (2018)

[14] Cao, Z., Simon, T., Wei, S.E., Sheikh, Y.: Realtime multi-person 2D pose estimation using part affinity fields. Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017 **2017-Janua**, 1302–1310 (2017) https://doi.org/10.1109/CVPR.2017.143 arXiv:1611.08050

[15] Borgia, A., Hua, Y., Kodirov, E., Robertson, N.M.: GAN-based Pose-aware Regulation for Video-based Person Re-identification (2019)

[16] Zhang, C., Zhu, L., Zhang, S.C., Yu, W.: PAC-GAN: An effective pose augmentation scheme for unsupervised cross-view person re-identification. Neurocomputing **387**, 22–39 (2020) https://doi.org/10.1016/j.neucom.2019.12.094 arXiv:1906.01792

[17] Ni, Z., Pei, J., Zhao, Y.: Affine transform for skew correction based on generative adversarial network method for multi-camera person re-identification. ACM International Conference Proceeding Series **PartF16898**, 89–95 (2021) https://doi.org/10.1145/3449365.3449380

[18] Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., Wojna, Z.: Rethinking the Inception Architecture for Computer Vision. Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition **2016-Decem**, 2818–2826 (2016) https://doi.org/10.1109/CVPR.2016.308 arXiv:1512.00567

[19] Zheng, Z., Zheng, L., Yang, Y.: Unlabeled Samples Generated by GAN Improve the Person Re-identification Baseline in Vitro. Proceedings of the IEEE International Conference on Computer Vision **2017-Octob**, 3774–3782 (2017) https://doi.org/10.1109/ICCV.2017.405 arXiv:1701.07717

[20] Ainam, J.P., Qin, K., Liu, G., Luo, G.: Sparse Label Smoothing Regularization

for Person Re-Identification. IEEE Access **7**, 27899–27910 (2019) https://doi.org/10.1109/ACCESS.2019.2901599 arXiv:1809.04976

[21] Eom, C., Ham, B.: Learning disentangled representation for robust person re-identification. Advances in Neural Information Processing Systems **32** (2019) arXiv:1910.12003

[22] Hussin, S.H.S., Yildirim, R.: StyleGAN-LSRO Method for Person Re-identification. IEEE Access, 13857–13869 (2021) https://doi.org/10.1109/ACCESS.2021.3051723

[23] Karras, T., Aittala, M., Laine, S., Härkönen, E., Hellsten, J., Lehtinen, J., Aila, T.: Alias-Free Generative Adversarial Networks (NeurIPS) (2021) arXiv:2106.12423

[24] DImitrakopoulos, P., Sfikas, G., Nikou, C.: Wind: Wasserstein Inception Distance for Evaluating Generative Adversarial Network Performance. ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings **2020-May**, 3182–3186 (2020) https://doi.org/10.1109/ICASSP40776.2020.9053325

[25] Jiang, Z., Zhao, L., Shuaiyang, L.I., Yanfei, J.I.A.: Real-time object detection method for embedded devices. arXiv **3**(October), 1–11 (2020)

[26] Yu, Y., Zhang Weibin, D., Yun: Frechet Inception Distance (FID) for Evaluating GANs (September), 0–7 (2021)

[27] Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P.: Image quality assessment: From error visibility to structural similarity. IEEE Transactions on Image Processing **13**(4), 600–612 (2004) https://doi.org/10.1109/TIP.2003.819861

[28] Alaluf, Y., Patashnik, O., Wu, Z., Zamir, A., Shechtman, E., Lischinski, D., Cohen-Or, D.: Third Time's the Charm? Image and Video Editing with StyleGAN3 (2022) arXiv:2201.13433

[29] Kettunen, M., Härkönen, E., Lehtinen, J.: E-LPIPS: Robust Perceptual Image Similarity via Random Transformation Ensembles (2019) arXiv:1906.03973

[30] Zhao, H., Gallo, O., Frosio, I., Kautz, J.: Loss Functions for Neural Networks for Image Processing (November) (2015) arXiv:1511.08861

[31] He, K., Fan, H., Wu, Y., Xie, S., Girshick, R.: Momentum Contrast for Unsupervised Visual Representation Learning. Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 9726–9735 (2020) https://doi.org/10.1109/CVPR42600.2020.00975 arXiv:1911.05722