

Article

HybridCharacter Generation via Shape Control Using Explicit Facial Features

Jeongin Lee ¹, Jihyeon Yeom ¹, Heekyung Yang ^{2,*} and Kyungha Min ^{1,*}¹ Department of Computer Science, Sangmyung University, 20, Hongjimoong 2 gil, Jongro-gu, Seoul 03016, Republic of Korea;² Division of SW Convergence, Sangmyung University, 20, Hongjimoong 2 gil, Jongro-gu, Seoul 03016, Republic of Korea

* Correspondence: yangkh@smu.ac.kr (H.Y.); minkh@smu.ac.kr (K.M.); Tel.: +82-2-2287-6156 (H.Y.); +82-2-2287-5377 (K.M.)

† These authors contributed equally to this work.

Abstract: We present a hybrid approach for generating a character by independently controlling its shape and texture using an input face and a styled face. To effectively produce the shape of a character, we propose an anthropometry-based approach that defines and extracts 37 explicit facial features. The shape of a character's face is generated by extracting these explicit facial features from both faces and matching their corresponding features, which enables the synthesis of the shape with different poses and scales. We control this shape generation process by manipulating the features of the input and styled faces. For the style of the character, we devise a warping field-based style transfer method using the features of the character's face. This method allows an effective application of style while maintaining the character's shape and minimizing artifacts. Our approach yields visually pleasing results from various combinations of input and styled faces.

Keywords: style transfer; character generation; energy minimization; face features

MSC: 03E72; 68T27; 03T52



Citation: Lee, J.; Yeom, J.; Yang, H.; Min, K. Hybrid Character Generation via Shape Control Using Explicit Facial Features. *Mathematics* **2023**, *11*, 2463. <https://doi.org/10.3390/math11112463>

Academic Editor: Xiaofeng Liu

Received: 21 April 2023

Revised: 17 May 2023

Accepted: 24 May 2023

Published: 26 May 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

In recent years, the increasing demand for various digital content, including webtoons, animations, and games makes it crucial for creators to design diverse characters that accurately reflect their intended purpose. Many creators follow a two-step technique in character creation, which involves selecting a real person's face that closely resembles the desired characters. Creators first choose individuals that fit the character's desired role and personality and then deform the shape of their selected individuals to produce the character's shape according to the creator's style. Afterwards, they apply appropriate colors and shading to the character's shape in order to achieve the desired character's final appearance.

The recent advancements in deep learning and generative models have introduced a stylization technique that applies styles captured from a styled face to a content face [1–3]. However, this approach has a serious limitation regarding manipulating the stylization process as it often produces artifacts. To address this issue, several studies based on StyleGAN [4], a seminal achievement in producing visually pleasing contents, have been proposed for creating characters with a desired style [5–9]. However, these methods have limitations in accurately controlling facial structures. To overcome this limitation, a series of warping field-based studies [10–14] and landmark-based studies [15,16] have been presented, allowing for direct control of facial components. While successful in controlling artifacts, these methods have limitations in producing characters from a pair of faces whose scale and pose do not match.

To generate a character, a two-step approach can be taken: (i) designing the character's shape and (ii) applying the character's style. In digital content such as webtoons, character styles remain consistent, while their shapes vary according to their roles and characteristics. To this end, many creators use the faces of real people whose characters and roles are similar to those of their target characters to determine their shape. However, existing deep learning-based methods, which learn a required style and apply it to a face, have limitations in controlling the shape of a character. Since they learn the style in a texture-based representation, it is challenging to explicitly maintain the morphological features of the input face. An approach that can explicitly control the facial features of a target character is necessary to overcome this limitation.

We propose a novel two-step hybrid approach for generating a character by controlling both the texture and shape of the input face and the styled face. In the first step, we define explicit facial features that quantitatively describe a person's face based on the literature on anthropometry. The features are determined according to the relative shapes, lengths, and positions of facial components, such as the face shape, eyes, nose, mouth, and eyebrows. This approach allows scale-free and pose-independent matching between the features of the input and styled face. We define 37 explicit facial features that can be measured using their corresponding formulas. Based on these explicit features, we generate the shape of the character's face while reflecting both the shape of the input face and the styled face. We design an energy minimization method to control the generated shape. The internal energy maintains a smooth shape, while the external energy controls the degree to which the explicit features of the input and styled faces are reflected. With this approach, we propose an effective and robust method for generating a character's face while appropriately controlling the shape of input and styled faces that differ in scale or pose.

In the second step, a deep-learning-based style transfer model is devised to determine the texture of the character from the texture of the input and styled faces. In this process, the shape of the character's face acts as a guide that controls the texture of the character, such as a warping field. To effectively apply the texture and prevent artifacts, we propose a method that applies the texture while gradually transforming the input face into the character's face. By using content loss and style loss in this process, the texture applied to the final character face can be controlled. Figure 1 illustrates our results. From the input face and styled face, we can generate various characters whose shape and texture interpolates the shapes and textures of the input face and styled face.

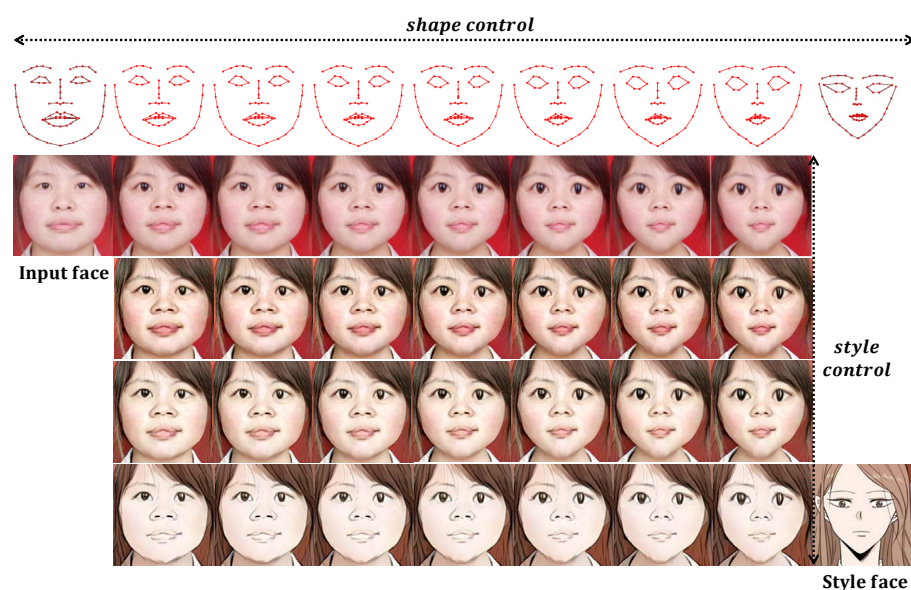


Figure 1. A teaser image of our approach: we can generate characters by controlling shape and style of the input face and the styled face through explicit facial features.

Our approach presents the following contributions:

1. We define a set of explicit facial features that effectively represents facial characteristics. This feature has the advantage of being measured regardless of the pose and size of facial images. The explicit facial features can be utilized not only in our research but also in various anthropometry fields, such as facial aesthetics, forensic science, facial recognition, and facial measurement.
2. We propose a novel method to generate the shape of a character's face by appropriately controlling the shapes of input and styled faces using explicit facial features. This method is designed based on an energy minimization technique that utilizes the input and styled faces to generate a smooth version of the character's face. This method produces the shape of the face, which can serve as an effective guideline for various generative models, including warping-field-based models and diffusion models.
3. We propose a deep-learning-based method that effectively applies texture-based style to a character's face by employing the shape of the face as a guideline. This method is applied step by step to the components of the face, providing the advantage of effective style transfer with artifact control.

This paper is organized as follows. In Section 2, we briefly survey the related works. In Section 3, an overview of our framework is presented. In Section 4, we define explicit facial features. In Sections 5 and 6, we explain how the character's face is produced and stylized, respectively. In Section 7, we present implementation details and results. We evaluate and discuss our results in Section 7. Finally, we conclude this paper and draw a future direction in Section 9.

2. Related Work

This section aims to provide a comprehensive overview of the related literature, which can be broadly categorized into three distinct groups. In Section 2.1, we present an overview of general schemes that employ diverse styles on a portrait. In Section 2.2, we delve deeper into various studies that employ styles on a face using feature maps defined in a latent space. Lastly, we introduce a set of methods that enable the application of styles on a face by leveraging a set of feature points that offer precise quantitative control in Section 2.3.

2.1. General Schemes for Face Style Transfer

In the early days, the studies that applied artistic styles to portrait images were developed based on a CNN-based approach. Gatys et al. [1] extracted the feature maps from the layers of a VGG-19 model and applied them to each channel of a target image. The style texture contained in a styled image was transferred to a target image through the feature map, which was encoded in the Gram matrix. The content of the target image was maintained by minimizing the feature values of the stylized image and the input image. The style was transferred by minimizing the correlation value between the features of the transferred image and the style image. Huang and Belongie [17] replaced the batch normalization scheme employed in Gatys et al.'s work [1] with an adaptive instance normalization (AdaIN) scheme to improve the optimization of the performance of style transfer. They transformed the style textures by aligning the mean and variance of each channel extracted from the target image to match the mean and variance of the styled image.

Generative adversarial networks (GAN) present a new technical background for allowing style transfer by learning styles from unpaired datasets. Zhu et al. [2] presented CycleGAN, which transfers styles between two domains that are not paired datasets. They proposed a cycle consistency loss in order to minimize the difference between the generated images and the images in the target domains. Their discriminator distinguishes the generated images and the target images in both directions in order to improve the performance of CycleGAN. CycleGAN, which is very effective in transferring texture-based styles, has a limitation in applying a style with large morphological deformations. Therefore, CycleGAN cannot properly transfer styles in certain image domains, such as cartoon characters. Kim et al. [3] proposed adaptive layer instance normalization (AdaLIN),

which learns the ratio between instance normalization and layer normalization to address the limitation that only texture-based style is transferred in the existing works. They minimized class activation mapping (CAM) loss, which learns the attention between the real face and the character face domain. Therefore, their model with AdaLIN and the CAM loss successfully deforms the input face according to the shape of the target face. They proved their model by producing character faces used in various animations and cartoons. These GAN-based methods are effective for learning the shape and textures in the image domain instead of an individual image but have a limitation in that they cannot control how the details of a desired example image are transferred to an input image.

2.2. Face Style Transfer Using Feature Map Modulation

Many researchers presented latent space modulation schemes for controlling the shape and texture of style transfer from an input face into a target face.

Pinkney and Adler [5] combined StyleGAN, which was presented by Karras et al. [4], and a style transfer model for a cartoon character's face. PGGAN plays the role of preserving the structure of an input face for generating character's faces. Their model employs the latent vector encoded from the input face as their input for generating a convincing character face. Liu et al. [7] employed a weighted blending module (WBM) to control the structure and texture of the input face and the styled face. Their module synthesizes target faces using a generative model whose input is a weighted average from the input face and styled face. Song et al. [6] proposed a hierarchical variational autoencoder (HVA) to retain features of different scales captured from input faces. A target face is generated by employing the extracted latent code as an input to the attribute-aware generator, which is developed from StyleGAN2. Their model successfully controls facial attributes, including age and gender. Jang et al. [18] proposed StyleCariGAN, which transforms the prominent features of an input face into an exaggerated form using a shape exaggeration block (SEB). They generated a caricature face using a hybrid model of StyleGAN and CaricatureStyleGAN. Their input are the latent vectors extracted from an input face and the exaggerated features, which are extracted using the SEB. These models show their effectiveness in generating visually convincing results. However, they suffer from the problem of collecting a lot of training data for a domain.

Yang et al. [8] proposed DualStyleGAN, which generates a stylized face from a single example style. This model consists of an intrinsic style path (ISP) for learning input facial features and an extrinsic style path (ESP) for learning styled facial features. They synthesized target faces in a stepwise approach by gradually fine-tuning the features learned in both paths. Zhu et al. [9] generated a target face from a single style image by controlling the attributes using a CLIP-based embedding space. The CLIP embedding of an input face and a styled face controls the desired attributes in synthesizing a target face. They extended the generation module from Zhu et al.'s work [19]. These latent-space-based schemes can control the shape of a target face through the encoded latent vector. However, they are limited to controlling the desired shape in a quantitative way.

2.3. Face Style Transfer Using Spatial Interpolation

Many researchers have proposed shape interpolation schemes that generate a target face through direct control. They include schemes that train warping fields or manipulate control points or landmarks.

2.3.1. Warping-Field-Based Schemes

Gong et al. [10] presented a framework that deforms the shape of an input face using a warping field. The warping field is generated from SENet50, a CNN model for face recognition. This warping field that distorts the shape of an input face is trained from the salient facial features. After the input face is distorted, the style of a style example is applied to complete style transfer for an input face.

Gu et al. [11] proposed a multi-exaggeration warper that deforms the shape of an input face in various forms. After training the warper on input faces and styled faces, a multi-exaggeration warper (MEW) deforms an input face to a random shape. The style texture of a styled face is then applied to the deformed face.

Liu et al. [13] proposed a framework that matches the features of an input face and the style example faces in order to implement shape-example-based face style transfer. A warping field trained using the corresponding features between an input face and a style example is applied to deform the shape of an input face. Finally, the renderer trained with the texture of the style example applies the texture from the style example to the deformed input face.

Warping-field-based schemes can control facial shape in detail. However, they have a limitation in the transfer of style to a style example whose shape shows great morphological differences from the input face. They produce many artifacts for faces whose shape is different on a large scale.

2.3.2. Control-Point-Based Schemes

Shi et al. [12] presented a control-point-based scheme that employs a warp controller for the deformation of the input face. They devised a style encoder that extracts texture features from a style example and a content encoder that extracts the shape of an input face. The warp controller deforms the shape of an input face by locating control points and displacements on the face. The extracted text features are applied to the deformed face to complete the style transfer.

Kim et al. [14] presented a scheme that employs the matched control points between an input face and a style example face in order to preserve the shape of both faces simultaneously. They devised a style loss that applies the texture on the styled face and a warp loss that preserves the similarity of the control points between the two faces. These loss terms are minimized to complete the style transfer.

These control-point-based methods depend on the control points. In case the initial control points are improperly selected or the matching between the control points of the faces is incorrect, they tend to produce unwanted results.

2.3.3. Landmark-Based Schemes

Cao et al. [15] presented CariGAN, which is composed of CariGeoGAN for shape transformation and CariStyleGAN for texture generation. CariGeoGAN, which exaggerates the prominent features in a face, produces a caricature from an input face. It exaggerates the features by employing landmarks on both faces. They devised a characteristic loss, which is defined by the difference between the landmarks on both faces. CariGeoGAN was trained to minimize the characteristic loss between an input face and a caricature. CariStyleGAN applies the style texture captured from a caricature to the deformed face. This GAN-based approach shows its effectiveness in processing a series of face images. However, it suffers in transferring a single face image.

Yaniv et al. [16] presented a landmark detector that catches facial landmarks on a style example face. They apply these landmarks to the paired landmarks extracted from an input face. They deform the input face using the pair of landmarks and apply the style texture to the deformed face. This landmark-based scheme synthesizes stylized faces from a single style example image. However, it shows limitation in preserving the identity of the input face.

In addition to the facial features, hand-based features have been studied to produce digital characters [20,21].

3. Overview

In this paper, we present a novel approach for creating a target character using a pair of input images consisting of a real face and a styled face. In our first step, we extract landmarks on both images, which we then use to obtain explicit facial features. These

features allow us to capture various aspects of the face, such as shape, eye, eyebrow, nose, and lip. By comparing the explicit facial features of both images, we generate the landmark of the target face using an energy minimization scheme. We then apply the style of the styled face to the generated landmarks to complete the target character. This entire process is illustrated in Figure 2.

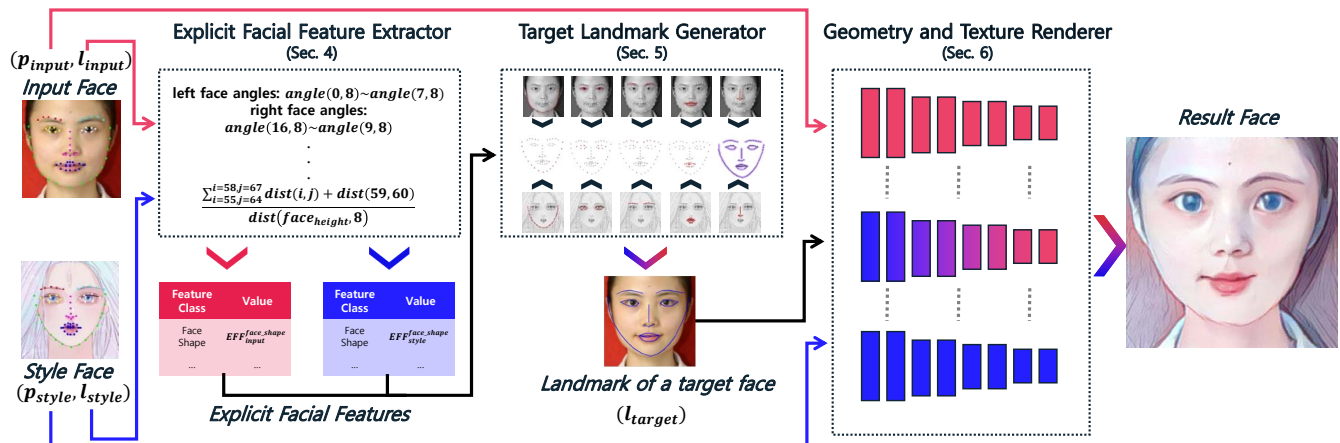


Figure 2. The overview of our approach.

4. Explicit Facial Feature Extractor

In Section 4, we propose a novel method that defines explicit features to enable effective control over facial features. To this end, we provide a comprehensive overview of anthropometric backgrounds pertaining to explicit facial features in Section 4.1. Building upon this, in Section 4.2, we define explicit features for various facial components and derive the corresponding formulas for these features.

4.1. Anthropometry-Based Explicit Facial Features

Anthropometry is the scientific study of the measurements and proportions of the human body, which has been employed in various fields such as aesthetics [22,23], forensics [24,25], and anthropology [26,27]. Farkas [28] used anthropometry to compare human faces, successfully recognizing facial features such as age and gender in a quantitative manner. Similarly, Merler et al. [29] extracted facial features by defining the anthropometrical expression of the quantitative face coding system.

In this research, we utilize anthropometry to measure facial features and generate the geometry of a character by considering the geometries of both the input face and the styled face. Our method explicitly categorizes facial features into components such as face shape, eyes, eyebrows, lips, and nose. Furthermore, these features can be categorized based on their properties such as location, length, and shape. Through this approach, we aim to improve the accuracy and precision of facial feature recognition for various applications.

4.2. Defining Explicit Facial Features

4.2.1. Preparation

In order to define explicit facial features, we first performed frontalization and face alignment on the input face. To achieve frontalization, we utilized the rotate-and-render scheme [30] and extracted 68 facial landmark points from the frontalized input face using the method proposed by King et al. [31]. The set of landmarks on a frontalized face is illustrated in Figure 3. However, King et al.'s approach has a limitation in extracting landmarks from styled faces that are not rendered in a photorealistic style. To overcome this limitation, we adopted Yaniv et al.'s method [16] for landmark extraction from styled faces rendered in various artistic styles. We then define 37 features based on the 68 landmark points estimated on the face. The definition of these features is shown in Figure 4.

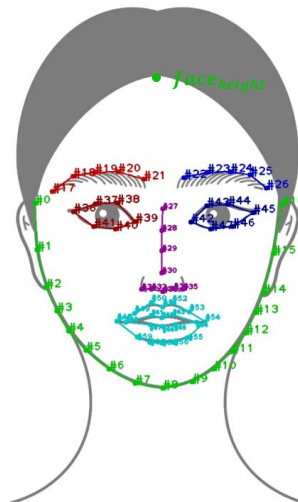


Figure 3. The landmarks of our approach.

4.2.2. Definition of Face Shape Features

The explicit features of face shape are defined as the angles of the landmark points on the face's contour. Sixteen landmark points on the face contour are identified, and the angle between the vector and the horizontal line is measured for each point. Landmark points #0 to #7 are located on the left contour, while points #9 to #15 are on the right contour. The resulting angles are then averaged to generate an overall measurement of face shape.

4.2.3. Definition of Eyebrow Features

We have defined nine explicit features for the eyebrows (#17 to #25). Feature #17 specifies the position of the eyebrow, which is measured as the relative distance between the eyebrow and the eye divided by the face height. Feature #18 defines the distance between the eyebrows, while feature #19 specifies the length of the eyebrow, divided by the face width. Finally, features #20 to #25 describe the shape of the eyebrow and are defined as the angles of the eyebrow line.

4.2.4. Definition of Eye Features

We have defined five explicit features for the eyes (#26 to #30). Feature #26 specifies the distance between the eye and the ends of the forehead. Features #27 and #28 measure the shape of the eye and are defined as the ratio of the height of the upper and lower eyelids to the eye width. Feature #29 describes the distance between the eyes, which is defined as the ratio of the width between the glabella landmarks and the face width. Feature #30 measures the length of the eye.

4.2.5. Definition of Nose Features

We have defined three explicit features for the nose (#31 to #33). Feature #31 specifies the position of the nose and is measured as the ratio of the height of the philtrum to the height of the face. Features #32 and #33 measure the length and height of the nose, respectively. Feature #32 is defined as the ratio of the width of the nose to the width of the face, while feature #33 is simply the height of the nose.

4.2.6. Definition of Lip Features

We have defined four explicit features for the lips (#34 to #37). Feature #34 specifies the position of the lips and is measured as the ratio of the height of the chin to the height of the face. Feature #35 defines the length of the lips and is measured as the ratio of the width of the lips to the width of the face. Finally, features #36 and #37 define the height of the upper and lower lips, respectively.

Fig.	1	2	3	4	5	6	7
visual							
category	feature	Fig.	formula		reference		
face	face angles	1	#1~#8 left face angles: $angle(0,8) \sim angle(7,8)$		#9~#16 right face angles: $angle(16,8) \sim angle(9,8)$ [Sunhem2016]		
eye-brow	distance between eyebrows and eyes	2	#17 $\frac{dist(19,37)+dist(24,44)}{dist(face_{height},8)}$		[Merler2019], [Xie2015]		
	distance between eyebrows		#18 $\frac{dist(21,22)}{dist(0,16)}$		[Xie2015]		
	eyebrows length		#19 $\frac{dist(21,17)+dist(22,26)}{dist(0,16)}$		-		
	eyebrow angles	3	left eyebrow angles: #20 $angle(17,18,18,19)$ #21 $angle(18,19,19,20)$ #22 $angle(19,20,20,21)$	right eyebrow angles: #23 $angle(22,23,23,24)$ #24 $angle(23,24,24,25)$ #25 $angle(24,25,25,26)$	-		
eye	distance between eyes and ends of forehead	4	#26 $\frac{dist(face_{height},38)}{dist(face_{height},8)} + \frac{dist(face_{height},43)}{dist(face_{height},8)}$		[Merler2019], [Alsawwaf2022], [Hong2022], [Ramanathan2006], [Alzahran2021], [Xie2015], [Zheng2022], [Porter2001], [Farkas2005]		
	eye upper curvature height		#27 $\frac{\sum_{i=37}^{38} dist(i,36,39) + \sum_{i=43}^{44} dist(i,42,45)}{dist(face_{height},8)}$		[Merler2019], [Verma2021], [Porter2001], [Farkas2005]		
	eye lower curvature height		#28 $\frac{\sum_{i=40}^{41} dist(i,36,39) + \sum_{i=46}^{47} dist(i,42,45)}{dist(face_{height},8)}$		[Merler2019], [Xie2015], [Verma2021], [Kukharev2020], [Alrubaih2020], [Alzahran2021], [Chen2021], [Porter2001]		
	distance between eyes	5	#29 $\frac{dist(42,39)}{dist(0,16)}$		[Merler2019]		
	eye length		#30 $\frac{dist(39,36) + dist(42,45)}{dist(0,16)}$		-		
	nose	distance between nose and lip	6	#31 $\frac{dist(33,51)}{dist(face_{height},8)}$		[Farkas1994], [Merler2019], [Xie2015]	
nose length		#32 $\frac{dist(31,35)}{dist(0,16)}$		[Sunhem2016]			
nose height		#33 $\frac{\sum_{i=27}^{29} dist(30,i)}{dist(face_{height},8)}$		[Merler2019], [Xie2015]			
lip	distance between lip and chin	7	#34 $\frac{dist(57,8)}{dist(face_{height},8)}$		[Xie2015]		
	lip length		#35 $\frac{dist(48,54) + dist(60,64)}{dist(0,16)}$		-		
	upper lip height		#36 $\frac{\sum_{i=49,j=60}^{i=53,j=64} dist(i,j)}{dist(face_{height},8)}$		-		
	lower lip height		#37 $\frac{\sum_{i=55,j=64}^{i=58,j=67} dist(i,j)+dist(59,60)}{dist(face_{height},8)}$		[Merler2019], [Alsawwaf2022], [Hong2022], [Ramanathan2006], [Alzahran2021], [Xie2015], [Zheng2022], [Porter2001], [Farkas2005]		

Figure 4. Definition of the explicit facial features.

5. Target Landmark Generator

Our target landmark generator produces the landmark of a target character by accommodating the explicit facial features of the input face and the styled face. Accommodating the landmarks from two misaligned faces may cause unexpected results. Therefore, we build the target landmark from the explicit facial features of the faces instead of their landmarks. The process of target landmark generation is illustrated in Figure 5. The target landmark, which is initiated as the landmarks of the input face, progressively reflects the styled face. This process is implemented using an energy minimization scheme [32]. The total energy of a target landmark is defined as the sum of the external energy term that constrains the target landmark but should accommodate the landmarks of both faces and the internal energy term that preserves the smoothness of the target landmark. We formulate the total energy L_{target} as follows:

$$L_{target} = L_{target}^{ext} + L_{target}^{int}, \quad (1)$$

where L_{target}^{ext} and L_{target}^{int} are the external energy and the internal energy of the target landmark, respectively.

The explicit facial features of a target character are defined as a linear combination of the features from both faces:

$$EFF_{target} = \alpha EFF_{style} + (1 - \alpha) EFF_{input}, \quad (2)$$

where α is assigned as 0.5.

L_{target}^{ext} is minimized when the explicit facial features extracted from the target landmarks match EFF_{target} . However, L_{target}^{ext} does not guarantee to produce smooth landmarks. Therefore, the smoothness of the produced landmarks is constrained from the internal energy, L_{target}^{int} , which is defined as the sum of the curvature of the line segments connecting the landmark points,

$$L_{target}^{int} = \sum_i \kappa_i, \quad (3)$$

where κ_i is defined as the finite difference among the 1-ring neighborhood points of the i -th point.

The target landmark, which is initialized as the landmark of the input face, is progressively deformed to minimize the total energy, L_{target} .

In addition to the progressive minimization of the energy term, we also devise a sequence of minimization for the facial components. Since the facial components are dependent on other components, the deformation of a component may affect the other component. For example, the explicit facial features of the eyes are defined by considering the face width. Therefore, deforming the eye landmarks before the face shape landmarks may cause incorrect results if the landmarks on face shape are deformed afterward. Therefore, we devise an order of application of the components for applying the energy minimization process. At each minimization step, the landmarks for face shape are determined first. Then, the landmarks for eye and lip are determined. Finally, the landmarks for eyebrows and noses are determined.

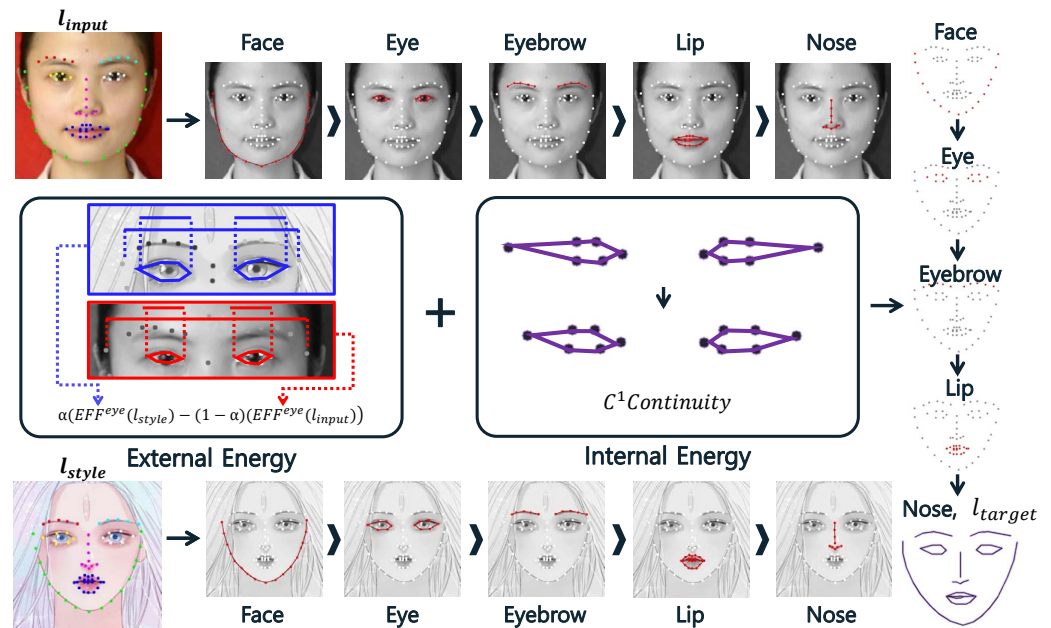


Figure 5. Generation of landmarks of the target character's face.

6. Stylizer with Landmarks

We generate character faces in two steps: (i) we generate the shape of a character (P_{warp}) by warping the input face using the character's landmarks, and (ii) we complete the character ($P_{warp \rightarrow style}$) by applying the style extracted from the styled face to P_{warp} . This process of stylization is illustrated in Figure 6.

In the previous section, we generated l_{target} , which is the landmark of a target character, from the landmarks extracted from the input face and the styled face. We generate a warping field from the landmarks l_{target} and l_{input} . This warping field is applied to the input face, which is gradually deformed to the target character's shape (P_{warp}) [14]. This scheme, however, that warps the shape of a target's shape simultaneously has limitations in that the warped shape may not follow the warping field or that artifacts may appear in the results. To resolve this limitation, we propose a component-wise warping that applies warping fields. The warping field is applied according to the face components, including the face shape, eyes, eyebrows, nose, and lips.

The target character is completed by applying the style from the styled face to P_{warp} . For this, we apply the VGG-16 model as a feature extractor to the input face (P_{input}), styled face (P_{style}), and the shape of the target (P_{warp}) and extract the features f_{input} , f_{style} , and f_{warp} , respectively. Using these features, we produce $P_{input \rightarrow style}$ by applying f_{style} to P_{input} and $P_{warp \rightarrow style}$ by applying f_{style} to P_{warp} . This stylization process iterates until the loss terms measuring the differences of content and style are minimized. We define four loss terms, as follows:

$$L_{content}^{input} = w_{input} \left(\frac{1}{n^2} \sum \left| \frac{Dist_{ij}^{P_{input \rightarrow style}}}{\sum_i Dist_{ij}^{P_{input \rightarrow style}}} - \frac{Dist_{ij}^{P_{input}}}{\sum_i Dist_{ij}^{P_{input}}} \right| \right) \quad (4)$$

$$\begin{aligned} L_{style}^{input} &= REMD(P_{input \rightarrow style}, P_{style}) \\ &+ \frac{1}{w_{input}} REMD(Color(P_{input \rightarrow style}), Color(P_{style})) \\ &+ MM(P_{input \rightarrow style}, P_{style}) \end{aligned} \quad (5)$$

$$L_{\text{content}}^{\text{warp}} = w_{\text{warp}} \left(\frac{1}{n^2} \sum \left| \frac{\text{Dist}_{ij}^{P_{\text{warp} \rightarrow \text{style}}}}{\sum_i \text{Dist}_{ij}^{P_{\text{warp} \rightarrow \text{style}}}} - \frac{\text{Dist}_{ij}^{P_{\text{warp}}}}{\sum_i \text{Dist}_{ij}^{P_{\text{warp}}}} \right| \right) \quad (6)$$

$$\begin{aligned} L_{\text{style}}^{\text{warp}} &= \text{REMD}(P_{\text{warp} \rightarrow \text{style}}, P_{\text{style}}) \\ &+ \frac{1}{w_{\text{warp}}} \text{REMD}(\text{Color}(P_{\text{warp} \rightarrow \text{style}}), \text{Color}(P_{\text{style}})) \\ &+ \text{MM}(P_{\text{warp} \rightarrow \text{style}}, P_{\text{style}}) \end{aligned} \quad (7)$$

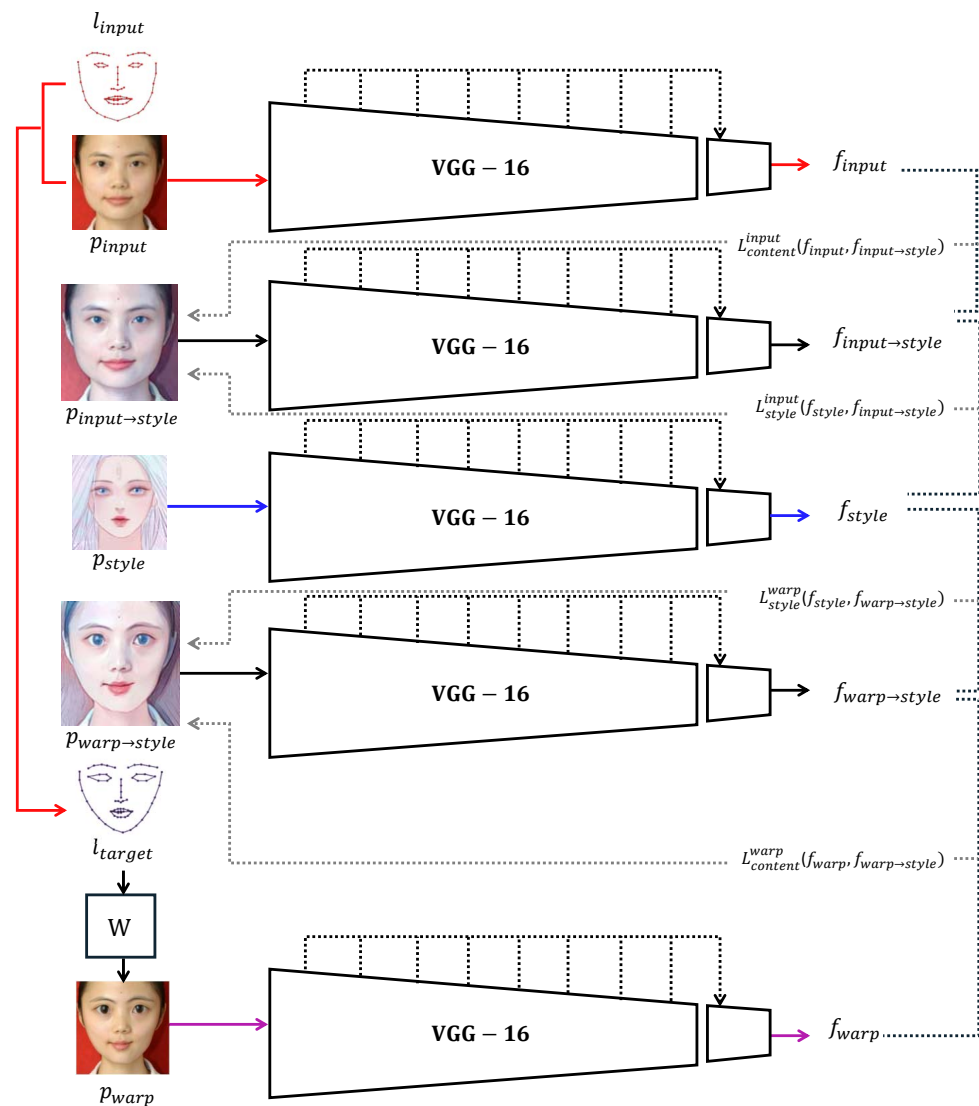


Figure 6. Synthesis of target character's face.

We assign 0.5 for w_{input} and w_{warp} . Dist_{ij}^P denotes the pairwise cosine distance between all the feature vectors extracted from P . In the above formula, $\text{REMD}(P_{a \rightarrow b}, P_b)$, which represents the relaxed earth mover's distance [33], and $\text{MM}(P_a, P_b)$, which denotes moment matching, are defined as follows:

$$\text{REMD}(P_{a \rightarrow b}, P_b) = \max \left(\frac{1}{n} \sum_i \min_j \text{dist}(P_{a \rightarrow b_i}, P_{b_j}), \frac{1}{m} \sum_j \min_i \text{dist}(P_{a \rightarrow b_i}, P_{b_j}) \right). \quad (8)$$

$$MM(P_a, P_b) = \frac{1}{d} \|\mu_{a \rightarrow b} - \mu_b\|_1 + \frac{1}{d^2} \|\Sigma_{a \rightarrow b} - \Sigma_b\|_1, \quad (9)$$

where μ is mean and Σ is the covariance. $dist(P_a, P_b)$ is defined as the Euclidean distance between the color vectors P_a and P_b .

We utilized Kim et al.'s model [14] as the backbone network for the texture-based stylization of the characters. To extract texture features, we used the VGG-16 model and set W_{input} to 0.3. To minimize the loss function, root-mean-square propagation (RMSProp) was employed with 250 iterations and a learning rate of 0.001. On average, our model takes 3 min to generate an image of size 512×512 . We trained our model using 10,000 input faces and 500 styled faces. We discuss the procedures of preparing the input and styled faces in Section 7.2 and present the results in Figures 7 and 8.

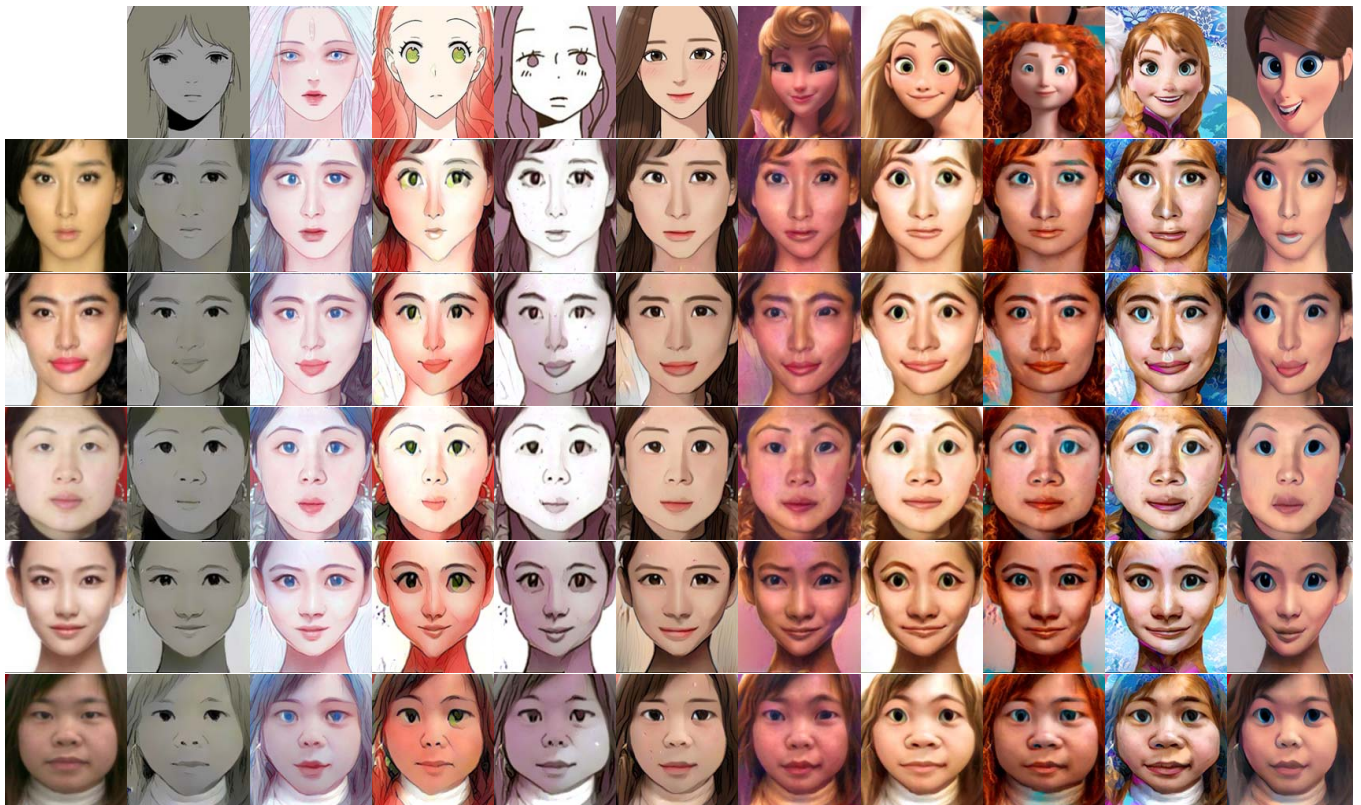


Figure 7. The result of our approach (1).

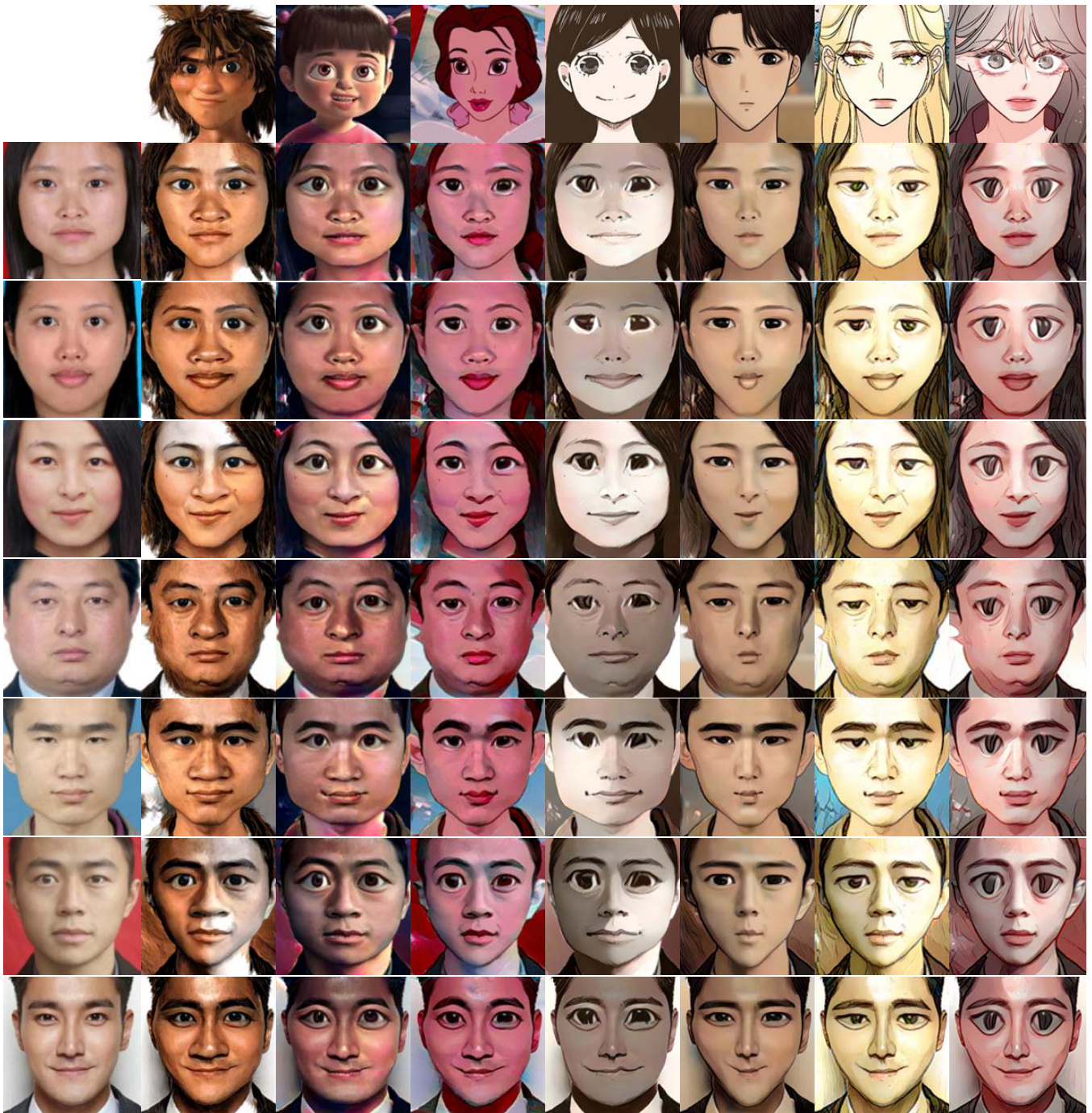


Figure 8. The result of our approach (2).

7. Implementation and Results

Section 7 provides a summary of the implementation environment, the dataset employed in this study, and our results. In Section 7.1, we provide the computational environment used for implementing our method. In Section 7.2, we describe the dataset used for training our model and the corresponding preprocessing procedures. Finally, in Section 7.3, we present our results, which demonstrate the effectiveness of our approach in generating digital characters.

7.1. Implementation Environment

We utilized Kim et al.'s model [14] as a backbone network for character generation. Our model was implemented using the Pytorch library and was trained on the Google

Cloud environment, which employs an Intel Xeon 2.2 GHz CPU and an Nvidia Tesla T4 GPU.

7.2. Training Data

The input dataset for this study consisted of two different sets of facial images: SCUT-FBP5500 [34], containing 5500 images of front-facing individuals, and AFAD [35], consisting of 160,000 images of Asian faces. Extracting landmark-based features from the faces in a consistent way requires the faces in the images to be aligned. To achieve this alignment, we utilized the frontalization model [30] for the faces in the dataset. As preparation for extracting explicit facial features, we used the dlib [31] library to generate 68 landmarks from the faces. For the style dataset, we collected 1370 images of front-facing cartoon characters from various webtoon sites and used 317 cartoon images from Pinkney et al.'s work [5]. To generate landmarks from the webtoon faces, we used the landmark detector from Yaniv et al.'s work [16]. We further refined the landmarks manually to ensure accuracy.

7.3. Results

In Figure 7, we demonstrate the application of ten styled faces to five input faces. Five of the styled faces were sampled from webtoons, while the remaining five were from 3D animations. Both sets of styled faces underwent appropriate transformations to reflect the intended appearance of the creator, resulting in deformations in facial features, such as the eyes, nose, mouth, and overall face shape. Furthermore, these two groups of styled faces exhibited differences in skin color and shading, thus showcasing diverse textures. Our method generated characters that reflect the appropriate shape and texture of the styled face while preserving the shape of the input face and accounting for variations in the shapes and textures of the styled images.

In Figure 8, we present the additional results of character generation by applying seven styled faces to seven input faces. The styled faces include a classical animation face, and four male faces were included to the input faces to explore diverse character generation. Figures 7 and 8 show the characters whose shape reflects the shapes from the input face and styled face and whose texture comes from the styled face. We execute a further analysis on the results in Section 8.

8. Comparison and Analysis

We provide Figures 9–12 for qualitative comparison. While Gatys et al.'s work [1] applies only textures for stylization, it is not suitable for styles with low amounts of texture and strict shapes, such as webtoons and cartoons. Kolkin et al. [33] applied textures with spatial consideration to reduce artifacts but did not perform geometric transformations. Yaniv et al.'s work [16] applied the style transfer scheme from [1] after warping the faces with style landmarks. To compare the style transfer results under the same conditions of shape transformation, we employed Kolkin et al.'s work [33]. Yaniv et al.'s work [16] produces artifacts when faces are imprecisely detected or when they undergo significant shape transformations. Kim et al.'s work [14] progressively warps the face using style landmarks before applying the style transfer scheme from [33], which reduces artifacts due to progressive warping. However, when there are small differences in face angles, such as the styled faces in the 6th and 10th columns of Figures 9 and 10, artifacts occur around the face shape or mouth. Our method is less sensitive to styled face angles as it uses landmarks determined by explicit facial features rather than style landmarks. Kim et al. [14] optimized all 68 landmarks simultaneously, making it impossible to select which face part to transform and by how much. Our method allows for adjusting the transformation degree by α for each facial part, as shown in Figure 13.

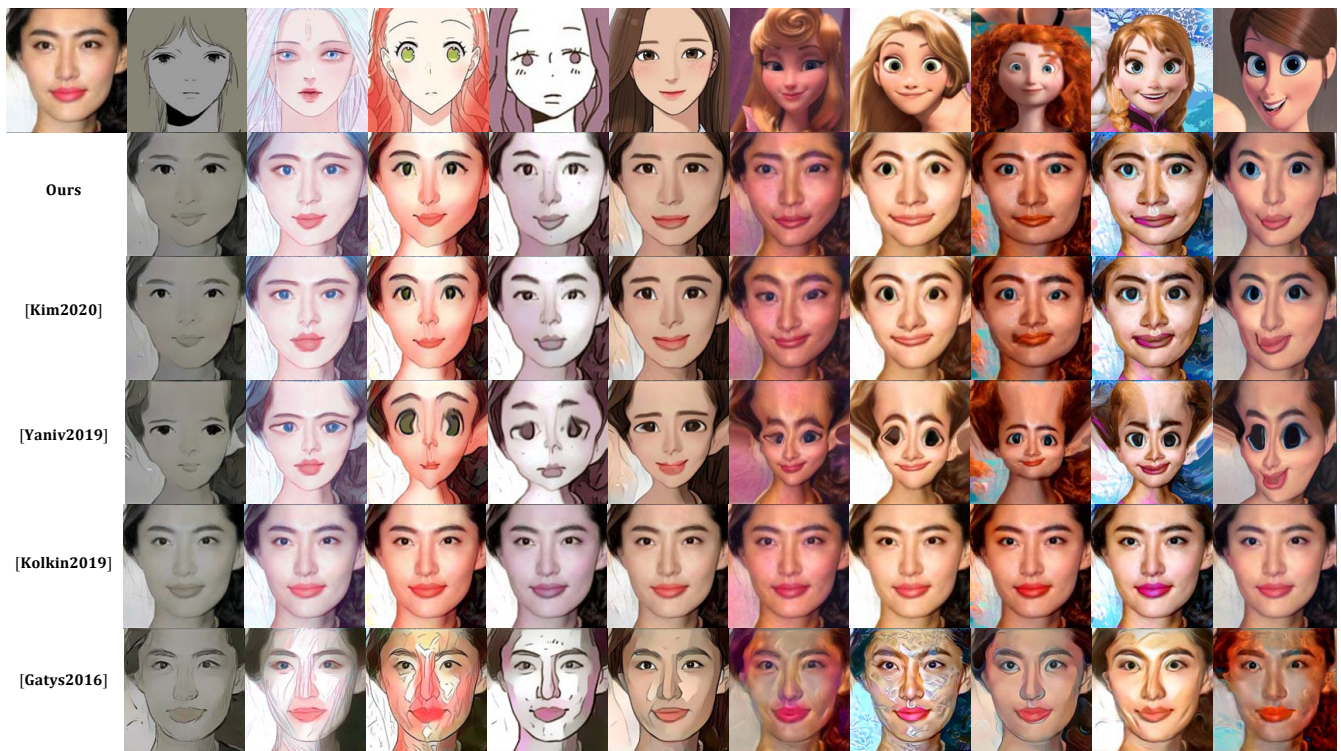


Figure 9. A comparison of our results with others (1).

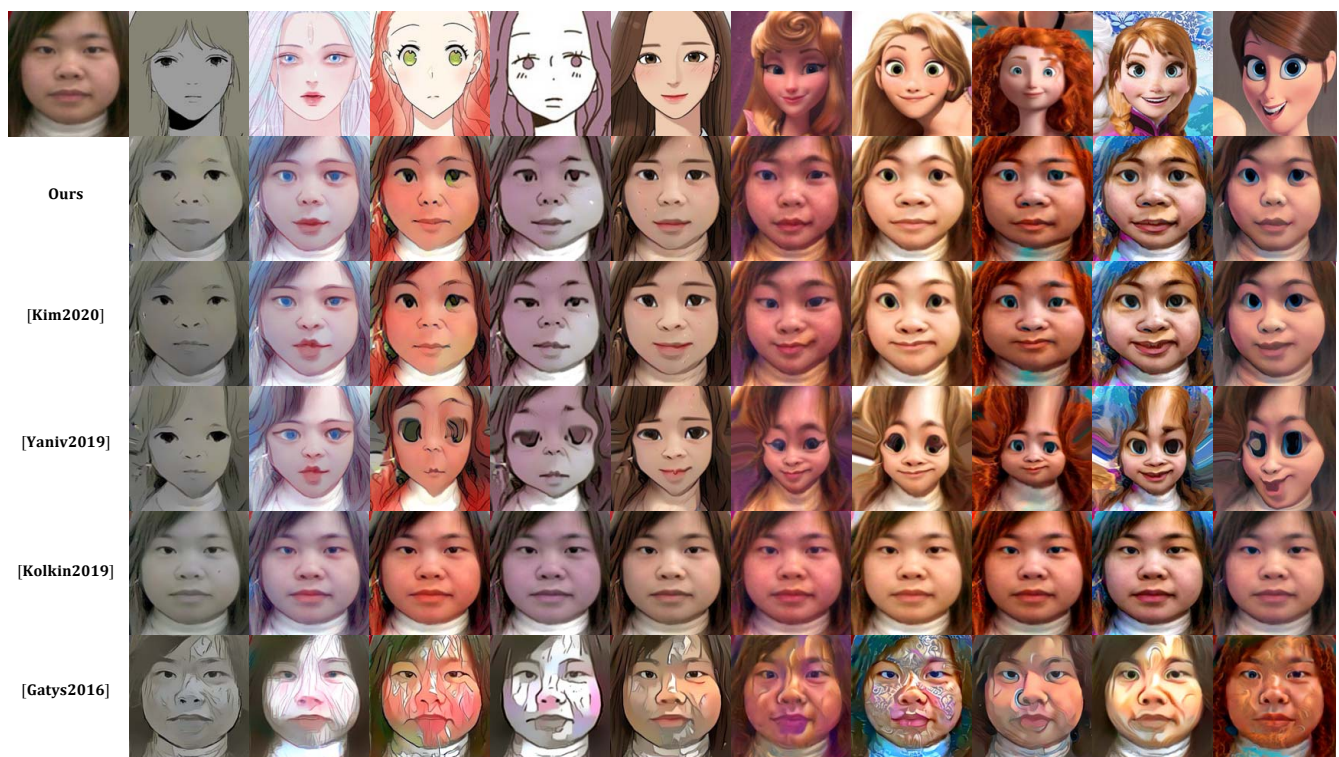


Figure 10. A comparison of our results with others (2).

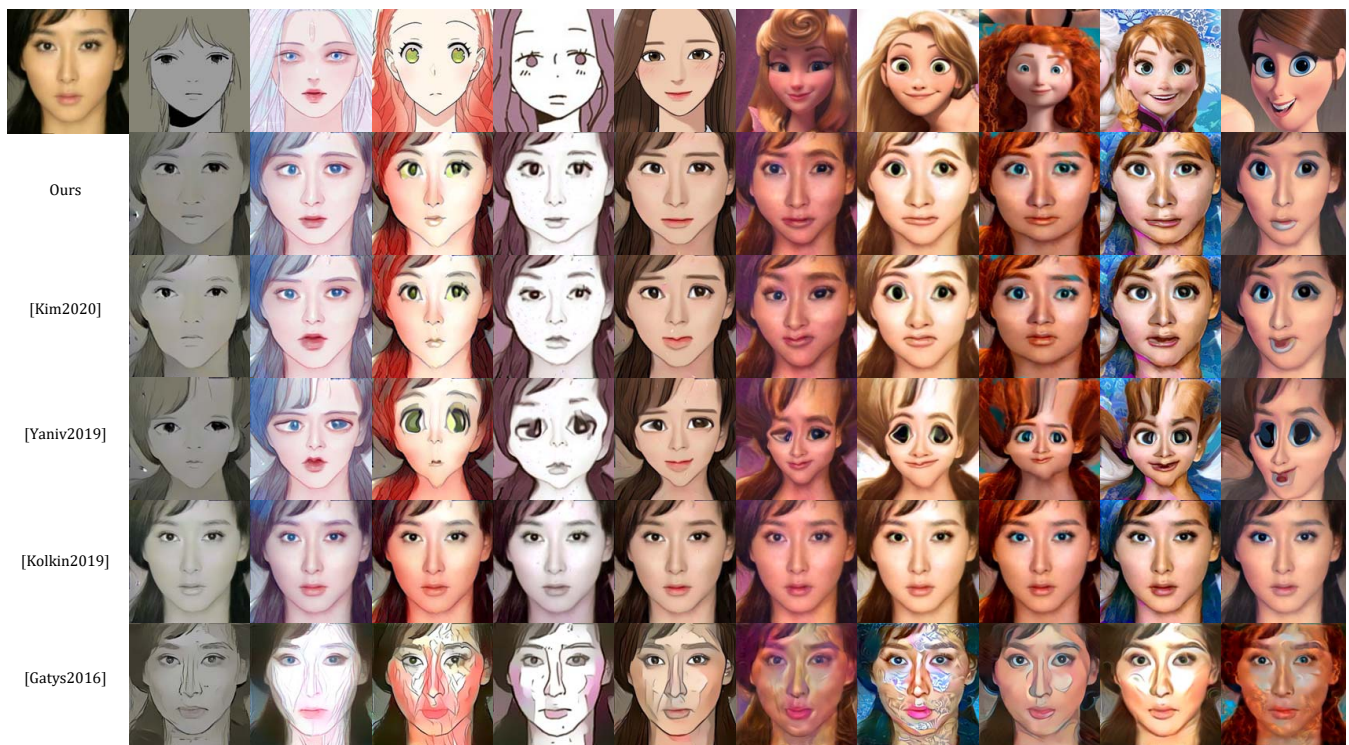


Figure 11. A comparison of our results with others (3).

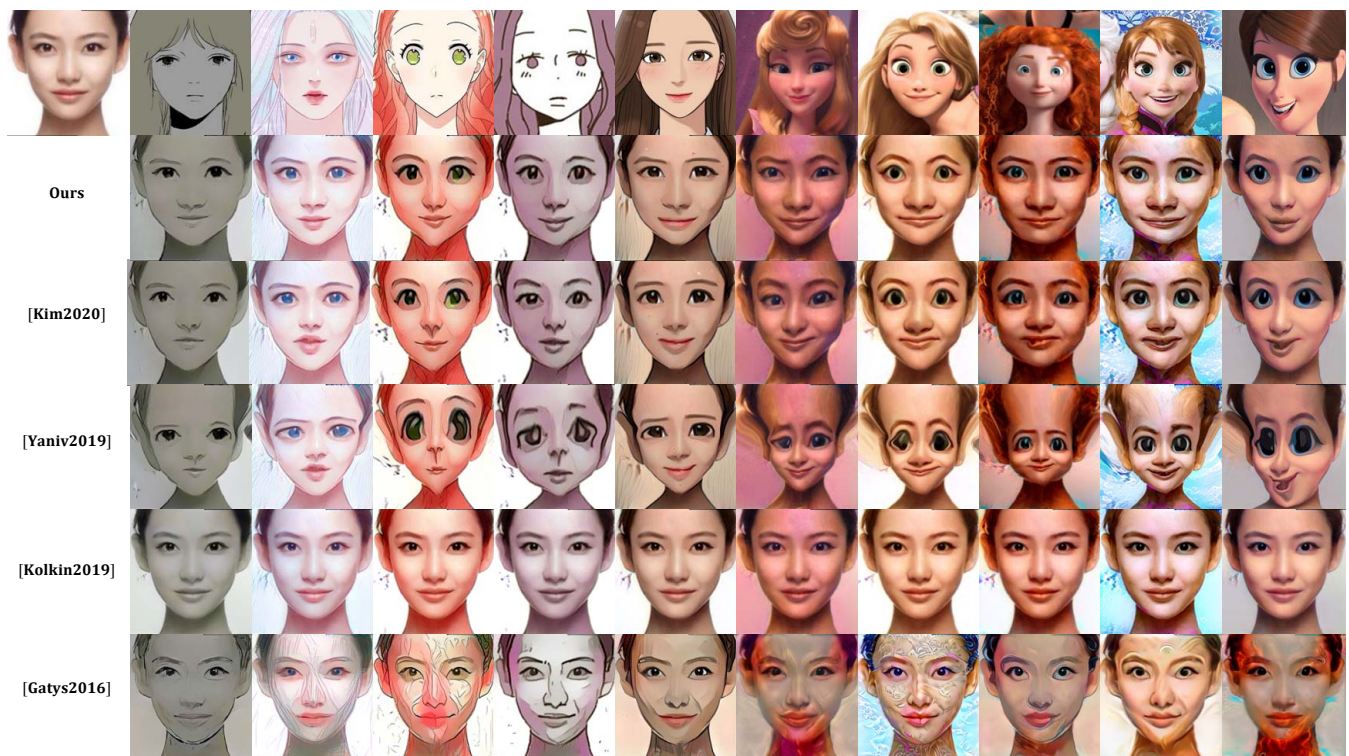


Figure 12. A comparison of our results with others (4).

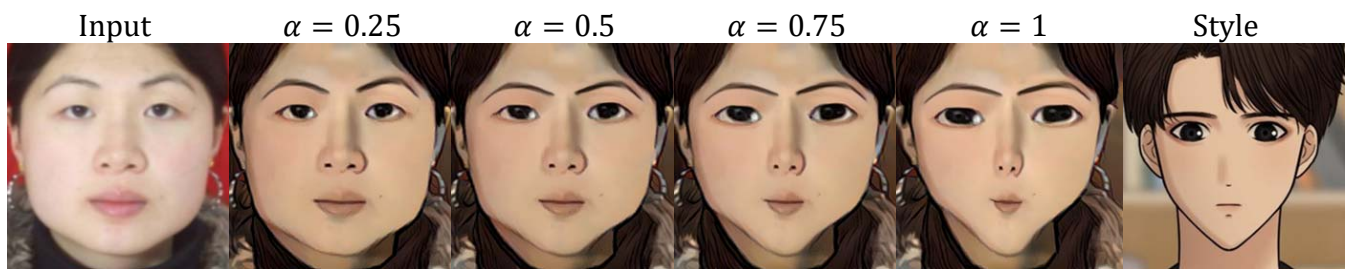


Figure 13. Generating interpolated target character faces by controlling α .

8.1. Evaluation

For the evaluation of our approach in a quantitative manner, we employed two metrics: FID and KID.

8.1.1. Evaluation Using FID

In this section, we used the Frechet inception distance (FID) to quantitatively compare our results with those of selected existing studies [1,14,16,33]. The FID measures the similarity between the distributions of two comparing image sets. Table 1 compares three FID scores measured from 25 faces generated using the existing studies; inputFID is measured by comparing the generated faces and the input faces, and styleFID is measured by comparing the generated faces and the styled faces. The meanFID score is calculated by averaging inputFID and styleFID. The formula for the FIDs are defined as follows:

$$InputFID = |\mu_{input} - \mu_{result}|^2 + Tr(C_{input} + C_{result} - 2C_{input}C_{result})^{1/2} \quad (10)$$

$$StyleFID = |\mu_{style} - \mu_{result}|^2 + Tr(C_{style} + C_{result} - 2C_{style}C_{result})^{1/2} \quad (11)$$

$$MeanFID = \frac{InputFID + StyleFID}{2} \quad (12)$$

Table 1. Comparison of three FID values for five methods, including ours. Red figure denotes the minimum value, and blue figure denotes the second minimum value.

Method	InputFID	StyleFID	MeanFID
ours	176.41	221.94	194.79
Kim et al.'s [14]	184.2	223.84	198.93
Yaniv et al.'s [16]	234.71	189.50	212.11
Kolkin et al.'s [33]	83.96	261.27	172.61
Gatys et al.'s [1]	249.36	250.63	249.99

8.1.2. Evaluation Using KID

When the distribution of a sample is not a normal distribution, it can be difficult to rely on FID as a reliable evaluation metric, since statistical metrics such as mean and variance are limited in their utility. Therefore, we employed kernel inception distance (KID) for additional quantitative evaluation. To calculate KID, we extracted two sample data, x and x' , from both the input image set and the generated image set and calculated the value using a polynomial kernel function K . We also calculated the value by applying K to sample data x extracted from the input image set and sample data x' extracted from the generated image set.

We used this method to calculate values for the input face data and styled face data, which we define as InputKID and StyleKID, respectively. MeanKID is calculated as the average of InputKID and StyleKID. This process is defined in the following equation:

$$\begin{aligned} \text{InputKID} = & E_{x,x' \sim \text{input}}[K(I_x, I_{x'})] + E_{x,x' \sim \text{result}}[K(I_x, I_{x'})] \\ & - 2E_{x \sim \text{input}, x' \sim \text{result}}[K(I_x, I_{x'})] \end{aligned} \quad (13)$$

$$\begin{aligned} \text{StyleKID} = & E_{x,x' \sim \text{style}}[K(I_x, I_{x'})] + E_{x,x' \sim \text{result}}[K(I_x, I_{x'})] \\ & - 2E_{x \sim \text{style}, x' \sim \text{result}}[K(I_x, I_{x'})] \end{aligned} \quad (14)$$

$$\text{MeanKID} = \frac{\text{InputKID} + \text{StyleKID}}{2} \quad (15)$$

Table 2 compares three KID scores measured from 25 faces generated using the existing studies. The figures in Table 2 show similar patterns to those in Table 1.

Table 2. Comparison of three KID values for five methods, including ours. Red figure denotes the minimum value, and blue figure denotes the second minimum value.

Method	InputKID	StyleKID	MeanKID
Ours	0.42	0.44	0.43
Kim et al.'s [14]	0.43	0.45	0.44
Yaniv et al.'s [16]	0.52	0.41	0.46
Kolkin et al.'s [33]	0.25	0.59	0.42
Gatys et al.'s [1]	0.55	0.46	0.50

8.1.3. Discussion

The InputFID and InputKID values obtained from the characters generated using Kolkin et al.'s method [33] exhibit the lowest values, since this method does not significantly alter the shape of the input face. In comparison, our method achieves the lowest InputFID and InputKID values when compared to methods that deform the shape of the input face [14,16], indicating that our approach preserves the features of the input face.

Yaniv et al.'s method [16] exhibits the lowest StyleFID and StyleKID values, as it generates characters with facial landmarks that match those of the styled face. In contrast, our results utilize explicit facial features to reflect both the input and style characteristics, resulting in higher values than Yaniv et al.'s method [16] but lower values than other methods [14,33]. This indicates that the faces generated by our method reflect the shape of the style.

Our study aims to maintain both the shape and texture of the input and styled faces, making MeanFID and MeanKID values important when evaluating the results. When comparing MeanFID and MeanKID values, our approach achieves superior results compared to methods that alter the shape of the input face [14,16].

8.2. User Study

We executed a user study to assess the superiority of the character faces generated by our research over the faces generated by the existing studies [1,14,16,33]. The goal of our study is to generate a target character face that reflects both the geometric and textural features of the styled face. In this user study, we evaluated whether the generated character face accurately reflects the geometric and textural features of the styled face. We prepared five input faces and ten styled faces for the user tests. Applying our method and the methods used in the four studies [1,14,16,33], we generated a total of 250 results, with 50 for each method. Therefore, we present 50 rows consisting of the input face, the styled face, and the faces generated from our scheme and the existing studies (shown in an example row).

For the user study, we recruited 50 participants, including 24 males and 26 females, who were mostly in their 20s. Overall, 39 of them were undergraduates, and 11 were

graduate students. Participants were presented with 20 randomly selected rows from the prepared 50 rows and were asked three questions for each row:

- Q1. Please select the image that reflects the shape of both the input and styled faces.
- Q2. Please select the image that is most similar to the styled face in terms of textural aspects.
- Q3. Please select the image that reflects both the shape of the input and styled faces and is most similar to the styled face in terms of textural aspects.

From the scores of the user study, where 50 participants participated, we can conclude that our scheme outperforms the comparative studies to a great degree. For the three questions, our results show higher scores than those from the compared studies. The results of the user study are presented in Table 3. Among the questions, the answers from Q1, which asks about the resemblance of the shape, shows the lowest score compared to other questions. We can assess that the results from [14] resemble both input and styled faces. However, our scheme shows better scores than Kim et al.'s work [14] since we can resolve the artifacts observed in the results from [14]. Our scheme that employs explicit facial features, which allows scale-independent shape control, can resolve the unwanted artifacts between the faces whose scales are significantly different.

Table 3. The results of user study.

Method	Q1	Q2	Q3
Ours	30	39	41
Kim et al.'s [14]	18	9	8
Yaniv et al.'s [16]	0	1	0
Kolkin et al.'s [33]	2	1	1
Gatys et al.'s [1]	0	0	0

8.3. Ablation Study

As explained in Section 5, we propose a method for generating facial landmarks using an energy optimization technique, where energy is defined as the sum of internal and external energies. The external energy is utilized to control explicit facial features, while the internal energy is used to maintain the smoothness of facial component shapes. We defined the internal energy for each facial component separately. Figure 14 illustrates the comparison between considering internal energy for all facial components and removing the internal energy for each component separately. The results generated by removing the internal energy show that, in most cases, the transformed landmarks are not evenly distributed or smooth. Furthermore, the shape of the landmark is not maintained. For instance, removing the internal energy for the nose resulted in some landmark points overlapping.

As explained in Section 6, we propose a method for transforming facial shape using component-wise warping where different energy functions are defined for each facial component. In contrast, other landmark-based face transformation methods, such as [14], optimize an identical equation for 68 landmarks. Figure 14 compares the optimization results of these methods applied at a fixed number of iterations. The circular points represent the input facial landmarks, while the square points represent the target landmarks, and the asterisks represent the optimized points. The conventional methods show limitations when optimizing the landmarks for the face components with greater differences, such as face shape or eyes, due to the landmarks for the components with smaller differences. In contrast, our component-wise optimization method optimizes each component independently, which leads to the effective optimization of all landmarks.

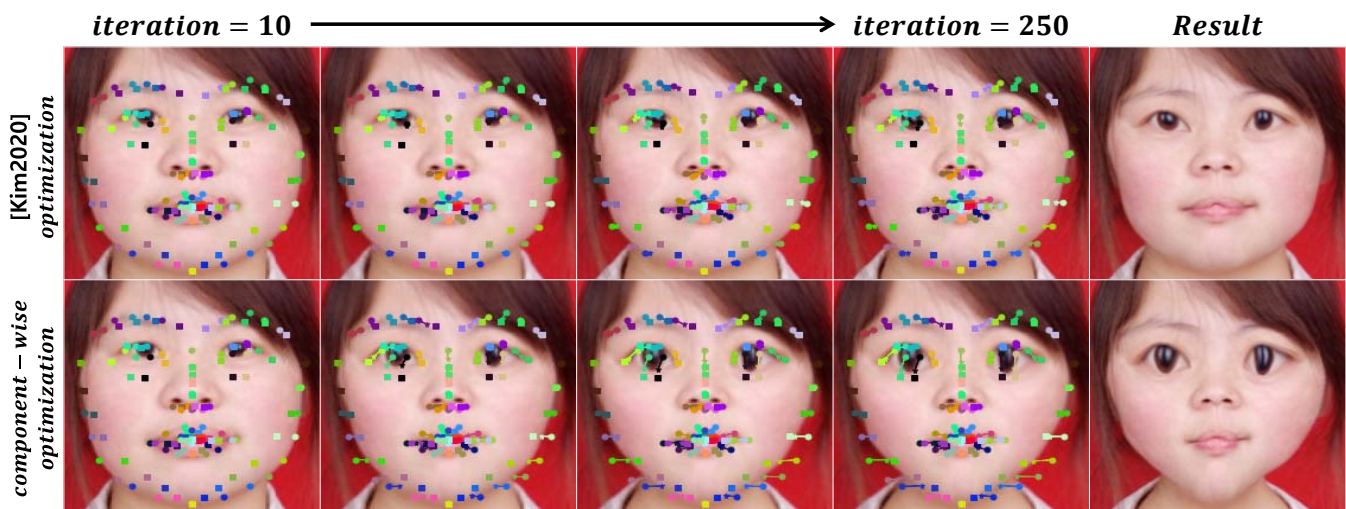


Figure 14. Ablation study comparing the optimization processes.

8.4. Applications

In very recent years, image generation technology based on the diffusion model has come to support various control methods. By applying the input face image to a diffusion model [36] in order to generate various results, the images in the first row of Figure 15 are produced. The faces in this row shows faces of various identities while maintaining the features of the input face image. In the diffusion model [36], a text guide called a prompt was used to generate a face whose features reflect the prompt. The images generated with the attributes such as “small lips”, “big eyes”, and “sharp face shape” are presented in the second row of Figure 15. These prompts are given to the model with the input face used in the first row. Furthermore, we synthesized a face with the above attributes and applied the face as an input to the diffusion model; the resulting images are presented in the third row of Figure 15. When comparing the results in the second and third rows, the sharp face shape attribute is not reflected in the second row’s results, and unwanted deformations occur in features such as the eyebrows or pupils. Therefore, stable and precise shape control is most effectively achieved when the user employs an input image that has been deformed into the desired shape, as shown in this study. This comparison demonstrates that our scheme can be extended to facial generation research that requires explainable deformation.

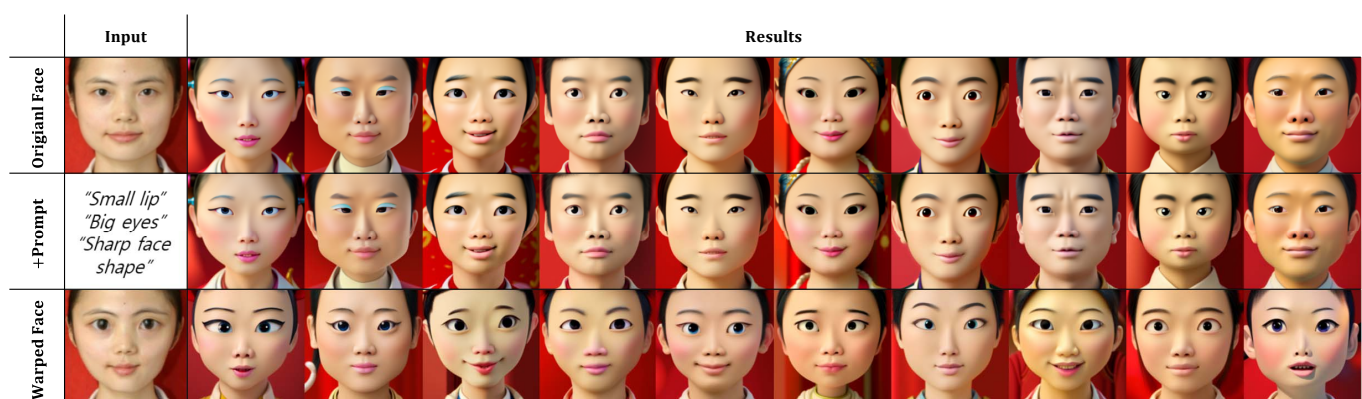


Figure 15. Application using diffusion model. In the top row, the input face is employed to control the generated faces. In the second row, prompts including “small lips”, “big eyes”, and “sharp face shape” are employed to control the shape of the generated face. In the bottom row, the face generated by our approach is employed to control the generated faces that obey both principles. As shown, the bottom row shows the most effective result images.

8.5. Limitations

Our method has three limitations related to the use of landmark-based features and optimization algorithm. Firstly, faces with extreme poses are not feasible for our framework since our framework relies on stable and reliable facial landmark extraction. Extracting reliable landmarks from highly rotated faces is challenging, which results in a limitation in extracting and matching explicit facial features for such faces. A similar limitation is observed for applying styles of a styled face whose components are not visible. Figure 16 reveals this limitation. For the styled faces whose foreheads are obscured by their hair, the produced character's forehead shows unexpected artifacts. Secondly, accessories such as glasses and skin color cannot be detected by landmarks, so additional modules are needed to reflect them in the results. Thirdly, using an optimization-based approach for a single example style makes it difficult to produce real-time results. To achieve a higher response time for real-time applications, our character generation method needs to be improved.

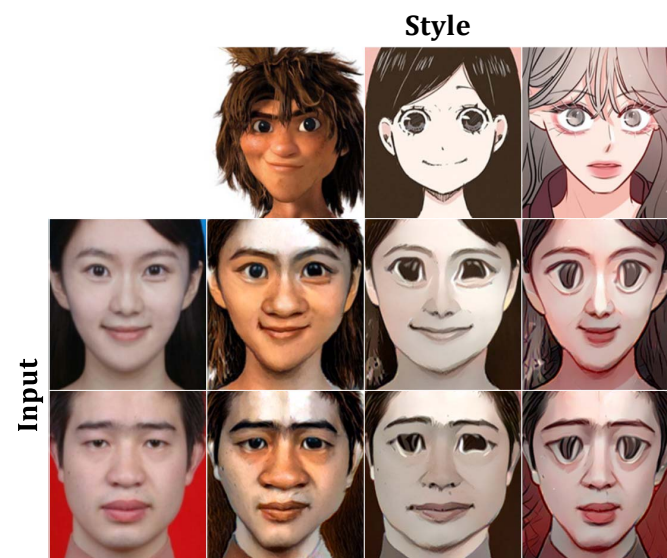


Figure 16. An example of the limitations of our approach. The forehead of styled faces, which is obscured by hair, raises unwanted artifacts in the forehead of the produced characters.

9. Conclusions and Future Work

This paper presents a novel method for generating character faces that incorporates explicit features from both the input and styled faces. To achieve this, we developed a series of formulas based on existing studies on anatomical features of the face and used it to extract explicit facial features from the landmarks obtained from the input and styled faces. We then utilized the generated explicit features to produce the shape of the target character's face and applied the texture of the styled face to complete the target character's face. To account for the varying importance of individual facial components, we adapted the optimization function to optimize for each face component during shape manipulation.

Our future work involves developing a model that can generate character faces by controlling the shape of facial components, allowing us to create faces that reflect the shape of the styled face while preserving the identity-defining features of the input face, such as a square jaw or small eyes. Additionally, we are exploring the use of image guides as a replacement for prompt guides, similar to Rombach et al.'s work [36], to enable real-time style transformation in conjunction with our model, potentially leading to performance improvements.

Author Contributions: Methodology, J.L. and J.Y.; Writing—original draft, H.Y.; Writing—review & editing, K.M. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by Sangmyung University at 2021.

Data Availability Statement: Our data is not available.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Gatys, L.A.; Ecker, A.S.; Bethge, M. Image style transfer using convolutional neural networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 2414–2423.
2. Zhu, J.Y.; Park, T.; Isola, P.; Efros, A. Unpaired image-to-image translation using cycle-consistent adversarial networks. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017; pp. 2223–2232.
3. Kim, J.; Kim, M.; Kang, H.; Lee, K. U-gat-it: Unsupervised generative attentional networks with adaptive layer-instance normalization for image-to-image translation. In Proceedings of the 8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, 26–30 April 2020; pp. 26–30.
4. Karras, T.; Laine, S.; Aila, T. A style-based generator architecture for generative adversarial networks. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 16–17 June 2019; pp. 4401–4410.
5. Pinkney, J.N.; Adler, D. Resolution dependent GAN interpolation for controllable image synthesis between domains. *arXiv* **2020**, arXiv:2010.05334.
6. Song, G.; Luo, L.; Liu, J.; Ma, W.C.; Lai, C.; Zheng, C.; Cham, T.J. AgileGAN: stylizing portraits by inversion-consistent transfer learning. *ACM Trans. Graph.* **2021**, *40*, 117. [\[CrossRef\]](#)
7. Liu, M.; Li, Q.; Qin, Z.; Zhang, G.; Wan, P.; Zheng, W. BlendGAN: Implicitly GAN blending for arbitrary stylized face generation. In Proceedings of the NeurIPS 2021, on-line conference, 6–14 December 2021; pp. 29710–29722.
8. Yang, S.; Jiang, L.; Liu, Z.; Loy, C.C. Pastiche Master: Exemplar-based high-resolution portrait style transfer. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), New Orleans, LA, USA, 18–24 June 2022; pp. 7693–7702.
9. Zhu, P.; Abdal, R.; Femiani, J.; Wonka, P. Mind the gap: Domain gap control for single shot domain adaptation for generative adversarial networks. *arXiv* **2021**, arXiv:2110.08398.
10. Gong, J.; H.-G.; Y.; Lu, J. Autotoon: Automatic geometric warping for face cartoon generation. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV), Snowmass Village, CO, USA, 1–5 March 2020; pp. 360–369.
11. Gu, Z.; Dong, C.; Huo, J.; Li, W.; Gao, Y. CariMe: Unpaired caricature generation with multiple exaggerations. *IEEE Trans. Multimed.* **2021**, *24*, 2673–2686. [\[CrossRef\]](#)
12. Shi, Y.; Deb, D.; Jain, A.K. WarpGAN: Automatic caricature generation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019; pp. 10762–10771.
13. Liu, X.C.; Yang, Y.L.; Hall, P. Learning to warp for style transfer. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Nashville, TN, USA, 20–25 June 2021; pp. 3702–3711.
14. Kim, S.S.; Kolkin, N.; Salavon, J.; Shakhnarovich, G. Deformable style transfer. In Proceedings of the European Conference on Computer Vision, Glasgow, UK, 23–28 August 2020; pp. 246–261.
15. Cao, K.; Liao, J.; Yuan, L. CariGANs: Unpaired photo-to-caricature translation. *ACM Trans. Graph.* **2018**, *37*, 244. [\[CrossRef\]](#)
16. Yaniv, J.; Newman, Y.; Shamir, A. The face of art: landmark detection and geometric style in portraits. *ACM Trans. Graph.* **2019**, *38*, 60. [\[CrossRef\]](#)
17. Huang, X.; Belongie, S. Arbitrary style transfer in real-time with adaptive instance normalization. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017; pp. 1501–1510.
18. Jang, W.; Ju, G.; Jung, Y.; Yang, J.; Tong, X.; Lee, S. StyleCariGAN: caricature generation via StyleGAN feature map modulation. *ACM Trans. Graph.* **2021**, *40*, 116. [\[CrossRef\]](#)
19. Zhu, J.; Shen, Y.; Zhao, D.; Zhou, B. In-domain GAN inversion for real image editing. In Proceedings of the European Conference on Computer Vision, Glasgow, UK, 23–28 August 2020; pp. 592–608.
20. Rungruanganukul, M.; Siriborvornratanakul, T. Deep learning based gesture classification for hand physical therapy interactive program. In Proceedings of the International Conference on Human-Computer Interaction, Copenhagen, Denmark, 19–24 July 2020; pp. 349–358.
21. Miah, A.; Musa, S.; Mehedi, Hasan, A.; Shin, J. Dynamic Hand Gesture Recognition using Multi-Branch Attention Based Graph and General Deep Learning Model. *IEEE Access* **2013**, *11*, 4703–4716. [\[CrossRef\]](#)
22. Xie, D.; Liang, L.; Jin, L.; Xu, J.; Li, M. Scut-fbp: A benchmark dataset for facial beauty perception. In Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics, Hong Kong, China, 9–12 October 2015; pp. 1821–1826.
23. Wei, W.; Ho, E.S.L.; McCay, K.D.; Damaševičius, R.; Maskeliūnas, R.; Esposito, A. Assessing facial symmetry and attractiveness using augmented reality. *Pattern Anal. Appl.* **2022**, *25*, 635–651. [\[CrossRef\]](#)
24. Moreton, R. Forensic Face Matching. In *Forensic Face Matching: Research and Practice*; Oxford Academic: Oxford, UK, 2021; p. 144.

25. Sezgin, N.; Karadayi, B. Sex estimation from biometric face photos for forensic purposes. *Med. Sci. Law* **2022**, *63*, 105–113. [[CrossRef](#)] [[PubMed](#)]
26. Porter, J.P.; Olson, K.L. Anthropometric facial analysis of the African American woman. *Arch. Facial Plast. Surg.* **2001**, *3*, 191–197. [[CrossRef](#)] [[PubMed](#)]
27. Maalman, R.S.-E.; Abaidoo, C.S.; Tetteh, J.; Darko, N.D.; Atuahene, O.O.-D.; Appiah, A.K.; Diby, T. Anthropometric study of facial morphology in two tribes of the upper west region of Ghana. *Int. J. Anat. Res.* **2017**, *5*, 4129–4135. [[CrossRef](#)]
28. Farkas, L. *Anthropometry of the Head and Face*; Raven Press: New York, NY, USA, 1994.
29. Merler, M.; Ratha, N.; Feris, R.S.; Smith, J.R. Diversity in faces. *arXiv* **2019**, arXiv:1901.10436.
30. Zhou, H.; Liu, J.; Liu, Z.; Liu, Y.; Wang, X. Rotate-and-render: Unsupervised photorealistic face rotation from single-view images. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 13–19 June 2020; pp. 5911–5920.
31. King, D.E. Dlib-ml: A machine learning toolkit. *J. Mach. Learn. Res.* **2009**, *10*, 1755–1758.
32. Kass, M.; Witkin, A.; Terzopoulos, D. Snakes: Active contour models. *Int. J. Comput. Vis.* **1988**, *1*, 321–331. [[CrossRef](#)]
33. Kolkin, N.; Salavon, J.; Shakhnarovich, G. Style transfer by relaxed optimal transport and self-similarity. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019; pp. 10051–10060.
34. Liang, L.; Lin, L.; Jin, L.; Xie, D.; Li, M. SCUT-FBP5500: A diverse benchmark dataset for multi-paradigm facial beauty prediction. In Proceedings of the IEEE 24th International conference on pattern recognition (ICPR), Beijing, China, 20–24 August 2018; pp. 1598–1603.
35. Niu, Z.; Zhou, M.; Wang, L.; Gao, X.; Hua, G. Ordinal regression with multiple output cnn for age estimation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 4920–4928.
36. Rombach, R.; Blattmann, A.; Lorenz, D.; Esser, P.; Ommer, B. High-resolution image synthesis with latent diffusion models. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), New Orleans, LA, USA, 18–24 June 2022; pp. 10684–10695.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.