LEGAN: Disentangled Manipulation of Directional Lighting and Facial Expressions whilst Leveraging Human Perceptual Judgements

Sandipan Banerjee¹, Ajjen Joshi¹, Prashant Mahajan⁴, Sneha Bhattacharya⁵, Survi Kyal¹ and Taniya Mishra⁴
¹Affectiva, USA, ²Amazon, USA, ³Silver Spoon Animation, USA, ⁴SureStart, USA
{firstname.lastname}@affectiva.com, prmhp@amazon.com, snehabhattac@umass.edu, taniya.mishra@mysurestart.com

Abstract

Building facial analysis systems that generalize to extreme variations in lighting and facial expressions is a challenging problem that can potentially be alleviated using natural-looking synthetic data. Towards that, we propose LEGAN, a novel synthesis framework that leverages perceptual quality judgments for jointly manipulating lighting and expressions in face images, without requiring paired training data. LEGAN disentangles the lighting and expression subspaces and performs transformations in the feature space before upscaling to the desired output image. The fidelity of the synthetic image is further refined by integrating a perceptual quality estimation model, trained with face images rendered using multiple synthesis methods and their crowd-sourced naturalness ratings, into the LEGAN framework as an auxiliary discriminator. Using objective metrics like FID and LPIPS, LEGAN is shown to generate higher quality face images when compared with popular GAN models like StarGAN and StarGAN-v2 for lighting and expression synthesis. We also conduct a perceptual study using images synthesized by LEGAN and other GAN models and show the correlation between our quality estimation and visual fidelity. Finally, we demonstrate the effectiveness of LEGAN as training data augmenter for expression recognition and face verification tasks.

1. Introduction

Deep learning [59] has engendered tremendous progress in automated facial analysis, with applications ranging from face verification [72, 22, 11] to expression classification [91]. However, building robust and accurate models that generalize effectively in-the-wild is still an open problem. A major part of it stems from training datasets failing to represent the “true” distribution of real world data [54, 68, 92, 6] (e.g. extreme lighting conditions [17, 73, 38]); or the training set may be non-uniformly distributed across classes, leading to the long-tail problem [63].

Figure 1: LEGAN jointly manipulates lighting and expression in face images while preserving the subject identity of the input.

One way to mitigate the imbalance problem, shown to work in multiple domains [62, 9, 15, 99], is to introduce synthetic samples into the training set. Many approaches for generating synthetic data exist [18, 8, 42], none as successful as GANs [37] in generating realistic face images [16, 51, 52, 25]. Thus, in this work, we design a GAN model for synthesizing variations of an existing face image with the desired illumination and facial expression, while keeping the subject identity and other attributes constant. These synthetic images, when used as supplemental training data, can help build facial analysis systems that better generalize across variations in illumination and expressions.

One drawback of GAN-based face generation is the absence of an accurate and automated metric to judge the perceptual quality of synthesized images [57, 20]. In order to solve this problem, we also introduce a quality estimation model that can serve as a cheap but efficient proxy for human judgment while evaluating naturalness of synthetic face images. Instead of generating a single score for a distribution of synthetic images [105, 45, 79] or for image pairs [103, 75], our goal is to infer an image-quality score on a continuum for a single synthetic face image. With this in mind, we run an Amazon Mechanical Turk (AMT) experiment where turkers are instructed to score the naturalness of synthetic face images, generated using different 3D-model [9] and GAN-based [51, 52, 78, 2] synthesis approaches. We then build a feed forward CNN to learn representations from these images that map to their corresponding percep-
In addition to a traditional discriminator [37], this trained quality model is then used as an auxiliary discriminator in the synthesis framework, named LEGAN (Lighting-Expression GAN), that we propose in this paper. Instead of intertwining the two tasks [25], LEGAN decomposes the lighting and expression sub-spaces using a pair of hourglass networks (encoder-decoder) that generate transformation masks capturing the intensity changes required for target generation. The desired output image is then synthesized by a third hourglass network from these two masks.

To demonstrate the effectiveness of LEGAN, we qualitatively and quantitatively compare its synthesized images with those produced by two popular GAN based models [25, 26] using objective metrics like FID [45], LPIPS [103], SSIM [94] and face match score. We also conduct a human rater study to evaluate the perceptual quality of LEGAN’s images and the contribution of the quality based auxiliary discriminator towards hallucinating perceptually superior images. Finally, we show the efficacy of LEGAN as training data augmenter by improving the generalizability of face verification and expression recognition models.

2. Related Work

Face Synthesis: Early approaches [18, 67] focused on stitching together similar looking facial patches from a gallery to synthesize a new face. Manipulating the facial shape using 3D models [49, 88, 63, 9] or deep features [27, 90] is another popular approach to generate new views. In recent times however researchers have pre-dominantly focused on using GANs [37] for synthesis, where an upsampling generator hallucinates faces from a noise vector, either randomly sampled from a distribution [37, 77, 51, 52] or interpreted from a different domain [71]. An existing face can also be encoded and then upsampled to obtain the desired attributes [48, 47, 95, 10, 70, 31, 35, 106, 23, 25].

Editing Expressions: Research in this domain started with modeling skin-muscle movements [33] for different facial expressions or swapping facial patches based on visual proximity [18]. With the advent of 3D face models, researchers used static [63] or morphable models [19, 101] to manipulate facial expressions with a higher degree of realism. Recently, the use of VAEs [13] and adversarial image-to-image translation networks have become extremely popular for editing facial expressions [48, 106, 25], with or without paired data for training. Some of these models use attention masks [76], facial shape information [36, 34] or exemplar videos [84] to guide the model in this task.

Editing Lighting: While methods like histogram equalization [74, 107] and gamma correction can shift the global luminance distribution and color encoding of an image, they cannot manipulate the direction of the light source itself. An early method [93] utilizes spherical harmonics to manipulate the directional lighting in 3D. In [24], local linear adjustments are performed on overlapping windows to change the lighting profile of an image. Deep learning based approaches have also been proposed where the reflectance, normal and lighting channels are disentangled and edited to relight images [80, 83, 82, 41]. Alternatively, the desired lighting can be passed to an encoder-decoder pair as the target for lighting manipulation in the input image [104, 85, 65, 69]. Recently, joint facial pose, lighting and expression manipulation has been proposed in [30, 58] where an input image can be manipulated by changing its attributes in feature space leveraging 3D parameters or latent information from synthetic images during training.

Quality Estimation of Synthetic Face Images: Synthetic image quality is commonly evaluated using metrics like the Inception Score [79] or FID [45], which compare statistics of real and synthetic feature distributions, and output a single score for the whole distribution rather than the individual image. The features themselves are extracted from the Inception-v3 model [86], usually pre-trained on objects from [29], and not specifically faces. As these metrics do
not take into account human judgments, they do not correlate well with perceptual realism [14, 20]. Consequently, researchers run perceptual studies to score the naturalness of synthetic images [102, 8]. These ratings are also used to design models that measure distortion between real and synthetic pairs [103, 75] or the coarse realism (‘real’ vs ‘fake’) of a synthetic image [105]. None of these evaluation models however are designed specifically for face images. Recently, [57] proposed a metric to rate the perceptual quality of a single image by using binary ratings from [105] as ground truth for synthetic face images generated by [51, 52]. However, their regression based model is trained on only 4,270 images and thus insufficient to reliably model the subjective nature of human judgements.

Unlike these methods, we build a synthetic face quality estimation model by leveraging perceptual ratings of over 37,000 images generated using five different synthesis techniques [51, 52, 78, 1, 9]. Our quality model takes into account the variability in human judgements and generate a realism score for individual images rather than the whole set. We leverage this model as an auxiliary discriminator in the LEGAN framework for simultaneous lighting and facial expression manipulation. This novelty together with LEGAN’s feature disentanglement improves the naturalness of the hallucinated images. Additionally, we do not require external 3DMM information or latent vectors during training nor do we need to fine-tune our model during testing on input images [30, 58].

3. Quality Estimation Model

Our quality estimator model is trained with synthetic face images assembled and annotated in two sequential stages, as described below.

Stage I: We first generate 16,507 synthetic face images using the StyleGAN [52] generator. These images are then annotated by labelers using Amazon Mechanical Turk (AMT) on a scale of 0 - 10 for naturalness, where a 0 rating represents an unnatural image and 10 a hyper-realistic one. The images are then binned into two broad groups - 'unnatural' for AMT ratings between 0 - 5 and 'natural' for 5 - 10. We extract descriptors for each image from the ‘avg_pool’ layer of the ResNet50 [44] model, pre-trained on VGGFace2 [22] and train a linear SVM [28] with the extracted features of around 12,000 images from this dataset and use the rest for parameter tuning. Post training, we use this SVM as a rough estimator of naturalness.

Stage II: In this stage, we perform the same AMT experiment again with a larger set of synthetic face images, collected from the following datasets:

1. FaceForensics++[78] - we randomly sample 1000 frames from this dataset consisting of 1000 video sequences that have been manipulated with four automated face manipulation methods.
2. DeepFake[2, 3] - we use sampled frames from 620 manipulated videos of 43 actors from [1].
3. ProGAN [51] - we generate 10,000 synthetic face images of non-existent subjects by training NVIDIA’s progressively growing GAN model on the CelebA-HQ dataset [51].
4. StyleGAN [52] - we extract 100,000 hyper-realistic face images of non-existent subjects generated using the StyleGAN model that were pre-filtered for quality [4].

To focus on near-frontal faces, we remove images with yaw over 15° in either direction, estimated using [43]. Since

Figure 3: During training, $G$ takes an image $I_a$ and target expression and lighting tensor $f_b$ as input and disentangles the feature sub-spaces using a pair of hourglass networks generating transformation masks $M_a$ and $M_f$ that are concatenated and passed through a third hourglass to hallucinate the target output $G(I_a, f_b)$. To eliminate the need of paired training data, we augment $G(I_a, f_b)$ with the source expression and lighting tensor $f_a$ and pass it through the same generator to reconstruct the input $G(G(I_a, f_b), f_a)$, to compute the reconstruction error $L_{rec}$. Moreover, with the help of $D$ and the quality based auxiliary discriminator $Q$, we calculate the adversarial ($L_{adv}$), feature classification ($L_{cls}$) and quality ($L_{qual}$) losses respectively. Utilizing the identity network $T$, we compute the identity loss $L_{id}$. During testing, only $G$ is required to generate synthetic images.
gender information is absent in most of the above datasets, we group the synthetic images using gender predictions from a pre-trained model [60]. Our trained SVM (from Stage I) is also used to rate the coarse naturalness of the collected images, using their ResNet50 features. We ensure balance in our synthetic dataset by sampling evenly from the natural and unnatural sets, as estimated by the SVM, and the perceived gender classes. To focus solely on the facial region, the pixels outside the convex hull formed by the facial landmarks, estimated using [21], of an image are masked. After the gender, facial yaw and naturalness based filtering, and the pre-processing step, we end up with 37,267 synthetic face images\footnote{Available here: \url{https://github.com/Affectiva/LEGAN_Perceptual_Dataset}} for our second AMT experiment.

Again, we ask Turkers to rate each image for naturalness on a scale of 0 - 10. Each image is shown to a Turk for 60 seconds to allow them time to make proper judgement even with slow network connection. We divide the full set of images into 72 batches such that each batch gets separately rated by 3 different Turkers. Post-crowd-sourcing, we compute the mean ($\mu$) and standard deviation ($\sigma$) from the 3 scores and assign them as naturalness label for an image.

To train the quality estimation model, we use 80\% of this annotated data and the rest for validation and testing. For augmentation, we only mirror the images as other techniques like translation, rotation and scaling drastically change their appearance compared to what the Turkers examined. Our model downsamples an input image using a set of strided convolution layers with Leaky ReLU\footnote{\url{https://github.com/Affectiva/LEGAN}} activation followed by two fully connected layers with linear activation and outputs a single realness scoring. Since both $\mu$ and $\sigma$ are passed as image labels, we try to capture the inconsistency in the AMT ratings (i.e. the subjective nature of human perception) by formulating a margin based loss for training. The model weights are tuned such that its prediction is within an acceptable margin, set to $\sigma$, from the mean rating $\mu$ of the image. The loss $L_N$ can be represented as:

$$L_N = \frac{1}{n} \sum_{i=1}^{n} \left( \sigma - \| \mu - Q(I_i) \|_2^2 \right)^2$$

where $n$ is the batch size, $Q(I_i)$ is the model prediction for the $i$-th image in the batch. Since the model is trained on the mean rating $\mu$ (regression) as the target rather than fixed classes (classification), $L_N$ pushes the model predictions towards the confidence margin $\sigma$ from $\mu$.

4. LEGAN

We build LEGAN as a lightweight network that works with unpaired data, similar to the StarGAN family, to focus more on assessing the effect of our quality estimation model ($Q$) as a perceptual loss. Unlike [30, 38], LEGAN does not require additional networks to regress 3DMM parameters or fine-tuning during inference. We describe the architecture and objective functions of LEGAN in this section, an overview of which can be seen in Figure 3.

4.1. Architecture

**Generator:** Our generator $G$, composed of three hourglass networks (encoder-decoder), starts with an input RGB face image $I_a$ and a target attributes vector $f_b$, that corresponds to expression and lighting conditions $c_e$ and $c_l$ respectively. The first hourglass receives $I_a$ concatenated with $c_e$ while the second one receives $I_a$ concatenated with $c_l$, thus disentangling the transformation task. Inside each hourglass, the concatenated tensor is downscaled using strided convolutions and then passed through a set of residual blocks [44] before being upsampled using pixel shuffling layers [81]. Each convolution layer is followed by instance normalization [89] and ReLU activation [64] for non-linearity. These upsampled images are the transformation masks $M_e$ and $M_l$ that map the changes in pixel intensity required to translate $I_a$ to conditions specified in $f_b$. $M_e$ and $M_l$ are concatenated together and fed to the third hourglass to generate the output image $G(I_a, f_b)$. The objective of dividing the generation process into two stages and hallucinating the transformation masks is two fold - (a) easing the task of each hourglass by simply making it focus on registering the required expression or lighting changes instead of both registration and hallucination, and (b) making the transformation process more explainable, with salient pixels prominent in $M_e$ and $M_l$, as can be seen in Figure 4.

**Discriminator:** The discriminator $D$ takes the output image $G(I_a, f_b)$ and predicts not only its realness score but also classifies its attributes $f_b$. $D$ is composed of strided convolution layers with Leaky ReLU\footnote{\url{https://github.com/Affectiva/LEGAN}} activation that downsamples the image to extract its encoded feature map. We use a patch discriminator [48] that takes this encoded feature map and passes it through a single channel convolution to get the realness map $D_{src}$. This feature map is also operated by a conv layer with $k$ filters to get the attributes prediction map $D_{str}$, where $k = \text{no. of channels in } f_b$.

**Auxiliary Discriminator:** We integrate the perceptual quality model $Q$, described in Section 3, into the LEGAN model graph to further refine the naturalness of the images synthesized by $G$. Unlike $D$, we do not train $Q$ jointly with
G but use the weights of a pre-trained snapshot.

Identity Network: We also add a pre-trained identity preserving network \( T \) to estimate the deviation of the output identity from that of the input. \( T \) is trained offline on face images with different pose, expression and lighting and its weights are kept frozen throughout LEGAN’s training.

4.2. Loss Function

1. Adversarial Loss: \( D \) is trained to distinguish a real face image \( I_a \) from its synthetic counterpart and judge the realness of the hallucinated image \( G(I_a, f_b) \). To stabilize the gradients and improve quality, we use the WGAN [7] based objective \( L_{adv} \) for this task with a gradient penalty [40], set as:

\[
L_{adv} = \mathbb{E}_{I_a}[D_{src}(I_a)] - \mathbb{E}_{I_a, f_b}[D_{src}(G(I_a, f_b))] - \lambda_{gp}(\|\nabla_{\hat{D}}D_{src}(\hat{I})\|_2 - 1)^2
\]

where \( \hat{I} \) is sampled uniformly from real and synthetic images and \( \lambda_{gp} \) is a tunable parameter. While \( D \) tries to minimize this to separate the synthetic from the real, \( G \) tries to maximize it by fooling \( D \).

2. Classification Loss: To ensure the target lighting and expression are correctly rendered by \( G \) and enable LEGAN to do many-to-many translations, we formulate a classification loss using \( D \)’s predictions, in the form of \( D_{cls} \). The loss \( L_{cls} \) is computed as:

\[
L_{cls} = \mathbb{E}_{I_a, f_a}[-\log D_{cls}(f_a \mid I_a)] + \mathbb{E}_{I_a, f_b}[-\log D_{cls}(f_b \mid G(I_a, f_b))] \tag{3}
\]

where \( f_a \) and \( f_b \) are the original and target attributes of an input image \( I_a \).

3. Identity Loss: To preserve the subject identity without using paired data, we add an identity loss between the input and the translated output \( G(I_a, f_b) \) by utilizing representations from \( T \). Both \( I_a \) and \( G(I_a, f_b) \) are passed through \( T \) for feature extraction and we set the objective \( L_{id} \) to minimize the cosine distance between these two features as:

\[
L_{id} = \mathbb{E}_{I_a, f_b}[1 - \frac{T(I_a) : T(G(I_a, f_b))}{\|T(I_a)\|_2 \|T(G(I_a, f_b))\|_2}] \tag{4}
\]

Ideally, the cosine distance between these two feature vectors should be 0, as they belong to the same identity.

4. Reconstruction Loss: To keep non-translating features from the input intact in the output image, we use a cyclic reconstruction loss [106] \( L_{rec} \) between \( I_a \) and its reconstruction \( G(G(I_a, f_b), f_a) \), computed as:

\[
L_{rec} = \mathbb{E}_{I_a, f_a}(|I_a - G(G(I_a, f_b), f_a)|) \tag{5}
\]

5. Quality Loss: We use \( Q \)’s predictions for further improving with perceptual realism of the synthetic images. Masked versions of the input image \( I_a \), the synthesized output \( G(I_a, f_b) \), and the reconstructed input \( G(G(I_a, f_b), f_a) \), produced using facial landmarks ex-
Table 1: Quantitative comparison with popular GAN models on held out CMU-MultiPIE [39] test set.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>StarGAN [25]</td>
<td>38.745</td>
<td>0.126</td>
<td>0.539</td>
<td>0.633</td>
<td>5.200</td>
<td>22.3%</td>
</tr>
<tr>
<td>StarGAN w/ $L_{\text{qual}}$ [25]</td>
<td>34.045</td>
<td>0.123</td>
<td>0.567</td>
<td>0.647</td>
<td>5.391</td>
<td>34.7%</td>
</tr>
<tr>
<td>StarGAN-v2 [26]</td>
<td>54.842</td>
<td>0.212</td>
<td>0.415</td>
<td>0.202</td>
<td>5.172</td>
<td>3.75%</td>
</tr>
<tr>
<td>LEGAN</td>
<td>29.964</td>
<td>0.120</td>
<td>0.649</td>
<td>0.649</td>
<td>5.853</td>
<td>39.3%</td>
</tr>
<tr>
<td>Real Images</td>
<td>12.931</td>
<td>-</td>
<td>-</td>
<td>0.739</td>
<td>5.921</td>
<td>-</td>
</tr>
</tbody>
</table>

Table 2: Quantitative performance of popular GAN models, trained only on CMU-MultiPIE [39], when tested on AFLW [56] images.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>StarGAN [25]</td>
<td>43.987</td>
<td>0.302</td>
<td>0.639</td>
<td>0.622</td>
<td>6.41</td>
</tr>
<tr>
<td>StarGAN w/ $L_{\text{qual}}$</td>
<td>41.641</td>
<td>0.268</td>
<td>0.624</td>
<td>0.614</td>
<td>6.73</td>
</tr>
<tr>
<td>StarGAN-v2 [26]</td>
<td>59.328</td>
<td>0.445</td>
<td>0.279</td>
<td>0.213</td>
<td>6.60</td>
</tr>
<tr>
<td>LEGAN</td>
<td>38.794</td>
<td>0.271</td>
<td>0.622</td>
<td>0.628</td>
<td>6.68</td>
</tr>
<tr>
<td>Real Images</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

5. Experiments and Results

Training Data. We utilize 36,657 frontal RGB images from the CMU-MultiPIE dataset [39], with 20 different lighting conditions and 6 acted facial expressions, to build our model. For training we use 33,305 images of 303 subjects and the remaining 3,352 images of 34 subjects for testing. The training data is highly skewed towards ‘Neutral’ and ‘Smile’ compared to the other 4 expressions but the distribution is almost uniform for the lighting classes. We align each image using their eye landmarks extracted with [21] and resize to $128 \times 128 \times 3$. We do not fine-tune LEGAN on any other data and solely rely on its generalizability for the different experimental tasks.

Testing Data. Along with MultiPIE’s held out test set, we also utilize the AFLW [56] and CelebA [61] datasets to test the robustness of LEGAN towards in-the-wild conditions. We do not train or fine-tune LEGAN on these datasets and use the model trained on MultiPIE for translation tasks.

Augmentation: Recoloring. Since MultiPIE [39] was acquired in a controlled setting, it has more or less uniform hue and saturation across all images. To artificially inject some diversity in the overall image color, and prevent visual overfitting, we build an image colorization model. Inspired by [102], we train two separate pix2pix [48] style GAN models where the input is a grayscale image and the target is set as its colored counterpart (i.e. original image). To manipulate the color style of the same grayscale image differently, we train these two models with randomly sampled 10K images from the UMDFaces [12] and FFHQ [51] datasets respectively. The trained generators are used to augment the color style of the MultiPIE training set.

Augmentation: White Balancing. In addition to changing the color style, we also artificially edit the white balance of the training images by utilizing the pre-trained model from [5]. This model can automatically correct the white balance of an image or render it with different camera presets. For each training input, we randomly use it as is or select its recolored or color corrected version, and pass it to LEGAN.

Implementation Details. To learn the model, we use the Adam optimizer [55] with a learning rate of 0.0001 and parameters $\beta_1$ and $\beta_2$ set to 0.5 and 0.999 respectively. The different loss weights $\lambda_{\text{adv}}, \lambda_1, \lambda_2, \lambda_3, \lambda_4$ and $\lambda_5$ are set empirically to 10, 20, 10, 0.5 and 0.000001 respectively. As done in [76], we train $D$ 5 times for each training iteration of the $G$. We train the LightCNN-29 model [97] on the CASIA-WebFace dataset [100] and utilize it as our identity network $T$. Features from its penultimate layer are used to compute $L_{\text{id}}$. LEGAN is trained with a batch size of 10 on a single Tesla V100 GPU for 100 epochs.

Comparison with Other GAN Models: For the training, we train StarGAN separately with the auxiliary discriminator added (StarGAN w/ $L_{\text{qual}}$). 

Metrics for Quality Estimation. To evaluate the quality of
Table 3: Quantitative performance of popular GAN models, trained only on CMU-MultiPIE [39], when tested on CelebA [61] images.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>StarGAN [25]</td>
<td>42.089</td>
<td>0.173</td>
<td>0.623</td>
<td>0.620</td>
<td>6.98</td>
</tr>
<tr>
<td>StarGAN w/ $L_{qual}$</td>
<td>35.189</td>
<td>0.145</td>
<td>0.614</td>
<td>0.621</td>
<td>7.15</td>
</tr>
<tr>
<td>StarGAN-v2 [26]</td>
<td>50.360</td>
<td>0.311</td>
<td>0.312</td>
<td>0.295</td>
<td>6.89</td>
</tr>
<tr>
<td>LEGAN</td>
<td>29.059</td>
<td>0.184</td>
<td>0.662</td>
<td>0.635</td>
<td>7.07</td>
</tr>
<tr>
<td>Real Images</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>7.31</td>
</tr>
</tbody>
</table>

Figure 6: LEGAN synthesized output images from randomly sampled images from the AFLW [56] and CelebA [61] datasets. Although, trained only on frontal MultiPIE images, LEGAN can realistically manipulate lighting and expression in images with variance in pose, resolution and domain (sketch).

synthetic face images generated by LEGAN and other GAN models, we compare the synthesized output with the corresponding target image\(^2\) using these metrics - (1) FID [45] and (2) LPIPS [103] to gauge the realism, (3) SSIM [94] to measure noise, and (4) face match score using pre-trained ResNet50 [44, 22] features and Pearson correlation coefficient. We also use our trained quality estimator to directly extract the (5) quality score of real and synthetic images.

Human Evaluation. We also run a perceptual study using face images generated by these models where we ask 17 non-expert human raters to pick an image from a lineup that best matches - (1) a target facial expression and (2) a target lighting condition, while (3) preserving the identity for 30 different MultiPIE [39] subjects. The raters are first shown real examples of the target expressions and lighting conditions. Each lineup consists of an actual image of the subject with bright lighting and neutral expression and the same subject synthesized for the target expression and lighting by the StarGAN [25], StarGAN w/ $L_{qual}$, StarGAN-v2 [26] and LEGAN, presented in a randomized order. We aggregate the rater votes across all rows and normalize them for each model (rightmost column of Table 1).

Quantitative Results. As can be seen from Table 1, LEGAN synthesizes perceptually superior face images (FID, LPIPS, Quality Score) while retaining subject identity (Match Score) better than the other GAN models. As validated by the human evaluation, LEGAN also effectively translates the input image to the target lighting and expression conditions. Surprisingly, the StarGAN-v2 fails to generate realistic images in these experiments. This can be attributed to the fact that our task of joint lighting and expression manipulation presents the model with a much higher number of possible transformation domains (101 to be exact). The StarGAN-v2 model does learn to separate and transform lighting and expression to a certain degree but fails to decouple the other image attributes like identity, gender and race\(^3\). Therefore, it synthesizes images that can be easily picked out by human perusal or identity matching. We also find adding $L_{qual}$ to StarGAN improves almost all its metric scores underpinning the value our quality estimator $Q$ even when coupled with off-the-shelf models. On top of enhancing the overall sharpness, $L_{qual}$ removes bullet-hole artifacts, similar to [53], from the peripheral regions of the output face. Such an artifact can be seen in (row 4, col 2, forehead-hair boundary) of Figure 5, which is eliminated by adding $L_{qual}$ (row 4, col 3).

For the unseen, in-the-wild datasets, the performance of LEGAN is mostly superior to the other models, as shown in Tables 2 and 3 respectively. Due to the non-uniform nature of the data, especially the facial pose, most of the metrics deteriorate from Table 1. However, the boost in quality score overall suggests the high quality images from AFLW [56] and especially CelebA [61] to be visually more appealing than images from MultiPIE [39]. Some sample results have been shared in Figure 6.

Effectiveness as Training Data Augmenter: We examine

\(^2\)Since target images are not available for the AFLW [56] and CelebA [61] datasets, we evaluate metrics between the source and output images.

\(^3\)We are not the first to encounter this issue, as shared here: https://github.com/clovaai/stargan-v2/issues/21
the use of LEGAN as training data augmenter for face verification and expression recognition tasks using the IJB-B [96], LFW [46] and AffectNet [66] datasets. For face verification, we use the CASIA-WebFace dataset [100] and the LightCNN-29 [97] architecture due to their popularity in this domain. We randomly sample 439,999 images of 10,575 subjects from [100] for training and 54,415 images for validation. We augment the training set by randomly manipulating the lighting and expression of each image (Table 4, row 2). The LightCNN-29 model is trained from scratch separately with the original and augmented sets and its weights saved when validation loss plateaus across epochs. These saved snapshots are then used to extract features from a still image or video frame in the IJB-B [96] dataset. For each IJB-B template, a mean feature is computed using video and media pooling operations [62] and match score between such features is calculated with Pearson correlation. For the LFW [46] images, we simply compare features between similar and dissimilar identities. To measure statistical significance of any performance benefit, we run each training 3 separate times. We find the model trained with the augmented data to improve upon the verification performance of the baseline (Table 4). This suggests that the LEGAN generated images retain their original identity and can boost the robustness of classification models towards intra-class variance in expressions and lighting.

For expression classification, we use a modified version of the AU-classification model from [32] (Leaky ReLU [98] and Dropout added) and manually annotated AffectNet [66] images for the (‘Neutral’, ‘Happy’, ‘Surprise’, ‘Disgust’) classes, as these 4 expressions overlap with MultiPIE [39]. The classification model is trained with 204,325 face images from AffectNet’s training split, which is highly skewed towards the ‘Happy’ class (59%) and has very few images for the ‘Surprise’ (6.2%) and ‘Disgust’ (1.6%) expressions. To balance the training distribution, we populate each sparse class with synthetic images generated by LEGAN from real images belonging to any of the other 3 classes. We use the original and augmented (balanced) data separately to train two versions of the model for expression classification. As there is no test split, we use the 2,000 validation images for testing, as done in other works [91]. We find the synthetic images, when used in training, to substantially improve test performance especially for the previously under-represented ‘Surprise’ and ‘Disgust’ classes (Table 5). This further validates the realism of the expressions generated by LEGAN.

### 6. Conclusion

We propose LEGAN, a GAN framework for performing many-to-many joint manipulation of lighting and expressions of an existing face image without requiring paired training data. Instead of translating the image representations in an entangled feature space like [25], LEGAN estimates transformation maps in the decomposed lighting and expression sub-spaces before combining them to get the desired output image. To enhance the perceptual quality of the synthetic images, we directly integrate a quality estimation model into LEGAN’s pipeline as an auxiliary discriminator. This quality estimation model, built with synthetic face images from different methods [51, 52, 78, 2, 9] and their crowd-sourced naturalness ratings, is trained using a margin based regression loss to capture the subjective nature of human judgement. The usefulness of LEGAN’s feature disentangling towards synthesis quality is shown by objective comparison [45, 103, 94] of its synthesized images to that of the other popular GAN models like StarGAN [25] and StarGAN-v2 [26]. These experiments also highlight the usefulness of the proposed quality estimator in LEGAN and StarGAN w/ L\textsubscript{qual}, specifically comparing the latter to vanilla StarGAN.

As a potential application, we use LEGAN as training data augmenter for face verification on the IJB-B [96] and LFW [46] datasets and for facial expression classification on the AffectNet [66] dataset (Tables 4 and 5). An improvement in the verification scores in both datasets suggests LEGAN can enhance the intra-class variance while preserving subject identity. The boost in expression recognition performance validates the realism of the LEGAN generated facial expressions. The output quality, however, could be further improved when translating from an intense expression to another. We plan to address this by - (1) using attention masks in our encoder modules, and (2) building translation pathways of facial action units [87] while going from one expression to another. Another future goal is to incorporate a temporal component in LEGAN for synthesizing a sequence of coherent frames.
References


