Facial Expression Neutralization With StoicNet

William Carver and Ifeoma Nwogu
Rochester Institute of Technology
1 Lomb Memorial Dive, Rochester, NY
wtc9653,ionvcs@rit.edu

Abstract

Expression neutralization is the process of synthetically altering an image of a face so as to remove any facial expression from it without changing the face's identity. Facial expression neutralization could have a variety of applications, particularly in the realms of facial recognition, in action unit analysis, or even improving the quality of identification pictures for various types of documents. Our proposed model, StoicNet, combines the robust encoding capacity of variational autoencoders, the generative power of generative adversarial networks, and the enhancing capabilities of super resolution networks with a learned encoding transformation to achieve compelling expression neutralization, while preserving the identity of the input face. Objective experiments demonstrate that StoicNet successfully generates realistic, identity-preserved faces with neutral expressions, regardless of the emotion or expression intensity of the input face.

1. Introduction

Expression neutralization is the process of synthetically altering an image of a face so as to remove any emotion from it. When done properly the identity of the face should not change, only the expression. This requires some intuition about how faces differ, whether dynamically through the display of emotion or in more fixed ways that constitute one's static facial features.

Algorithmically analyzing faces is an immense challenge due to the fact that human faces vary drastically in appearance from one person to the next. Problems like facial recognition and emotion analysis are made significantly harder by the fact that our appearance can vary as a function of both identity and facial expression. These tasks would likely be much easier if one could always be provided a neutral version of faces to work with. Unfortunately, a specific individual's neutral face is not always available, so the next best option is to generate one using facial expression neutralization. In addition to helping with facial recognition or expression analysis, expression neutralization could also be used for the generation of identification documents like government issued licenses or missing person photos.

StoicNet tackles this problem by providing a means of neutralizing any facial expression through a learned transformation. The model combines the robust encoding capacity of the VAE, the generative power of GANs, and the enhancing capabilities of super resolution networks with a simple learned encoding transformation. This enables it to achieve compelling expression neutralization. It is also fast enough to be practical as a preprocessing step for other applications.

2. Related Work

Face alteration methods go as far back as the mid-2000's pre-deep learning techniques, when computer graphic researchers began investigating the notion of face transfers. Face transfer involves techniques for mapping pose and expressions obtained from one individual to the underlying 3D model of another [23] and applying the known textures on the modified face structure. These models require the use of three-dimensional (3D) facial meshes which can be very expensive to obtain and manipulate, especially when the expression adjustment is required for only one or a small number of facial images. A whole slew of related techniques ensued in the following years and a summary of these can be found in [25]. Such models proved useful in aiding face recognition, irrespective of facial expressions [7].

A related problem [20] involves a single-sample face recognition system in which selected source and transfer images are projected into a feature space via locality preserving projection (LPP). The learned transfer projection matrix is then applied to training samples to transfer the macro characteristics learned. These characteristics involved facial expressions and pose. Specifically, the technique was used to transfer smiles to neutral faces, and face

recognition was accomplished in the embedding space using nearest neighbor classification. Although the recognition was accomplished via nearest classes being preserved, the resulting images were very blurry not visually appealing. Many other works built off this but focused more on the face recognition aspect, rather than the face altering component of the work.

With the advent of deep learning methods and large publicly available training datasets, face altering techniques are currently dominated by the use of VAEs and GANs. Face style transfer is one of the earlier techniques for face altering. More specifically, some problems in face alteration can also be formulated as an instance of style transfer where attributes of a face such as hair style, facial expression, beard/no-beard can be transferred from one image to another. Zhu *et al*. [27] apply CycleGANs for image-to-image translation task and achieved good performance, especially in their image generation task.

A somewhat related task is that of targeted face aging where the authors pay particular emphasis on preserving the identity of the source image in the resulting aged version of the face. To this end, Wang *et al*. [24] implemented an identity-preserved conditional GAN which functioned as the face generator, and an age classifier forced the face generation at the target age. Also, along similar lines, Antipov *et al*. [3] proposed an aging mechanism which also applies a conditional GAN and used a local manifold adaptation (LMA) technique to for identity preservation. In a more recent work, the same authors included an additional module that solved an LBFGS optimization problem for each image at inference time [2], as an improvement over the LMA, but this was not very efficient.

3. Method

3.1. Model Design

StoicNet is at its core a VAE-GAN. An image x is fed into the encoder network, which applies several strided convolution layers to encode it into Gaussian distributions, represented as two vectors containing each latent feature's mean μ and standard deviation σ. These distributions are then each sampled using the reparameterization technique to get the latent vector representation z.

The encoding is then passed through the neutralizer, which contains a single dense perceptron layer. This creates the neutral encoding z_n. This encoding is then given as the input to the decoder, which uses a series of fractionally-strided transposed convolution layers to convert the encoding back into the image space, producing the initial generated image x_l.

This initial image is of relatively low quality thanks to VAE's tendency to produce blurry images. To remedy this, the low quality image is cleaned up using the enhancer network to produce the final higher quality output x_h. The enhancer's architecture is based on that of super-resolution networks, featuring multiple residual blocks and skip connections. This helps to fill in finer face details that may have been blurry in the initial VAE output. To further improve image clarity, outputs are post-processed with a simple sharpening filter.

During the training process described in the next section, a discriminator network is also used. The discriminator is trained along with the previous networks, attempting to differentiate between real and generated imagery. Its architecture is roughly the same as the encoder except with additional fully connected layers ending with a single output value representing a "realness" rating.

3.2. Training

StoicNet's loss function is a combination of several smaller loss functions. All of these require balancing to achieve the desired output. This is done with the set of weights W. The complete loss function is summarized in equation 1.

$$L = W_{kl}L_{kl} + W_rL_{rec1} + (1 - W_g)L_{rec2} + W_gL_{gan} \quad (1)$$

This function will be broken down in the following subsections. Note that the neutralization is trained last, completely separately from the other networks, and is thus not considered in the equation above. No neutralization is performed during the VAE and Enhancer training stages. They are simply trying to recreate the input images. The complete StoicNet training diagram for these stages is shown in Figure 1.

3.2.1 VAE Loss

The VAE's encoder and decoder have separate loss functions. Both are primarily composed of the reconstruction loss. This is calculated using mean squared error (MSE) between the pixel values of the real input image and the low quality generated image.

$$L_{rec1} = \sqrt{||x - Dec(Enc(x))||_2^2} \quad (2)$$

$$L_{Dec} = W_rL_{rec1} \quad (3)$$

For the encoder, the loss function also includes the KL divergence of the encoding $q(z|x)$ from the prior $p(z)$, a normal distribution with mean 0 and standard deviation 1. It's weight, W_{kl} is slowly ramped up from 0 to its full value during the first several epochs of training.

$$L_{kl} = D_{KL}(q(z|x)||p(z)) \quad (4)$$

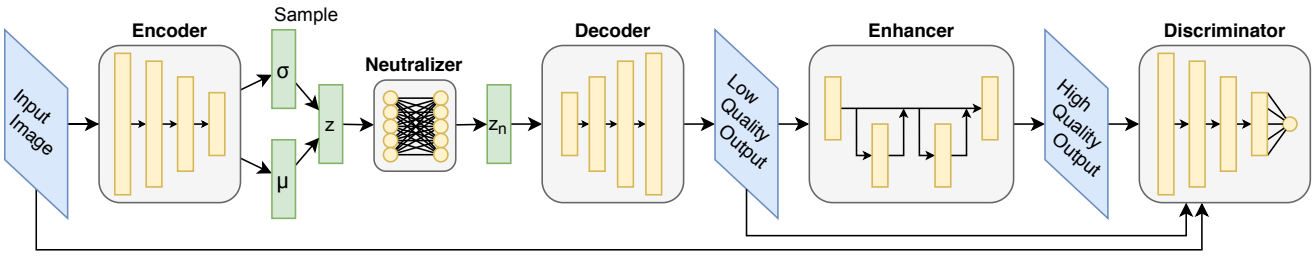

Figure 1. StoicNet Data Flow Diagram

$$L_{Enc} = W_{kl}L_{kl} + W_r L_{rec1} \tag{5}$$

W_{kl} is roughly equivalent to the hyperparameter β used in [9]. This is increased to encourage the encoder to produce efficient disentangled encodings. Achieving thorough disentanglement between the elements of the encoding is crucial for the task of neutralization as it is imperative that emotions are encoded separately from identity. If that doesn't happen then altering how much somebody is smiling could also alter their hair color, making identity preserving neutralization on the encoding utterly impossible.

3.2.2 Discriminator Loss

The discriminator is formulated for use as a LSGAN [19], with its output using a linear activation instead of a sigmoid. This provides smoother training by helping to avoid the vanishing gradient when the discriminator becomes too accurate. This is immensely helpful due to the fact that the discriminator should ideally be as accurate as possible in order to provide the best guidance for the generator.

As such, the loss function for the discriminator L_{dis} is simply the MSE between the true realness labels and the discriminator's predicted labels.

$$L_{Dis} = \frac{(D(x)-1)^2 + D(d(e(x)))^2 + D(E(d(e(x))))^2}{3} \tag{6}$$

where, for shorthand, $D(\cdot)$ is the discriminator, $E(\cdot)$ is the Enhancer, $d(\cdot)$ is the decoder, and $e(\cdot)$ is the encoder.

This is calculated for both the real images and both the high and low quality generated images. Both outputs are used to prevent the enhancer from becoming stuck in a cycle. When the discriminator is not given the VAE's unenhanced outputs, the enhancer will slowly learn improvements on the original blurry images. However, it reaches a point where the discriminator has mostly forgotten the original blurry faces. At that point it becomes advantageous to simply let the original low quality image pass through the skip connections relatively unimpeded, starting the cycle over again. By giving the discriminator the blurry unenhanced images, the enhancer is prevented from falling into this lazy cycle.

3.2.3 Enhancer Loss

The enhancer is trained in two stages. For the first stage it acts purely as an extension of the VAE's decoder, using the same kind of reconstruction loss on its own output image.

$$L_{Enh} = \sqrt{||x - Enh(Dec(Enc(x)))||_2^2} \tag{7}$$

The second stage involves two changes. First, the enhancer's reconstruction loss is tweaked to be based on the low quality image $Dec(Enc(x))$ instead of the input x. This is done to make the enhancer act as a standalone module instead of an extension of the decoder.

$$L_{rec2} = \sqrt{||Dec(Enc(x)) - Enh(Dec(Enc(x)))||_2^2} \tag{8}$$

The second change is the addition of the GAN's adversarial loss from the discriminator. This is balanced against the reconstruction loss using the GAN loss weight, W_g. This can be thought of as balancing content and style, with content being the identity of the face and style being the image clarity.

$$L_{gan} = (Dis(Enh(Dec(Enc(x)))) - 1)^2 \tag{9}$$

$$L_{Enh} = (1 - W_g)L_{rec2} + W_g L_{gan} \tag{10}$$

During stage 2, the loss functions for the Encoder and Decoder are kept the same, and the neutralizer is still not used. This changes during the third and final stage.

3.2.4 Neutralizer Loss

The final stage of StoicNet's training involves freezing the now-trained encoder, decoder, and enhancer networks and training the neutralizer network. Up until this point the encoding vector z was being sent directly to the decoder without modification. To achieve expression neutralization that encoding must be altered in such a way as to only change

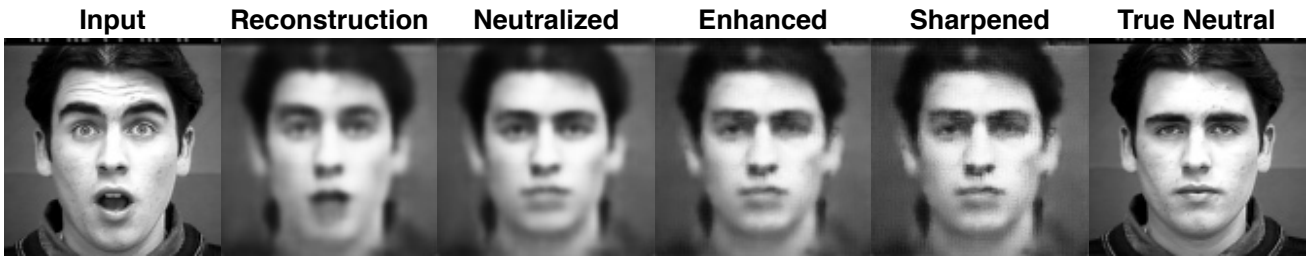

Figure 2. Depiction of the results at each stage of StoicNet's execution. The first reconstruction is the result of decoding without the neutralizer being applied. Neutralized and enhanced images are the low quality and high quality outputs. Input and True Neutral images ©Jeffrey Cohn.

the elements that encode emotion and not those that encode identity. Rather than figuring out this alteration manually, StoicNet learns the transformation.

To learn neutralization, two input images are used, x_e containing an expressive face, and x_n containing the same individual but with a neutral face. Both are encoded to produce their respective encodings. The neutralizer network is then given the expressive encoding $Enc(x_e)$ and the output is compared against the neutral encoding $Ntr(Enc(x_n))$ using MSE. This gives the following loss function for the neutralizer:

$$L_{Ntr} = \sqrt{||Enc(x_n) - Ntr(Enc(x_e))||_2^2} \qquad (11)$$

Thanks to the efficient and disentangled encoding enforced by the L_{kl} (see equation 4 above), learning neutralization is quite straightforward. Once the neutralizer is trained, the model is complete. For actual use the VAE can be more precise by not sampling the encoding from distributions. Instead z is simply taken from μ directly.

4. Experiments and Results

4.1. Datasets

4.1.1 Cohn-Kanade

The Extended Cohn-Kanade (CK+) dataset provides 593 sequences of 123 different actors transitioning from neutral faces to expressive faces of various emotions[14][18]. This makes is particularly well suited for learning identity-preserving expression neutralization. These sequences are preprocessed by using OpenCV to crop in a square around each faces. Since the majority of the dataset consists of black and white images, those that are provided in color are also converted to grayscale.

For the first two stages of training, images are used individually. For training neutralization, the images are used in pairs combining every image with the first image of the sequence it is taken from. This provides pairings of neutral and expressive images of varying intensity from the same person.

It was found that the CK+ dataset alone was not well suited for training a generative model due to the small number of different people provided, especially with such consistent positioning and lighting. For this reason a second dataset was added.

4.1.2 Labeled Faces in the Wild

The Labeled Faces in the Wild dataset features 13233 images of 5749 different people collected from the web[12][17]. A variation of the dataset in which the faces are all aligned using image funneling was chosen for use in the training of StoicNet[11]. Similar to the CK+ dataset, the images are preprocessed by converting to grayscale and cropping around the faces.

These are used to augment the CK+ imagery during the first two training stages in order to prevent simple memorization. The greater number of identities and wider variety of angles and lighting force the VAE to learn a more robust feature-based encoding. The LFW imagery is not used in the third stage of training since the images are not labeled by emotion.

4.2. Results & Evaluation

As shown in Figure 2, StoicNet is able to effectively reconstruct, neutralize, and enhance facial imagery. Figure 3 further demonstrates that StoicNet is robust enough to work on a variety of emotions for both male and female subjects.

Objective analysis is done in two ways to evaluate the two key functions of StoicNet, neutralization and identity preservation. Unless otherwise noted, a split of the dataset not used for training was used for these analyses, consisting of 6424 images.

4.2.1 Neutralization Analysis

For evaluating neutralization, images are analyzed using the Facial Action Unit Coding System (FACS), which describes the atomic facial movements known as action units (AU) that can combine to make any facial expression[8]. Com-

Figure 3. Neutralization of various expressions in both male and female faces. Top rows are the input images and the bottom rows are the enhanced output images (no sharpening applied). Neutral faces are kept as such, while expressive faces are effectively neutralized. Input images ©Jeffrey Cohn.

Image	All	Top 25%	Top 10%
Expressive Input	4.31	9.22	11.44
StoicNet Neutralized	2.10	2.32	2.45
True Neutral	1.42	1.82	1.92

Table 1. Average emotion intensities in expressive, StoicNet neutralized, and true neutral images. Includes breakdown of results for subsets of the most expressive input imagery.

bined they allow a thorough analysis of the facial expression an individual is making.

To determine the magnitude of action units, images are fed through OpenFace[4][5]. The action unit magnitudes for each image are summed together to serve as an approximation of the total expressiveness of that face. These sums are then averaged across all samples to produce the average emotion intensity of the inputs, neutralizations, and true neutral images.

As shown in Table 1, the images produced by StoicNet are far more neutral than the inputs. Note that even the ground truth neutral images are not completely absent of perceived facial movement. This is largely due to the great variety of resting faces that people have, and the fact that some individual's resting faces can still appear somewhat emotive. It should also be noted that the inputs are made up of a range of expression intensities, from neutral to full emotion intensity. For this reason, a breakdown is included in the table to show the results on the top 25% (1606 samples) and the top 10% (642 samples) most expressive samples.

4.2.2 Identity Analysis

To objectively evaluate StoicNet's ability to preserve identity in the neutralized face, facial recognition is done with OpenFace[1]. The evaluation samples were used to generate 6424 sets of three pairs of images with the expressive

Figure 4. Examples of poor output quality and identity preservation. Leftmost input image ©Jeffrey Cohn.

Pair	Identity Distance
Positive	0.192
StoicNet Neutralized	0.660
Negative	1.493

Table 2. Average identity encoding distance compared to input image

input face. The "positive" pair is with the same individual's real neutral face, the "neutralized" pair with StoicNet's generated neutral face and the "negative" pair with a different person's face. These three pairs are given to OpenFace and the distances in identities of each pair are compared. Table 2 shows the average results of this comparison. As shown, the average distance for the generated images is closer to the average positive distance than the average negative distance.

Based on this data, the threshold distance for being considered a match for facial verification is 0.842. Of the 6424 samples used for evaluation, the distance between the anchor and the generated sample was better than this threshold and therefore considered a match 75.6% of the time. Furthermore, for 1:2 facial recognition, the distance between the anchor and generated image was less than the distance between the anchor and the random negative 94.4% of the time.

4.2.3 Biases and Shortcomings

Both of the datasets used for training suffer from an over representation of younger Caucasians, and this bias is clearly visible in the outputs of StoicNet, with outputs being of considerably worse quality for inputs containing older or non-Caucasian individuals. A sample of the lesser image quality is shown in Figure 4.

Furthermore, both datasets featured relatively few individuals with glasses, so StoicNet tends to ignore them. This effect is magnified by the blurry VAE output, which would likely make thin-framed glasses disappear even if the data contained more glasses-wearing individuals.

5. Conclusion and Future Work

It has been shown to be advantageous to use a deep convolutional feature distance instead of pixel distance for reconstruction loss. This is also referred to as perceptual loss[13]. Perceptual loss is calculated by comparing the distances between latent features of the input and output images at certain layers within pre-trained networks like VGG19 or Alexnet. While this provides a richer metric for reconstruction loss, it can be computationally expensive compared to pixel-based distances.

By encouraging disentangled encodings, StoicNet encodes the identities of faces separately from their emotion. Using an learned neutralization transformation in this latent encoding space, it can eliminate the expression of emotion without disrupting the identity of the individual being depicted. By including an enhancer network to clean up the initially blurry images, StoicNet is able to produce higher quality outputs than a standalone VAE. Together, these enable it to effectively perform facial expression neutralization.

StoicNet's design is not inherently constrained to the task of neutralizing faces and can easily generalize to other applications. With a different dataset the model could easily be used for other transformations such as the removal of glasses or facial hair. Like StoicNet, these could also be immensely beneficial for tasks like facial recognition or the generation of enhanced images for identification cards.

6. Background for StoicNet

6.1. Variational Auto-Encoders

The Variational Auto-Encoder (VAE) is an improvement on the auto-encoder model that encodes inputs into a set of distributions rather than directly into discrete values[15]. By encoding into distributions, the VAE is forced to learn a more continuous latent encoding space. This is crucial when later altering the encodings as it helps to ensure that novel values in the latent space decode into viable points in

the image space. VAEs can be used for image data quite effectively by implementing them with convolutional layers instead of fully connected layers.

A VAE is composed of two separate networks, an encoder and a decoder. The encoder takes an input image and outputs two vectors, one representing the means (μ) and one representing the standard deviations (σ) for each latent feature. These distributions are then sampled to create the latent feature vector z. This sampling is done using a reparameterization trick to allow the gradient to pass through the sampling operation:

$$z = \mu + \sigma \odot \epsilon, \tag{12}$$

where $\epsilon \sim N(0, I)$.

The z vector is then fed into the decoder to transform the data back into image space. The bottleneck of the encoding forces the VAE to learn an efficient encoding of the data

VAEs typically use a two part loss function. The first component is how well it recreates the input image, referred to as the reconstruction loss. The second component is the KL Divergence of the latent distributions compared to the normal distribution ($N(0, 1)$). This component prevents the VAE from encoding the data into points that are spread far apart with little variation, which would defeat the purpose of encoding into distributions in the first place.

Work has been done to play with the ratio of weights between these two components by adding a β multiplier to the KL Divergence [9]. Increasing β has been shown to improve the efficiency of the encoding, promoting disentangling of the encoded features. This disentangling means that features of the input space are encoded separately in the latent vector. For example, the first index of the latent vector might encode the color of something, the second might encode the shape, and the third might encode size. Without disentangling the distinctions of which elements correspond to which features might not be so clear. It has also been shown that there are benefits to providing a warm-up period to slowly ramp up β during the beginning of training to allow the VAE time to start creating more accurate encodings before being penalized too harshly [22].

One of the primary advantages of a VAE over a traditional auto-encoder is the ingrained ability to sample from the latent distributions. Whereas a traditional auto-encoder might map features to any range of values within the latent space, the KL-Divergence loss of the VAE encourages a denser encoding. This makes image manipulation much easier, as it allows for smoother interpolation within the latent space [10].

6.2. Generative Adversarial Networks

A Generative Adversarial Network (GAN) is a model that pits a generator network against a discriminator network, with the generator attempting to create synthetic images and the discriminator trying to distinguish whether images are real or synthetic. When being applied to images it is standard for both of these two networks to utilize convolutional layers, leading to the title of being a Deep Convolutional Generative Adversarial Network (DCGAN)[21]. All of the approaches discussed here are variations on DCGANs.

Compared to VAEs, GANs are usually able to produce much sharper imagery. This is because while standard VAEs work with explicit loss functions, a GAN's discriminator is able to provide a more robust, learned loss function for the generator. This is much better for highly multimodal data like faces as it avoids the blurriness that results from the tendency to average the data when using explicit loss functions.

GANs aren't inherently designed for the manipulation of imagery. To achieve this, conditions are often injected into the input data, creating what is known as a conditional GAN. These conditions can be injected as part of the encoding[2] or as additional layers of the input image[24]. However, when dealing with manipulation of data, it is important to include some form of reconstruction loss to ensure that the output is in some way related to the input. For example, while GANs can create realistic face images, they usually generate *a person* rather than *a specific person*. However, it has been shown that by augmenting the generator's discriminator-based loss function with an identity preservation rating, one can produce highly compelling age alterations to specific target individuals [24][2].

6.3. Hybrid Models

Drawing inspiration from a multitude of different models, many have explored the idea of using hybrid approaches. One such hybrid is to incorporate architecture similar to those used in super resolution networks[6][27]. By incorporating residual blocks and skip connections into the decoder, it is possible to greatly reduce the blurry outputs that standard VAEs are notorious for. Inspired by the results of the multi-stage VAE proposed by Cai et al.[6], StoicNet adopts a similar multi-stage architecture.

Another hybrid approach is to combine convolutional VAEs and DCGANs into one single model[16][26]. With these VAE-GAN models the decoder of the VAE doubles as the generator of the GAN. StoicNet expands upon the multi-stage VAE architecture by also including a discriminator during training in order to provide adversarial loss for its image enhancing.

Acknowledgments

This material is based upon work partially supported by the National Science Foundation under Grant No. 1846076.

References

[1] Brandon Amos, Bartosz Ludwiczuk, and Mahadev Satyanarayanan. Openface: A general-purpose face recognition library with mobile applications. Technical report, CMU-CS-16-118, CMU School of Computer Science, 2016.

[2] Grigory Antipov, Moez Baccouche, and Jean-Luc Dugelay. Face aging with conditional generative adversarial networks. *CoRR*, abs/1702.01983, 2017.

[3] Grigory Antipov, Moez Baccouche, and Jean-Luc Dugelay. Boosting cross-age face verification via generative age normalization. In *IJCB 2017, International Joint Conference on Biometrics, October 1-4, 2017, Denver, Colorado, USA*, Denver, ÉTATS-UNIS, 10 2017.

[4] T. Baltrusaitis, A. Zadeh, Y. C. Lim, and L. Morency. Openface 2.0: Facial behavior analysis toolkit. In *2018 13th IEEE International Conference on Automatic Face Gesture Recognition (FG 2018)*, pages 59–66, 2018.

[5] T. Baltrušaitis, M. Mahmoud, and P. Robinson. Cross-dataset learning and person-specific normalisation for automatic action unit detection. In *2015 11th IEEE International Conference and Workshops on Automatic Face and Gesture Recognition (FG)*, volume 06, pages 1–6, 2015.

[6] Lei Cai, Hongyang Gao, and Shuiwang Ji. Multi-stage variational auto-encoders for coarse-to-fine image generation. *CoRR*, abs/1705.07202, 2017.

[7] Baptiste Chu, Sami Romdhani, and Liming Chen. 3d-aided face recognition robust to expression and pose variations. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2014.

[8] P Ekman and W Friesen. *Facial Action Coding System: A Technique for the Measurement of Facial Movement*. Consulting Psychologists Press, Palo Alto, 1978.

[9] Irina Higgins, Loïc Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew M Botvinick, Shakir Mohamed, and Alexander Lerchner. beta-vae: Learning basic visual concepts with a constrained variational framework. *ICLR*, 2017.

[10] Xianxu Hou, LinLin Shen, Ke Sun, and Guoping Qiu. Deep feature consistent variational autoencoder. *CoRR*, abs/1610.00291, 2016.

[11] Gary B. Huang, Vidit Jain, and Erik Learned-Miller. Unsupervised joint alignment of complex images. In *ICCV*, 2007.

[12] Gary B. Huang, Manu Ramesh, Tamara Berg, and Erik Learned-Miller. Labeled faces in the wild: A database for studying face recognition in unconstrained environments. Technical Report 07-49, University of Massachusetts, Amherst, October 2007.

[13] Justin Johnson, Alexandre Alahi, and Fei-Fei Li. Perceptual losses for real-time style transfer and super-resolution. *CoRR*, abs/1603.08155, 2016.

[14] T. Kanade, J. F. Cohn, and Yingli Tian. Comprehensive database for facial expression analysis. In *Proceedings Fourth IEEE International Conference on Automatic Face and Gesture Recognition (FG'00)*, pages 46–53, Grenoble, France, 2000.

[15] Diederik P Kingma and Max Welling. Auto-encoding variational bayes, 2013.

[16] Anders Boesen Lindbo Larsen, Søren Kaae Sønderby, and Ole Winther. Autoencoding beyond pixels using a learned similarity metric. *CoRR*, abs/1512.09300, 2015.

[17] Gary B. Huang Erik Learned-Miller. Labeled faces in the wild: Updates and new reporting procedures. Technical Report UM-CS-2014-003, University of Massachusetts, Amherst, May 2014.

[18] Patrick Lucey, Jeffrey F. Cohn, Takeo Kanade, Jason M. Saragih, Zara Ambadar, and Iain A. Matthews. The extended cohn-kanade dataset (ck+): A complete expression dataset for action unit and emotion-specified expression. In *Proceedings of the Third International Workshop on CVPR for Human Communicative Behavior Analysis (CVPR4HB 2010)*, pages 94–101, San Francisco, USA, 2010.

[19] Xudong Mao, Qing Li, Haoran Xie, Raymond Y. K. Lau, and Zhen Wang. Multi-class generative adversarial networks with the L2 loss function. *CoRR*, abs/1611.04076, 2016.

[20] Jie Pan, Xuesong Wang, and Yuhu Cheng. Single-sample face recognition based on lpp feature transfer. *IEEE Access*, 4:1–1, 01 2016.

[21] Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. In Yoshua Bengio and Yann LeCun, editors, *4th International Conference on Learning Representations, ICLR 2016, San Juan, Puerto Rico, May 2-4, 2016, Conference Track Proceedings*, 2016.

[22] Casper Sønderby, Tapani Raiko, Lars Maaløe, Søren Sønderby, and Ole Winther. How to train deep variational autoencoders and probabilistic ladder networks. *ICML*, 02 2016.

[23] Daniel Vlasic, Matthew Brand, Hanspeter Pfister, and Jovan Popovic. Face transfer with multilinear models. *ACM Trans. Graph.*, 24(3):426–433, 2005.

[24] Zongwei Wang, Xu Tang, Weixin Luo, and Shenghua Gao. Face aging with identity-preserved conditional generative adversarial networks. *CVPR*, pages 7939–7947, 06 2018.

[25] Fei Yang, Jue Wang, Eli Shechtman, Lubomir D. Bourdev, and Dimitris N. Metaxas. Expression flow for 3d-aware face component transfer. *ACM Trans. Graph.*, 30(4):60, 2011.

[26] Zhifei Zhang, Yang Song, and Hairong Qi. Age progression/regression by conditional adversarial autoencoder. *CoRR*, abs/1702.08423, 2017.

[27] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. *CoRR*, abs/1703.10593, 2017.