A Hierarchical Generative Model for Eye Image Synthesis and Eye Gaze Estimation

Kang Wang Rui Zhao Qiang Ji
ECSE, Rensselaer Polytechnic Institute, Troy, NY, USA
{wangk10, zhaor, jiq}@rpi.edu

Abstract

In this work, we introduce a Hierarchical Generative Model (HGM) to enable realistic forward eye image synthesis, as well as effective backward eye gaze estimation. The proposed HGM consists of a hierarchical generative shape model (HGSM), and a conditional bidirectional generative adversarial network (c-BiGAN). The HGSM encodes eye geometry knowledge and relates eye gaze with eye shape, while c-BiGAN leverages on big data and captures the dependency between eye shape and eye appearance. As an intermediate component, eye shape connects knowledge-based model (HGSM) with data-driven model (c-BiGAN) and enables bidirectional inference. Through a top-down inference, the HGM can synthesize eye images consistent with the given eye gaze. Through a bottom-up inference, HGM can infer eye gaze effectively from a given eye image. Qualitative and quantitative evaluations on benchmark datasets demonstrate our model's effectiveness on both eye image synthesis and eye gaze estimation. In addition, the proposed model is not restricted to eye images only. It can be adapted to face images and any shape-appearance related fields.

1. Introduction

Human eye plays an important role in perceiving the world around us, expressing our intent, emotion, and communicating with each other. Human can infer rich information from the appearance of the eye and the eye gaze directions. Being able to synthesize realistic eye appearances and track eye gaze directions has numerous applications: videoconferencing [4], human computer interface [15], social attention [13, 29], virtual reality [8], graphical animation [31] and gaming [3] to name a few.

Realistic eye image synthesis is however challenging. Recent development of eye image synthesis can be divided into graphics-based and warping-based approaches. Graphics-based approaches [38, 39] leverage on graphical engine to render eye textures on top of a 3D eye model. The advantage is the full geometry control over the 3D eye model to simulate variations in terms of gaze direction, head orientation, etc. However, there is a gap between the synthesized eye texture distribution and real eye texture distribution. The overall synthesized eye looks like cartoon/animated image. On the other hand, warping-based methods take an eye image as input, and warp the input image to have a desired output appearance. Among them, [7] warped the image to be under different head orientations, and [41] further extended to different gaze directions and head orientations. In [33], the authors refined the input unrealistic images to be more realistic. And [1, 20] warped the input eye images to have specific gaze angles (gaze correction). Despite the realism inherited from the input image or additional model, such synthesis is task-dependent or subject-specific, it cannot produce sufficient eye geometry variations and appearance variations.

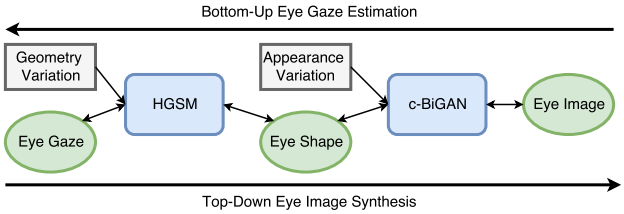


Figure 1. Overview of hierarchical generative model.

Existing gaze estimation approaches fall into appearance-based and model-based approaches. The appearance-based approaches [18, 43, 21] estimate eye gaze directly from eye appearance or extracted eye features. They need sufficient amount of data with groundtruth labels to produce accurate results. Model-based approaches [12, 34, 16, 37] leverage on anatomical eye model knowledge to estimate eye gaze from detected eye features. They are sensitive to the variation of personal eye parameters such as eyeball radius. A subject-specific personal calibration is often required for each person to estimate these personal eye parameters. They also require explicit detection of eye features such as pupil center and cornea reflections.

There is little work on a unified model for joint top-down eye image synthesis and bottom-up eye gaze estimation,

despite their close relationship. To address this issue, we introduce a novel hierarchical generative model to accomplish the goal. As shown in Fig. 1, we first introduce HGSM, a Bayesian hierarchical model inspired by eye anatomy knowledge, to relate eye gaze with eye shape. Through a top-down inference, it can synthesize eye shape consistent with the given eye gaze with subject eye geometry variations. Through a bottom-up inference, it can estimate eye gaze effectively from the given eye shape. The hierarchical perspective of the model is to account for subject-dependent eye geometry variations through the introduction of hyperparameters. Next we propose the c-BiGAN to relate eye shape with eye image. As a bidirectional model, c-BiGAN enables realistic eye image synthesis from eye shape with different appearance variations, as well as effective eye shape estimation from eye images. To summarize, our main contributions include:

- Combine knowledge-based Bayesian model (HGSM) and data-driven deep model (c-BiGAN) for unified eye image synthesis and eye gaze estimation.
- Enable synthesizing large amount of gaze-annotated eye images with subject eye geometry variations and appearance variations.
- Incorporate eye anatomy knowledge into eye gaze estimation, and achieve better generalization than pure data-driven approaches.
- Develop a generic top-down image synthesis framework, which can be applied beyond eye images, to any shape-appearance related fields.

2. Related Work

Image Synthesis. Image synthesis is an arising research topic [19, 25, 5]. The most related work to ours are the Generative Adversarial Network (GAN) [9, 28], its variant conditional-GAN [30], bidirectional-GAN [6] and [2, 22] for generating images consistent with high-level latent representations. Different from their applications like text-to-image synthesis, we focus on human eye synthesis conditioned on eye shape. Instead of performing pure data-driven image synthesis, we studied eyeball anatomy and 3D eye gaze model to explore how visual attention affects eye appearance. In addition, besides image synthesis, the proposed model also allows bottom-up inference from image to high-level latent representations.

For eye image synthesis, graphics-based methods [38, 39] can produce high resolution eye images, but the artificial eye texture makes them unrealistic. For warping-based methods, [41, 7] can synthesize additional eye images with different gaze directions and head poses. However, their per-image or per-subject based synthesis cannot produce large eye geometry and appearance variations. In [33], the authors proposed to warp eye images generated from graphics engine to make them more realistic. Because of the use of GAN and graphics engine generated eye images, they can synthesize realistic eye images with different variations. However, they lose the precise control over the gaze angles of the generated images as they directly inherit the gaze angles from the input images. On the contrary, our synthesis framework is from high level latent representation (gaze), and can synthesize eye images with arbitrary gaze angles. This is of significant importance in many applications. For example, we can synthesize meaningful eye movements like horizontal/vertical movement by providing the corresponding gaze angles. This is challenging and cumbersome for methods like [33], unless they search their input image database and find images with required gaze angles.

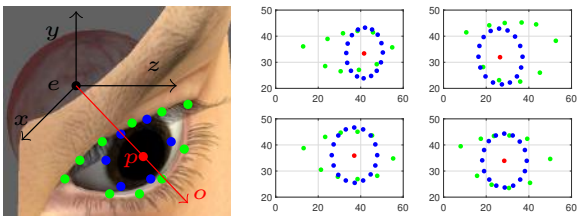
Gaze Correction. Another related research topic is gaze correction [1, 20], where the goal is to synthesize an eye image only differ by gaze from the input eye image, while keeping all other nuisance parameters the same. Despite the similarity in generating new eye images, our goal is, however, to synthesize large amount of eye images with variations in terms of subjects, gaze directions, appearances, etc.

Gaze Estimation. Traditional model-based methods [12, 36, 16, 42] leverage on 3D eye model to estimate eye gaze with detected features like pupil center, corneal reflections, facial landmarks, etc. They can achieve good accuracy in controlled environments but cannot adapt to challenging in-the-wild scenarios. Recent approaches [43, 21] tackle the challenge by training deep neural networks on large amount of eye images and show promising results. However, their model cannot generalize well on data with different distributions (cross-dataset). One similar work to ours is [24], where the authors also built a generative model to capture the dependence between eye gaze and eye image. However, besides replacing its simple tri-color appearance model with a GAN that can synthesize realistic appearances, we also make significant extensions to their shape model: 1) We introduce a hierarchical model to model subject eye geometry variations, while [24] does not involve any hierarchy and can only handle one specific subject. As a result, we can build one unified hierarchical model and apply it to all subjects, while [24] needs to train many subject-specific models. 2) We employ a comprehensive eye geometry model with 27 eye shape points, while [24] only includes 4 control points. We can therefore better capture gaze-shape correlation and control the deformations of eye shape.

3. Model Description

3.1. Hierarchical Generative Shape Model

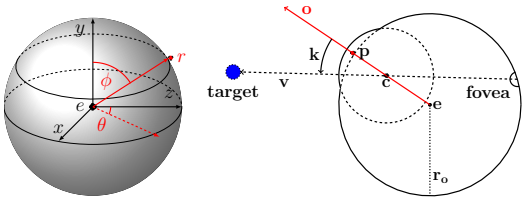
We identify 27 eye related landmarks to represent eye shape, including 10 eyelid points, 16 iris contour points and 1 pupil center. For top down eye shape synthesis, we first



(a) (b)

Figure 2. (a) 3D eye shape with eyelid points (green), iris contour points (blue) and pupil center (red). (b) Projected 2D eye shapes with different gaze directions (best view in color).

use 3D eye geometry model to construct 3D eye shape as in Fig. 2(a), 2D eye shape in Fig. 2(b) can be obtained by projecting 3D eye shape to image plane.



(a) (b)

Figure 3. (a) Eyeball coordinates system (ECS). (b) Anatomical eye geometry model to relate eye gaze and pupil center.

The 3D eye shape is represented in Eyeball Coordinate system (ECS) (Fig. 3 (a)), where eyeball center is located at the origin. A point $\mathbf{t} \in \mathcal{R}^{3\times 1}$ in ECS can be represented by $[x_t, y_t, z_t]^T$ in Euclidean coordinates or $[r_t, \phi_t, \theta_t]^T$ in spherical coordinates. They are related by the transformation function $\mathbf{f}(\cdot)$ such that $[x_t, y_t, z_t]^T = \mathbf{f}([r_t, \phi_t, \theta_t]^T) = [r_t \sin(\phi_t)\sin(\theta_t), r_t \cos(\phi_t), r_t \sin(\phi_t)\cos(\theta_t)]^T$. In this work, the angles ϕ_t and θ_t represent pitch and yaw angles respectively, and r_t represents eyeball radius.

3.1.1 Gaze-to-Shape Synthesis

Eye shape synthesis depends on personal eye parameters. To avoid constructing subject-dependent models, we propose one hierarchical model (Fig. 4) to capture the gaze-to-shape synthesis process for all $\mathcal{M}$ subjects. The personal eye parameters $\{\phi_k, \theta_k, r_o\}$ are regarded as random variables, whose distributions are controlled by their hyperparameters $\{\hat{\phi}_k, \hat{\theta}_k, \hat{r_o}\}$.

Specifically considering the shape generation process of a particular subject, we first need to determine the personal eye parameters, which can be drawn from their prior distributions. Then given personal parameters and eye gaze, we generate 3D eye shape. Finally, we project 3D eye shape to 2D eye shape. Now we discuss in details of each step.

Pupil center generation. Pupil center reveals the most

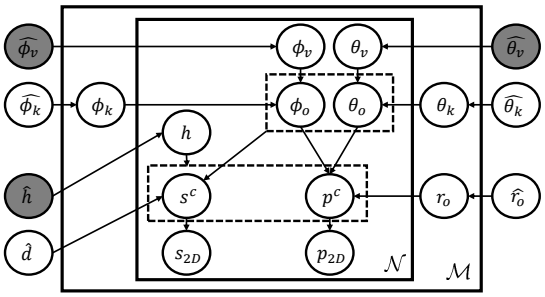


Figure 4. Graphical model of eye shape synthesis process, where the shaded nodes are known during learning and inference. $\mathcal{M}$ represents number of subjects and $\mathcal{N}$ represents number of images per subject.

important information about human attention. We systematically study eyeball anatomy to relate pupil center with human attention. As shown in Fig. 3(b), supposing human's attention is drawn by a visual target, the eyeball will rotate towards the target so that the field of vision will be focused in the fovea region. Fovea is a small depression in the retina of the eye where visual acuity is the highest. The line of sight passing fovea and visual target is called visual axis (eye gaze) $\mathbf{v} = [\phi_v, \theta_v]$. However, the pupil position is not directly determined by $\mathbf{v}$, but rather the optical axis $\mathbf{o} = [\phi_o, \theta_o]$, which can be obtained by subtracting the angle difference $\mathbf{k} = [\phi_k, \theta_k]$ from visual axis. The angle difference $\mathbf{k}$, as well as the eyeball radius r_o, are both subject-dependent parameters. Finally, pupil center in ECS can be represented as $\mathbf{p}^e = (r_o, \phi_o, \theta_o)$.

Eyelid and iris contour generation. Eyelids and iris contour points are correlated with each other to form reasonable and realistic human eye shapes. In addition, eyelids and iris contour also strongly correlate with eye gaze. For instance, upper eyelids will go up as we look up, and iris contour will always move towards the gaze direction. To encode eye shape correlation and its dependence on eye gaze, we propose a linear Gaussian regression model such that continuous eye gaze is the independent variable $\mathbf{o}$ and the continuous eye shape vector is the dependent variable $\mathbf{s}^e$:

$$\mathbf{s}^e = \mathbf{A}\mathbf{o} + \mathbf{b} + \epsilon \tag{1}$$

where $\mathbf{s}^e$ represents the coordinates of eyelids and iris contour points in ECS, $\mathbf{A}$ is the regression matrix, $\mathbf{b}$ is the bias vector and $\epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{\Sigma})$ is the noise term, which accounts for shape variations under the same gaze direction. In practice, we only care about dominant shape variations, therefore $\epsilon = \mathbf{B}\boldsymbol{\tau}$, where $\mathbf{B}$ contains the dominant eigenvectors of covariance matrix $\mathbf{\Sigma}$, and $\boldsymbol{\tau}$ is the coefficient. We denote the linear Gaussian regression model as $\hat{\mathbf{d}} = \{\mathbf{A}, \mathbf{B}, \mathbf{b}, \hat{\boldsymbol{\tau}}\}$, where $\hat{\boldsymbol{\tau}}$ is the prior of the coefficients $\boldsymbol{\tau}$.

2D eye shape synthesis. Given one point in ECS $\mathbf{t}^e$ with its Euclidean coordinates, we first get its coordinates

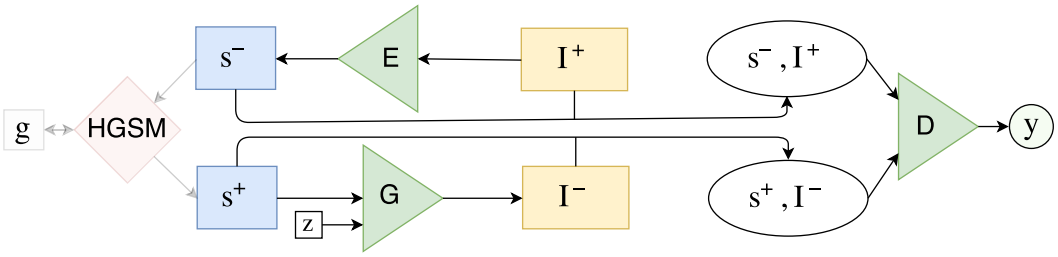


Figure 5. Flowchart of c-BiGAN. The left opague part shows its relationship with the HGSM. $\mathbf{z}$ denotes the random vector input in GAN, $\mathbf{s}$ and $\mathbf{I}$ represent eye shape and eye image, $\{\mathbf{G}, \mathbf{E}, \mathbf{D}\}$ represent generator, encoder and discriminator respectively. $\mathbf{y}$ denotes the probability that the shape and image tuple ($\mathbf{s}$, $\mathbf{I}$) is true/consistent.

in camera coordinate system (CCS) $\mathbf{t}^c = \mathbf{R}\mathbf{t}^e + \mathbf{e}$ with head pose $\mathbf{h} = \{\mathbf{R}, \mathbf{e}\}$. The 2D eye shape (see Fig. 2 (b)) can be obtained by projecting $\mathbf{t}^c$ using the camera intrinsic parameters $\mathbf{W}$, assuming orthographic projection.

Algorithm 1: Eye Shape Synthesis from Eye Gaze

1. Input: hyper-parameters: $\{\hat{\phi_v}, \hat{\theta_v}, \hat{h}, \hat{\phi_k}, \hat{\theta_k}, \hat{r_o}, \hat{d}\}$
2. Output: 2D eye shape $\mathbf{s}_{2D}$
3. Specify subject by drawing personal parameters: $\phi_k = \mathcal{N}(\hat{\phi_k}, \sigma_k), \theta_k = \mathcal{N}(\hat{\theta_k}, \sigma_k), r_o = \mathcal{N}(\hat{r_o}, \sigma_{r_o})$
4. Draw visual axis and head pose $\mathbf{h} = \{\mathbf{R}, \mathbf{e}\}$: $\phi_v = \mathcal{U}(\phi_l, \phi_h), \theta_v = \mathcal{U}(\theta_l, \theta_h), \mathbf{h} = \mathcal{U}(\mathbf{h}_l, \mathbf{h}_h)$
5. Optical axis: $\mathbf{o} = [\phi_o, \theta_o] = [\phi_v - \phi_k, \theta_v - \theta_k]$
6. Pupil center in ECS and CCS: $\mathbf{p}^e = \mathbf{f}(r_o, \phi_o, \theta_o), \mathbf{p}^c = \mathbf{R}\mathbf{p}^e + \mathbf{e}$
7. Eyelids and iris contour in ECS and CCS: $\boldsymbol{\tau} = \mathcal{N}(\hat{\boldsymbol{\tau}}, \sigma_{\boldsymbol{\tau}}), \mathbf{s}^e = \mathbf{A}\mathbf{o} + \mathbf{B}\boldsymbol{\tau} + \mathbf{b}, \mathbf{s}^c = \mathbf{R}\mathbf{s}^e + \mathbf{e}$
8. 2D eye shape: $\mathbf{s}_{2D} = \frac{1}{\lambda}\mathbf{W}\mathbf{s}^c, \mathbf{p}_{2D} = \frac{1}{\lambda}\mathbf{W}\mathbf{p}^c$

Mathematical formulation. To summarize, the overall top-down shape synthesis process is outlined in Alg. 1, where $\mathcal{N}(\cdot)$ represents Gaussian distribution. Linear Gaussian regression model and hyperparameters can both be learned from data. For visual axis and head pose, their values can be drawn from an uniform distribution $\mathcal{U}(\cdot)$.

3.2. Conditional-BiGAN

We propose a conditional-BiGAN (Fig. 5), which is inspired by two recent work conditional-GAN (Reed *et al.* [30]) and bidirectional-GAN (Donahue *et al.* [6]).

The motivation is two-fold. First, we want to synthesize gaze-annotated eye images, and naturally we would treat eye gaze as condition. However, we want to explicitly incorporate subject eye geometry variations (through HGSM), therefore we treat the intermediate eye shape as our condition. Actually, shape-to-image is more generic than task-dependent label-to-image synthesis, and allows us to generalize the image synthesis framework to different fields. Second, we also want to infer eye gaze from eye image. The bidirectional idea in [6] inspired us to consider BiGAN to recover conditions from images. Differently, [6] is used for feature learning to recover the latent representation $\mathbf{z}$ from images, while we want to recover the condition (eye shape $\mathbf{s}$) from images. Therefore we construct the shape-image tuple $(\mathbf{s}, \mathbf{I})$, which is fed to the Discriminator D and output the probability y, the probability is high only if $\mathbf{I}$ is realistic and $\mathbf{s}$, $\mathbf{I}$ are consistent with each other. When learning is finished, the generator G can map eye shape and the random vector to eye image ($\mathbf{I} = G(\mathbf{z}, \mathbf{s}; \mathbf{w}_g)$), while the encoder E can map eye image back to eye shape ($\mathbf{s} = E(\mathbf{I}; \mathbf{w}_e)$).

4. Learning and Inference

4.1. HGSM Parameter Learning

We first learn the linear Gaussian regression model parameters $\hat{\mathbf{d}}$ from 3D eye shapes $\{\mathbf{s}_i^e\}_{i=1}^N$ and corresponding optical axes $\{\mathbf{o}_i\}_{i=1}^N$. The data is extracted from the UnityEye dataset [39]. $\mathbf{A}$ and $\mathbf{b}$ can be easily solved with least square regression: $\{\mathbf{A}^*, \mathbf{b}^*\} = \sum_{i=1}^N ||\mathbf{s}_i^e - \mathbf{A}\mathbf{o}_i - \mathbf{b}||^2$. The regression error for each data sample $(\mathbf{s}_i^e - \mathbf{A}^*\mathbf{o}_i - \mathbf{b}^*)$ is concatenated into a large matrix $\mathbf{L}$. We perform a PCA analysis on $\mathbf{L}$ and only keep the first K dominant eigenvectors to form the bases matrix $\mathbf{B}$. The mean and variance of the corresponding coefficients from PCA analysis are computed to determine $\hat{\boldsymbol{\tau}}$.

To learn the hyperparameters $\boldsymbol{\alpha} = \{\hat{\phi_k}, \hat{\theta_k}, \hat{r_o}\}$, we first specify their prior distributions, i.e. hyperpriors: $p(\boldsymbol{\alpha}) = \prod_{i=1}^3 \mathcal{N}(\alpha_i | b_i, \sigma_i^2)$, where α_i is one of the three hyperparameters, b_i and σ_i^2 represent the mean and variance for the hyperpriors. In our work, b_i is set to empirical human average values [12] and σ_i^2 is set to a fixed large value to account for the variations in the entire population. We can then solve $\boldsymbol{\alpha}$ by maximizing its posterior: $\boldsymbol{\alpha}^* = \arg\max_{\boldsymbol{\alpha}} \prod_{i=1}^N p(\{\mathbf{s}_{2D}^i, \mathbf{p}_{2D}^i\} | \boldsymbol{\alpha}) P(\boldsymbol{\alpha})$.

Solving the problem analytically is intractable because of the hierarchical architecture and the presence of intermediate latent variables. To address this challenge, we propose an iterative algorithm as in Alg. 2. Hyperparameters are first initialized by sampling from their prior distribution. Given hyperparameters and observations, we leverage on numerical sampling method, the No-U-Turn Sampler (NUTS) [14], to

Algorithm 2: Hyperparameter Learning Algorithm

1. Input: Prior $p(\boldsymbol{\alpha})$ and data from M subjects: $D = \{D_j\}_{j=1}^{M}$, where $D_j = \{\mathbf{s}_{2D}^{i,j}, \mathbf{p}_{2D}^{i,j}\}_{i=1}^{N_j}$, N_j is the number of samples from j^{th} subject.
2. Output: $\boldsymbol{\alpha} = \{\hat{\phi}_k, \hat{\theta}_k, \hat{r_o}\}$.
3. Initialization: Sample hyperparameters $\boldsymbol{\alpha}^0$ from their prior $p(\boldsymbol{\alpha})$.
while *not converge* **do**
  - **for** *subject j in {1, ..., M}* **do**
    -Estimate subject-dependent parameters $\boldsymbol{\beta}_j^t = \{\phi_k, \theta_k, r_o\}_j^t$ given $\boldsymbol{\alpha}^t$ and D_j.
    $\boldsymbol{\beta}_j^t = \arg\max_{\boldsymbol{\beta}_j} p(\boldsymbol{\beta}_j|\boldsymbol{\alpha}^t)p(D_j|\boldsymbol{\beta}_j)$
  -Update hyperparameters $\boldsymbol{\alpha}^{t+1}$ given parameters $\{\boldsymbol{\beta}_j^t\}_{j=1}^{M}$ and prior $p(\boldsymbol{\alpha})$:
  $\boldsymbol{\alpha}^{t+1} = \arg\max_{\boldsymbol{\alpha}} p(\boldsymbol{\alpha})\Pi_{j=1}^{M} p(\boldsymbol{\beta}_j^t|\boldsymbol{\alpha})$

estimate the parameters. Given parameters $\{\phi_k, \theta_k, r_o\}$ and the prior $p(\boldsymbol{\alpha})$, hyperparameters $\boldsymbol{\alpha}$ can be updated correspondingly. The algorithm will iterate until convergence.

4.2. c-BiGAN Parameter Learning

We first consider following 4 different shape-image tuples:

- Matched real shape and real image: $(\mathbf{s}^+, \mathbf{I}^+)$
- Mismatched wrong shape and real image: $(\mathbf{s}^w, \mathbf{I}^+)$
- Real shape and synthesized image: $(\mathbf{s}^+, \mathbf{I}^-)$
- Inferred shape and real image: $(\mathbf{s}^-, \mathbf{I}^+)$

where the first two tuples are selected from the training dataset, while the last two tuples are generated from the model as in Fig. 5.

For discriminator D, only the first tuple $(\mathbf{s}^+, \mathbf{I}^+)$ is true while the other 3 tuples are false, the discriminator can be learned by maximumly distinguishing true and false tuples. For generator G, we want the synthesized image $\mathbf{I}^-$ or the third tuple $(\mathbf{s}^+, \mathbf{I}^-)$ to be true, and generator can be learned by maximizing the true probability. For encoder E, we first want the inferred $\mathbf{s}^-$ or the fourth tuple $(\mathbf{s}^-, \mathbf{I}^+)$ to be true, but we also want the encoder to reconstruct the shape accurately, therefore we also introduce a reconstruction loss term besides the adversarial loss term. Overall, the c-BiGAN training is summarized in Alg. 3. The Discriminator, Generator and Encoder are updated alternatively until final convergence.

4.3. Top-down Eye Image Synthesis

Eye shape inference from eye gaze. We can first sample the prior probability of gaze to generate an eye gaze direction. We can then follow Alg. 1 to synthesize 2D eye shape for the gaze direction.

Algorithm 3: c-BiGAN Parameter Learning

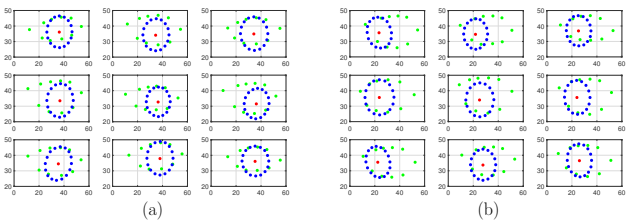
1. Input: batches of real image $\mathbf{I}^+$, matched real shape $\mathbf{s}^+$, mismatched wrong shape $\mathbf{s}^w$, learning rate α and balance factor λ.
2. Initialization: weight $\{\mathbf{w}_d, \mathbf{w}_g, \mathbf{w}_e\}$.
3. **while** *not converge* **do**
  - $\mathbf{z} \sim \mathcal{N}(0, 1)$: draw random vector
  - $\mathbf{I}^- = G(\mathbf{z}, \mathbf{s}^+; \mathbf{w}_g)$: eye image synthesis
  - $\mathbf{s}^- = E(\mathbf{I}^+; \mathbf{w}_e)$: eye shape inference
  - Generate probability for the 4 shape-image tuples: $p^r = D(\mathbf{s}^+, \mathbf{I}^+; \mathbf{w}_d)$; $p^f = D(\mathbf{s}^w, \mathbf{I}^+; \mathbf{w}_d)$; $p^I = D(\mathbf{s}^+, \mathbf{I}^-; \mathbf{w}_d)$; $p^s = D(\mathbf{s}^-, \mathbf{I}^+; \mathbf{w}_d)$
  - Discriminator loss: $\mathcal{L}_D = \log(p^r) + \log(1 - p^f) + \log(1 - p^I) + \log(1 - p^s)$
  - $\mathbf{w}_d \leftarrow \mathbf{w}_d - \alpha\partial\mathcal{L}_D/\partial\mathbf{w}_d$
  - Generator loss: $\mathcal{L}_G = \log(p^I)$
  - $\mathbf{w}_g \leftarrow \mathbf{w}_g - \alpha\partial\mathcal{L}_G/\partial\mathbf{w}_g$
  - Encoder loss: $\mathcal{L}_E = \log(p^s) + \lambda||\mathbf{s}^- - \mathbf{s}^+||^2$
  - $\mathbf{w}_e \leftarrow \mathbf{w}_e - \alpha\partial\mathcal{L}_E/\partial\mathbf{w}_e$

Eye image synthesis from eye shape. Given 2D eye shape $\mathbf{s}$ and random vector $\mathbf{z}$, we can synthesize the eye image through the generator G of c-BiGAN (Sec. 3.2): $\mathbf{I} = G(\mathbf{z}, \mathbf{s}; \mathbf{w}_g^*)$, where $\mathbf{w}_g^*$ are the learned generator parameters.

4.4. Bottom-up Eye Gaze Inference

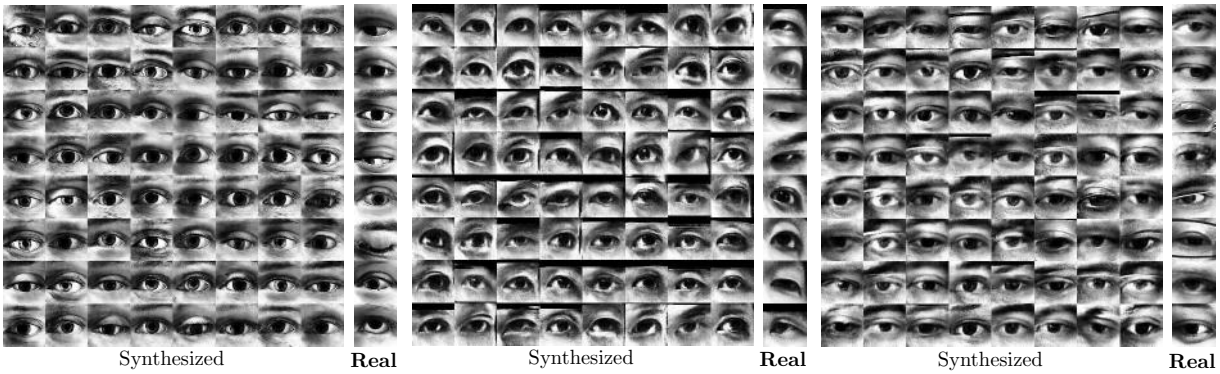
Inferring eye shape from eye image. Given an observed real eye images $\mathbf{I}$, we can estimate the eye shape with the encoder E of c-BiGAN (Sec. 3.2): $\mathbf{s} = E(\mathbf{I}; \mathbf{w}_e^*)$, where $\mathbf{w}_e^*$ are the learned encoder parameters.

Inferring eye gaze from eye shape. Given eye shape, we are able to infer the two angles $\{\phi_v, \theta_v\}$ of eye gaze (Fig. 4). Define $\eta := \{\phi_k, \theta_k, h, s^c, p^c, \phi_o, \theta_o, r_o\}$, $\pi := \{\hat{\phi}_k, \hat{\theta}_k, \hat{r}_o, \hat{d}, \hat{h}\}$. We can perform MAP inference to get gaze angles: $\{\phi_v^*, \theta_v^*\} = \arg\max_{\phi_v,\theta_v} p(\phi_v, \theta_v|\mathbf{s}_{2D}, \mathbf{p}_{2D}, \hat{\phi}_v, \hat{\theta}_v, \pi) = \arg\max_{\phi_v,\theta_v} \int_\eta p(\mathbf{s}_{2D}, \mathbf{p}_{2D}|\phi_v, \theta_v, \eta)p(\phi_v, \theta_v, \eta|\hat{\phi}_v, \hat{\theta}_v, \pi)d\eta$. Similar to parameter learning, we use NUTS to draw samples of $\{\phi_v, \theta_v\}$ from its posterior, and the sample mean is used as final estimation.



(a) (b)

Figure 6. Synthesized eye shapes given eye gaze. (a) Different eye shapes with same frontal gaze direction and (b) Different eye shapes with same gaze direction (looking left).



(a) UnityEye (b) UTMultiview (c) MPIIGaze

Figure 7. Synthesized eye images with same gaze (pitch=yaw=0). For reference, right side of each sub-figure shows real images with different gaze angles.

5. Experiments and Analysis

We consider three benchmark datasets UnityEye [39], UTMultiview[41] and MPIIGaze[43]. We extract 180K images from 60 subjects in UnityEye, 128K images from 50 subjects in UTMultiview, and 214K images from 15 subjects in MPIIGaze. Sample images can be seen on right side of each sub-figure in Fig. 7. For HGSM, PyMC3 [26] is used for NUTS as inference engine to perform sampling. All eye images are resized to 64×64. The groundtruth landmarks/shape are provided in UnityEye, but are not in MPIIGaze and UTMultiview. Their shape are alternatively detected by adapting the approach in [10]. For c-BiGAN, our implementation is based on *DCGAN-tensorflow*[1]. The generator has 3 standard deconvolutional layers, the encoder has 5 convolutional layers and 3 fully connected layers, and the discriminator has 4 layers of stride-2 convolutions. Leaky-RELU activation and batch normalization are also used. We typically obtain realistic eye images after 20 epochs.

5.1. Top-down Eye Image Synthesis

5.1.1 Qualitative Evaluation

Eye geometry variation. Subject-dependent eye geometry variation is caused by different eyeball radius, different eyelid shapes, etc. By keeping the same gaze direction, and sampling coefficients from prior $\hat{\boldsymbol{\tau}}$, we are able to generate different eye shapes as illustrated in Fig. 6. This helps introduce diversity in the synthesized eye images.

Appearance variation. The appearance variation can be introduced by sampling the noise vector $\mathbf{z}$. Given the same shape vector, we can obtain eye images with different appearances as shown in Fig. 7. Appearance variations are very important in gaze-annotated image synthesis. We want not only accuracy (matching between gaze and image), but also diversity to approximate real eye image distributions.

[1]https://github.com/carpedm20/DCGAN-tensorflow

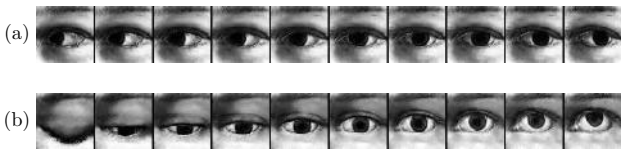


Figure 8. Eye movement synthesis.(a) Horizontal movement from left to right and (b) vertical movement from close to open. See supplementary materials for animated gif images.

Gaze annotation. To demonstrate synthesizing gaze-annotated eye shapes/images, we uniformly sample 8 pitch angles from $[-20°, 30°]$ and 8 yaw angles from $[-30°, 30°]$, the corresponding synthesized shapes/images on the 3 datasets are shown in Fig. 9. Notice due to the randomness in the input vector $\mathbf{z}$, there might be some poor samples like (row 8, col 7) in Fig. 9(a) and (row 7, col 2) in Fig. 9(b). But overall, the synthesized eye images and eye shapes are well matched, as well as the given gaze angles.

Eye movement synthesis. By providing a sequence of gaze angles as conditions, we can easily synthesize sequences of eye movement as shown in Fig. 8. The model is trained on UnityEye to cover large range of gaze angles. Notice we use the same random vector $\mathbf{z}$ for all images, so that the sequence looks like from the same subject.

5.1.2 Quantitative Evaluation on Gaze-Annotation

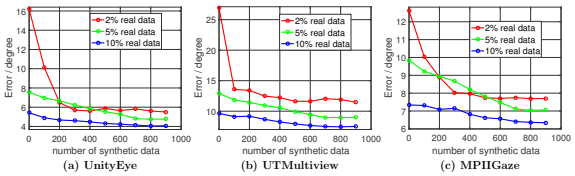
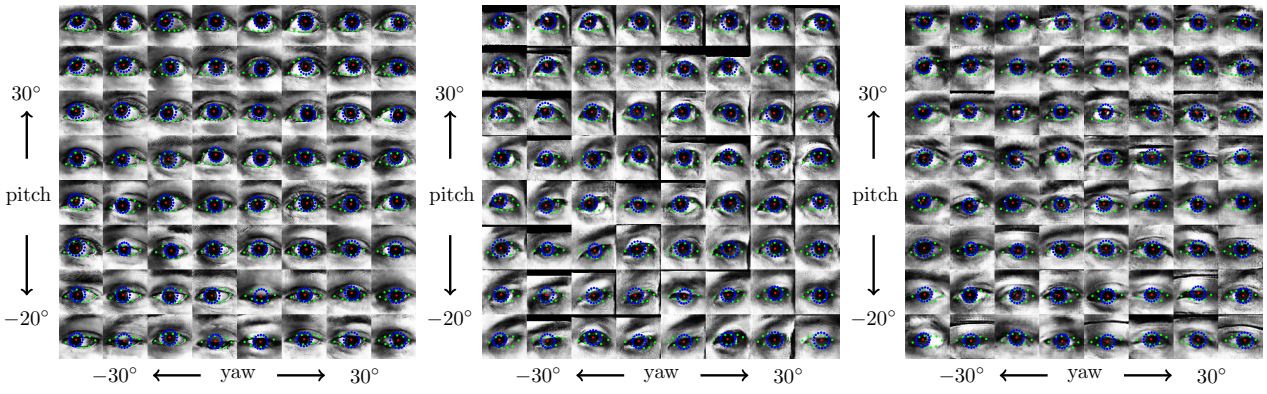


Figure 10. Gaze estimation error with real and synthetic data (best view in color).

We first evaluate whether synthesized eye images help



(a) UnityEye (b) UTMultiview (c) MPIIGaze

Figure 9. Synthesized eye shapes and eye images given different gaze directions (best view in color).

gaze estimation with small amount of real data. We train a gaze estimator (LeNet in [43]) on synthesized image-gaze pairs and test on reserved 10^4 real eye images. Training data consists of $k\%$ of 10^4 real images and n synthetic eye images. As shown in Fig. 10, the model is under-fitted with 2% real data which cause poor performance. However, as we continuously add synthetic eye images, the performance keeps improving until saturation. Final error reduces about 53.4% compared to no synthetic images. If we increase the number of real data to 5% and 10%, the improvement is less significant, with 32.1% and 20.3% performance gain. This demonstrates the synthesized eye images indeed capture the correlation between eye gaze, eye shape and eye appearance, and can be helpful for applications with small amount of real data or less well annotated data.

Table 1. Comparison with [33] on image synthesis

Dataset	MPIIGaze		UTMultiview	
Method	[33]	Ours	[33]	Ours
Error / degree	7.8	7.6	8.9	8.8

Next we compare with state-of-the-art method [33] on eye image synthesis. The image synthesis models are learned on MPIIGaze and UTMultiview separately. The synthesized images are used to train the same gaze estimator and test on remaining images from the two datasets. As shown in Tab. 1, we achieve comparable results as [33] for both datasets. The reason is that both methods use GAN as the key component to generate images. However, compared to [33], besides synthesizing similar quality images, the proposed method yields additional advantages:

- Our model is a generic two-step synthesis framework, from high-level latent representation to shape and from shape to image, while [33] is based on refining existing images and is restricted to synthetic images.
- Direct image synthesis from latent representations enables broader applications, like the eye movement synthesis in Fig. 8, but this is difficult for [33].
- The proposed method also supports effective bottom-up gaze inference, which is infeasible for [33].

5.2. Bottom-up Eye Gaze Estimation

5.2.1 Evaluation of Eye Shape Inference

Table 2. Comparison of eye shape prediction error.

Method / # training	1000	3000	10000
[11]	2.7	2.3	1.8
Proposed	4.5	3.1	1.4

We first evaluate the detection error of the 27 eye landmarks on UnityEye. As shown in Tab. 2, with smaller number of training samples (1000 and 3000), [11] achieves better results, but we can achieve much better results (1.4 pixel) with more training data (10000), demonstrating the effectiveness of the proposed shape estimation method.

Table 3. Comparison of pupil detection rate.

Method	$d_{eye} \leq 0.05$	$d_{eye} \leq 0.10$	$d_{eye} \leq 0.25$
[11] (R)	91.2%	99.4%	99.8%
Proposed (R)	90.3%	99.3%	99.9%
Proposed (R+S)	92.1%	99.9%	99.9%

Next we compare the pupil detection rate on benchmark dataset BioID [17]. In Tab. 3, R and S denote real and synthetic images respectively. When trained with 4000 real samples (second row in Tab. 3), we achieve reasonable detection rate but is not as good as [11]. However, the proposed method outperforms [11] on all three metrics when we add 10000 synthesized image-shape pairs (third row in Tab. 3). This not only demonstrates the effectiveness of shape estimation, but also shows the potential of using synthesized image-shape pairs to learn shape estimator.

5.2.2 Comparison with Appearance-based Methods

We compare with the state-of-the-art appearance-based method [43] and a variant of the proposed method (G-c-

Table 4. Comparison with appearance-based methods on within-dataset and cross-dataset experiments.

Dataset	Category	CNN[43]	G-c-BiGAN	HGM
UnityEye	within	3.4	5.6	4.5
	cross	19.5	21.3	15.9
UTMultiview	within	6.8	7.7	8.4
	cross	18.3	19.9	14.2
MPIIGaze	within	6.1	8.5	7.5
	cross	12.3	13.9	7.7

BiGAN), where we train c-BiGAN conditioned on eye gaze and directly infer eye gaze given eye images. We perform both within-dataset and cross-dataset experiments. Cross-dataset means the model is trained on the rest two datasets other than the testing dataset.

As shown in Tab. 4, compared to deep CNN model, our HGSM is a shallow model with much fewer parameters (the deep c-BiGAN only provides intermediate eye shape), thus it cannot perform as well as CNN on within-dataset experiments. The reason is that CNN tends to over-fit on the dataset with large amount of parameters. However, CNN cannot generalize well on cross-dataset experiments, while HGSM encodes universal eye model knowledge and can generalize better than CNN for cross-dataset evaluation. Similarly to CNN, HGM also outperforms the data-driven based G-c-BiGAN, as it directly captures the gaze-appearance correlation, without taking eye shape, which encodes eye model knowledge, into consideration.

5.2.3 Comparison with Model-based Methods

Table 5. Comparison with model-based methods.

Dataset	MPIIGaze		EyeDiap		
Methods	[40]	Ours	[23]	[35]	Ours
Error / degree	47.1	7.5	18.3	17.3	15.2

We also compare with two state-of-the-art model-based methods on within dataset experiments as in Tab. 5. [40] performs poorly on MPIIGaze, and we also outperforms [23, 35] on EyeDiap. Their methods rely on detected features like iris contour, pupil center and facial landmarks. These features are heavily affected by illumination variations, large head poses or limited image resolution, which causes poor feature/edge detections. Our eye shape is inferred from the global eye appearance, which is more robust to the mentioned variations. In addition, our HGSM is a probabilistic model that uses prior information and observed data to jointly perform gaze inference, which is more robust and accurate than deterministic model-based methods.

5.3. Beyond Eye Images

The proposed HGM can also be extended to face image synthesis. We modify HGSM to generate 2D face shape from a 3D deformable face model [27], and experimentally trained the c-BiGAN with 2.8K face images from [32]. After training, we can sample 2D shapes from the deformable model and synthesize corresponding face images as shown in Fig. 11. The quality of the synthesized face can be improved with more training data and advanced synthesis framework. Here we demonstrate that the synthesized images align well with the given shape, even with large pose variations ((Row, Col) = (1, 1), (1, 3), (4, 6), (4, 8), etc) or facial expressions ((6, 2), (8, 1), etc). In fact, shape-related high-level latent representations (pose, expression, etc) can be seamlessly incorporated into the HGM framework and bring in more control for task-oriented image synthesis.

Figure 11. Synthesized face images given facial landmarks.

6. Conclusion

To summarize, we propose a hierarchical generative model for eye image synthesis and eye gaze estimation. With a top-down inference, we are able to synthesize realistic gaze-annotated eye images that reflect eye geometry and appearance variations. Quantitatively compared to state-of-the-art image synthesis methods, we can achieve comparable gaze estimation accuracy, but with a more generalized framework and with less restrictions. With a bottom-up inference, we are able to predict eye gaze accurately from eye images. Benefited from combining knowledge-based Bayesian model with data-driven deep model, the proposed method gives better accuracy than model-based methods and shows better generalization capability than appearance-based methods. Finally, the proposed two-step image synthesis framework can be generalized beyond eye images, demonstrating potential applications for shape-image related fields.

Acknowledgements: This work was supported in part by a National Science Foundation grant (IIS 1539012) and in part by RPI-IBM Cognitive Immersive Systems Laboratory (CISL), a center in IBM's AI Horizon Network.

References

[1] Y. G. B, D. Kononenko, D. Sungatullina, and V. Lempitsky. DeepWarp : Photorealistic Image Resynthesis for Gaze Manipulation. *ECCV*, 2016. 1, 2

[2] X. Chen, Y. Duan, R. Houthooft, J. Schulman, I. Sutskever, and P. Abbeel. InfoGAN: Interpretable Representation Learning by Information Maximizing Generative Adversarial Nets. *arXiv*, 2016. 2

[3] P. M. Corcoran, F. Nanu, S. Petrescu, and P. Bigioi. Real-time eye gaze tracking for gaming design and consumer electronics systems. *TCE*, 2012. 1

[4] A. Criminisi, J. Shotton, A. Blake, and P. H. S. Torr. Gaze manipulation for one-to-one teleconferencing. *ICCV*, 2003. 1

[5] E. Denton, S. Chintala, A. Szlam, and R. Fergus. Deep Generative Image Models using a Laplacian Pyramid of Adversarial Networks. *Arxiv*, 2015. 2

[6] J. Donahue, P. Krähenhübl, and T. Darrell. Adversarial feature learning. *ICLR*, 2017. 2, 4

[7] L. Feng, Y. Sugano, T. Okabe, and Y. Sato. Gaze Estimation From Eye Appearance: A Head Pose-Free Method via Eye Image Synthesis. *TIP*, 2015. 1, 2

[8] M. Garau, M. Slater, V. Vinayagamoorthy, A. Brogni, A. Steed, and M. A. Sasse. The impact of avatar realism and eye gaze control on perceived quality of communication in a shared immersive virtual environment. In *SIGCHI*, 2003. 1

[9] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative Adversarial Nets. *NIPS*, 2014. 2

[10] C. Gou, Y. Wu, K. Wang, F.-Y. Wang, and Q. Ji. Learning-by-synthesis for accurate eye detection. In *ICPR*, 2016. 6

[11] C. Gou, Y. Wu, K. Wang, K. Wang, F.-Y. Wang, and Q. Ji. A joint cascaded framework for simultaneous eye detection and eye state estimation. *Pattern Recognition*, 67:23–31, 2017. 7

[12] E. D. Guestrin and M. Eizenman. General theory of remote gaze estimation using the pupil center and corneal reflections. *TBE*, 2006. 1, 2, 4

[13] Q. Guillon, N. Hadjikhani, S. Baduel, and B. Rogé. Visual social attention in autism spectrum disorder: Insights from eye tracking studies. *Neuroscience & Biobehavioral Reviews*, 42:279–297, 2014. 1

[14] M. D. Homan and A. Gelman. The no-u-turn sampler: Adaptively setting path lengths in hamiltonian monte carlo. *The Journal of Machine Learning Research*, 2014. 4

[15] T. E. Hutchinson, K. P. White Jr, W. N. Martin, K. C. Reichert, and L. A. Frey. Human-computer interaction using eye-gaze input. *Systems, Man and Cybernetics, IEEE Transactions on*, 1989. 1

[16] T. Ishikawa. Passive driver gaze tracking with active appearance models. 2004. 1, 2

[17] O. Jesorsky, K. J. Kirchberg, and R. W. Frischholz. Robust Face Detection Using the Hausdorff Distance. *In Proc. Third International Conference on Audio- and Video-based Biometric Person Authentication*, 2001. 7

[18] K. -H. Tan, D.Kriegman, and N. Ahuja. Appearance-based eye gaze estimation. *In Proc. 6th IEEE Workshop on Applications of Computer Vision*, 2002. 1

[19] D. P. Kingma and M. Welling. Auto-Encoding Variational Bayes. 2013. 2

[20] D. Kononenko and V. Lempitsky. Learning to look up: Realtime monocular gaze correction using machine learning. In *CVPR*, 2015. 1, 2

[21] K. Krafka, A. Khosla, and P. Kellnhofer. Eye Tracking for Everyone. *CVPR*, 2016. 1, 2

[22] T. Kulkarni and W. Whitney. Deep Convolutional Inverse Graphics Network. *NIPS*, 2015. 2

[23] J. Li and S. Li. Gaze estimation from color image based on the eye model with known head pose. 2016. 8

[24] K. Mora and J.-M. Odobez. Geometric generative gaze estimation (g3e) for remote rgb-d cameras. In *CVPR*, 2014. 2

[25] S. Nie, M. Zheng, and Q. Ji. The deep regression bayesian network and its applications: Probabilistic deep learning for computer vision. *IEEE Signal Processing Magazine*, 35(1):101–111, 2018. 2

[26] A. Patil, D. Huard, and C. J. Fonnesbeck. Pymc: Bayesian stochastic modelling in python. *JSS*, 2010. 6

[27] P. Paysan, R. Knothe, B. Amberg, S. Romdhani, and T. Vetter. A 3d face model for pose and illumination invariant face recognition. *AVSS*, 2009. 8

[28] G.-J. Qi. Loss-sensitive generative adversarial networks on lipschitz densities. *arXiv preprint arXiv:1701.06264*, 2017. 2

[29] G.-J. Qi, C. C. Aggarwal, and T. S. Huang. On clustering heterogeneous social media objects with outlier links. In *Proceedings of the fifth ACM international conference on Web search and data mining*, pages 553–562. ACM, 2012. 1

[30] S. Reed, Z. Akata, X. Yan, L. Logeswaran, B. Schiele, and H. Lee. Generative Adversarial Text to Image Synthesis. *ICML*, 2016. 2, 4

[31] K. Ruhland, C. Peters, S. Andrist, J. Badler, N. Badler, M. Gleicher, B. Mutlu, and R. McDonnell. A review of eye gaze in virtual agents, social robotics and hci: Behaviour generation, user interaction and perception. In *CGF*, 2015. 1

[32] C. Sagonas, E. Antonakos, G. Tzimiropoulos, S. Zafeiriou, and M. Pantic. 300 Faces In-The-Wild Challenge: database and results. *Image and Vision Computing*, 2015. 8

[33] A. Shrivastava, T. Pfister, O. Tuzel, J. Susskind, W. Wang, and R. Webb. Learning from Simulated and Unsupervised Images through Adversarial Training. *arXiv*, 2016. 1, 2, 7

[34] K. Wang and Q. Ji. Real time eye gaze tracking with kinect. In *Pattern Recognition (ICPR), 2016 23rd International Conference on*, pages 2752–2757. IEEE, 2016. 1

[35] K. Wang and Q. Ji. Real time eye gaze tracking with 3d deformable eye-face model. In *Computer Vision (ICCV), 2017 IEEE International Conference on*. IEEE, 2017. 8

[36] K. Wang and Q. Ji. 3d gaze estimation without explicit personal calibration. *Pattern Recognition*, 2018. 2

[37] K. Wang, S. Wang, and Q. Ji. Deep eye fixation map learning for calibration-free eye gaze tracking. In *Proceedings of the Ninth Biennial ACM Symposium on Eye Tracking Research & Applications*, pages 47–55. ACM, 2016. 1

[38] E. Wood, T. Baltruaitis, X. Zhang, Y. Sugano, P. Robinson, and A. Bulling. Rendering of Eyes for Eye-Shape Registration and Gaze Estimation. In *ICCV*, 2015. 1, 2

[39] E. Wood, T. Baltrušaitis, L.-P. Morency, P. Robinson, and A. Bulling. Learning an appearance-based gaze estimator from one million synthesised images. In *ETRA*, 2016. 1, 2, 4, 6

[40] E. Wood and A. Bulling. Eyetab: Model-based gaze estimation on unmodified tablet computers. In *ETRA*, 2014. 8

[41] Y. Sugano, Y. Matsushita, and Y. Sato. Learning-by-synthesis for appearance-based 3d gaze estimation. *CVPR*, 2014. 1, 2, 6

[42] H. Yamazoe, A. Utsumi, T. Yonezawa, and S. Abe. Remote gaze estimation with a single camera based on facial-feature tracking without special calibration actions. In *ETRA*, 2008. 2

[43] X. Zhang, Y. Sugano, M. Fritz, and A. Bulling. Appearance-based gaze estimation in the wild. In *CVPR*, 2015. 1, 2, 6, 7, 8