Dual In-painting Model for Unsupervised Gaze Correction and Animation in the Wild

Jichao Zhang1, Jingjing Chen2, Hao Tang1, Wei Wang3, Yan Yan4, Enver Sangineto1, Nicu Sebe1,5
1University of Trento, 2Shandong University, 3Ecole Polytechnique Fédérale de Lausanne
4Texas State University, 5Huawei Research Ireland

ABSTRACT

We address the problem of unsupervised gaze correction in the wild, presenting a solution that works without the need of precise annotations of the gaze angle and the head pose. We created a new dataset called CelebAGaze consisting of two domains X, Y, where the eyes are either staring at the camera or somewhere else. Our method consists of three novel modules: the Gaze Correction module (GCM), the Gaze Animation module (GAM), and the Pre-trained Autoencoder module (PAM). Specifically, GCM and GAM separately train a dual in-painting network using data from the domain X for gaze correction and data from the domain Y for gaze animation. Additionally, a Synthesis-As-Training method is proposed when training GAM to encourage the features encoded from the eye region to be correlated with the angle information, resulting in gaze animation achieved by interpolation in the latent space. To further preserve the identity information e.g., eye shape, iris color, we propose the PAM with an Autoencoder, which is based on Self-Supervised mirror learning where the bottleneck features are angle-invariant and which works as an extra input to the dual inpainting models. Extensive experiments validate the effectiveness of the proposed method for gaze correction and gaze animation in the wild and demonstrate the superiority of our approach in producing more compelling results than state-of-the-art baselines. Our code, the pretrained models and supplementary results are available at: https://github.com/zhangqianhui/GazeAnimation.

CSCS CONCEPTS

• Computing methodologies → Computational photography; Image representations.

KEYWORDS

Deep Learning, Generative Adversarial Networks, Image Translation, Gaze Correction, Gaze Animation

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

ACM Reference Format:

Jichao Zhang1, Jingjing Chen2, Hao Tang1, Wei Wang3, Yan Yan4, Enver Sangineto1, Nicu Sebe1,5. 2020. Dual In-painting Model for Unsupervised Gaze Correction and Animation in the Wild. In Proceedings of the 28th ACM International Conference on Multimedia (MM ’20), October 12–16, 2020, Seattle, WA, USA. ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3394171.3413981

1 INTRODUCTION

Gaze correction aims at manipulating the eye gaze with respect to the desired direction. This is important in some real-life scenarios where there is the need to “move” the person’s gaze into the camera. For example, shooting a good portrait is challenging as the subject may be too nervous to stare at the camera. Another scenario is videoconferencing where the eye contact is extremely important as the gaze can express attributes such as attentiveness and confidence. Unfortunately, eye contact and gaze awareness are lost in most videoconferencing systems, as the participants look at the monitors and not directly into the camera.

Early works in gaze correction relied on special hardware, such as stereo cameras [3, 46], Kinect sensor [21] or transparent mirrors [18, 30]. Recently, some learning-based methods have produced very high-quality synthetic images with a corrected gaze. For instance, Kononenko and Lemptisky [20] propose to solve the problem of monocular gaze correction using decision forests. DeepWarp [6] uses a deep network to directly predict an image-warping flow field with a coarse-to-fine learning process. However, despite its good qualitative results, this method fails in generating photo-realistic images when redirecting the gaze for large angles. Additionally, it produces unnatural eye shapes as the L1 loss is exploited to learn the flow field without any geometry regularization. To solve this problem, PRGAN [9] proposes to exploit adversarial learning with a cycle-consistent loss to generate more plausible gaze redirection results. However, it is challenging for these methods [6, 9, 20] to obtain high-quality gaze redirection results in the wild when there are large variations in the head pose, gaze angles etc. Another category of works is based on a 3D model without training data such as GazeDirector [44]. The main idea of GazeDirector is to

Figure 1: A few example images of the CelebAGaze dataset. Left: People staring at the camera for domain X. Right: People staring somewhere else for domain Y. These examples show the diversity in the head pose, the gaze angle and the illumination conditions of the proposed dataset.
model the eye region in a 3D reference system instead of predicting a flow field directly from the input image. However, modeling in 3D has strong assumptions that do not hold in reality. More importantly, the previous methods need training labels of the head pose and gaze angles, but these are hard to obtain in the wild. To solve these problems, we collected the CelebAGaze dataset which consists of two domains: eyes staring at the camera for domain X and eyes staring somewhere else for domain Y, as shown in Fig. 1. Note that no paired samples exist in the CelebAGaze dataset. We propose an unsupervised learning method for gaze correction and animation consisting of three modules: 1) GCM is an in-painting model, trained on the domain X, which learns how to fill in the missing eye regions with the new content which has the corrected eye gaze; 2) GAM exploits the other in-painting model, trained on the domain Y. To generalize the gaze animation with various angle directions, we propose the Synthesis-As-Training Method which encourages the features encoded from the eye region to be correlated with the angle information. Finally, gaze animation can be achieved by interpolating this feature in the latent space; 3) PAM encodes the angle-invariant content features, e.g., iris color, eye shape. Specifically, we pretrain an autoencoder by self-supervised mirror learning where the bottleneck features are extracted as an extra input of the dual in-painting model to preserve the identity of the corrected results. Finally, both the qualitative and quantitative evaluations demonstrate that our model achieves more compelling results than the state-of-the-art baselines in gaze correction and gaze animation. To recap, our main contributions are:

1) We propose a simple yet effective dual in-painting method for gaze correction and animation.
2) For gaze animation, we introduce a novel Synthesis-As-Training Method which encourages the features encoded from the eye region to be correlated with the angle information.
3) We design a novel Self-Supervised Mirror Learning for pretraining an autoencoder to extract content features to preserve the identity of the corrected results.
4) We make available to the research community a new gaze dataset for gaze correction and animation.

2 RELATED WORK

Generative Adversarial Networks: Generative Adversarial Networks [7] are powerful generative models which learn a distribution that mimics a given target distribution. They have been applied to many fields, such as low-level image processing tasks (image in-painting [12, 36], image super-resolution [22, 23, 42]), high-level semantic and style transfer (image translation [14, 26, 28, 33, 40, 41, 55], person image synthesis [38, 39], image manipulation [34]).

Image Inpainting: Image inpainting, an important task in computer vision and graphics, aims at filling the missing pixels of an image with plausibly synthetic content. Recently, CNN-based and GAN-based methods have shown promising performance on image inpainting [13, 24, 35, 50]. Thus, some works try to apply an inpainting model for facial attribute manipulation, such as hair, mouth and eye [4, 15, 31]. Similar to these methods, our approach also follows the deep inpainting methods, but it does not require the data to be labeled or other additional information, such as a semantic segmentation mask, a sketch and even a reference image.

Gaze Correction: Previous work for gaze correction can be divided into three classes: 1) hardware-driven, 2) rendering and synthesis, 3) learning-based.

The hardware support is indispensable in early research. Kolarits et al. [18] tried to make use of half-silvered mirrors to allow the camera to be placed on the optical path of the display. Yang et al. [47] aimed to address the eye contact problem with a novel view synthesis, and they use a pair of calibrated stereo cameras and a face model to track the head pose in 3D. Generally speaking, these hardware-based methods are expensive.

Some works render the eye region based on a 3D fitting model, which replaces the original eye with new synthetic eyeballs. Bani et al. [1] uses an example-based approach for deforming the eyelids and slide the iris across the model surface with texture-coordinate interpolation. To fix the limitation caused by the use of a mesh, where the face and eyes are joined, GazeDirector [44] regards the face and eyeballs as separate parts, synthesizing more high-quality images, especially for large redirection angles. These methods can be applied to eye correction, but they struggle when rendering eyes realistically is a challenge. Additionally, modeling methods have strong assumptions that usually do not hold in reality.

The core idea for most of the learning-based methods is to use a large paired training dataset to train the model [19, 20, 32, 48]. Some methods [19, 20] learn to generate the flow field which is used to relocate the eye pixels in the original image. Gannin et al. [6] proposes to use a convolutional network to learn the flow field warping the input image for redirecting the gaze to the desired angle. However, the model fails to generate photo-realistic and natural shape results of gaze redirection, as it uses only pixel-wise differences between the input and the ground truth as the loss. To solve this problem, He et al. [9] propose to use adversarial learning jointly with a cycle-consistent loss, which can improve the visual quality and redirection precision. However, these methods can hardly generate plausible results in the wild, coping with large variations in head pose, gaze angle and illumination. In contrast, we propose to utilize an in-painting model to correct the gaze angle which can generate high-quality gaze correction and gaze animation in the wild.

3 METHOD

The overview of our method is shown in Fig. 2 and it consists of three modules: the Gaze correction Module (GCM), the Gaze Animation Module (GAM) and the Pretrained Autoencoder Module (PAM). Before introducing the details, we first clarify the notation.

- \( z \in Z \) indicates an image instance represented in \( Z = \mathbb{R}^{m \times n \times c} \), where \( m, n, c \) are the height, the width and the number of channels.
- \( Z \) is divided into two domains: \( X \) containing image instances with gaze staring at the camera and \( Y \) containing image instance with gaze staring somewhere else.
- \( M \in \mathbb{R}^{m \times n \times c} \) denotes a binary mask of the eye region and \( M' \) defines the operation of extracting a sub-image (eye region) for a rectangular region.
- \( P_x, P_y \) denote distributions of data from the domains \( X \) and \( Y \), respectively. \( P_m \) denotes the distribution of data \( M(z) \) with the eye regions removed. \( M(x) \) and \( M(y) \) have the same distribution, as
the images \(x \in X\) and \(y \in Y\) have the only discrepancy in the eye region. Thus, \(M(x) \sim P_m\), \(M(y) \sim P_m\).  
- \(r \in \mathbb{R}^2\); \(c \in \mathbb{R}^{256}\) denote the angle and the content features (angle-invariant) where both are extracted by different encoders.  
- \(F\) denotes the image horizontal flipping operation (Mirroring).

### 3.1 GCM with Generator \(G_x\)

As shown in Fig. 2, \(G_x\) is trained on data \(x\) from domain \(X\) and it aims to complete the masked image \(x_m = M(x)\) by synthesizing the eye region. In theory, \(G_x\) can learn the mapping from \(P_m\) to \(P_x\), thus, \(\hat{x} = G_x(M(x))\) and \(\hat{y}^x = G_x(M(y))\). Then, \(\hat{x} \sim P_x\), \(\hat{y}^x \sim P_x\), as \(M(x)\) and \(M(y)\) have the same distribution \(P_m\). This is the theoretical basis of our correction module. Specifically, we can use \(G_x\) to correct the gaze to stare at the camera for data \(y\) from domain \(Y\). It can be represented as:

\[
\begin{align*}
    c_x &= E_c(M'(x)), c_y = E_c(M'(y)) \\
    \hat{x} &= G_x(M(x), c_x), \hat{y}^x &= G_x(M(y), c_y),
\end{align*}
\]

(1)

where \(c_x\) and \(c_y\) are the content features encoded from \(M'(x)\) and \(M'(y)\) by \(E_c\). \(E_c\) is the encoder of \(G_{pre}\) which will be introduced in Section 3.3.

We use a pixel-wise loss (L1) for training GCM, shown in Fig. 2. It is defined as:

\[
\ell_{recon}^x = \mathbb{E}_{x \sim P_X} \| x - \hat{x} \|_1.
\]

(2)

### 3.2 GAM with Generator \(G_y\)

In addition to correct the gaze to stare at the camera, it is more valuable and efficient to redirect the gaze into any direction for gaze animation. Thus, we propose a novel Synthesis-As-Training Method, in which we use the synthetically corrected data as training data of the other generator \(G_y\) to learn gaze animation.

In detail, this module can be divided into two stages. In the first stage we train the inpainting model \(G_y\) to fill-in the masked image \(y_m = M(y)\) and produce \(\hat{y}\). Different from \(G_x\), \(G_y\) encodes the eye region into the latent code \(r \in \mathbb{R}^2\) by exploiting \(E_r\), where \(r\) is an extra input of the decoder in \(G_y\).

\[
\begin{align*}
    r_y &= E_r(M'(y)), c_y = E_c(M'(y)) \\
    \hat{y} &= G_y(r_y, c_y).
\end{align*}
\]

(3)

The reconstruction loss for PAM is defined as:

\[
\ell_{recon}^y = \mathbb{E}_{y \sim P_Y} \| y - \hat{y} \|_1.
\]

(4)

In the second stage, we use \(G_x\) to correct the gaze of \(y\) and produce the synthetic sample \(\hat{y}^x\) with the corrected gaze, shown in the right side of Fig. 2. Then, \(\hat{y}^x\) is used for training \(G_y\), just like \(y\) does. With the paired samples \((y, \hat{y}^x)\), which have the same incomplete region \(M(y)\) but different eye region \(i.e., (y, \hat{y}^x))\), we train GAM and we ensure that the encoded feature \(r\) has a highly correlation with the gaze angle.

\[
\begin{align*}
    r_{\hat{y}^x} &= E_r(M'(\hat{y}^x)), c_{\hat{y}^x} = E_c(M'(\hat{y}^x)) \\
    \hat{\hat{y}} &= G_y(M(\hat{y}^x), r_{\hat{y}^x}, c_{\hat{y}^x}).
\end{align*}
\]

(5)
where the new angle representation is fed to $E$ as shown in the top-right of Fig. 2, we use two paired samples to eye shape, iris color, etc., likewise for $\gamma$. We have paired images: left eye $y_l$, the right eye $y_r$, and the flipping result of left eye ($F(y_l)$).

The corresponding loss is defined as:

$$\ell_{\text{recon}} = \mathbb{E}_{y \sim p_y} [||\hat{y}^x - \hat{y}||].$$

We exploit $\ell_{\text{recon}} + \ell_{\text{adv}}$ as objective functions for training $E_r$ and $G_y$. And we interpolate this feature $r$ between $E_r(y_l)$ and $E_r(y_r)$ to produce the gaze animation which will be introduced later.

### 3.3 PAM with Self-Supervised Mirror Learning

This inpainting process has difficulties in preserving the consistency of the identity information (e.g., iris color, eye shape). Consequently, we propose to pretrain $G_{\text{pre}}$ to learn the content features (c). Then, $c$ is used as the guidance for $G_x$ and $G_y$ to preserve the identity information of the inpainted results ($c$ denoted with a blue box in Fig. 2).

Although our training dataset is collected from the Internet, the majority of the images it is composed of, are close to the frontal image (first column in Fig. 4). As shown in columns 2-5 of Fig. 4, we have paired images: left eye $y_l$, and the mirror $F(y_l)$ of the right eye $y_r$, where both have different angles, but they have a very similar eye shape, iris color, etc., likewise for $y_r$. Based on this observation, as shown in the top-right of Fig. 2, we use two paired samples to pretrain $G_{\text{pre}}$ using the following objective functions:

$$\ell_{\text{pre}} = \mathbb{E}_{y \sim p_y} [||y_l - G_{\text{pre}}(y_l)||_1 + ||y_r - G_{\text{pre}}(F(y_l))||_1 + ||y_r - G_{\text{pre}}(F(y_l))||_1].$$

After training, we found that the bottleneck feature $c$ of $G_{\text{pre}}$ is quasi angle-invariant, containing only the content information (iris color, eye shape). Thus, we use the encoder network $E_c$ of $G_{\text{pre}}$ as extra input of $G_x$ and $G_y$ to extract content features. With these features as guidance, the inpainted results are more consistent with the input in identity information.

### 3.4 Global and Local Discriminators for Adversarial Learning

Since the $L_1$ loss tends to produce blurry results, we use two discriminators $D_x$ and $D_y$ adversarially trained. $D_x$, $D_y$ do not share the weights of all the layers. Moreover, inspired by [13], we use a global discriminator that takes the entire face as input and a local discriminator which takes the local eye region as input. The global part is used to make coherent the entire image as a whole, while the local part is used to make the local region more realistic and sharper. We concatenate the final fully-connected feature map of both parts into one output which is used as the input of a sigmoid function to predict the probability of the image being real. The objective function for $D_x$ and $G_x$ is defined as:

$$\min_{G_x} \max_{D_x} \ell_{x \text{adv}} = \mathbb{E}_{x \sim p_x} [\log D_x(x, M'(x))]
+ \mathbb{E}_{x \sim p_x} [\log (1 - D_x(x, M'(x)))]
+ \mathbb{E}_{y \sim p_y} [\log (1 - D_x(y, M'(y)))].$$

The objective function for $D_y$ and $G_y$ is defined as:

$$\min_{G_y} \max_{D_y} \ell_{y \text{adv}} = \mathbb{E}_{y \sim p_y} [\log D_y(y, M'(y))]
+ \mathbb{E}_{y \sim p_y} [\log (1 - D_y(y, M'(y)))].$$

### 3.5 Overall Loss

Similar to [11], we use a reconstruction loss for the content features in the latent space to further preserve the identity information between the input image and the corrected output. This loss ($\ell_{fp}$) is defined as:

$$\ell_{fp} = \mathbb{E}_{y \sim p_y} [||y - E_r(M'(y))||_1 + ||\hat{y}^x - E_r(M'(\hat{y}^x))||_1].$$
As described above, we randomly select 300 samples from domain \( \mathcal{X} \) as the training set and 100 samples from domain \( \mathcal{Y} \) as the test set, the remaining as the training set. Note that this dataset is unpaired and it is not labeled with the specific eye angle or the head pose information. We show some samples of the CelebA gaze dataset in Fig. 1.

### 4.2 Training Details

We first train the PAM module. Then, the discriminators \( D_x \) and \( D_y \) and the generators \( G_x \) and \( G_y \) are jointly optimized. We use the Adam optimizer with \( \beta_1 = 0.5 \) and \( \beta_2 = 0.999 \). The batch size is 8. The initial learning rate for DAM is 0.0005 and 0.0001 for the discriminators and the generators in the first 20000 iterations, and linearly decayed to 0 over the remaining iterations. All the weight coefficients \( \lambda_1, \lambda_2, \lambda_3, \lambda_4, \lambda_5 \) are set to 1. To stabilize the network training in the adversarial learning, we use spectral normalization [29] for all the convolutional layers of the discriminators \( D_x \) and \( D_y \) but not for the generator \( G_x \). Our model is implemented in Tensorflow and takes ten hours to be trained with a single NVIDIA Titan X GPU.

### 4.3 Baseline Models

**Gaze Correction**: PRGAN [9] achieved state-of-the-art gaze redirection results on the Columbia dataset based on a single encoder-decoder network with adversarial learning, similarly to StarGAN [2]. The original PRGAN is trained on paired samples with labeled angles (Columbia Gaze [37]). To train PRGAN on the proposed CelebA gaze dataset, we remove the VGG perceptual loss of PRGAN and learn the domain translation between \( X \) and \( Y \). Note that we train PRGAN only with the local eye region, the same way as in the original paper.

**Facial Attribute Manipulation**: Gaze correction and animation can be regarded as a sub-task of facial attribute manipulation. Recently, StarGAN [2] achieved very high-quality results in facial attribute manipulation. We train StarGAN on the CelebA gaze dataset to learn the translation mapping between domain \( X \) and domain \( Y \). To further improve the ability of StarGAN on this task, we employ a variant of StarGAN as an additional baseline: StarGAN with spatial attention learning [49] referred to as StarGANA. Moreover, gaze correction can be considered as an image translation task. Thus, we adopt CycleGAN as another baseline for our experiments. Note that we do not compare GazeGAN with SGGAN [51], AttGAN [10],

| Metrics | MSSSIM | LPIPS | User Studies |
|---------|--------|-------|--------------|
| Other   | 0.969  | 0.0732| 3.40 %       |
| StarGAN [2] | 0.998  | 0.0022| 6.67 %       |
| CycleGAN [55] | 0.991  | 0.0267| 15.00 %      |
| PRGAN [9] | 1.0    | 0.0   | 8.33 %       |
| GazeGAN | 1.0    | 0.0   | 42.40 %      |
| GT      | 1.0    | 0.0   | 100%         |

### Table 1: Columns 1-2: the MSSSIM and LPIPS scores computed on the background regions of the gaze correction results using different models. Column 3: the user study results. Higher is better for MSSSIM and the user study; lower is better for LPIPS.

*Note: The table is not directly transcribed as an image.*
Figure 5: Qualitative comparison for the gaze correction task. The first row shows the input images and the following rows show the gaze correction results of StarGAN [2], StarGANA, CycleGAN [55], PRGAN [9] and GazeGAN.

4.4 Gaze correction

In this section, we first qualitatively compare the proposed method with state-of-the-art methods on the CelebAGaze dataset for the task of gaze correction. Then, we choose some metrics to quantitatively evaluate the gaze correction results.

Qualitative results: As shown in the last row of Fig. 5, GazeGAN can correct the eyes to look at the camera while preserving the identity information such as the eye shape and the iris color, validating the effectiveness of the proposed method. The 2nd row shows the results of StarGAN [2]. We note that StarGAN could not produce precise gazes staring at the camera and it suffers for the low-quality generation with lots of artifacts. The results of CycleGAN are shown in the 4th row. Although the results of CycleGAN are very realistic and with low artifacts in the eye region, this method does not produce a precise correction for the gaze. We believe that this is because both StarGAN and CycleGAN are based on the cycle-consistency loss, which requires that the mapping between $X$ and $Y$ should be continuous and invertible. According to the invariance of the Domain Theorem 4, the intrinsic dimensions of the two domains should be the same. However, the intrinsic dimension of $Y$ is much larger than $X$, as $Y$ has more variations for the gaze angle than $X$. As shown in the 7th columns of Fig. 5, both StarGAN and CycleGAN cannot preserve the irrelevant regions of the face. Another stronger baseline is StarGANA, which combines spatial attention [49] with StarGAN, and successfully preserves the irrelevant region (shown in the 3rd row). However, the results of StarGANA are not faithfully and realistic, with lots of deformation artifacts, compared with GazeGAN.

Moreover, we compare GazeGAN with PRGAN [9], which is trained using only local eye regions (same as in the original paper), which may be helpful to focus on the translation in the eye region. The results of PRGAN are shown in the 5th row of Fig. 5. Compared with GazeGAN, PRGAN does not produce precise and realistic correction results, similarly to StarGAN and CycleGAN. Additionally, PRGAN suffers from the boundary mismatch problem between the local eye region and the global face.

Quantitative Evaluation Protocol: The qualitative evaluation has validated the effectiveness and the superiority of our proposed GazeGAN in gaze correction. To further support the previous evaluation with a quantitative evaluation, we employ MSSSIM [43] and LPIPS [52] to measure the preservation ability of the irrelevant regions, i.e., faces without the eye region ($M(y)$). Specifically, we compute the mean MSSSIM and LPIPS scores between $M(y)$ and $M(\tilde{y})$ across all the test data from domain $Y$.

In addition to the metrics aforementioned, we conduct a user study to assess the results of the gaze correction from different models. In detail, given an input face image in the CelebAGaze dataset, the user is presented with a pair of images, one showing the corrected gaze and the other showing the original face. The user is asked to indicate which image appears more natural and realistic. The results of the user study indicate that GazeGAN outperforms all other methods, achieving a higher user rating and a lower rating for the other methods. This further supports the effectiveness and the superiority of our proposed GazeGAN in gaze correction.
**Quantitative results:** The first two columns of Table 1 show the MSSSIM and LPIPS scores evaluating the preservation ability of the corrected images across different models. GazeGAN and PRGAN obtain the best results with 1.0 for MSSSIM and 0.0 for LPIPS. In fact, the original irrelevant regions are integrated with the generated eye region in both models using binary masks. StarGAN, StarGANA and CycleGAN obtain the worse irrelevant region preservation scores. The last column of Table 1 shows the evaluation results of the user study. The user average vote for GazeGAN is 42.40%, which is higher than all the other methods, i.e., 3.40% higher than StarGAN, 6.67% higher than StarGANA, 15.00% higher than CycleGAN, 8.33% higher than PRGAN. Overall, the qualitative and quantitative evaluations demonstrate the effectiveness and the superiority of the proposed approach.

### 4.5 Gaze Animation

In Fig. 6 we show the gaze animation results using input images with an arbitrary gaze which are corrected to stare at the camera. We observe that the interpolation results are smooth and plausible in each row. Each column has a different gaze angle, but the identity
ℓ-As-Training Method, and the Latent Reconstruction Loss $\ell_{fp}$ with respect to GazeGAN W/O $B$ and GazeGAN, respectively. $r[0]$ and $r[1]$ mean the 1st and 2nd value of the $r$ feature vector, respectively. Column 3 shows the mean ($X$-coordinate) and the variance ($Y$-coordinate) of the content representation differences between GazeGAN W/O $C$ and GazeGAN.

Figure 8: A visualization of the angle (columns 1-2) and the content (column 3) representations. Columns 1-2 correspond to the results of GazeGAN W/O $B$ and GazeGAN, respectively. $r[0]$ and $r[1]$ mean the 1st and 2nd value of the $r$ feature vector, respectively. Column 3 shows the mean ($X$-coordinate) and the variance ($Y$-coordinate) of the content representation differences between GazeGAN W/O $C$ and GazeGAN.

Figure 9: A qualitative comparison between GazeGAN and GazeGAN W/O $A$. The first row shows the input images and next two rows are the corrected results of GazeGAN W/O $A$ and GazeGAN (Zoomed left eyes).

...information is overall preserved (e.g., the eye shape, the iris color). Next, we show gaze animation results with more various directions by extrapolating the features $r$, in addition to interpolation methods. As shown in Fig. 7, our model not only achieves high-quality interpolation results but also achieves gaze animation with more directions where the angles are out of the range between the input and corrected output.

4.6 Ablation Study

In this section, we conduct extensive ablation studies to investigate the three key components of the proposed GazeGAN, i.e., the Pretrained Autoencoder for content extraction, the Synthesis-As-Training Method, and the Latent Reconstruction Loss $\ell_{fp}$. For brevity, we refer to these components as $A$, $B$, and $C$, respectively.

Pretrained Autoencoder: We propose this pretrained model to extract the content features to guide the process of gaze correction while preserving the identity information. We observe that GazeGAN has a stronger ability to preserve identity information with respect to GazeGAN W/O $A$ in Fig. 9. To quantitatively evaluate this, we use $G_x$ to reconstruct the input image $x$ from the test dataset in the domain $X$ and we measure the differences between the input images and the correction results in the local eye regions by employing MSSSIM and LPIPS. Table 2 shows that GazeGAN achieves better scores than GazeGAN W/O $A$, validating our design motivation.

Training-As-Synthesis Method: The results in Fig. 6 have shown the effectiveness of this method for gaze animation. Additionally, we illustrate three groups of features $r$ extracted from $E_r$ with real test samples $y$, real test samples $x$, the corrected results $\hat{y}$ as input. In Fig. 8, the 1st and the 2nd column correspond to the results of GazeGAN W/O $B$, and GazeGAN respectively. We observe that the points corresponding to $\hat{y}$ are closer to the points corresponding to $x$ in the 2nd figure with respect to what happens in the 1st column, which indicates that features $r$ have a strong correlation with angle information.

Latent Reconstruction Loss $\ell_{fp}$: We use $G_y$ to complete the input images $y$ from the test data of $Y$ and we measure the differences between the input images and the results using MSSSIM and LPIPS. In Fig. 3, GazeGAN obtains higher scores than GazeGAN W/O $C$, which shows that $\ell_{fp}$ further improves the ability to preserve identity information. Additionally, we visualize the mean $\mu$ ($X$-coordinate) and the variance $\sigma$ ($Y$-coordinate) of the arithmetical differences $c^y - \hat{c}^y$ extracted from $G_x$ across all the test data from $Y$. As shown in the right of Fig. 8, we see that our full model GazeGAN has a lower variance than GazeGAN W/O $C$, as the loss encourages to reduce the differences of the content features.

5 CONCLUSION

In this paper we introduced a new gaze dataset in the wild, CelebA Gaze, which is characterized by a large diversity in the head poses, gaze angles and illumination conditions. Moreover, we presented a novel unsupervised model, GazeGAN, for gaze correction and animation, trained and tested on this dataset. GazeGAN generates more high-quality and precise gaze correction results than state-of-the-art methods. Furthermore, it can redirect the gaze into the desired direction, producing gaze animation by interpolation in the latent space. Future work includes extending the proposed model to gaze redirection tasks in videos.

ACKNOWLEDGMENTS

This work has been partially supported by the Italy-China collaboration project TALENT.
