Eye-Opening GANs

Faiq Iftikhar Awan
Aena Nuzhat Lodi
Ahmed Usman Khattak
Saarland University
Germany

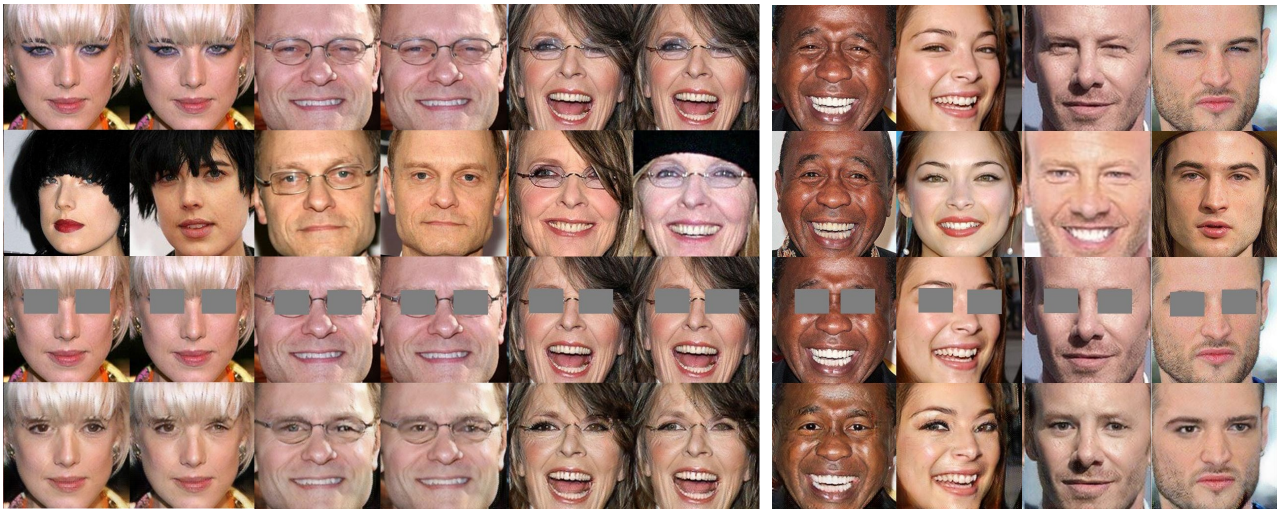

Figure 1. Examples of ExGAN

Abstract

This report is a study which looks into some of the publicly available Generative Adversarial Networks which tries to solve the problem of closed eyes in social media pictures. With people generating millions of photos each day, and they may not have time to manually retouch and beautify each image and try to fix the closed eyes. In this project, we explore the possibility of using various types of well-reputed GANs to modify such photos to make the eyes of the subject appear open, and try to see which of them performs best. There are a lot of general purpose GANs which can perform very well in this scenario but we are looking into simple GANs which can be trained relatively quickly and produce good results. For this purpose we compared two types of GANs which not only differ in architecture but also the approaches they use. Exemplar GAN (ExGAN) [1] is a type of conditional GAN that utilizes exemplar information to produce high quality, personalized in-painting results which heavily relies on the reference image. On the other hand, Gaze GAN [3] tries to solve the same problem by using two GANs and using synthesis-as-training method to improve the performance. In this report we would try to compare their results in terms of performance and time to see which one performs better and quick to be used in either an editing tool for mobile devices.

CCS Concepts: • **Computing methodologies** → **Reconstruction**.

Keywords: Computer Vision; Generative Adversarial Network; Eye-Inpainting

ACM Reference Format:
Faiq Iftikhar Awan, Aena Nuzhat Lodi, Ahmed Usman Khattak. . Eye-Opening GANs. In *Proceedings of ACM Conference (Conference'22).* ACM, New York, NY, USA, 5 pages.

1 Introduction

In many real world scenarios, photographers may find that the subject in their photos has blinked at the time of capturing the photo, causing the photos to come out with eyes half-closed, closed or looking not straight into the camera. Modern cameras often have built-in fail-safes, enhancements and filters for taking better photos. Modifications such as red eye removal and low-light adjustments have become the norm and such problems are rarely encountered these days as a result. A similar problem is when the subject of a photo closed their eyes during the taking of the photo, and in the result their eyes appear closed. This is a problem that can be solved in a similar way - rather than having to retake the photo, the photo can simply be retouched to open the eyes of

Conference'22, July 2022, Saarbrucken, Saarland, DE
.

the subject in the photo. Implementing such a solution will help to save the photographer's valuable time, that would have otherwise been wasted checking the photo and then retaking it, perhaps multiple times.

2 Related Work

We found four research papers that employed various types of GAN architectures and that we suspected could be useful for our project. We will go over two of these in more detail in this section, as the paper on GazeCorrection [4] turned out to be a previous work which laid the foundation for the authors' later work with GazeGAN [3]. Meanwhile, whilst the paper titled Globally and Locally Consistent Image Completion [2] used an advanced GAN architecture, it was very generalized, and worked better for images of landscape and scenery, and was also quite heavy to consider running on mobile devices. We were looking for GANs which were specifically designed for filling in facial features, which need to be done with a high level of accuracy to provide realistic results and avoid results falling into 'uncanny valley' territory.

2.1 Eye In-Painting with Exemplar Generative Adversarial Networks [1]

The first GAN architecture (2) that we implemented made use of Exemplar GAN, which uses two images as input. The area of the eyes to in-paint on, is marked in the first input image, whilst the second input image is used as a reference or guide to in-paint the eyes from. The gradients of the discriminator and generator are then calculated and the discriminator error is back-propagated through the generator.

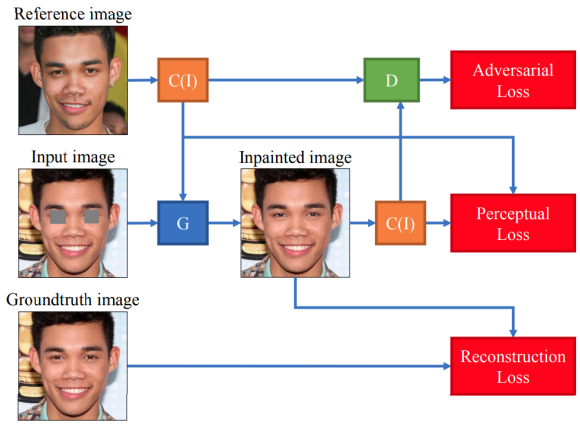

Figure 2. Exemplar GAN: General Architecture

2.2 Dual In-painting Model for Unsupervised Gaze Correction and Animation in the Wild[3]

In the second related paper, GazeGAN, the architecture (3) is made up of three modules. Images of people looking straight into the camera were compiled into a dataset and used to train the Gaze Correction Module. Meanwhile, images with people looking to the side, or with eyes closed, were compiled into another dataset and used to train the Gaze Animation Module, as well as the images that were synthesized from the Gaze Correction Module. In order to preserve the unique features of the eyes in each animated frame, the Pretrained Autoencoder Module extracts the eye features which are also then used as input to the Gaze Animation Module. So, the work in this paper demonstrates not only a system to correct eye gaze in photos but also a smooth animation from incorrect to corrected.

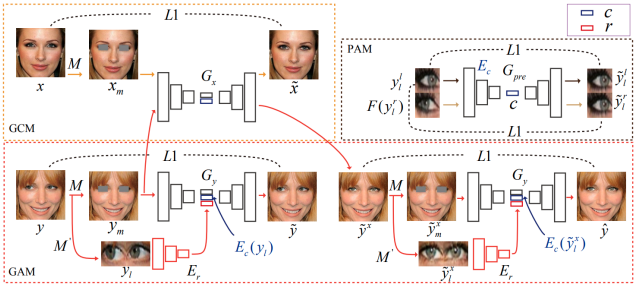

Figure 3. Gaze GAN: General Architecture

3 Method

3.1 Eye In-Painting with Exemplar Generative Adversarial Networks [1]

The ExGAN uses a custom dataset provided by the authors of the paper. The dataset is a huge set of in the wild images of popular internet personalities and actors around the world. The dataset was manually downloaded from the website www.famousbirthdays.com, which has multiple images of the same person in different lighting environment and poses. For the working of Exemplar GAN it is important to have atleast 3 images of the same person and atleast one of them with the person's eyes open. The dataset can be downloaded and it contains roughly 100,000 images with 17,000 unique persons. Furthermore the authors have created a JSON file that goes along with the data to extract exact position of eyes within each and every photos and tells some other useful information like, if the person is wearing glasses or not etc. The dataset is then divided in training and test based on the usability of images and if it meets the condition of the availability of image for the same person.

Due to the time constraints the GAN is trained for 34,000 steps over the dataset with a learning rate of 0.00001, batch size of 4 and the learning rate decay of 1 which linearly decreases after 20,000th step.

The exemplar GAN is highly dependent on the reference image provided to the GAN. If the reference image has some artifacts around the eyes, this would be shown in the resultant generated image. In the example shown in figure 5, the

image on the left end is the ground truth, the middle ones are two possible reference images one with glasses and the other one without glasses and in the far right we can see two images generated by GAN by each reference image. The image generated with reference image as without glasses is for more realistic then the one generated by the other image as reference image. This makes it a big issue for the user to give a clear image to GAN, only then the results would be realistic.

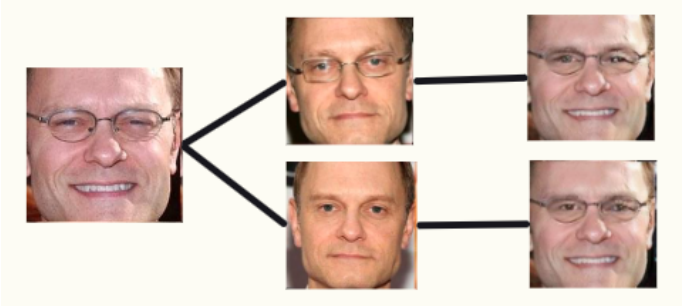

Figure 4. ExGAN: Effects of reference image

Also the custom dataset used by the authors has lot of images where the subject is looking away or they are not in the center of the shot. This evidently resulted in a "garbage in garbage out" situation. The dataset needed more refinement to be used properly by the GAN.

3.2 Dual In-painting Model for Unsupervised Gaze Correction and Animation [3]

GazeGAN uses the publicly available dataset *CelebA* and aligned the photos of every subject to be in the center of picture. The dataset specifically created by the authors of the paper is called *CelebAGaze* which is completely based on the original dataset with some orientation and allignment difference. CelebAGaze consists of 25,283 high-resolution celebrity images that are collected from CelebA and the Internet. It consists of 21,832 face images with eyes staring at the camera and 3,451 face images with eyes staring somewhere else. All images (256 × 256) are cropped and the eye mask region by dlib is computed. Note that this dataset is unpaired and it is not labeled with the specific eye angle or the head pose information.

Due to the time constraints the GAN is trained for 34,000 steps over the dataset with a learning rate of 0.00001, batch size of 16. Also this GAN uses pretrained VGG16 model for perceptual loss.

GazeGAN generates realistic and more believable results. Also, as we would look in the experiment section, the FID scores of GazeGAN is very low. On the other hand the complex nature of GazeGAN results in long training and testing times, almost 2.5 times more time than ExGAN. This can be catered down by improving the performance of the machine

Figure 5. GazeGAN: Results of Gaze GAN

but since we are looking for a solution for mobile devices, this is not possible.

4 Experiment

We trained both the GANs with the given values of hyperparameters in the method section. To compare both the models against each other we used the same number of steps i.e. 34,000. GazeGAN uses a pre-trained VGG16 for perceptual loss with the last layer removed. There is also a difference of dataset in the two models. GazeGAN uses custom dataset built on top of *CelebA* which makes sure that the subject is perfectly aligned and makes it easier for the GAN to generalize while ExGAN uses completely in the wild pictures and relies heavily on the reference image which causes the generated image to be unrealistic and with artifacts.

4.1 Results

We tried out two different metrics to evaluate our GAN. First we tried Mean Squared Error, but the results that it gave were difficult to interpret, because it penalizes some of the images that have been generated correctly. For example, if the subject's eyes were originally closed in an image, and in the output image their eyes were open, it would deduct points for differing from the input image even though this is precisely the result we wanted to achieve.

In the end, we went with the Frechet Inception Distance metric as recommended in the interim report. A lower FID score indicates a higher level of accuracy. An example shown on the side clearly depicts the issues raised by using MSE and were resolved by FID.

In the figure 6 we have plotted the FID score for Exemplar GAN against steps. We found the testing accuracy of Exemplar GAN to be around 50.

However, GazeGAN performed significantly better, achieving a FID score lower than 5; ten times as small as the score for Exemplar GAN, in the same number of steps. This score as shown in figure 7, along with the results that we were seeing left us with no doubts as to which GAN performed with better accuracy between the two.

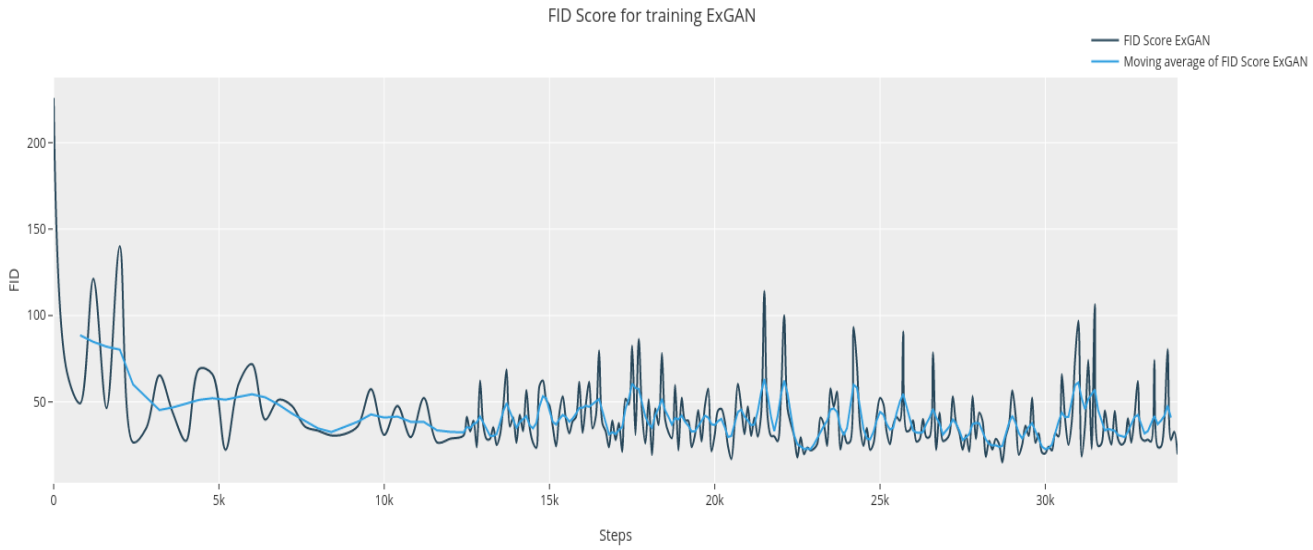

Figure 6. FID Score of ExGAN over 34,000 steps

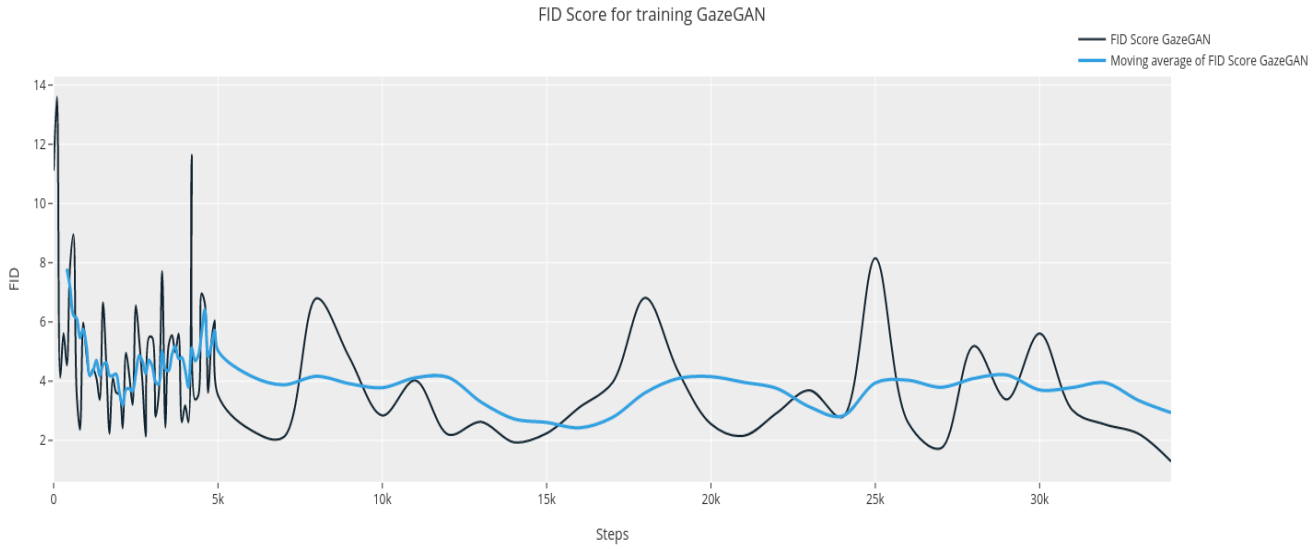

Figure 7. FID Score of GazeGAN over 34,000 steps

4.2 Analyses

GazeGAN's performance boost is mainly thanks to its complex design and custom dataset. The cleaner selection of data and the higher number of layers in the Gaze Correction Module greatly helped it to achieve the level of accuracy that it did.

On the flip side, Exemplar GAN runs much faster than GazeGAN, by a factor of 2.5. Exemplar GAN has a much more diverse dataset with more in the wild images that generalize the training, but its dependence upon the reference image makes its results quite unpredictable, and as a result brings down its overall performance.

GazeGAN not only trains two GANs for the problem but also has a very narrow and strict criteria for evaluation than Exemplar GAN. Exemplar GAN only performs very well using reference images where the eyes are completely free of occlusion - any images that were reconstructed using reference images with the subject wearing glasses, or with

hair covering their eyes, did not turn out as well as the ones that were generated with a less occluded reference image.

We also found that GazeGAN was better at generating symmetric eyes thanks to its rotationally-invariant feature detection, which it used to generate corrected gaze as well as animate it.

5 Conclusions

In conclusion, we found that GazeGAN provides the best type of architecture to open closed eyes in images, synthesizing the most convincing images with a high level of consistency. Exemplar GAN takes less time to run, but does not produce results with such high accuracy as GazeGAN.

References

[1] Brian Dolhansky and Cristian Canton Ferrer. 2018. Eye in-painting with exemplar generative adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 7902–7911.

[2] Satoshi Iizuka, Edgar Simo-Serra, and Hiroshi Ishikawa. 2017. Globally and locally consistent image completion. *ACM Transactions on Graphics (ToG)* 36, 4 (2017), 1–14.

[3] Jichao Zhang, Jingjing Chen, Hao Tang, Wei Wang, Yan Yan, Enver Sangineto, and Nicu Sebe. 2020. Dual in-painting model for unsupervised gaze correction and animation in the wild. In *Proceedings of the 28th ACM International Conference on Multimedia*. 1588–1596.

[4] Jichao Zhang, Meng Sun, Jingjing Chen, Hao Tang, Yan Yan, Xueying Qin, and Nicu Sebe. 2019. Gazecorrection: Self-guided eye manipulation in the wild using self-supervised generative adversarial networks. *arXiv preprint arXiv:1906.00805* (2019).