

Received 4 November 2023, accepted 4 December 2023, date of publication 7 December 2023,
date of current version 14 December 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3340788

RESEARCH ARTICLE

High-Quality Face Caricature via Style Translation

LAMYANBA LAISHRAM^{ID}, MUHAMMAD SHAHERYAR^{ID}, JONG TAEK LEE^{ID}, (Member, IEEE),
AND SOON KI JUNG^{ID}, (Senior Member, IEEE)

School of Computer Science and Engineering, Kyungpook National University, Daegu 41566, South Korea

Corresponding author: Soon Ki Jung (skjung@knu.ac.kr)

This work was supported in part by the Ministry of Science and Information and Communications Technology (MSIT), South Korea, through the Innovative Human Resource Development for Local Intellectualization Support Program, Supervised by the Institute for Information and Communications Technology Planning and Evaluation (IITP), under Grant IITP-2022-RS-2022-00156389; in part by the IITP Grant funded by the Korean Government (MSIT), Development of 5G-Based Predictive Visual Security Technology for Preemptive Threat Response, under Grant 2019-0-00203; and in part by the Brain Korea 21 (BK21) FOUR Project (AI-Driven Convergence Software Education Research Program) funded by the Ministry of Education, School of Computer Science and Engineering, Kyungpook National University, South Korea, under Grant 4199990214394.

ABSTRACT Caricature is an exaggerated form of artistic portraiture that accentuates unique yet subtle characteristics of human faces. Recently, advancements in deep end-to-end techniques have yielded encouraging outcomes in capturing both style and elevated exaggerations in creating face caricatures. Most of these approaches tend to produce cartoon-like results that could be more practical for real-world applications. In this study, we proposed a high-quality, unpaired face caricature method that is appropriate for use in the real world and uses computer vision techniques and GAN models. We attain the exaggeration of facial features and the stylization of appearance through a two-step process: Face caricature generation and face caricature projection. The face caricature generation step creates new caricature face datasets from real images and trains a generative model using the real and newly created caricature datasets. The Face caricature projection employs an encoder trained with real and caricature faces with the pre-trained generator to project real and caricature faces. Using the encoder and generator's latent space, we perform an incremental facial exaggeration from the real image to the caricature faces. Our projection preserves the facial identity, attributes, and expressions from the input image. Also, it accounts for facial occlusions, such as reading glasses or sunglasses, to enhance the robustness of our model. Furthermore, we comprehensively compared our approach with various state-of-the-art face caricature methods, highlighting our process's distinctiveness and exceptional realism.

INDEX TERMS Face caricature, facial exaggeration, image translation, GAN.

I. INTRODUCTION

A caricature is a visual portrayal of a person that simplifies or exaggerates their most visible characteristics through sketches or creative drawings [1], which are primarily used to express humor and for entertainment. In traditional practice, caricatures are manually created by artists who carefully analyze the variations between an individual's unique features and the standard human facial characteristics. It is becoming more intriguing and essential to explore the automated generation of caricatures from given real images,

The associate editor coordinating the review of this manuscript and approving it for publication was Joewono Widjaja^{ID}.

as crafting a caricature demands significant effort, labor, and skill from the artist.

Computer vision applications encompass a broad spectrum, including the capability to create caricatures without requiring an artist's direct involvement. Much like artists' process when creating caricatures, a computer vision-based approach can also be divided into two key phases. Firstly, it involves identifying and enhancing distinctive features, and secondly, infusing the exaggerated image with artistic styles to match the artist's preferences. This division into two independent categories adds flexibility and disentanglement, resulting in the creation of high-quality caricatures.

The automated generation of a caricature from real images is a non-trivial challenge. Apart from imbuing the photo with

a texture style reminiscent of caricatures, we should also take spatial exaggerations into consideration [2], [3]. Previous methods for creating facial caricatures required the expertise of professionals to achieve satisfactory outcomes [4]. The issue of exaggerating facial features remains an open problem in research areas like detection [5] and recognition [6].

Certain methods incorporate additional data, such as user interaction [7] or by increasing the shape representation's divergence from the average, as in the case of 2D landmarks or 3D meshes [8], [9], [10], [11] to tackle the exaggeration challenge. With the advancement in computer vision techniques, numerous automated caricature generation methods accomplish the exaggeration task by employing deep neural networks in an image-to-image translation manner [3], [10], [12], [13]. Certain approaches employ point-based warping techniques to convert real images into caricatures [3].

Furthermore, there has been considerable research into automatic portrait style transfer, which is based on image style transfer [14], [15], [16] and image-to-image translation [17]. Deep learning techniques have successfully performed image translation by learning from representation data examples [18] and [19]. Unfortunately, paired real and caricature are not commonly found. Training the translation process in a supervised manner is impractical, and creating such a dataset can be laborious. One of the readily accessible caricature datasets is WebCaricature [20], encompassing 6042 caricatures and 5974 photographs spanning 252 distinct identities. However, it's worth noting that the dataset's quality is subpar, with caricatures exhibiting inconsistent styles and exaggerations.

Due to the scarcity of paired image data, image-to-image translation is increasingly shifting towards training with unpaired images [19], [21], [22], as well as gaining insights from unpaired portraits and caricatures [13], [23]. Several studies [19], [24] have introduced unsupervised cross-domain image translation approaches, aiming to learn both geometric deformation and appearance translation simultaneously. However, training on unpaired images may introduce significant variations in exaggerations due to the substantial gap in shape and appearance between real and caricature images, often leading to unsatisfactory outcomes. Additionally, differences in poses and scales among images can make it challenging to differentiate facial features.

Neural style transfer techniques employ deep neural networks to transfer artistic styles from a reference to images and excel in stylizing appearances but do not enhance the geometric features [25], [26]. However, the advancement of Generative Adversarial Networks (GANs) [27] has led to the emergence of state-of-the-art face generators like StyleGAN [28], [29], which offer disentangled and high-fidelity images through transfer learning.

Our face caricature approach is different from the previous methods. Our main goal is to exaggerate facial features while keeping them realistic and usable in real-world scenarios. Following the work in [30], our caricature exaggerates the eye and mouth regions, keeping the face contour and other

facial features unchanged. We create our realistic caricatures with the focus on three goals: (1) realistic face caricature with exaggerated eyes and mouth region, (2) making sure our caricature identity is the same as the input face, (3) our caricature should be realistic enough to be usable in real-world scenarios, and (4) unconditional visual style transfers and conserving all facial attribute from the real image to the caricature faces.

We proposed a novel caricature creation with a realistic style applicable to the real world. Style translation refers to the conversion of one style representation to another style representation. We utilize an unpaired caricature learning method to achieve our goal. We exaggerate facial features and the stylization of appearance through a two-step process: face caricature generation and face caricature projection. The initial phase of the face caricature generation step is the creation of new caricature face datasets from real images. We then train a style generator using the real and the new caricature datasets, discussed in Section III-B. The face caricature projection employs an encoder trained with our pre-trained generator. The encoder is trained using real and our new caricature images to project similar real and caricature faces. Additionally, using the projected real and caricature images, we achieve an incremental facial exaggeration from the real to the caricature images, which provides flexibility in our method. The projection of the real and caricature images preserves the facial identity, attributes, and expressions from the input images, discussed in Section III-C. Our method also addresses facial occlusions like reading glasses or sunglasses to enhance the robustness of our model.

This work presents several significant contributions:

- 1) Our approach employs an unpaired learning procedure to produce caricatured faces from real facial images. We don't require a pair of real and caricature images. It accomplishes both facial exaggeration and style transfer from the real face image.
- 2) We produce caricature face datasets from real face images. We train StyleGAN using the real and the new caricature faces, enabling the synthesis of different styles of real and caricature faces. We further designed an encoder to get the full translation of expressions, poses, and attributes from the real faces.
- 3) Our caricature projection provides an incremental exaggeration of facial features. This incremental process provides flexibility in our caricature projection as the extent of facial exaggeration can be performed according to one's preference.
- 4) Our generated caricatures exhibit superior realism and quality compared to state-of-the-art methods. Our caricatures maintain high quality, making them more visually convincing when used in real-world scenarios.

The remainder of the paper is structured as follows: Section II provides the necessary reviews of related works; Section III outlines the two-stage proposed framework, explaining face caricature generation and face caricature projection;

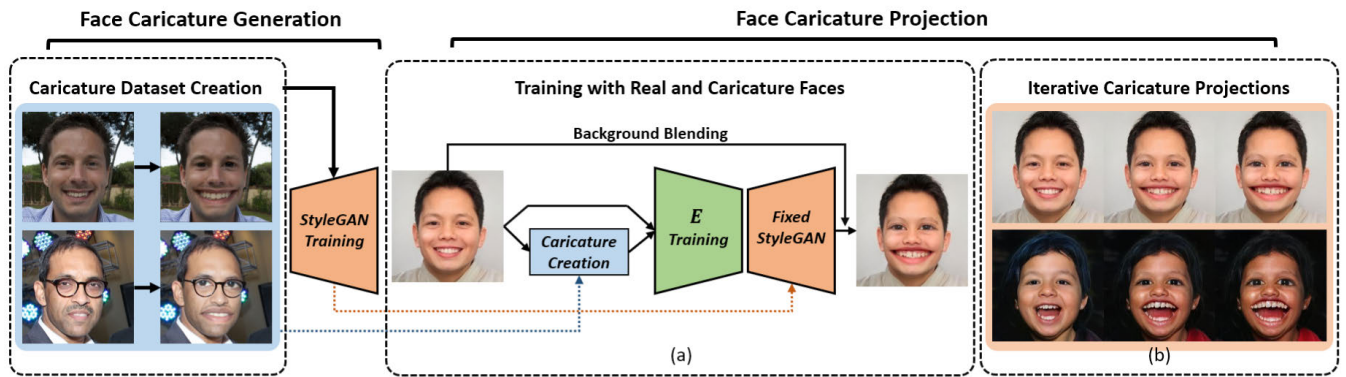


FIGURE 1. We present the overview of our proposed method. Our method consists of two key steps: **Face Caricature Generation** and **Face Caricature Projection**. In the first step, **Face Caricature Generation**, we create a caricature dataset from real faces. A StyleGAN is trained with real and caricature faces, which can produce face caricatures and real images with different styles. The second step, **Face Caricature Projection**, involves training an encoder using the pre-trained StyleGAN. (a) The encoder training process uses real and newly created caricature faces. (b) The first row shows the incremental facial exaggeration from real to caricature faces, and the second row shows the style change with facial exaggeration.

Section IV details implementation settings and dataset uses; Section V details experiment settings and evaluates the results; and Section VI concludes this paper with discussions.

II. RELATED WORK

A. CARICATURE CREATION

Creating caricatures entails the recognition and exaggeration of unique facial characteristics while preserving the individual's identity. Caricatures can be crafted using three approaches: distorting facial attributes, employing style transfer, or utilizing methods that combine both techniques.

Conventional techniques operate by amplifying the deviation from the average, achieved through methods such as explicitly identifying and warping landmarks [31], [32] or employing data-driven approaches to estimate unique facial attributes [33], [34]. As generative networks have advanced, some image-to-image translation methods [35], [36] have been undertaken to incorporate style transfer. Nevertheless, these networks are unsuitable for applications involving significant spatial variations, resulting in outputs with diminished visual quality. Zhang et al. [37] introduced an approach that aims to acquire a disentangled feature representation of various facial attributes, enabling the generation of realistic portraits that exhibit appropriate exaggerations and a rich diversity in style.

Cao et al. [13] employ two CycleGANs, trained on image and landmark spaces, to handle texture rendering and geometry deformation. WarpGAN [3] surpasses the visual quality and shape exaggeration, providing spatial variability flexibility for image geometry and texture. CariGAN [13], on the other hand, is a GAN trained with unpaired images, focusing on learning image-to-caricature translation. Shi et al. [3] introduce an end-to-end GAN framework that simultaneously trains warping and style. AutoToon [12] utilizes deformation fields to apply exaggerations and is trained in a supervised manner using paired data derived from artist-warped photos to learn warping fields. However, it is

limited to mapping to a single domain, making it unable to produce diverse exaggerations. Abdal et al. [38] introduced a technique for crafting 3D caricatures that permits modifying and animating personalized 3D avatars using artistic datasets.

B. STYLE TRANSFER

One aspect of image synthesis that poses a challenge is style transfer, which aims to create a content image with multiple styles. Numerous networks dedicated to style transfer have been developed thanks to the practical ability of convolutional neural networks (CNNs) [39] to extract semantic features. The initial style rendering process was introduced by Gatys et al. [40], who employed hierarchical features from a VGG network [41]. Gatys et al. [42] pioneered the first neural style transfer approach, utilizing a CNN to transfer style information from a style image to a content image. However, a drawback of this approach is that the style and content need to be similar, which is different from caricature images.

A promising area of research lies in the application of Generative Adversarial Networks (GANs) [27] for image synthesis, which has yielded cutting-edge results in various domains such as text-to-image translation [43] and image inpainting [44]. Methods like CycleGAN [22] have been utilized for unpaired image translation, leveraging a cycle consistency loss to achieve translation between different image domains. Additionally, approaches like StarGAN [21], [45] employ a single generator to learn mappings across various image domains. However, capturing the geometric transformations required for direct photo-to-caricature mapping in an image-to-image translation framework remains a challenging task.

StyleGAN [28], [46] excels at producing high-fidelity facial images with fine-grained control over hierarchical channel styles. Many techniques leverage StyleGAN for the generation of high-quality images and the manipulation of facial characteristics, as well as for various applications

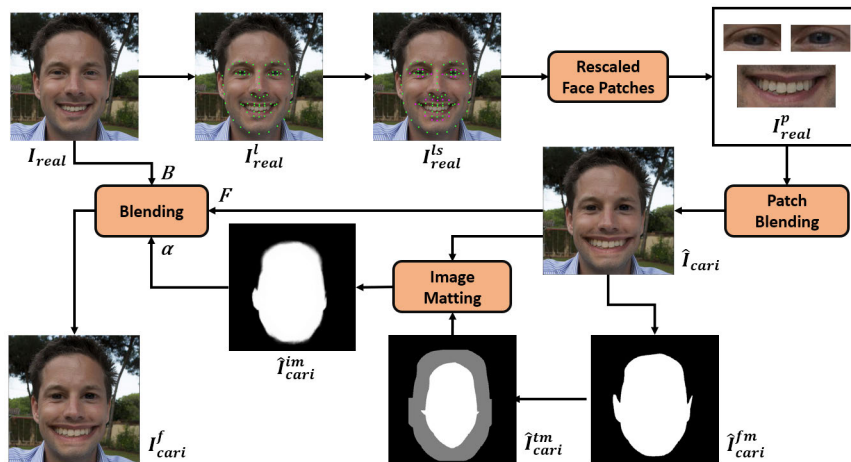


FIGURE 2. The steps for our face caricature data creation. We perform multiple image operations techniques to achieve our face caricature dataset.

related to faces, including swapping, restoration, de-aging, and reenactment [47], [48]. Pinkney and Adler [49] further enhanced StyleGAN using sparse cartoon data, demonstrating its effectiveness in generating lifelike cartoon faces. Additionally, DualStyleGAN [50] provides customizable control over dual style transfers, catering to both the extended artistic portrait domain and the original face domain. StyleCariGAN [51] produced shape exaggeration and stylization by mixing layers of photo and caricature styles.

III. STYLE BASED CARICATURE CREATION

Our proposed method operates transparently and understandably, comprising two distinct stages: face caricature generation and face caricature projection, as illustrated in Figure 1. In the initial stage, we focus on creating new facial caricature datasets, which exaggerate eye and mouth regions while preserving the facial contours. Subsequently, we train a style generator called StyleGAN [28] using the real and our new caricature datasets, which can generate highly realistic images in different styles. In the second stage, we design a projection model to produce high-quality caricature faces from real facial images and the incremental exaggeration of facial features. Our proposed method projects the genuine facial image into a caricature representation, emphasizing the unique and exaggerated facial features that constitute an individual's appearance.

A. BACKGROUND

1) StyleGAN

We use StyleGAN2's [46], a style-based network that controls the synthesis of images. To train StyleGAN, a large dataset of real images is used, which is then processed to learn the underlying patterns and characteristics of the data. We use real and caricature faces for our method. The model learns to

generate real and caricature images visually similar to those in the training set.

2) PROJECTION TECHNIQUES

We can project an input image into an equivalent output image using the StyleGAN architecture by employing two distinct approaches: latent code optimization and encoder-based methods. Our approach is predominantly focused on encoder-based methods for several compelling reasons. Firstly, these methods offer significant speed advantages, as they can map the latent code in a single forward pass. The encoder-based approach contrasts with the optimization-based approach, which can be computationally demanding for each image. Secondly, the output of an encoder resides within a compact and well-defined space, rendering it more suitable for subsequent editing and manipulation tasks.

3) NATURE OF StyleGAN LATENT SPACE

The StyleGAN latent space plays a crucial role in creating and manipulating the characteristics of the generated real and caricature faces. The latent space vectors control various aspects of image generation, like facial features, colors, textures, etc. Another characteristic of StyleGAN latent space is the nature of disentanglement, where each latent space direction corresponds to specific features or attributes of the generated image. A smooth interpolation between two points in the latent space creates images that transition between different attributes.

B. FACE CARICATURE GENERATION

In the face caricature generation, datasets are formed by creating exaggerated facial representations featuring enlarged eyes and mouths using real face images. Here, we also discuss generating exaggerated faces with facial obstacles, like the faces with eyeglasses. After creating caricature faces,

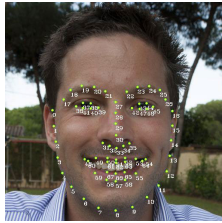


FIGURE 3. Face landmarks representation: The landmark position with the 68 indexes representation on the FFHQ [28] dataset.

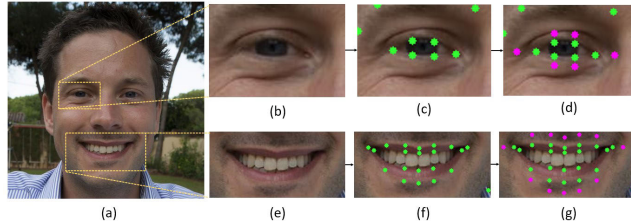


FIGURE 4. Face landmark enlargement for patch generation. (b-d) represents the eye landmark enlargement. (e-g) represents the mouth landmark enlargement. The landmarks produced by the detector are represented in green, and the enlarged landmarks are represented in pink.

We followed the generation process by training a style generator with our caricature face dataset.

1) FACE CARICATURE DATASET CREATION

To create our caricature faces, we utilize real-face images randomly sampled from the FFHQ [28] and CelebA-HQ [52] dataset. Both datasets provide a diverse range of genders, races, ages, expressions, and poses, ensuring the variety and representation of our caricature faces. The pipeline for caricature creation is divided into three stages: (i) facial landmark enlargement, (ii) face patch rescaling, and (iii) image matting, as illustrated in Figure 2.

In the first stage, we use landmark detectors to detect the facial landmarks in the real input image I_{real} . Specifically, we employ a pre-trained detector from the Dlib library [53], which estimates the location map of the facial structure. The Dlib library detects 68 facial landmarks, each assigned specific (x, y) coordinates ranging from 0 to 67. These landmarks correspond to different parts of the face, such as the eyes, eyebrows, nose, mouth, and face contour, as depicted in Figure 3. We represent the input face image with identified landmarks as I_{real}^l , and the coordinates are highlighted with green markers. This initial stage of landmark detection provides crucial information about the facial structure, enabling us to proceed to the subsequent steps of face patches and blending, as well as image matting and blending.

In the second stage, face patch rescaling, we perform several operations, including producing face patches, exaggerating them, and blending them into the original image to create the caricature effect. Our focus for exaggeration



FIGURE 5. (a) The blurring of face patches after the blending process due to extreme head pose. (b) The result of removal of blurring after the image matting.

is on the eyes and mouth regions of the face. To target the eye regions accurately, we group the landmark indexes into the left and right eyes, as depicted in Figure 3. The mouth area comprises the upper and lower lips; we consider specific landmark indexes for these regions. In the case of the mouth, we utilize the top landmark indexes for the upper lip and the bottom indexes for the lower lip. Using these landmark indexes, we produce face patches that will undergo exaggeration. We achieve this by enlarging the coordinates of the landmarks corresponding to the eye and mouth regions, as illustrated in Figure 4. The resulting image I_{real}^{ls} displays the face with enlarged landmark coordinates, which are highlighted in pink. To further enhance the exaggeration effect, we scale the face patches to a factor of 1.5, resulting in exaggerated face patches represented as I_{real}^p . These exaggerated patches seamlessly blend into the original image I_{real} .

For the blending process, we employ the Poisson image editing technique [54], which ensures seamless and natural integration of the exaggerated patches with the original image. This technique considers factors such as image illumination and texture, resulting in a visually pleasing caricature effect. By applying these operations, we can generate the final caricature image \hat{I}_{cari} , where the distinctive exaggerated features, such as enlarged eyes and mouth, seamlessly blend into the original face image while maintaining a natural appearance. The Poisson editing method influences both image illumination and texture and is represented as follows:

$$v = argmin_v \sum_{i \in S, j \in N_i \cap S} ((v_i - v_j) - (s_i - s_j))^2 + \sum_{i \in S, j \in N_i \cap \bar{S}} ((v_i - t_j) - (s_i - t_j))^2, \quad (1)$$

where v represents the pixel values of the new image, s corresponds to the pixel values of the source image, t represents the pixel values of the target image, S signifies the destination domain, and N_i denotes a set of neighboring pixels of i .

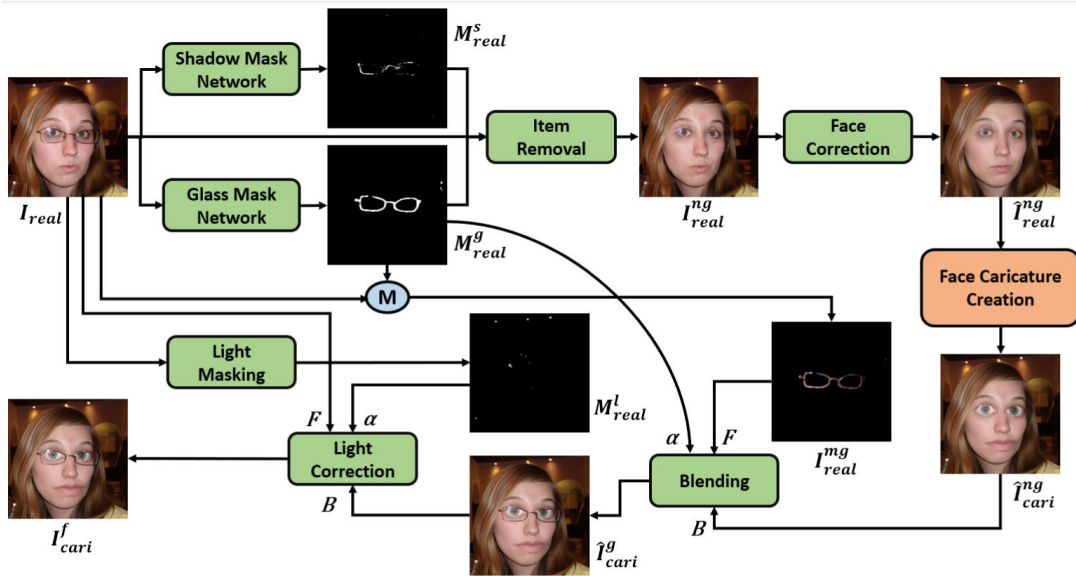


FIGURE 6. The workflow of creating the reading glass caricature face dataset. We perform additional steps to overcome the face occlusion challenges for reading glass caricature creation.

In the third stage, we tackle the problem of blurriness that can occur along the contours of the face, especially in cases where faces exhibit extreme poses during the blending process in stage two. To mitigate this blurring effect, we apply an image matting technique. First, we generate face masks from the previously obtained caricature image \hat{I}_{cari} using a face segmentation method [55], resulting in a mask image \hat{I}_{cari}^m . In this mask, the foreground corresponds to the face region, while the background encompasses the remaining areas. Notably, we perform segmentation only for the face region, excluding the hair, as blurring tends to occur mainly in the hair-background region.

Next, we generate a trimap mask \hat{I}_{cari}^{tm} from the face mask \hat{I}_{cari}^m using trimap mask generation process [56]. It involves applying a series of erosion and expansion operations to the foreground region of the face mask, using specific parameter values tailored to our method. With the face caricature image \hat{I}_{cari} and the trimap mask \hat{I}_{cari}^{tm} in hand, we apply an image matting method [57]. This technique effectively addresses the blurring issue by enhancing the sharpness and clarity of the face contours in the caricature image. The image matting process utilizes both the caricature image and the trimap mask to generate a refined caricature image \hat{I}_{cari}^{im} . By employing this image matting stage, we can improve the overall visual quality of the caricature image by reducing blurring effects around the face contours, resulting in a more polished and realistic appearance.

The blurring on the face contour is removed by performing alpha blending. The image alpha blending technique requires a foreground, a background, and an alpha mask. We set the \hat{I}_{cari} as foreground, I_{real} as background, and \hat{I}_{cari}^{im} as alpha mask. The alpha blending can be performed using the

following equation:

$$I_p = \alpha_p F_p + (1 - \alpha_p) B_p, \quad (2)$$

where α_p denotes the matte and within the range value of $[0,1]$, and F_p and B_p correspond to the pixel values for the foreground and background, respectively. When $\alpha_p = 1$ or 0 , it signifies that the pixel at that position unequivocally belongs to the foreground or background respectively. Otherwise, such a pixel is termed a partial or mixed pixel. Following the ultimate blending procedure, we produce our caricatured face denoted as \hat{I}_{cari} . Figure 5 illustrates eliminating blurring after the matting process.

2) FACE CARICATURE DATASET WITH OCCULSION

For generating our caricature dataset, we use various images, including faces with eyeglass occlusions. To enrich our caricature dataset, we address the caricature generation for face occlusion caused by eyeglasses. The faces with eyeglasses can be categorized into (i) Reading glasses and (ii) sunglasses. We organized all the transparent glasses as reading glasses and the remaining as sunglasses.

Face Caricature with Reading Glasses: The whole pipeline for reading glass caricature generation is shown in Figure 6. We can divide the reading glass caricature generation into five stages: (i) glass removal, (ii) correction, (iii) caricature generation, (iv) putting back glasses, and (iv) lighting correction.

The first stage is glass removal, where we remove both the reading glass and the cast shadow from the face image. We employ two networks, Shadow Mask Network and Glass Mask Network [58], for the glass removal. Given an input image I_{real} , we generate two masks: a glass mask M_{real}^g using the glass mask network and a shadow mask M_{real}^s using the



FIGURE 7. Examples of our dataset produced in the face caricature creation.

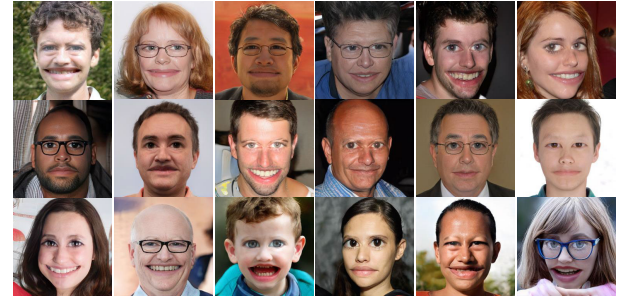


FIGURE 8. Random sample of caricature faces generated by the StyleGAN.

shadow mask network. We use item removal from [58] to remove both the eyeglass and the shadow from the face image and generate a new face image I_{real}^{ng} with no eyeglass and cast shadow.

The second stage is the correction stage, where information on the image that was lost during the item removal stage is retrieved. We use an image restoration method [59] to restore the degraded image and restore lost details. The corrected image \hat{I}_{real}^{ng} restores both quality and fidelity and shows robustness to the degraded parts.

The third stage is the caricature generation process. The face image now has no reading glasses, so it performs the caricature generation method discussed in the previous section and generates appropriate caricature \hat{I}_{cari}^{ng} .

We put the glasses from I_{real} into \hat{I}_{cari}^{ng} in the fourth stage. We first generate a glass image I_{real}^{mg} with only glasses using a bitwise AND mask operation using I_{real} and M_{real}^g . We perform the alpha blending presented in Equation 2 to put back the reading glass from I_{real} in our caricature face. We set the I_{real}^{mg} as foreground, \hat{I}_{cari}^{ng} as background, and M_{real}^g as alpha mask. The generated caricature face with reading glasses is represented as \hat{I}_{cari}^g .

The final step is the lighting correction. During this caricature process, we must preserve the light illumination from I_{real} . We generate a light mask M_{real}^l from I_{real} by keeping a specific threshold that only the illuminated area is highlighted. We perform the alpha blending technique in Equation 2 to retrieve the lost light illumination from I_{real} . We set I_{real} as foreground, \hat{I}_{cari}^g as background, and M_{real}^l as alpha mask. Finally, we create our reading glass caricature image, I_{cari}^f .

Face Caricature with Sunglasses: We consider the face with sunglasses where it can't be see-through. The caricature generation of faces with sunglasses is a simple, straightforward process where we exaggerate face patches only for the mouth region. After the landmark detection in Figure 2, only the mouth landmark has been enlarged, represented in Figure 4. The patch blending presented in Equation 1 is performed only for the mouth patch and generates a caricature face with sunglasses. The remaining steps are the same as in Section III-B.



FIGURE 9. After StyleGAN training, we represent different styles for a specific face. Each row represents one identity.

3) FACE CARICATURE DATASET

We have successfully generated a diverse collection of caricature face images encompassing various attributes such as gender, race, age, expression, pose, illumination, etc. We use the FFHQ [28] and CelebA-HQ [52] datasets for our caricature creation. Figure 7 illustrates some examples of our caricature dataset.

4) STYLE GENERATOR

The final step for the caricature generation is the training of StyleGAN [28], [29] architecture. The StyleGAN architecture comprises two networks: a mapping network and a synthesis network. The mapping network, denoted as f , is an 8-layer Multi-Layer Perceptron (MLP) responsible for mapping a given latent code z from the set Z to generate w in the set W . It can be represented as $f : Z \rightarrow W$. The synthesis network, g , consists of 18 convolutional layers, with each layer being controlled via adaptive instance normalization (AdaIN) [60]. AdaIN incorporates the learned affine transformation “A” derived from the latent code w at each layer. Additionally, a scalable Gaussian noise input “B” is introduced into each layer of the synthesis network g .

The architectural design ensures that each style influences only a single convolution. Random latent codes serve as a means to control the styles of the generated images. The StyleGAN training process exclusively used the real and newly created face caricature images. Following the training of StyleGAN, the generator can produce real and caricature

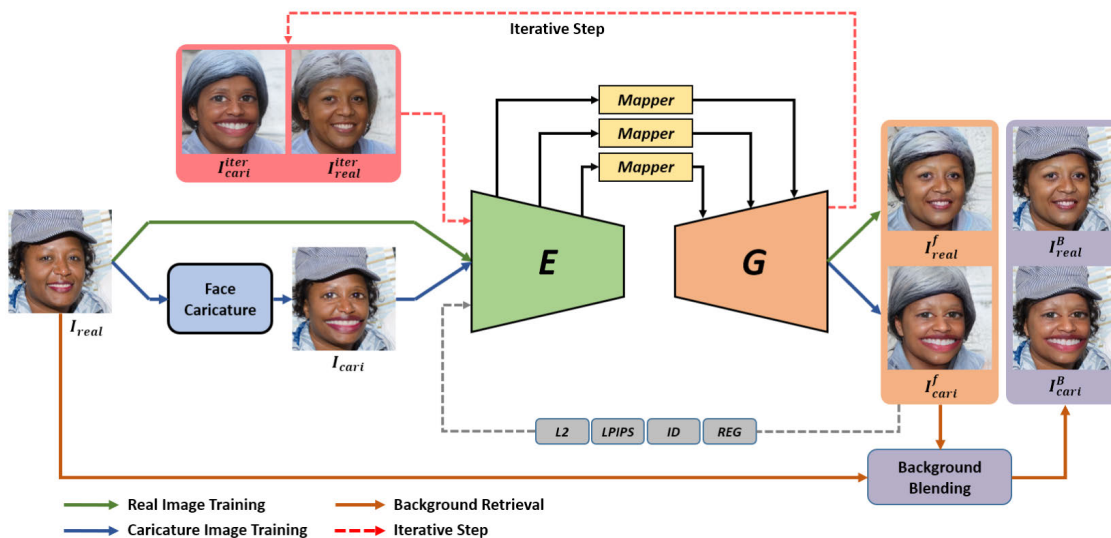


FIGURE 10. The encoder training for our Face caricature projection. We employ an encoder-based method with our pre-trained StyleGAN, G . The encoder E is trained using real and new caricature faces. We perform iterative steps to enhance the quality of our generated images and make them more faithful to the input faces. We perform a background blending process to get the background information the projected image cannot generate from the input image.

images with diverse facial attributes, including variations in skin tone, hair color, and shapes. It’s crucial to underscore that our caricature generation generator stands out from previous approaches in a notable manner in terms of realism and usability. After training the StyleGAN, we generated random samples from the latent space to visualize how our caricature performed. The results are high quality and realistic, as shown in Figure 8. We can also generate different styles for different identities, and some examples are shown in Figure 9.

C. FACE CARICATURE PROJECTION

Our caricature projection technique employs an encoder trained with two different datasets. The encoder is trained using real and caricature images with our pre-trained StyleGAN from Section III-B4. The training framework of the encoder is shown in Figure 10. For training the encoder, denoted as E , with our pre-trained StyleGAN generator, represented as G , given an input source image I_{real} , we first create a corresponding caricature from the real input face, I_{cari} , following the process in Section III-B1. The newly created caricature faces with the real faces are used in the training of E with the primary objective of $I_{cari}^f = G(E(I_{cari}))$, such that $I_{cari}^f \approx I_{cari}$ and $I_{real}^f = G(E(I_{real}))$, such that $I_{real}^f \approx I_{real}$. To enhance the quality of our generated images and make them more faithful to the input, we perform two forward passes through the encoder, E , and generator G . Our goal is to efficiently and effectively produce high-quality real and caricatured faces while preserving the desired characteristics and visual resemblance to the input images.

Follows a methodology similar to the PSP [61] and e4e [62] approaches. We utilize a Feature Pyramid Network [63] built upon a ResNet [64] backbone, extracting style features from three intermediate levels. Our pre-trained StyleGAN is

kept fixed during the caricature projection process. Much like the PSP network, we employ “Mapper”, a small mapping network, which is trained to extract learned styles from the corresponding feature maps for each of the 14 target styles (for 256×256 images). This small mapping network is fully convolutional, downsampling the feature map to generate the corresponding 512-dimensional style input. It achieves this through a series of 2-strided convolutions followed by LeakyReLU activations. Specifically, the small feature map from the Mapper generates styles $W_0 - W_2$, the medium feature map generates styles $W_3 - W_6$, and the large feature map generates styles $W_7 - W_{17}$. We incorporate Restyle’s [65] iterative refinement method to enhance the reconstruction quality with each iterative step. We perform a single training iteration per batch with our model trained. The iterative outcome for I_{real} is I_{real}^{iter} and I_{cari} is I_{cari}^{iter} . To further transfer the background from I_{real} , a common approach involves applying a blending technique, as depicted in Equation 2, as a post-processing step that swaps the inner face of I_{real}^f and I_{cari}^f with I_{real} . We execute the image matting technique, followed by the alpha blending process discussed in Section III-B, to produce our final projection I_{real}^B and I_{cari}^B .

1) INCREMENTAL CARICATURE PROJECTION

We utilize the disentangled nature of the StyleGAN latent space to create caricature faces of desired facial exaggeration. The disentangled latent spaces also facilitate smooth and predictable transitions between real and caricature faces. We employ our trained encoders, E , and our pre-trained StyleGAN to perform the incremental facial exaggeration process. Figure 11 shows the overview of our incremental facial exaggeration. Given an input image I_{real} , we create the corresponding caricature of the real face, I_{cari} . We fed the

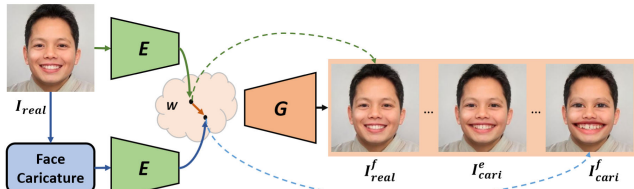


FIGURE 11. After training the encoder, we perform incremental caricature projection by performing latent walking from the real image toward the direction of the corresponding caricature image.

I_{real} and I_{cari} to the encoder E . We Project two latent codes in the StyleGAN latent space W , one for I_{real} , represented as $E(I_{real}) = z^{real}$ where z^{real} is the real latent code, and another for I_{cari} , represented as $E(I_{cari}) = z^{cari}$ where z^{cari} is the caricature latent code. We perform a latent walk from z^{real} to z^{cari} with the objective of $I_{cari}^f = G(z^{real} + n_{cari})$ if $n_{cari} = 1$, where n_{cari} is the incremental latent steps of z^{cari} direction. We can perform a uniform iteration represented as I_{cari}^e . We visualize more results in Section V-C.

2) LOSSES

To achieve our objective, we employ a variety of losses during the training of our encoder. We incorporate the non-saturating GAN loss [66] along with R_1 regularization [67] as the adversarial loss, proposed in [68].

The purpose of regularization is to encourage the encoder to produce latent-style vectors that are closer to the average latent vector. The formulation of the regularization loss is as follows:

$$\mathcal{L}_{reg} = \| G(E(x)) - \bar{w} \|_2, \tag{3}$$

where \bar{w} represents the average style vector obtained from our pre-trained generator.

We employ the pixel-wise L2 loss,

$$\mathcal{L}_2 = \| x - G(E(x)) \|_2, \tag{4}$$

To preserve the perceptual similarity, we use LPIPS [69] loss. The image is preserved better [70] as compared to the traditional approach [25].

$$\mathcal{L}_{LPIPS} = \| F(x) - F(G(E(x))) \|_2, \tag{5}$$

We employ identity loss to generate a caricature face that retains similar facial characteristics. This involves integrating a dedicated recognition loss, which assesses the cosine similarity between the resulting image and its source.

$$\mathcal{L}_{ID}(x) = 1 - \langle A(x), A(G(E(x))) \rangle, \tag{6}$$

where A represents the pre-trained ArcFace [71] network. Collectively, our overall loss function is presented as

$$\mathcal{L}(x) = \lambda_1 \mathcal{L}_2(x) + \lambda_2 \mathcal{L}_{LPIPS}(x) + \lambda_3 \mathcal{L}_{ID}(x) + \lambda_4 \mathcal{L}_{reg}(x), \tag{7}$$

where $\lambda_1, \lambda_2, \lambda_3,$ and λ_4 are constants defining the loss weights.

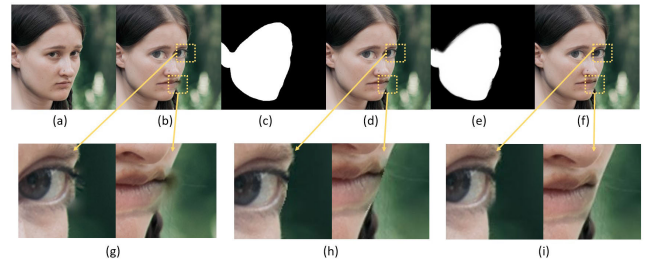


FIGURE 12. Experiment results with various masks during the blending process. (a) The real input face, (b) our face caricature dataset before blurring removal, (c) the facial mask obtained from image segmentation, (d) the blending of the original image with the facial mask, (e) the facial mask acquired through image matting, (f) the blending of the original image with the image matting mask, (g) blurring effect to the eyes and mouth region, (h) display of unnatural blending results, and (i) demonstrating the appropriate and natural blending of the image matting face mask with the original face.

IV. IMPLEMENTATION

A. DATASET

To showcase the efficacy of our approach, we produce caricature datasets and conducted experiments using a diverse dataset that encompasses two widely recognized datasets: FFHQ [28] and CelebA-HQ [52]. The FFHQ dataset comprises 70,000 high-quality facial images, which we segmented into three groups based on the presence of eyeglasses: no glasses, reading glasses, and sunglasses. Specifically, we assigned approximately 56,500 images to the no-glasses group, 10,600 images to the reading glasses group, and 2,900 images to the sunglasses group. Similarly, the CelebA-HQ dataset contains 30,000 high-quality facial images, and we also categorized these into three groups: no glasses, reading glasses, and sunglasses. Here, we allocated approximately 28,500 images to the no-glasses group, 1,000 images to the reading glasses group, and 500 images to the sunglasses group.

We use our new caricature and real faces from the FFHQ dataset to train our StyleGAN model. The FFHQ dataset's considerable size and high-quality image content render it suitable for effectively training a robust and representative caricature generator. For the encoder E training, we use the FFHQ real and our new caricature dataset as the training set and the CelebA-HQ real and our new caricature dataset as the testing set. By using these diverse datasets and splitting them into different groups based on eyeglass presence, we aimed to assess the ability of our approach to handle various scenarios and generate accurate caricatured faces across different styles and eyeglasses.

B. IMPLEMENTATION DETAILS

We trained a StyleGAN model [28], [46] using real and our caricature datasets. Since our hardware resources are limited, the input and output image resolution for our caricature generation task was set to 256×256 pixels. The training process for the StyleGAN model was conducted on four Nvidia Titan Xp GPUs, each with 12 GB of RAM. It took approximately eight days to train the model using a batch size



FIGURE 13. Results of various stages in the process of generating a reading glass caricature dataset: (a) The real input face, (b) the result after removing reading glasses and cast shadows, (c) the result after the face correction technique, (d) after applying our face caricature, and (e) the reading glass caricatured face.

of 16. For the encoder E training, we utilized the ResNet-IRSE50 architecture from Arcface [71], a pre-trained model commonly used for facial recognition tasks. In our E training process, we set the values of the constants λ as follows: $\lambda_1 = 1$, $\lambda_2 = 0.8$, $\lambda_3 = 0.5$, and $\lambda_4 = 0.005$. These constants were used to control and balance different aspects of the training process. We set other training details the same as [61].

V. EXPERIMENTS

A. EXPERIMENTS ON CARICATURE GENERATION

The process of patch blending is of utmost importance in creating the face caricature dataset. The quality of the StyleGAN-generated images greatly depends on the seamless blending of these patches. However, the blending process can sometimes be blurry when dealing with extreme head poses. To address this blurring issue, we employ a face mask that eliminates all blurriness, resulting in a more natural-looking image. Additionally, we introduce a face mask in conjunction with a matting mask and compare the outcomes, as demonstrated in Figure 12. The image matting mask successfully eliminates blurriness along the facial contours, yielding a more natural appearance than just the face mask. It's worth noting that the face segmentation mask tends to produce unnatural edges, which can adversely impact the final result.

While removing reading glasses, the facial details concealed behind the glasses are inevitably lost. We employ a correction technique to address this issue and recover the lost information. We illustrate the various stages of the reading glass removal process in Figure 13. After the



FIGURE 14. The projection result of the encoder trained with real and caricature images with background blending. (a) The real input face, (b) The real projected face, (c) The caricature input face, and (d) The caricature projected face.

eyeglass removal, a significant portion of the information in the eye region is degraded, which can adversely impact caricature generation. However, the correction method not only restores both quality and fidelity but also exhibits remarkable resilience in handling the deteriorated portions, ultimately enhancing the creation of a superior caricature dataset.

B. EXPERIMENTS ON CARICATURE PROJECTION

Following the training of our caricature projection using the encoders trained on real and our caricature datasets, we conducted a series of experiments to assess the efficacy of our projection method. We evaluate the encoder results by comparing the input and output faces for the real and caricature images, as illustrated in Figure 14. The outcome of projecting real and caricature images demonstrates the effectiveness of our approach, as the input and the resulting projected images exhibit minimal differences. It is evident that our real and caricature projections consistently yield attractive and aesthetically pleasing outcomes. Table 1 shows the evaluation of our projection results.

Each iterative step enhances the image quality during our projection method, as illustrated in Figure 15. The iterative process enhanced the eyeglasses information, as demonstrated in rows 1, 2, 3, and 6. It's noteworthy that there is a substantial improvement in head pose and facial expression, as observed in row 4. There is also an improvement in the

TABLE 1. Comparison between the real and caricature projections results.

Projection	↓ LPIPS	↓ L2	↑ SSIM
Real Face Projection	0.049	0.008	0.91
Caricature Face Projection	0.056	0.009	0.90



FIGURE 15. Result of our iterative steps during our caricature projection. (a) The real input face, (b) the initial projected face after caricature creation from the corresponding real face, (c) the final iterative step, and (d) our projected caricature face (with background blending).

lighting and skin color resemblance with the input image, as observed in row 5.

C. EXPERIMENTS ON INCREMENTAL PROJECTION

We perform an incremental caricature projection method in which we gradually exaggerate facial features, as demonstrated in Figure 16. The visual result shows that the exaggeration affects the eyes and mouth, leaving all other facial attributes unchanged. The exaggeration steps are crucial in our method as they hold great significance in our approach, as the extent to which individuals prefer facial exaggeration varies. This step provides flexibility and robustness in our caricature projection process.

Furthermore, we introduce a style-mixing element into the exaggeration process. We can select and incorporate the desired style during the exaggeration process, as shown in Figure 17. The desired style can be blended by mixing the style codes within the finer layers of StyleGAN.

D. COMPARISON TO STATE-OF-THE-ART METHOD

We evaluate the performance of our caricature projection method by comparing it to the state-of-the-art technique.



FIGURE 16. Result of our incremental facial exaggeration steps from the real face to the corresponding caricature face. (a) The real input face, and (b) the final caricature face.



FIGURE 17. Result of our incremental facial exaggeration steps with desirable style change from the real face to the corresponding caricature face. (a) The real input face.

This comparison encompasses all the encoder-based to assess the efficacy of our method comprehensively. We conduct



FIGURE 18. Qualitative comparison results of different face projection techniques using the StyleGAN, which is trained on our caricature dataset. (a) The real input face, (b) hyperstyle encoder, (c) e4e encoder, (d) restyle encoder and (e) our projected caricature face.

experiments utilizing various approaches in our qualitative evaluation, as depicted in Figure 18. We explore three encoder-based methods: Hyperstyle [72], e4e [62] and Restyle [65]. These encoders are trained using a pre-trained StyleGAN, which is trained using only our caricature images. Hyperstyle tends to generate caricatures that closely resemble the original image in terms of structure, as it tunes the StyleGAN weights to retrieve the original image rather than caricature faces. Conversely, the caricatures produced by the e4e encoder yield superior caricature results when compared to Hyperstyle, and the results provide convincing caricature results. The Restyle results resemble real images more than the caricature faces. Finally, our method outperforms all techniques, particularly facial exaggeration, expression, and head pose, which more closely resemble the real image. Furthermore, our approach excels at handling occluded faces and produces caricatures that resemble the original eyeglasses more closely. Overall, our approach demonstrates superior results compared to existing techniques, offering a more faithful representation of the original image's characteristics while achieving high-quality caricature results.

Moreover, we conducted a qualitative analysis of various state-of-the-art caricature methods, comparing our outcomes with those of WarpGAN [3], StyleCariGAN [51], and DualStyleGAN [50], as shown in Figure 19. All results



FIGURE 19. Qualitative comparison results of different face caricature methods. (a) The input real face, (b) WarpGAN, (c) StyleCariGAN, (d) DualStyleGAN, and (e) our projected caricature face.

TABLE 2. Quantitative results for different caricature creation methods.

Method	↑Identity	↓FID	↓LPIPS	↓L2	↑SSIM
WarpGAN [3]	0.26	74.60	0.48	0.65	0.25
StyleCariGAN [51]	0.11	52.35	0.47	0.41	0.32
DualStyleGAN [50]	0.08	104.51	0.42	0.51	0.33
Our	0.37	38.08	0.06	0.01	0.85

were generated using the pre-trained models provided by the respective authors using CelebA-HQ [52]. The WarpGAN struggled to produce caricatures with proper facial structures and weakly stylized images. The StyleCariGAN had difficulty preserving the original image's identity and heavily relied on the chosen style. The DualStyleGAN yielded convincing results but was limited in retaining the original attributes. In contrast, our caricature results excelled in quality, maintaining both style and the facial attributes of the original image. The exaggeration achieved in our projected caricature faces holds promise for practical applications in real-world scenarios. We also performed a quantitative evaluation to assess the degree of resemblance between real and caricature images, as shown in Table 2. The identity similarity calculation uses the ArcFace [71] method. Our method demonstrated the most favorable score, effectively exaggerating facial features and aligning with our primary goal of making it applicable in real-world scenarios.

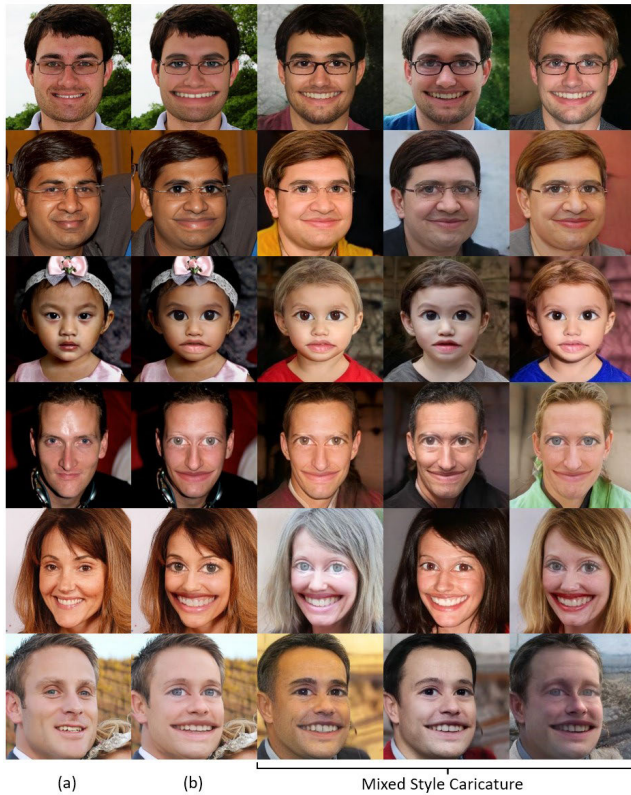


FIGURE 20. Visual results obtained using our method. (a) The input real face, (b) our projected caricature face, and mixed style caricature faces.

We showcase the outcomes of our approach using various facial images captured in diverse conditions, as shown in Figure 20. Our methodology consistently delivers outstanding facial caricature results marked by realism and the retention of the original facial attributes. We visualize the generation of different style types that can be incorporated with our caricature face. Moreover, our method generates faces with occlusions, such as reading glasses and sunglasses. Furthermore, it displays versatility by producing caricatured faces across different age groups and adapting to various styles.

VI. CONCLUSION

In this paper, We generate realistic facial caricatures featuring exaggerated features suitable for real-world applications. Our methodology is carefully crafted to emphasize exaggerating the eyes and mouth while preserving the original facial contours. We have introduced an innovative caricature generation method that comprises two key stages: face caricature generation and face caricature projection. We construct caricature datasets using real images in the face caricature generation phase. Subsequently, we train a StyleGAN to synthesize various styles of real and caricatured faces.

The face caricature projection step transforms input images into corresponding real and caricatured faces. Our caricature projection process excels at producing highly

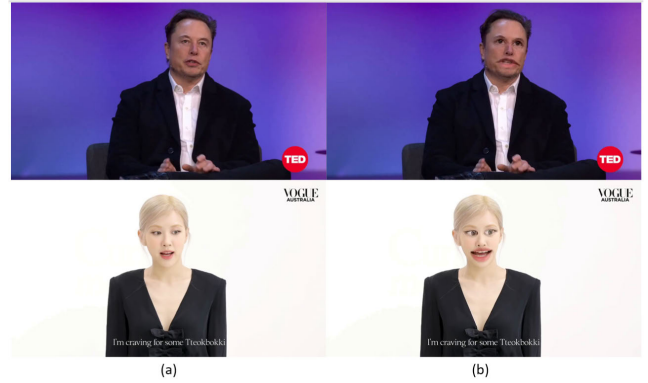


FIGURE 21. Visualization for protecting the privacy of an individual using our caricature faces in a full image. (a) The input real face, and (b) our projected caricature face.

realistic results while faithfully retaining the original facial attributes and identity. We also perform an incremental caricature projection method, gradually exaggerating facial features. We emphasize the importance of the exaggeration steps in our technique because different people have varying preferences regarding facial exaggeration. This process gives our caricature projection process flexibility and resilience. Our caricatures stand out in their superior realism and quality compared to previous methods. They offer visually convincing results suitable for real-world applications.

Our approach introduces an innovative method for crafting exaggerated facial representations while maintaining a realistic style. Our future work includes exploring face de-identification using our caricature-projected faces to conceal the important facial features that can be used to identify the individual. Our core concept revolves around using caricature faces to protect individuals’ privacy. We illustrate one example of protecting the privacy of an individual using our caricature faces in Figure 21.

A. LIMITATIONS

It’s crucial to explore the adaptability of our method in various applications. However, it’s essential to acknowledge that our approach is tailored to specific applications and does have inherent limitations when applied in broader contexts. Notably, we emphasize exaggerating features in the eyes and mouth region, limiting the range of caricatures we can generate. Additionally, our method relies on real images, restricting its stylistic diversity and making it less suitable for producing different out-of-domain caricatures.

Our method is entirely automated, and future improvements could involve enhancing controllability through additional caricature examples or user interaction. Nonetheless, our approach holds great promise for specific applications, and we are eager to refine and expand its capabilities in the future.

REFERENCES

- [1] S. B. Sadimon, M. S. Sunar, D. Mohamad, and H. Haron, "Computer generated caricature: A survey," in *Proc. Int. Conf. Cyberworlds*, Oct. 2010, pp. 383–390.
- [2] K. Cao, J. Liao, and L. Yuan, "CariGANs: Unpaired photo-to-caricature translation," *ACM Trans. Graph.*, vol. 37, no. 6, pp. 1–14, Dec. 2018.
- [3] Y. Shi, D. Deb, and A. K. Jain, "WarpGAN: Automatic caricature generation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 10754–10763.
- [4] E. Akleman, J. Palmer, and R. Logan, "Making extreme caricatures with a new interactive 2D deformation technique with simplicial complexes," in *Proc. Vis.*, vol. 1, 2000, p. 2000.
- [5] J. Yaniv, Y. Newman, and A. Shamir, "The face of art: Landmark detection and geometric style in portraits," *ACM Trans. Graph.*, vol. 38, no. 4, pp. 1–15, Aug. 2019.
- [6] H.-C. Shin, J. Hee Park, and S.-D. Kim, "Combination of warping robust elastic graph matching and kernel-based projection discriminant analysis for face recognition," *IEEE Trans. Multimedia*, vol. 9, no. 6, pp. 1125–1136, Oct. 2007.
- [7] L. Liang, H. Chen, Y.-Q. Xu, and H.-Y. Shum, "Example-based caricature generation with exaggeration," in *Proc. 10th Pacific Conf. Comput. Graph. Appl.*, 2002, pp. 386–393.
- [8] S. E. Brennan, "Caricature generator: The dynamic exaggeration of faces by computer," *Leonardo*, vol. 18, no. 3, pp. 170–178, 1985.
- [9] Z. Mo, J. P. Lewis, and U. Neumann, "Improved automatic caricature by feature normalization and exaggeration," in *Proc. ACM SIGGRAPH Sketches*, 2004, p. 57.
- [10] X. Han, K. Hou, D. Du, Y. Qiu, S. Cui, K. Zhou, and Y. Yu, "CaricatureShop: Personalized and photorealistic caricature sketching," *IEEE Trans. Vis. Comput. Graphics*, vol. 26, no. 7, pp. 2349–2361, Jul. 2020.
- [11] Q. Wu, J. Zhang, Y.-K. Lai, J. Zheng, and J. Cai, "Alive caricature from 2D to 3D," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 7336–7345.
- [12] J. Gong, Y. Hold-Geoffroy, and J. Lu, "AutoToon: Automatic geometric warping for face cartoon generation," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Mar. 2020, pp. 349–358.
- [13] K. Cao, J. Liao, and L. Yuan, "CariGANs: Unpaired photo-to-caricature translation," 2018, *arXiv:1811.00222*.
- [14] C. Li and M. Wand, "Combining Markov random fields and convolutional neural networks for image synthesis," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 2479–2486.
- [15] A. Selim, M. Elgharib, and L. Doyle, "Painting style transfer for head portraits using convolutional neural networks," *ACM Trans. Graph.*, vol. 35, no. 4, pp. 1–18, Jul. 2016.
- [16] J. Liao, Y. Yao, L. Yuan, G. Hua, and S. B. Kang, "Visual attribute transfer through deep image analogy," 2017, *arXiv:1705.01088*.
- [17] J. Kim, M. Kim, H. Kang, and K. Lee, "U-GAT-IT: Unsupervised generative attentional networks with adaptive layer-instance normalization for image-to-image translation," 2019, *arXiv:1907.10830*.
- [18] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, no. 5786, pp. 504–507, Jul. 2006.
- [19] X. Huang, M.-Y. Liu, S. Belongie, and J. Kautz, "Multimodal unsupervised image-to-image translation," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 172–189.
- [20] J. Huo, W. Li, Y. Shi, Y. Gao, and H. Yin, "WebCaricature: A benchmark for caricature recognition," 2017, *arXiv:1703.03230*.
- [21] Y. Choi, M. Choi, M. Kim, J.-W. Ha, S. Kim, and J. Choo, "StarGAN: Unified generative adversarial networks for multi-domain image-to-image translation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 8789–8797.
- [22] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2242–2251.
- [23] R. Wu, X. Tao, X. Gu, X. Shen, and J. Jia, "Attribute-driven spontaneous motion in unpaired image translation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 5922–5931.
- [24] M.-Y. Liu, T. Breuel, and J. Kautz, "Unsupervised image-to-image translation networks," *Adv. neural Inf. Process. Syst.*, vol. 30, 2017.
- [25] J. Johnson, A. Alahi, and L. Fei-Fei, "Perceptual losses for real-time style transfer and super-resolution," in *Proc. 14th Eur. Conf. Comput. Vis. (ECCV)*, Amsterdam, The Netherlands. Cham, Switzerland: Springer, Oct. 2016, pp. 694–711.
- [26] J. Liao, Y. Yao, L. Yuan, G. Hua, and S. B. Kang, "Visual attribute transfer through deep image analogy," *ACM Trans. Graph.*, vol. 36, no. 4, pp. 1–15, Jul. 2017.
- [27] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial networks," *Commun. ACM*, vol. 63, no. 11, pp. 139–144, 2020.
- [28] T. Karras, S. Laine, and T. Aila, "A style-based generator architecture for generative adversarial networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 4396–4405.
- [29] T. Karras, S. Laine, M. Aittala, J. Hellsten, J. Lehtinen, and T. Aila, "Analyzing and improving the image quality of StyleGAN," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 8107–8116.
- [30] L. Laishram, M. Shaheryar, J. T. Lee, and S. K. Jung, "A style-based caricature generator," in *Proc. Int. Workshop Frontiers Comput. Vis.* Singapore: Springer, 2023, pp. 71–82.
- [31] B. Gooch, E. Reinhard, and A. Gooch, "Human facial illustrations: Creation and psychophysical evaluation," *ACM Trans. Graph.*, vol. 23, no. 1, pp. 27–44, Jan. 2004.
- [32] P.-Y. C. W.-H. Liao and T.-Y. Li, "Automatic caricature generation by analyzing facial features," in *Proc. Asia Conf. Comput. Vis. (ACCV)*, Jeju-do, South Korea, vol. 2, 2004, pp. 694–699.
- [33] J. Liu, Y. Chen, and W. Gao, "Mapping learning in eigenspace for harmonious caricature generation," in *Proc. 14th ACM Int. Conf. Multimedia*, Oct. 2006, pp. 683–686.
- [34] Y. Zhang, W. Dong, C. Ma, X. Mei, K. Li, F. Huang, B.-G. Hu, and O. Deussen, "Data-driven synthesis of cartoon faces using different styles," *IEEE Trans. Image Process.*, vol. 26, no. 1, pp. 464–478, Jan. 2017.
- [35] Z. Zheng, C. Wang, Z. Yu, N. Wang, H. Zheng, and B. Zheng, "Unpaired photo-to-caricature translation on faces in the wild," *Neurocomputing*, vol. 355, pp. 71–81, Aug. 2019.
- [36] W. Li, W. Xiong, H. Liao, J. Huo, Y. Gao, and J. Luo, "CariGAN: Caricature generation through weakly paired adversarial learning," *Neural Netw.*, vol. 132, pp. 66–74, Dec. 2020.
- [37] K. Zhang, W. Luo, L. Ma, W. Ren, and H. Li, "Disentangled feature networks for facial portrait and caricature generation," *IEEE Trans. Multimedia*, vol. 24, pp. 1378–1388, 2022.
- [38] R. Abdal, H.-Y. Lee, P. Zhu, M. Chai, A. Siarohin, P. Wonka, and S. Tulyakov, "3DAvatarGAN: Bridging domains for personalized editable avatars," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2023, pp. 4552–4562.
- [39] L. A. Gatys, A. S. Ecker, M. Bethge, A. Hertzmann, and E. Shechtman, "Controlling perceptual factors in neural style transfer," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 3730–3738.
- [40] L. Gatys, A. S. Ecker, and M. Bethge, "Texture synthesis using convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 28, 2015, pp. 262–270.
- [41] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*.
- [42] L. A. Gatys, A. S. Ecker, and M. Bethge, "Image style transfer using convolutional neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 2414–2423.
- [43] S. Reed, Z. Akata, X. Yan, L. Logeswaran, B. Schiele, and H. Lee, "Generative adversarial text to image synthesis," in *Proc. Int. Conf. Mach. Learn.*, 2016, pp. 1060–1069.
- [44] R. A. Yeh, C. Chen, T. Yian Lim, A. G. Schwing, M. Hasegawa-Johnson, and M. N. Do, "Semantic image inpainting with deep generative models," vol. 2, no. 3, 2016, *arXiv:1607.07539*.
- [45] Y. Choi, Y. Uh, J. Yoo, and J.-W. Ha, "StarGAN v2: Diverse image synthesis for multiple domains," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 8185–8194.
- [46] T. Karras, M. Aittala, J. Hellsten, S. Laine, J. Lehtinen, and T. Aila, "Training generative adversarial networks with limited data," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 33, 2020, pp. 12104–12114.
- [47] A. Melnik, M. Miasayedenkav, D. Makarovets, D. Pirshutuk, E. Akbulut, D. Holzmann, T. Renusch, G. Reichert, and H. Ritter, "Face generation and editing with StyleGAN: A survey," 2022, *arXiv:2212.09102*.
- [48] W. Xia, Y. Zhang, Y. Yang, J.-H. Xue, B. Zhou, and M.-H. Yang, "GAN inversion: A survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 3, pp. 3121–3138, Mar. 2023.
- [49] J. N. M. Pinkney and D. Adler, "Resolution dependent GAN interpolation for controllable image synthesis between domains," 2020, *arXiv:2010.05334*.

- [50] S. Yang, L. Jiang, Z. Liu, and C. C. Loy, "Pastiche master: Exemplar-based high-resolution portrait style transfer," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 7683–7692.
- [51] W. Jang, G. Ju, Y. Jung, J. Yang, X. Tong, and S. Lee, "StyleCariGAN: Caricature generation via StyleGAN feature map modulation," *ACM Trans. Graph.*, vol. 40, no. 4, pp. 1–16, Aug. 2021.
- [52] Z. Liu, P. Luo, X. Wang, and X. Tang, "Deep learning face attributes in the wild," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 3730–3738.
- [53] Dlib. (2022). *C++ Library*. [Online]. Available: <http://dlib.net>
- [54] P. Pérez, M. Gangnet, and A. Blake, "Poisson image editing," in *Proc. ACM SIGGRAPH Papers*, Jul. 2003, pp. 313–318.
- [55] C. Yu, J. Wang, C. Peng, C. Gao, G. Yu, and N. Sang, "Bisenet: Bilateral segmentation network for real-time semantic segmentation," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 325–341.
- [56] V. Gupta and S. Raman, "Automatic trimap generation for image matting," in *Proc. Int. Conf. Signal Inf. Process. (ICONSIP)*, Oct. 2016, pp. 1–5.
- [57] G. Park, S. Son, J. Yoo, S. Kim, and N. Kwak, "MatteFormer: Transformer-based image matting via prior-tokens," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 11686–11696.
- [58] J. Lyu, Z. Wang, and F. Xu, "Portrait eyeglasses and shadow removal by leveraging 3D synthetic data," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 3419–3429.
- [59] S. Zhou, K. Chan, C. Li, and C. C. Loy, "Towards robust blind face restoration with codebook lookup transformer," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 35, 2022, pp. 30599–30611.
- [60] X. Huang and S. Belongie, "Arbitrary style transfer in real-time with adaptive instance normalization," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 1510–1519.
- [61] E. Richardson, Y. Alaluf, O. Patashnik, Y. Nitzan, Y. Azar, S. Shapiro, and D. Cohen-Or, "Encoding in style: A styleGAN encoder for image-to-image translation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 2287–2296.
- [62] O. Tov, Y. Alaluf, Y. Nitzan, O. Patashnik, and D. Cohen-Or, "Designing an encoder for StyleGAN image manipulation," *ACM Trans. Graph.*, vol. 40, no. 4, pp. 1–14, Aug. 2021.
- [63] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 936–944.
- [64] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.
- [65] Y. Alaluf, O. Patashnik, and D. Cohen-Or, "ReStyle: A residual-based StyleGAN encoder via iterative refinement," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 6691–6700.
- [66] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in Neural Information Processing Systems*, vol. 27, Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K. Weinberger, Eds. Red Hook, NY, USA: Curran Associates, 2014.
- [67] L. Mescheder, A. Geiger, and S. Nowozin, "Which training methods for GANs do actually converge?" in *Proc. Int. Conf. Mach. Learn.*, 2018, pp. 3481–3490.
- [68] Y. Nitzan, A. Bermano, Y. Li, and D. Cohen-Or, "Face identity disentanglement via latent space mapping," *ACM Trans. Graph.*, vol. 39, no. 6, pp. 1–14, Nov. 2020.
- [69] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, "The unreasonable effectiveness of deep features as a perceptual metric," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 586–595.
- [70] S. Guan, Y. Tai, B. Ni, F. Zhu, F. Huang, and X. Yang, "Collaborative learning for faster StyleGAN embedding," 2020, *arXiv:2007.01758*.
- [71] J. Deng, J. Guo, N. Xue, and S. Zafeiriou, "ArcFace: Additive angular margin loss for deep face recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 4685–4694.
- [72] Y. Alaluf, O. Tov, R. Mokady, R. Gal, and A. Bermano, "HyperStyle: StyleGAN inversion with HyperNetworks for real image editing," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 18490–18500.



face swapping, face privacy, and surveillance systems.



learning, and generative models.



he has been an Assistant Professor with the School of Computer Science and Engineering, Kyungpook National University, Daegu, South Korea. His research interests include computer vision, video surveillance, and artificial intelligence in medicine.



From 2001 to 2002, he was a Research Associate, and from 2008 to 2009, he was a Visiting Faculty with the IRIS Computer Vision Laboratory, University of Southern California. He is the author of more than 200 articles on computer vision and graphics. He holds more than 20 patents deriving from the research. His research interests include improving the understanding and performance of intelligent vision systems and VR/AR systems, mainly through the application of 3D computer vision, computer graphics, visualization, and HCI. He serves as the Vice President of the Korean Computer Graphics Society, the Korean HCI Society, and the Korean Multimedia Society.

...