Title	スケッチによるアートスタイルの描画支援に関する研究
Author(s)	黄, 正宇
Citation	
Issue Date	2023-03
Type	Thesis or Dissertation
Text version	ETD
URL	http://hdl.handle.net/10119/18422
Rights	
Description	Supervisor: 宮田 一乗, 先端科学技術研究科, 博士

Doctoral Dissertation

STUDY ON SKETCH-BASED ART STYLE DRAWING ASSISTANCE

HUANG, Zhengyu

Supervisor: Professor Kazunori Miyata

Graduate School of Advanced Science and Technology
Japan Advanced Institute of Science and Technology
Information Science

March 2023

Abstract

Drawing has creative, expressive, and educational value. It remains fundamental to translate and analyze the world. However, traditional drawing requires sophisticated skills. For general users, it is not easy to access professional drawing skills due to lack of artistic training, which is time-consuming and labor-intensive. Nowadays, with the development of Artificial Intelligence (AI), those artistic drawing styles have been achieved by Non-Photorealistic Rendering (NPR) or Neural Style Transfer (NSF) techniques from images. However, recent studies have shown that the drawing process predicted by AI is definitely different from a human's behavior – there's still a long way to go to make AI understand the drawing and support users for artistic creation following their expectation.

The final goal of this dissertation is to let AI understand users' freehand rough sketches and provide suitable guidance to support users' art creativity interactively and extend users' drawing ability. As applications, this dissertation is dedicated to supporting the creation of artistic portraits for both realistic style and anime style. In order to achieve this goal, the major research question is how to translate the features extracted by machine learning or deep learning into a user-recognizable form that can be used to converse with users. From a mathematical perspective, this goal is essentially to utilize features extracted from AI to help the user explore the optimal solution in mind in the process of creating a new artistic drawing. If a user's response to the guidance given by the AI is regarded as a user-perception evaluation function, then the greatest problem in this dissertation is that the function is dynamically varying and non-differentiable, even with individual differences. How to maximize this user-perception function which only exists in one's mind with AI is the major research question in this dissertation.

To address this research problem, this paper proposes a User-AI cooperation paradigm which considers the user as a black-box part of the whole drawing assistance system and interactively approximates the above user-perception evaluation function by constructing an overall optimization function with a certain prior knowledge of this system. With this paradigm, the AI obtains more valuable input information, and the user's drawing ability is extended, making it a win-win situation for both the AI and the user through conversation. Depending on whether the extracted features are directly visualized as user feedback for conversation, the strategies for constructing the overall optimization function can be divided into two types: explicit strategy and implicit strategy.

The various works in this dissertation are centered on this paradigm, which can be summarized in the following three parts.

(1) Data preparation.

As there is no off-the-shelf sketch-art database available for deep learning yet so far, I proposed a sketch-Art pair generation framework based on style transfer for realistic style and anime-style artistic portraits. In particular, for line drawing generation in anime style, a one-shot line drawing style transfer approach from color illustrations is proposed to solve the limited data problem. Note that this one-shot framework is a prior-knowledge-based style transfer, which is derived from a feedback-free version of the above paradigm.

(2) AI-assisted Drawing with Explicit Conversation Strategy.

To achieve realistic style drawing assistance, "dualFace" was proposed, which decomposes the overall system optimization function into nested functions and designs a two-stage drawing assistance scheme - the AI offers sketch contour guidance in the global stage while providing detailed guidance in the local stage. To allow sketches to be converted to other recognizable input for realistic portrait style transfer with intermediate real as prior knowledge, a low-level feature-matching algorithm is proposed which converts rough sketches to semantics masks for real-style artistic portrait generation automatically and connects these two stages. Since the guidance given by both stages of dualFace relies heavily on the prior knowledge of real human faces as an intermediate, the method fails when one's drawing style differs significantly from real faces, such as an anime face. Therefore, this dissertation also designs the anime face drawing assistance system using implicit strategy.

(3) AI-assisted Drawing with Implicit Conversation Strategy.

Unlike the two-stage explicit strategy, this part proposes an implicit optimization function for the end-to-end sketch-guidance style transfer. An unsupervised stroke-level disentanglement training strategy for prior knowledge in StyleGAN is proposed so that rough sketches with sparse strokes can automatically match the corresponding local facial parts in anime portraits respectively. What's more, to analyze the correspondence between strokes and semantics in portraits for smooth conversation with users in anime style, a one-shot semantics-level matching framework is proposed in the final interactive drawing assistance system.

Besides the success of each part in the above, the validity of the User-AI cooperation paradigm is demonstrated by analyzing and discussing the relationship between system evaluation with objective metrics and user evaluations with user studies for art portrait drawing assistance of both realistic style and anime style in the final drawing assistance.

Keywords: Generative Adversarial Networks (GAN), GAN inversion, sketch comprehension, User-AI cooperation, sketch-based art creation.

Acknowledgment

I would like to thank the many people who supported me during the implementation of this research. My primary supervisor, Professor Kazunori Miyata, gave me many opinions from various angles and warmly watched over me throughout the research. I would like to express my deepest gratitude to him.

Then, I would like to express my sincere gratitude to Dr. Haoran Xie for his constant concern for the progress of my research and for his guidance and advice on the direction of my research. I would like to thank him from the bottom of my heart.

I would like to thank Professor Atsuo Yoshitaka and Professor Hideaki Kanai for being the associate supervisor of my major research and the on-campus supervisor of my minor research, respectively. I also thank Professor Shogo Okada, Professor Kazunori Kotani and Professor Yoshihiro Kanamori, for their valuable comments and suggestions on this dissertation.

Outside of JAIST, I would like to thank Professor Tsukasa Fukusato of the University of Tokyo for his professional guidance and valuable advice on sketch-based 3D modeling in my minor research as well as those studies about the human-computer interface. Also, I'd like to express my deep gratitude to him.

I would like to thank our parents for their care and support in my life so that I can devote myself to my research life.

Finally, I would like to thank everyone in and outside the MiyataLab for their generous cooperation and advice on experiments and research.

List of Abbreviations

AdaIN	Adaptive Instance Normalization
AI	Artificial Intelligence
BiSeNet	Bilateral Segmentation Network
CNN	Convolutional Neural Network
FDoG	Flow-based Difference of Gaussians
GALIF	Gabor Local Line-based Feature
GAN	Generative Adversarial Networks
HCML	Human-Centered Machine Learning
NPR	Non-Photorealistic Rendering
NSF	Neural Style Transfer
S2I	Sketch to Image
SBIR	Sketch-based Image Retrieval
UEL	User-perception Evaluation Loss
XDoG	Extended Difference of Gaussians

List of Figures

List of Tables

Contents

Chapter 1

Introduction

> "A painting is not thought out in advance. While it is being done, it changes as one's thoughts change. And when it's finished, it goes on changing, according to the state of mind of whoever is looking at it ." [2] - Pablo Picasso

1.1 Motivation

Drawing is an ancient medium of expression in the history of human civilization, which originates from life and records the development of human thought. As an important form of visual art, drawing has strong reproduction and realism, which is the artist's expression of his/her own emotions and thoughts. However, no matter how abstract a drawing is, it is based on a certain level of drawing skills. Satisfying this condition requires a lot of time and effort to practice painting repeatedly, which has become the main barrier for untrained amateur users who want to express themselves properly through drawing.

With the development of human-computer interaction technology, the utilization of artificial intelligence and deep learning to assist and expand user capabilities has become an important research topic which is also known as Human-Centered Machine Learning (HCML) [3]. Many research efforts have been made to apply AI to assist users in various tasks. An application of AI background music generation for short online videos was developed by Frid et al. [4], which allows video creators to interactively regenerate and mix AI-generated music based on the songs fed into the AI engine. In the same research area, Louie et al. [5], investigated how AI music generation tools can be tuned to minimize user burden through user experimentation. Balasubramanian et al. [6] developed an assistive AI tool for visually impaired people to master nonverbal cues and conducted user experiments to understand their perspectives. Kacorri et al. [7] also explore a mobile assistive tool with few-shot learning to help blind people extract customized user-defined information about which they are interested and concerned from the surrounding environment about visual objects using deep learning techniques. Similarly, Feiz et al. [8] used AI technologies

to develop an AI system Write-it-Yourself guide(WiYG) which makes it possible for blind people to fill out printed forms independently. Lee et al. [9] proposed an improved object recognition-based deep learning system for people with visual impairments by locating their hands as prior evidence to consider when focusing on an object in a frame. Zhao et al. [10] developed a face recognition application "Accessibility Bot" for visually impaired people, extracting identity and facial expressions information from their friends via their smartphone camera. As to drawing assistance, Drawing Apprentice is developed by Davis et al. [11] an intelligent drawing system which can improvise and collaborate on abstract sketches.

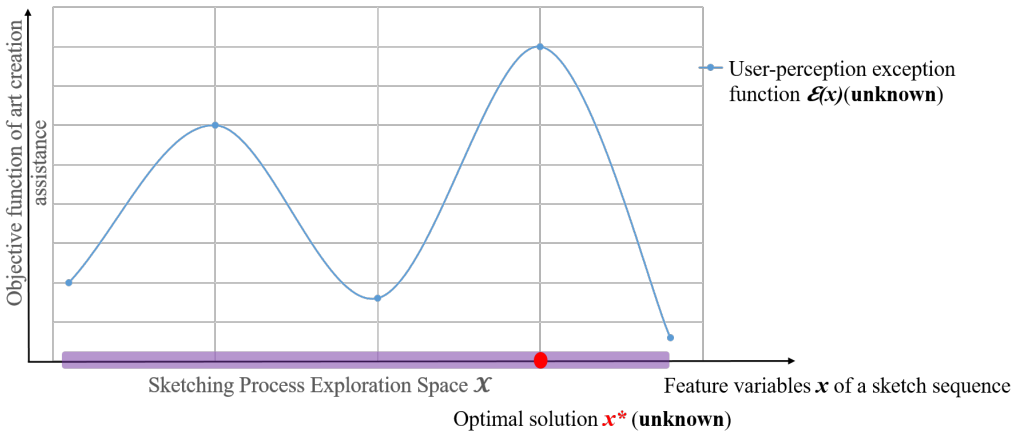
However, as far as the current research is concerned, the methods that enable providing support or guidance during the sketching process are almost always based on image retrieval techniques, which greatly limits the quality of the drawn objects and the creativity of the user. On the other hand, the input for sketch-based image generation studies is often a complete hand-drawn sketch or a pseudo sketch generated by traditional edge detection methods such as Canny, Sobel, etc. Those S2I technique is difficult to be used for user drawing process assistance because there is a big difference between reconstructing the sketch input by the user during the drawing process and image inpainting techniques which simply reconstruct masked regions on an image. Therefore, although there are many sketch-based assistance systems and sketch-based image editing applications, there is still a gap in the research on high-quality image generation assistance throughout the drawing process.

1.2 Research Objectives and Problem Formulation

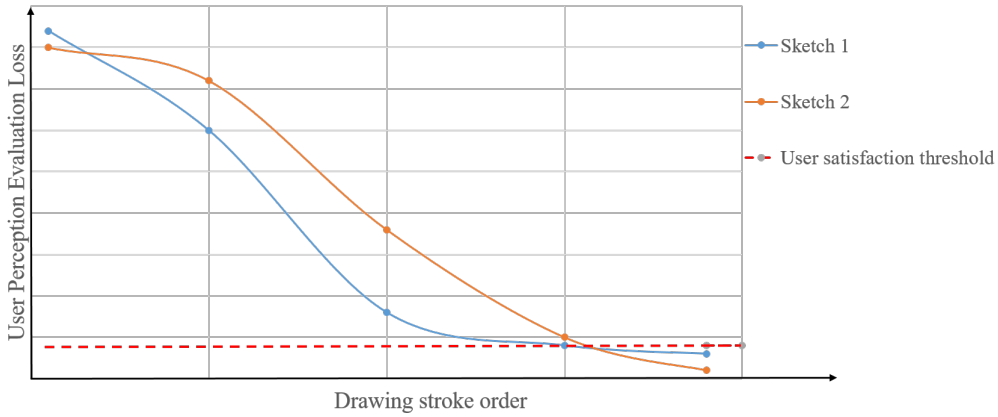
The final research objective of this dissertation is to make AI understand users' rough sketches and provide suitable guidance to support users' art drawing interactively and extend users' drawing ability as well as creativity.

To achieve this aim, the gap between the guidance promoted by the AI during the sketching process and the users' expectations needs to be measured mathematically at first. Different from pure sketch recognition and sketch-based image retrieval tasks, AI drawing assistance is more challenging because AI needs to create a "new" artwork with high-quality details to meet the user's desires based on incomplete sketches that do not exist in a given database. However, whether the guidance in the drawing process and the final artwork meet the user's expectations is a very abstract and subjective concept. Thus, this AI-assisted drawing problem can be formalized mathematically as follows:

Assuming that there is a sequence of n strokes $\{\boldsymbol{s}_1, \boldsymbol{s}_2, ..\boldsymbol{s}_n\}$ for a free-hand rough sketch $\boldsymbol{S}$ as the system input, then the AI first conducts with feature



(a) Original problem



(b) The ideal performance of AI in the simplified problem from (a)

Figure 1.1: The problem setting of this dissertation. As the user-perception exception function $\mathcal{E}(x)$ is unknown, non-differentiable, and dynamic, it is difficult to calculate the optimal solution $x*$. Instead, the problem in (a) is simplified into minimizing the UEL loss function which is defined in Equation (1.4). In this case, an ideal AI allows different users to input different sketches with increasing degrees of completeness to obtain satisfactory results. Just as shown in (b), the values of UEL from different sketches(Sketch 1 and Sketch 2) tend to be 0 stroke by stroke.

extraction operation $F_{\mathbf{FE}}(\cdot)$ as feature variables $\boldsymbol{x}$, and there is

$$F_{\mathbf{FE}}(\boldsymbol{S}) \rightarrow \boldsymbol{x} \in \boldsymbol{\mathcal{X}} \tag{1.1}$$

where $\boldsymbol{\mathcal{X}}$ denotes sketching process exploration space for guidance generation operation $\boldsymbol{\mathcal{G}}(\cdot)$ of AI. Then, the output guidance image $\mathbf{Im_g}$ is:

$$\mathbf{Im}_g = \boldsymbol{\mathcal{G}}(\boldsymbol{x}) \tag{1.2}$$

The AI-assisted system is considered valid if the user believes that the guidance image $\mathbf{Im}$ provided by the AI meets expectations. Therefore, we introduce the concept of user-perceived expectation function $\boldsymbol{\mathcal{E}}(\cdot)$, where a higher value represents a better performance of AI in the free-hand sketching process assistance. According to the above-mentioned notations, the drawing assistance task is described as an optimization problem:

$$\boldsymbol{x}^* = \arg\max \boldsymbol{\mathcal{E}}(\boldsymbol{x}) \tag{1.3}$$

where the optimal solution $\boldsymbol{x}^*$ corresponds to the most intelligent drawing assistance AI system. However, the distribution of function $\boldsymbol{\mathcal{E}}(\boldsymbol{x})$ depends on the subjective will of users – the unknown of $\boldsymbol{\mathcal{E}}(\boldsymbol{x})$ leads to the unknown of solution $\boldsymbol{x}^*$ which cannot be calculated directly with Equation (1.3) as Figure 1.1(a) shown. What's more, the function $\boldsymbol{\mathcal{E}}(\cdot)$ exists only in the user's own mind and cannot be observed in real-time beyond the user's reactions to the outside world, and is therefore non-differentiable. In addition, the function $\boldsymbol{\mathcal{E}}(\cdot)$ may be influenced by guidance from the AI system or changes as their prior knowledge changes with the increase of users' experience – it is dynamic with personal preferences.

For these reasons, this dissertation simplifies the problem by introducing a new loss function $L_{\mathbf{UE}}$ called "User-perception Evaluation Loss (UEL)" :

$$L_{\mathbf{UE}} = L_1(\boldsymbol{\mathcal{E}}(\boldsymbol{\mathcal{G}}(\boldsymbol{x})), \hat{\boldsymbol{\mathcal{G}}}(\boldsymbol{x})) \tag{1.4}$$

where the function $\boldsymbol{\mathcal{E}}$ takes the generated guidance $\boldsymbol{\mathcal{G}}(\boldsymbol{x})$ as the only independent variable, and $\hat{\boldsymbol{\mathcal{G}}}$ is a customized evaluation criterion corresponding to $\boldsymbol{\mathcal{G}}(\boldsymbol{x})$. As UEL represents the gap between the guidance and the users' expectations with L1 loss function $L_1(\cdot, \cdot)$, this problem can be converted into a more intuitive form with the following equation:

$$\boldsymbol{x}^* = \arg\min L_{\mathbf{UE}}(\boldsymbol{x}) \tag{1.5}$$

Figure 1.1(b) illustrates the requirements of the AI-assisted system we'd like to obtain in this simplified problem with a simple example. For any sketching process, the value of $L_{\mathbf{UE}}$ should tend to 0 as the number of strokes increases. Once the value of the function $L_{\mathbf{UE}}$ is below the user satisfaction threshold, it means users have obtained their desired results with the help of AI.

1.3 Solution Paradigm and Contribution

1.3.1 Challenges

According to the above-mentioned simple form of our problem in Equation (1.4), our main task is to find a suitable customized evaluation criterion function $\hat{\mathcal{G}}$ which allows the AI to provide effective assistance to the user continuously. In this process, this paper needs to face the following challenges:

The arbitrary and disorganization of free-hand sketches. During the free-hand sketching process, the stroke order of user sketches is unpredictable because there is no stroke order restriction, which forces AI to have the ability to cope with various stroke order input situations as well as unexpected ”bad” strokes.

The abstraction of sketches. While it is intuitive as an input sketch, sketches often do not contain detailed information, which makes a large difference between the input sketches and the output guidance or final drawing.

The incompleteness of the sketch input. Since AI needs to be provided during the drawing process, this predestines the input sketch to be incomplete in most cases. How to make AI create user-satisfying results based on input sketches under incomplete conditions is an ill-posed problem.

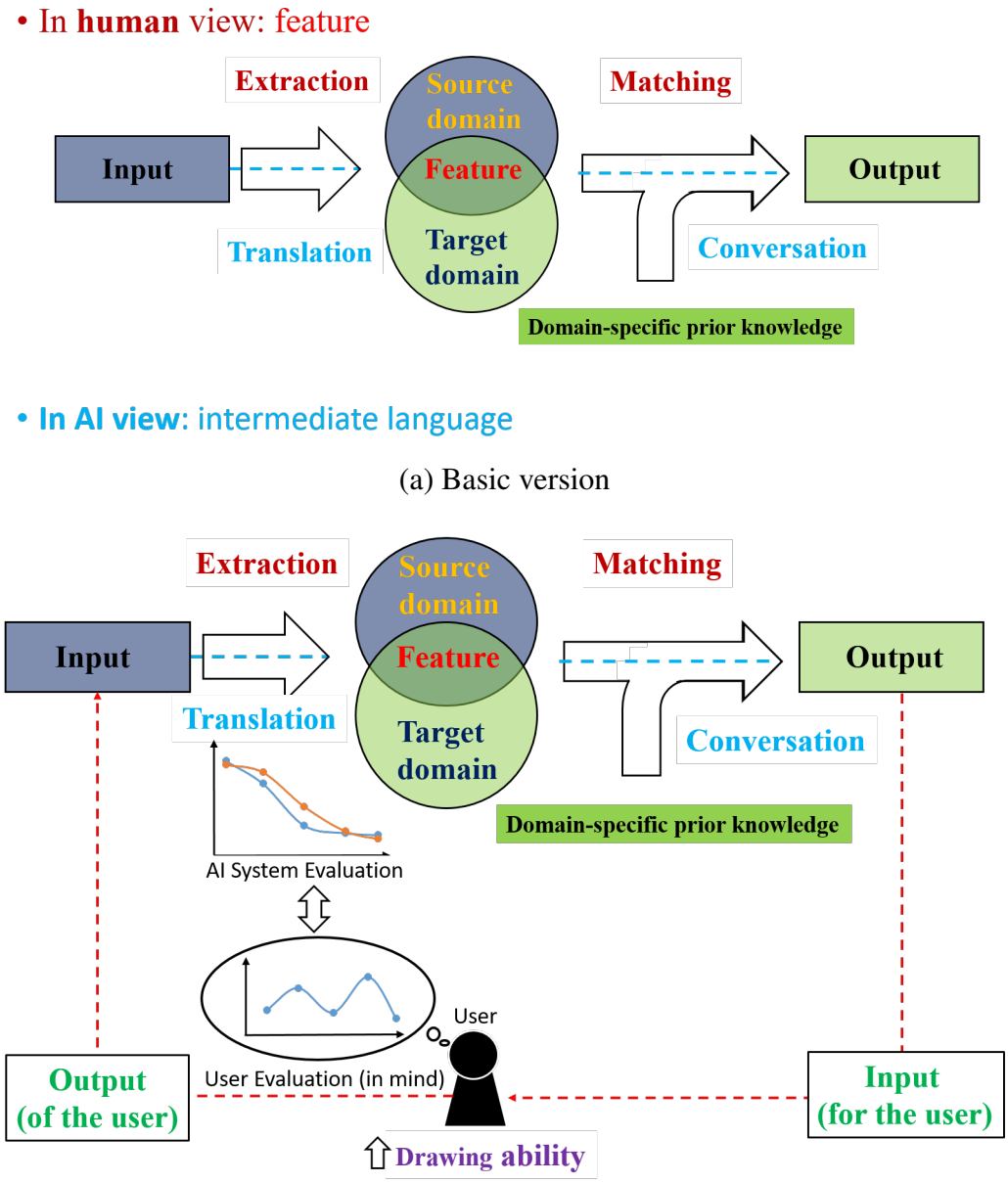
1.3.2 Solution Paradigm

In this dissertation, the sketch-based art drawing generation is considered as a style transfer task. For single drawing generation in our AI system, a prior knowledge-based style transfer paradigm is proposed as the basis of all work in this dissertation which is shown in Figure 1.2(a). The basic version of our paradigm in Figure 1.2(a) is based on the style transfer problem. The blue text indicates the system operation of the AI perspective while the dark red text indicates the system operation in the user's view. Thus, features for the user can be regarded as an intermediate language. In this situation, guidance generation operation $\mathcal{G}(\cdot)$ of AI and the output image $\mathbf{Im}_g^t$, at time t when the first t strokes of a sketch $\boldsymbol{S}_t$ is input, are expanded as:

$$\mathbf{Im}_g^t = \mathcal{G}(\boldsymbol{x_t}|\mathbf{PK}) \tag{1.6}$$

where $\mathbf{PK}$ is a given prior knowledge and features $\boldsymbol{x}_t$ at time t is extracted from $\boldsymbol{S}_t$ according to Equation (1.1).

Based on this paradigm, a User-AI cooperation version is proposed for AI-assisted drawing in this dissertation shown in Figure 1.2(b), which is a recursive system composed of AI and users together. Here, the user is regarded as a part



(a) Basic version

(b) User-AI cooperation version

Figure 1.2: The proposed solution paradigm for AI-assisted drawing of this dissertation. The key idea in the User-AI cooperation version is to let users and AI cooperate with each other, converting the user evaluation function of his/her own mind into an optimization function of the AI system. Leveraging this paradigm, users' drawing abilities can be expanded.

of the AI-assisted system – the output of AI is the input for the user and vice versa. In the drawing creation process, R denotes the user's response function to the guidance, and the sketch $\boldsymbol{S}_{t+1}$ input at the next time $t+1$ for AI is:

$$\boldsymbol{S}_{t+1} = R(\mathbf{Im}_g^t | \mathbf{Im}_o^t) \tag{1.7}$$

where response function $R(\cdot)$ is assumed to be related only to the guidance image $\mathbf{Im}_g^t$ and the ideal object $\mathbf{Im}_o^t$ that the user intends to draw. In other words, $R(\cdot)$ is considered to be a substitution function positively correlated with user-perceived expectation function $\boldsymbol{\mathcal{E}}(\cdot)$ which also can be simply denoted as $R|_{\boldsymbol{\mathcal{E}}(\cdot)}$. Note that the symbol t in $\mathbf{Im}_o^t$ means that the user-perceived expectation function $\boldsymbol{\mathcal{E}}(\cdot)$ is dynamic and can change over time. At the same time, due to the limitations of their own ability and drawing skills, there is a gap between what users imagine $\mathbf{Im}_o^t$ and what they actually draw $\boldsymbol{S}_{t+1}$. Ideally, when $t \to \infty$, $\mathbf{Im}_g^t$ will gradually approximate $\mathbf{Im}_o^t$, i.e., the loss between $\mathbf{Im}_g^t$ and $\mathbf{Im}_o^t$ will gradually become smaller with the assistance of AI:

$$L_1(\mathbf{Im}_g^{t+1}, \mathbf{Im}_o^t) < L_1(\mathbf{Im}_g^t, \mathbf{Im}_o^t) \tag{1.8}$$

which means the user-perceived expectation function $\boldsymbol{\mathcal{E}}(\mathbf{Im}_g^t)$ is gradually converging to the maximum. Since the user takes on the task of exploring the maximum of $\boldsymbol{\mathcal{E}}$, the AI, by contrast, needs to consistently give reasonable guidance to the user's sketches. Denote a loss function which is able to measure the degree of match between sketch $\boldsymbol{S}_t$ and guidance $\mathbf{Im}_g^t$ as L_m, then for any t, there is system objective function f:

$$f = L_m(\boldsymbol{S}_t, \mathbf{Im}_g^t) \to 0 \tag{1.9}$$

where 0 for L_m means the best match of $\boldsymbol{S}_t$ and $\mathbf{Im}_g^t$ is obtained.

Substituting Equations (1.9) and the simple form of (1.6) into Equation (1.9), then

$$f = L_m(R|_{\boldsymbol{\mathcal{E}}(\boldsymbol{x}_t)}, \boldsymbol{\mathcal{G}}(\boldsymbol{x_t}|\mathbf{PK})) \to 0 \tag{1.10}$$

At this point, Equation (1.4) has been transformed into the above form which is more. This Equation replaces the function of subjective user evaluation using the system's overall optimization function, providing a theoretical basis for AI-assisted drawing creation.

According to whether the extracted features $\boldsymbol{x}$ are directly visualized as guidance for users or not, the strategies for constructing the overall optimization function can be divided into two types: implicit strategy and explicit strategy, which are corresponding to Equation (1.9) in and Equation (1.10), respectively. This dissertation conducts explicit strategy Chapter 4 and implicit one in Chapter 5. The main contributions in this dissertation are as follows:

- Support general users for drawing creation with gradual guidance frameworks based on sketch analysis and feature matching from our User-AI cooperation paradigm.
- Lower the threshold of art creation for general users and make artistic drawing easier.
- Insight into the correspondence between users' rough sketches and artwork in the drawing process with AI drawing assistance and tutorial, which is beneficial to keep cultural succession.

1.4 Dissertation Outline

The rest of the dissertation is organized as follows.

Chapter 3 proposed a style transfer-based sketch-art paired data generation framework. Then, based on this framework, an efficient one-shot learning strategy for line drawing style transfer from color illustrations is proposed. Chapter 4 proposed a low-level feature matching-based sketch parsing approach and applied it to realistic portrait drawing assistance with explicit conversation strategy. In order to extend AI-assisted drawing to a more abstract style which differs significantly from the real human face, Chapter 5 proposed the first stroke-level S2I synthesis framework supporting the whole drawings process for high-quality anime portrait generation with implicit conversation strategy. What's more, a one-shot semantic labelling approach for StyleGAN is proposed for the conversation with the user and a comprehension-based drawing support system combining stroke-level feature manipulation from with semantic-level feature matching. Chapter 6 concludes the proposed approaches for drawing assistance, and discusses the future works at the end.

The relationship between chapters is shown in our research route in Figure. 1.3. The framework in data preparation in Chapter 3 provides adequate data support for the following work in Chapter 4 and Chapter 5. Chapter 5 is an extension and improvement of Chapter 4 in the case where the prior knowledge of realistic human faces can not be utilized. Both efforts are centered on the User-AI cooperation paradigm to explore a mutually beneficial way of existence between humans and AI.

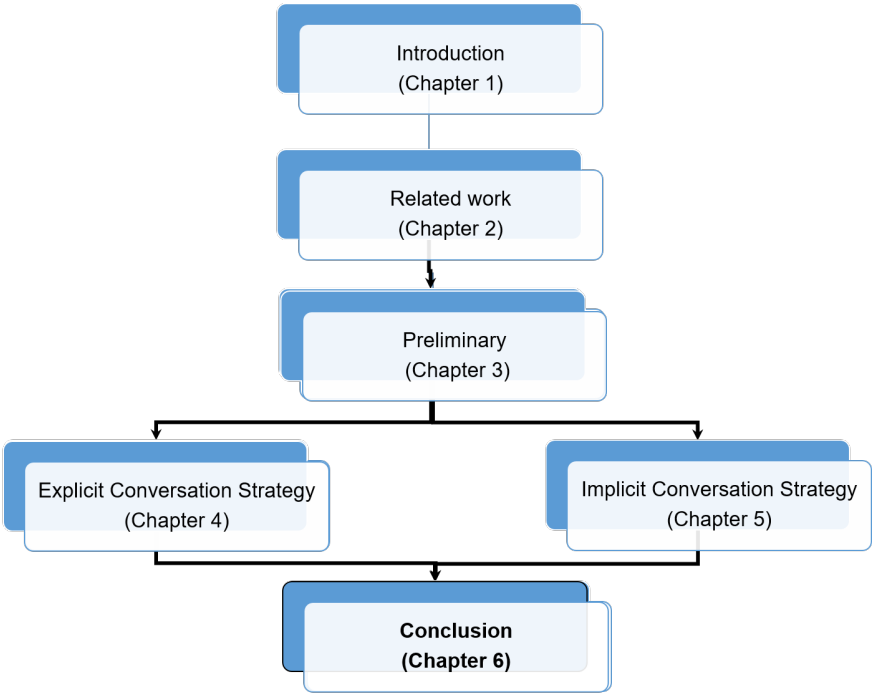


Figure 1.3: Research route in this dissertation.

Chapter 2

Related Work

2.1 Sketch Feature Extraction

The first step we take, as shown in Figure 1.2, is to extract the features of the sketch. In this section, some common features in computer vision are introduced. Features are important measurable information derived from raw data that can be specific to a certain attribute or object. Feature extraction is the process of eliminating redundant information and retaining only information that is valid for a particular task. These techniques have played essential roles in research fields such as machine learning, image retrieval, computer vision, object detection, image abstraction, data mining, and pattern recognition. There are also many studies that make efforts to extract features from sketches for downstream applications such as Sketch-based Image Retrieval (SBIR), sketch-based image editing and sketch-based 3D shape modeling so on. Similar to image features, sketch features are divided into two categories: hand-crafted features and deep features. Hand-crafted features are often elaborated by researchers, while deep features are extracted with deep neural networks. Next, these two features are presented separately.

2.1.1 Hand-crafted Features for Sketch

The first extensive exploration of hand-drawn sketches was initiated by Mathias et al., which analyzed the distribution of non-expert sketches of everyday objects [12]. Before that, there are many shape-related representations and descriptors for sketches have been studied [13]. Generally speaking, sketch representations can be grouped into 4 main categories: region-based representation [14, 15], contour-based representation [16, 17], skeleton-based representation [18], and hybrid-based representation [19, 20]. Cao et al. [17] described each edgel(edge pixel) in an image as a visual word with a triple (x, y, θ), which recorded the edgel orientation θ at that position (x, y). However, this descriptor leads to the loss of position-invariant information on the image unavoidably. A region-based point descriptor for sketch-based image retrieval is developed by Chatbri et al. [15]

combining information about the features of the support regions which is defined on every sketch point. Eitz et al. [21, 22] introduced a bag-of-features sketch descriptor that can represent sketches into local feature vectors by encoding distributions of orientation. Contour points distribution histogram (CPDH) is described by Shu et al. [23] to represent shape information. This descriptor is based on the point distribution on the contour of the object in the polar coordinate, evolving from the shape context. However, calculating the distance between CPDHs has high computational complexity due to its adoption of mirror matching and circular shift schemes in order to partially solve the rotation-invariance problem. Jing et al. [19] present a hybrid descriptor for freehand sketch retrieval by combining region-based features, contour-based features, and skeleton features in a weighted process.

In contrast to the above representations that describe local features, there is another descriptor that can explain the meaning of the object as a whole in the image, which is called the global descriptor. Belongie [24] proposed a Shape-context Descriptor (SCD) which can describe the global and geometric features of images. Histogram of Oriented Gradient (HOG), presented by Shu et al. [23], is effective in extracting edges and textures information from the input image. There are many variations based on the HOG, such as a Field Histogram for Oriented Gradients (GFHOG) proposed by Eitz et al. [22], a co-occurrence histogram of oriented gradients (CoHOG) from Watanabe et al. [25], Circular Histogram of Oriented Gradients (CHOG) developed by Skibbe and Reisert [26], Segmental Histogram of Orientated gradients (SHOG) Katoet al. [27], and Rectangular Histogram of Oriented Gradients (RHOG) from Porikli [28]. Each of them has its own advantages and can be employed in different applications, such as sketch retrieval, face recognition, etc.

In addition to figuring out new descriptors, a fusion of existing descriptors is also a common approach. For 3D model retrieval, Wen et al. [29] propose a joint description, which is invariant to scale, translate, and rotation, by fusing local statistical structures and global spatial features. Similarly, Zhao et al. [30] proposed a novel sketch descriptor by fusing multiple features with their statistic information and bag-of-features representation to achieve translation and scale invariance as well as rotational robustness.

2.1.2 Deep Features for Sketch

Phrase "Deep learning" is first used by Rina Dechter [31] in 1986 and is becoming popular with the introduction of Convolutional Neural Network (CNN) and the development of computing hardware in the last two decades. In the deep learning era, deep feature learning has outperformed hand-crafted features on various retrieval tasks in computer vision [32]. Unlike hand-crafted features, deep features

are end-to-end, in other words, the researcher only needs to focus on designing a suitable network architecture to obtain good features. Here, several representative networks are introduced as follows:

1) Sketch-a-Net was introduced by Yu et al. [33] in 2015, as a CNN dedicated to the recognition of free-hand sketching. It garnered attention as the first to achieve recognition rates beyond those of humans and contributed to the popularization of deep sketch analysis.

2) To leverage the sequential information during sketching, Sarvadevabhatla et al. [34] provides a sketch recognition network, in which the training sketches are redrawn to generate a continuous sequence stroke by stroke and the corresponding deep features extracted from AlexNet [35] would be sent into a Gated Recurrent Unit (GRU) [36] network in sequence. This network allows online recognition during sketching because it contains information about the drawing process of sketches. Similarly, as an improvement of this idea, Jia et al. [37] combine shape and texture features model to upgrade the performance of sketch recognition, in which both features are encoded by corresponding GRU networks respectively stroke by stroke and their outputs are weighted combined based on the respective time step. Furthermore, the deep visual sequential fusion (DVSF) net is proposed by He et al. [38] to obtain both the space and stroke pattern of the sketch.

3) A groundbreaking network called "SketchRNN" is proposed Ha and Eck [39]. SketchRNN learns from its sequential sketch generator based on variable inference [40] for representation. In contrast to the previous sub-image representations accumulated by strokes, SketchRNN takes the key points of strokes as input directly.

4) SketchMate is proposed by Xu et al. [41] as a sketch hashing network. SketchMate backbone contains both CNN and RNN branches, where the CNN is used to extract abstract visual features while the RNN simulates the human stroke order of sketch.

5) Multi-Graph Transformer (MGT) [42] is a novel Transformer GNN model that learns both stroke order process and geometric information from sketch graphs. The transformer architecture in MGT adopts multiple sparse-connected graphs instead of the fully connected graph in the original one. Then, domain-specific knowledge is injected into Graph Transformers via these sparse-connected graphs. What's more, input sketches are converted to extra-stroke graphs as well as multiple intra-stroke ones by MGT, corresponding to their global and local topological features, respectively.

Sketch-based generation studies with deep features or hand-crafted ones focus on the problem of image retrieval or complete sketch and image correspondence generation. The study of the sketching process is often limited to sketch-to-sketch generation, which can not provide a high-quality guide for users. This dissertation investigates the problem of high-quality guidance generation during the sketching

process for art creation and the utilization of both types of features is covered, filling the void in this research area.

2.2 Style Transfer

In this dissertation, the single process of generating guidance images from sketches for sketch-are drawing pair generation in Chapter 3 is considered a style transfer problem. This section describes the progress of research on style transfer techniques. Before deep learning rises, the style transfer methods were usually called texture transfer. which can be mainly divided into two types: Non-Photorealistic Rendering (NPR) and photorealistic rendering [43]. These algorithms provide methods and inspirations for later algorithms with deep learning. When texture transfer technique and deep learning are combined, Neural Style Transfer (NSF) is presented, which has become a highly influential field of research recently. The first complete and effective approach of NSF was proposed by Gatys [44] et al. where features extracted from a pre-trained CNN are divided into "style" and "content", and they are recombined together to generate an image with a similar style to the reference image. Studies most related to this dissertation in style transfer are portrait rendering and line drawing extraction, which are introduced as follows.

2.2.1 Portraits Rendering

In the field of NPR of portraits [45], existing approaches typically take one of two approaches. One approach is to extract contour lines from images [46–48]. While these can be useful for visual abstractions (e.g., preserving and enhancing local shapes), it is difficult to consider semantic constraints and capture specific styles. The other approach is to train a network that automatically generates artistic-like drawings from facial images [49–52]. In these problem settings, training a network requires pairs of facial images and portraits. However, it is challenging to construct pixel-based (dense) correspondence because facial components (e.g., eye and nose) in portraits are manually located by artists. Lie et al. [53] combine a global network (for images as a whole) and a local network (for each facial component recognition) and transform high-quality portraits while preserving facial components.

2.2.2 Line Drawing Style Transfer

Line Drawing Style Transfer, also known as line extraction, can be classified into two categories: edge detection methods and CNN-based approaches. Edge

detection methods such as Canny edge detector [54],Flow-based Difference of Gaussians (FDoG) [55] and Extended Difference of Gaussians (XDoG) [56] are highly dependent on the gradient. This property makes them able to extract useful information from real-world images. However, false contours are usually generated, because color illustrations are abstract. A CNN-based approach proposed a sketch simplification method for rough sketches [57]. A similar approach extracted structure lines from manga images successfully [58]. However, techniques for color illustrations as specific objects are absent in these works, which becomes the motivation of the work in Chapter 3.

2.3 GAN and StyleGAN

A typical GAN [59] exploits the contradiction between a generator and a discriminator for an adversarial game. The generator randomly generates pseudo-images from Gaussian noise in an attempt to interfere with the decision of the discriminator, while the discriminator is required to identify which of the input synthetic image and the real image is real/fake.

With advanced research in GAN design and training, the recent studies [60–62] have made it possible to generate high-fidelity images. To make GAN learn disentangled representations, many previous efforts have been made such as addition of regularization conditions [63], post-hoc disentanglement on the trained manifold [64], or creation of an architectural prior [61].

A milestone approach in regularisation is InfoGAN [63] where two groups of latent codes $\boldsymbol{c}$ and $\boldsymbol{z}$ tend to disentanglement with encouragement: one of two $\boldsymbol{c}$ learned structured information on data distribution while the other one $\boldsymbol{z}$ handles non-structural noise. This goal is achieved by maximizing the lower bound on the mutual information between the generated data and $\boldsymbol{c}$. As an extension of InfoGAN with discrete version, Mukherjee et al. [65] proposed Cluster-GAN where an inverse-mapping network is adopted to project the generated data into the latent space. During the inverse-mapping training, a clustering loss is used for supervised learning as a regularizer. StyleGAN is one of the most prominent Generative Adversarial Networks (GAN) models proposed by Karras et al. [61].

Motivated by Adaptive Instance Normalization (AdaIN) in style transfer [66], StyleGAN consist of a special generator network architecture which is able to generate high-quality images. A typical StyleGAN generator usually involves 3 types of latent space $\mathcal{Z}$, $\mathcal{W}$, and $\mathcal{W}+$. A random vector $\boldsymbol{z} \in \mathcal{Z}$ is often a white noise belonging to a Gaussian distribution, which is the same as the original GAN. In the StyleGAN, the $\boldsymbol{z}$ vector first passes through a mapping network, which is composed of 8 fully connected layers and is transformed to $\boldsymbol{w}$ embedding to an intermediate latent space $\mathcal{W}$. Note that both $\boldsymbol{z}$ and $\boldsymbol{w}$ are

512-dimensional vectors. Here, the introduction of this mapping network is to get rid of the influence of the input vector $\boldsymbol{z}$ by the distribution of the input data set and to better disentangle the attributes. Each layer of the StyleGAN generator can receive a vector $\boldsymbol{w}$ of input via AdaIN. As there are 18 such layers in the StyleGAN generator, StyleGAN can input up to 18 mutually different $\boldsymbol{w}$ vectors. In this case, this different $\boldsymbol{w}$ can be concatenated into a new vector $\boldsymbol{w}+$ with 18×512 dimensions and the corresponding latent space to $\boldsymbol{w}+$ is call $\boldsymbol{\mathcal{W}}+$. One of the applications of $w+$ is style mixing, which is also can be found in Section 5.3. In addition, latent space $\boldsymbol{\mathcal{W}}+$ is used for GAN Inversion [67, 68] which is described in the next section.

2.4 GAN Inversion

With the further development of GAN, how to use the latent space to achieve the manipulation of the outputs from pre-trained GANs has also become a hot research topic [69]. Among various of pre-trained GANs, StyleGAN [70] is usually the most common choice. One of the most important applications of latent space manipulations is face attribute editing. Chiu et al. [71] present a human-in-the-loop differential subspace search for exploring the high-dimensional latent space of GAN by letting the user perform searches in 1D subspaces. [72] identify latent directions with PCA (Principal Components Analysis), and create interpretable controls for image synthesis, such as viewpoint changing, lighting, and aging. By finding out facial semantic boundaries with a trained linear SVM (Support Vector Machine), [73] is able to control the expression and pose of faces. IALS (Instance-Aware Latent-Space Search) is performed to find semantic directions for disentangled attribute editing [74]. Pixel2Style2Pixel (pSp) [75] encoder implements GAN inversion without optimization by using feature pyramids and mapping networks. As it is not necessary to measure the loss between the input and the output of the GAN, this approach also allows semantic layout or sketch as input. Tov et al. [76] argued that adversarial loss and regularization of the latent code should be incorporated into the training of the encoder as an improvement to the editability of these encoder-based methods. In addition, ReStyle encoder [77] improves the reconstruction quality of inverted images by iteratively refining latent codes from the encoder. However, the GAN Inversion technique with incomplete free-hand sketches as input has not been yet investigated. In Chapter 5, we fill a niche in this research area.

2.4.1 GAN Evaluation Metrics

Inception Score (IS) [78] and Fréchet Inception Distance (FID) [79] are the most commonly used evaluation metrics for the image quality of GAN generation. Both of them depend on a pre-trained CNN called InceptionNet which is fed with a large-scale real-world photo image database "ImageNet" for feature extraction. IS calculates the relationship between the conditional class distribution and the marginal class distribution of the generated data with the Kullback-Leibler divergence. FID computes a Fréchet distance, which is also known as the Wasserstein distance between the multivariate Gaussian of the real and the potential space of the generated images projected by the Inception-v3 CNN. In this dissertation, FID is employed to measure the qualitative value of the generated image based on the sketch input as a qualitative evaluation.

2.5 Sketch-based Applications

The sketch is a high-level abstract visual representation without lots of visual details. By analyzing the intention behind users' freehand sketch, the sketch-based interaction allows users intuitive access to various applications such as image retrieval [1, 80, 81] and image editing [82–85], simulation control [86], block arrangement [87], and 3D modeling [88–90]. Among sketch-based systems, freehand portrait sketching is difficult for common users due to the required drawing skills and capabilities, which are inaccessible to novices (e.g., those with poor drawing skills). To address this issue, we aim to establish a user-friendly framework to support the process of the freehand drawing of human faces.

A sketching system's guidance has been thoroughly investigated [91–93]. Especially, displaying visual guidance that can be extracted from reference images (e.g., geometric structures [94, 95]) on the canvas enables one to support the process of the freeform drawing of objects by tracing over the guidance [96, 97]. However, the user must select reference images, which can be time-consuming. Lee et al. [1] and Choi et al. [98] dynamically search relevant images from a large-scale database based on intermediate drawing results at drawing time and generate shadow guidance that suggests a sketch completion to users. A similar drawing interface was designed for calligraphy practice [99]. With these retrieval-based approaches, visual guidance may limit to the predefined database. To overcome this issue, image generation approaches can increase the variations from simple strokes, such as Drawfromdrawings [100] and MaskGAN [101]. Similarly, shadow guidance is used in this dissertation to help users. The only difference is that our goal in this dissertation is to expand users' expressive capabilities to create "new" art drawings, rather than simply imitate them with image retrieval.

Chapter 3

Sketch-Art Pair Generation for Portrait

The first task of this dissertation is to obtain enough sketch-art drawing data pairs in order to provide data support for subsequent research. To accomplish this task, a sketch-art painting generation framework that is applicable to both realistic and anime-style portraits is proposed. Since the problem of style transfer to realistic line drawing in the framework has already been studied [53], the focus of this chapter will be on how to generate anime-style line drawings using a small number of samples. Thus, a one-shot line drawing style transfer for anime style from color illustrations is introduced.

3.1 Introduction

Line drawing, containing a lot of structural information, is an important medium for artistic expression and information abstraction. This dissertation, therefore, adopted the line drawing as guidance for an interactive system in both Chapter 4 and Chapter 5.4. To provide high-quality guidance to users, however, the first problem to be faced is the lack of suitable datasets.

Although there are many sketch-related databases available for deep learning, none of them are suitable for art drawing assistance. Back in 2012, in order to

Table 3.1: Sketch-related databases comparison

Databases	**Modalities & Sample Amount**	**Coarse vs. fine-grained**	**Annotations**	**Category**
TU-Berlin	20K sketches	Coarse	Class	250
QuickDraw	50M+ sketches	Coarse	Class	345
Sketchy	75K sketches, 12K photos	Fine-grained	Class, pairing	125
Da Vinci	71 sketch-line drawing pairs	Fine-grained	Pairing	-
Photo-Sketching	5K sketches, 1K photos	Fine-grained	Pairing	-
Tracing-vs-Freehand	1498 sketches for 100 prompts	Fine-grained	Pairing	-

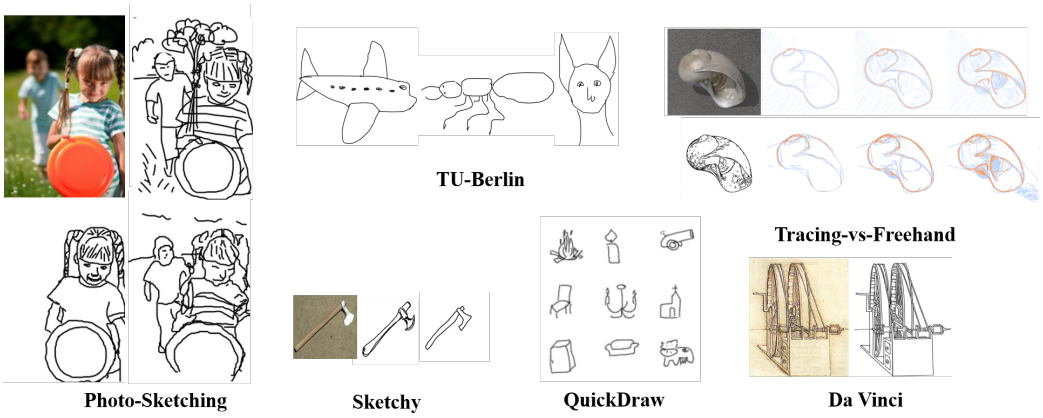


Figure 3.1: Samples from published databases. None of them provide sketch-art drawing pairs.

explore how humans sketch objects, Eitz et al. [12] created a large-scale sketching database called "TU-Berlin" by collecting more than 20,000 hand sketches of nearly 250 categories. Google published the large sketch dataset of Quick Draw for the research purpose by collecting users' rough sketches online [102]. The above databases consist of simple and rough sketches, which can generally only be used for image retrieval or recognition tasks. As to finding the correspondence between rough sketches and real pictures, the following databases are often used for comparison experiments to verify the effectiveness. Sangkloy et al. [103] published the first large-scale sketch-photo pairs called Sketchy, which consists of 75,471 sketches of 12,500 objects with 125 categories drawn by the human hand. Li et al. [104] collected 5000 relative high-quality drawings of 1000 outdoor images grabbed from Adobe Stock [105] establishing a one-to-many photo-sketching database. To evaluate freehand drawings with tracings, Wang et al. [106] create a dataset of 1,498 freehand drawings and tracings by 110 participants for 100 image prompts, whose drawings are all registered to the prompts and contain stroke order information. When converting paper sketches to digital ones, it is often necessary to suppress the effects of noise and clean up excess strokes. Thus, there are studies that have made efforts to create datasets for rough sketch cleanup. Sasaki et al. [107] provided a line drawing restoration dataset which consists of 71 sketch-line drawing pairs. Yan et al. [108] presented a benchmark for rough sketch cleanup with a dataset consisting of 281 sketches obtained in the wild and a curated subset of 101 sketches.

For a more intuitive illustration, samples from several popular databases are shown in Figure 3.1 and the comparison of these sketch-related datasets is listed in Table.3.1

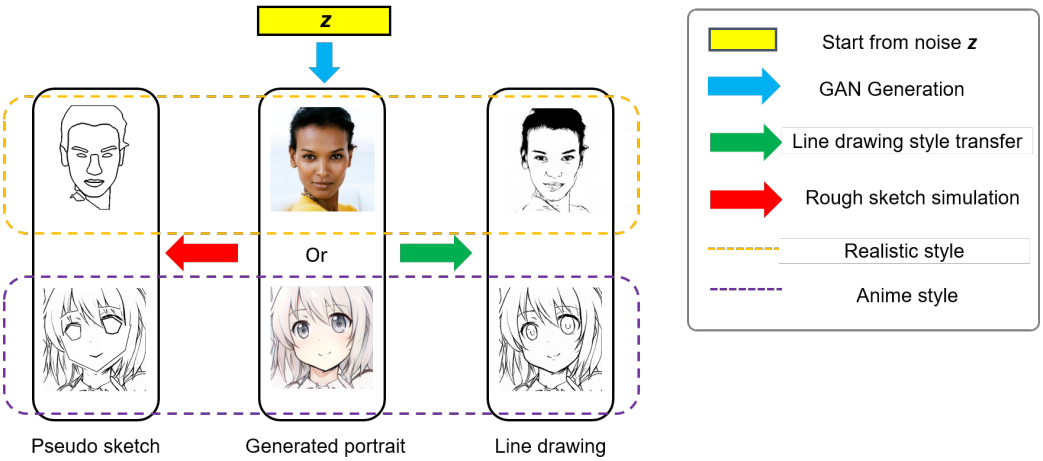


Figure 3.2: Proposed general framework for sketch-line drawing pair generation. This framework can be applied to both realistic and anime styles.

Since there is no data available for sketch-art pairs, this chapter proposes a new sketch-portrait pairs generation for both realistic style and anime style and put the concentration on converting illustrations to line drawing according to previous works.

The contributions of this chapter can be summarized as follows:

- This chapter proposes a new sketch-line drawing pair generation framework for artistic portraits.
- This chapter proposes a novel lightweight CNN-based pipeline for one-shot learning to extract structural lines from color illustrations with arbitrary input sizes.
- This chapter proposes an effective method for data augmentation to avoid over-fitting and the double-edge problem.

The rest of the chapter is organized as follows. Section 3.2 describes the framework for generating sketch-art drawing data pairs. Then, Section 3.3 focuses on the proposed method of line drawing style transfer from anime-style illustrations. At last, Section 3.4 summarizes this chapter.

3.2 General Framework for Pair Data Generation of Portraits

Figure 3.2 shows the proposed general framework for sketch-art generations. This dissertation view art line drawing generation as an Image-to-Image (I2I) transla-

tion task which can be processed with style transfer and the corresponding rough sketches are generated with facial contour information. The images generated from GAN-based models are regarded as bridges between art line drawings and the corresponding sketches.

Firstly, the basic formal description of the I2I task is defined. Given an input image I_A in a source domain A, if there exists a style-transfer mapping $F_{\mathbf{st}}$ such that

$$I_B = F_{\mathbf{st}}(I_A) \tag{3.1}$$

where I_B is the output image in a target domain B with I_A's intrinsic source content preserved and the extrinsic style transferred to domain B. According to the definition of I2I above, it is obvious that converting an image from one domain to a target domain covers a large scale of issues in computer graphics and computer vision.

This chapter concentrates on line drawing transfer for anime style, and the other steps are described in detail in Chapter 4 and Chapter 5.

3.3 One-shot Line Drawing Transfer from Color Illustrations

Although there are numerous works for line drawing from real-world images, extracting lines from color illustrations remains challenging, because there are various hand-painted styles, and additional details such as shadows and textures are difficult to be distinguished from structure lines. Moreover, manually tracing structure lines from color illustrations is labor-intensive and time-consuming. With the development of deep learning, some studies have been conducted to automatically extract lines. However, these CNN-based models require a large number of data for training and usually take seconds for processing. In addition, some of these models (e.g., [109]) extract a bold line as two edges which is known as the double-edge problem (Figure 3.8(b)). One reason these CNN-based methods cannot obtain desirable results is that techniques for colorization expression between training data and test data may be different – adequate training data with the same colorization techniques may be not easy to collect for those CNN-based approaches. This work explores a line drawing style transfer method from a cluster of color illustrations by learning features from only one similar example.

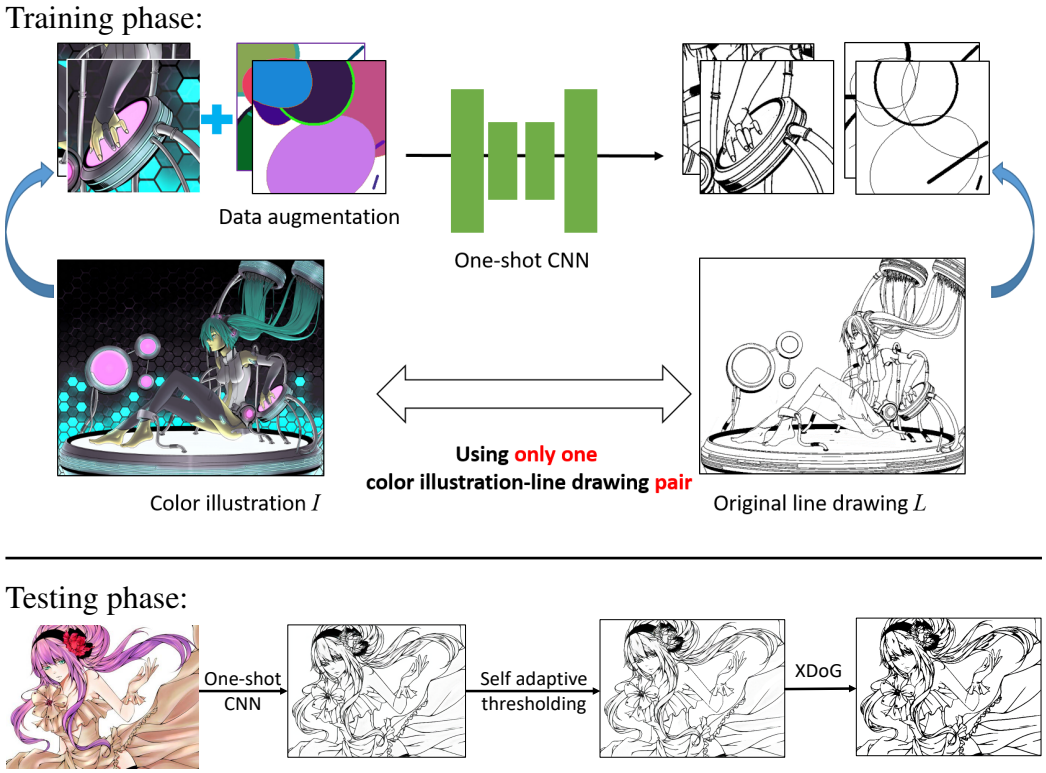


Figure 3.3: Pipeline of the proposed framework for line drawing style transfer from color illustrations, including training and testing phases. One-shot learning is conducted in the training phase.

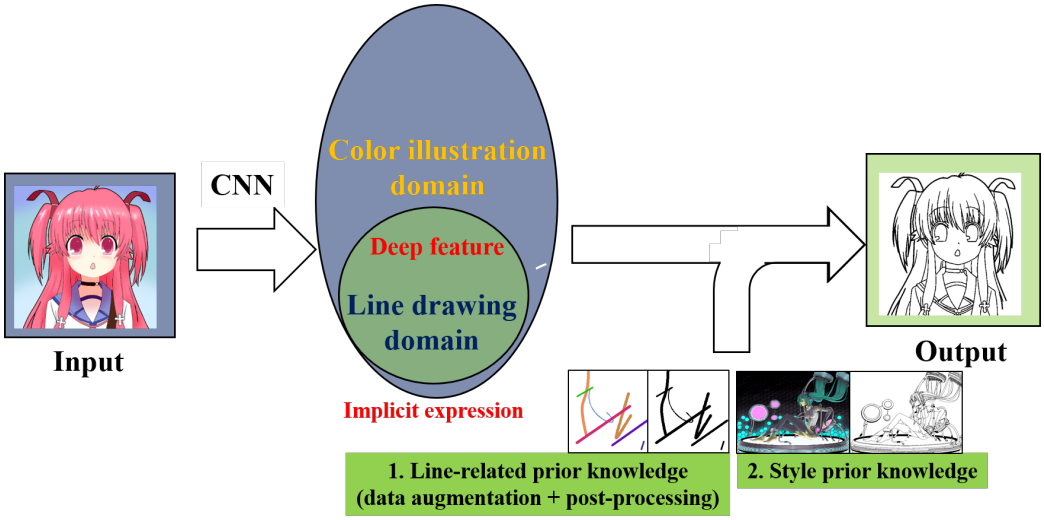


Figure 3.4: Proposed color illustration-line drawing framework for one-shot learning of anime styles. As these deep features are extracted from CNN which can't be understood and controlled by users intuitively, this form of feature is called Implicit expression.

3.3.1 Pipeline

Figure 3.3 illustrates the major processes in the proposed pipeline. Both the training phase and test phase for line drawing style transfer are conducted online. During the training phase, the only one color illustration-line drawing pair is used for one-shot learning after data augmentation processing.

3.3.2 Framework Analysis

According to the pipeline of one-shot learning, the corresponding framework is shown in Figure 3.4. This framework is based on the basic version of our paradigm shown in Figure 1.2(a). Equation (3.1) can be converted into

$$I_B = \mathcal{G}(F_{\mathbf{EF}}(I_A)|\mathbf{PK}) \tag{3.2}$$

where prior knowledge $\mathbf{PK}$ consists of two parts: line-related prior knowledge and style prior knowledge: in this work, line-related prior knowledge is included in the data augmentation for learning in Section 3.3.4 as well as the post-processing step in Section 3.3.5; Style prior knowledge is included in the hand-picked simple of line drawing-color illustration pair for training.

The CNN features of line drawings can be regarded as a proper subset of features extracted from the corresponding color illustrations. It is an implicit

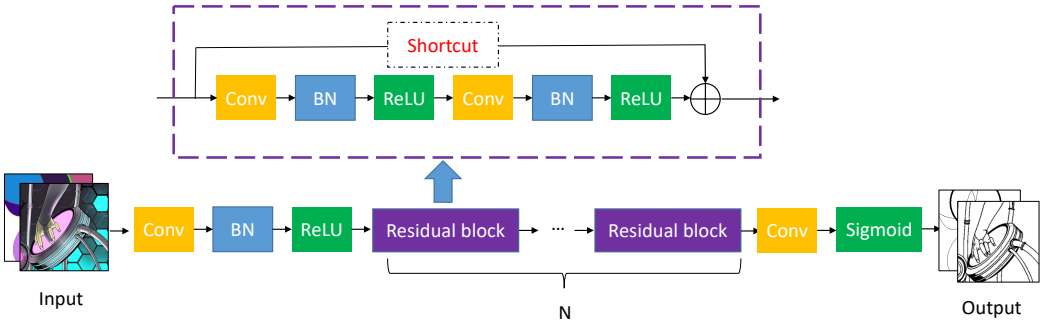


Figure 3.5: Network architectures of our proposed one-shot learning model.

expression of features because the user can not understand and manipulate these CNN features directly. The following Section 3.3.3 will introduce the architecture of CNN we used.

3.3.3 Architecture

Supervised CNNs usually try to capture every possibility of correct color-sketch relations in the learned weights, which train on a large number of pairs of color-sketch image examples. Therefore, these networks tend to be complex with huge numbers of parameters. In contrast, this work can build a much smaller and simpler pixel-wise network based on a single pair of images because the patterns of the color-sketch relations are significantly simple and consistent. The architecture of CNN is shown in Figure 3.5. It consists of seven residual block layers [110], which can preserve important information from input with shortcut mapping to make the gradient on the lower levels easy to propagate, and outperform the basic block (e.g., Conv-BN-ReLU). Note that the number of parameters in our model is 140,067, which is smaller than other pixel-wise CNNs (e.g. Pix2Pix [111] has more than 20M). There is no need to adopt downscaling and upscaling operations in our model which accelerate training at the expense of accuracy as U-net or other pixel-wise CNNs do. This work adopts the sigmoid function as the activation function at the last layer because the reference line drawing (our ground truth) image has been thresholded and normalized – each output pixel in ideal condition is 0.0 or 1.0, matching the range of the sigmoid.

3.3.4 One-shot Learning and Data Augmentation

A single supervised CNN is next to impossible to perform well for all types of color illustrations with different hand-painted styles. Although there are large numbers of color illustrations, the relationship between color illustrations and line

drawings is many-to-many – artists have their own habits and understanding of both sketch colorization and line drawing from color illustration. Therefore, it is not easy to collect adequate color illustrations with the same rendering methods for a data-intensive CNN. To solve this issue, this work aims to extract lines from color illustrations with similar rendering methods by learning the features from only one data. The main idea is to learn line features with data augmentation.

To make our model recognize the line feature in different color illustrations and avoid over-fitting, this work augmented the training data by combining primitive shapes. As shown in Figure 3.6, several primitive shapes, including rectangles, ellipses, lines, and cubic Bezier curves are selected randomly with random parameters and transformation matrices in each synthesized image as the augmented training data. Furthermore, the filling color and the line color of these primitive shapes are also randomly chosen using the color boundaries in the HSV (Hue, Saturation, and Value) space. Note that the white color with the HSV color value ranging from (0,0,221) to (180,30,255) in OpenCV was not selected as the line color. Therefore, our CNN can distinguish lines from filling colors successfully with only one pair of color illustrations and a line drawing for training.

During the training phase, this work adopts the mean square error (MSE) as the measurement metric to obtain smoother structure lines with nearly accurate intensity and pressure. The sampling ratio is 6:4 for training between the color illustration and line drawing pair and our augmentation method. ADAM solver (with learning rate=0.001, β_1=0.9, β_2=0.999) was adopted in our model for faster convergence.

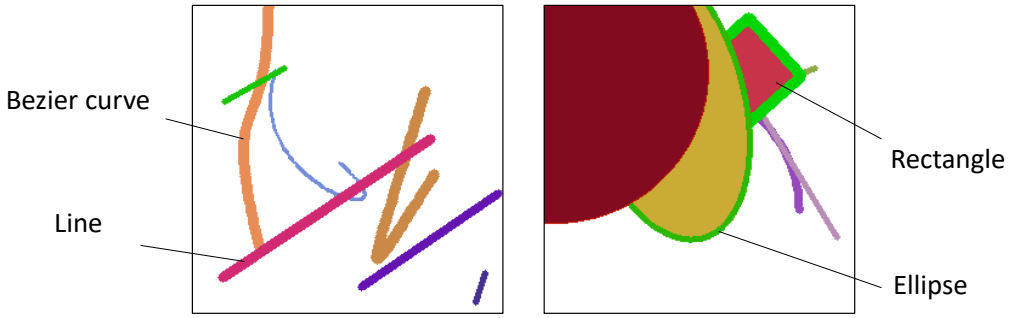


Figure 3.6: Examples of synthesized images for data augmentation.

3.3.5 Post-processing for Refinement

Although the output of our CNN model can enhance structure lines and weaken shadows and textures, a certain number or amount of light gray regions and

shadows remain to some degree. To remove additional details and extract the main structure lines, this work proposes a self-adaptive binary algorithm followed by XDoG [56]. Algorithm 3.1 shows details of the proposed self-adaptive binary algorithm. This work utilizes statistical information – mean value and standard deviation – from color illustrations to distinguish additional shadows from structure lines. Then, the XDoG operator is applied for further noise suppression and increases the aesthetic appeal. Figure 3.7 shows our refinement procedure and results if this work adopts only one of these operations to the source image (Figure 3.7(a)) as a comparison. With the following experiment, it is verified that our method is efficient to remove most of the textures and preserve structure lines from color illustrations.

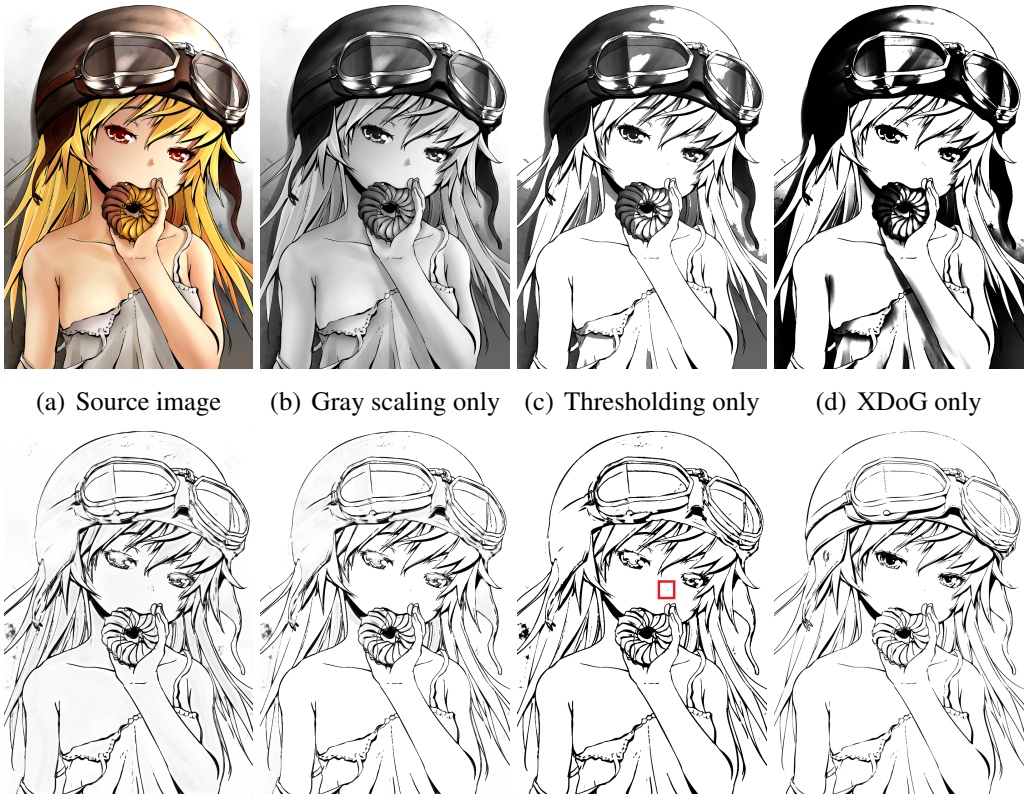
Algorithm 3.1 Post-processing

Input: Original color image C, Output image of one-shot CNN I
Output: Refined image R
Gray image $G \leftarrow$ ColorToGray(C)
Mean of G: $M_g \leftarrow Mean(G)$
Standard deviation of gray image G $Std_g \leftarrow std(G)$
Height of G: $h \leftarrow$ Shape(G)[0]
Width of G: $w \leftarrow$ Shape(G)[1]
for *k=1:3* **do**
  Threshold $t \leftarrow$ Mean(G.where(Pixel $p \in G$,
    if ($p < M_g + (k-2) \times Std_g$)))
  for *i=1:*$w \cdot h$ **do**
    if $I[i] > t$ **then**
      I[i]=Color_white;
    end
  end
end
R=XDoG(I)
return R

3.3.6 Experiments and Results

3.3.6.1 Quantitative Evaluation

To evaluate the proposed method, this work collected 20 Japanese-style "color illustration-line drawing" pairs from online resources. The original sizes ranged from (480, 640) to (2362, 2835). The one shown in Figure 3.3 was selected in the training phase to generate training data. Then, color illustrations from the



(a) Source image (b) Gray scaling only (c) Thresholding only (d) XDoG only

(e) One-shot CNN (f) (e)+Self-adaptive threshold (g) (f)+XDoG (h) Ground truth

Figure 3.7: Intermediate results of our post-processing procedure. Note that in (g) with our approach, the nose in the red box is expressed as a point, which was preserved successfully, and the shadow of the nose was cleared. As a reference, this work also showed results with a single operation mentioned in Algorithm.3.1, i.e. gray scaling, self-adaptive thresholding, and XDoG, respectively.

other pairs were test data and their corresponding sketch images were used as the ground truth. This work calculated the RMSE (root mean squared error) and the recall rate between the line sketches extracted by the framework with the ground truth as a quantitative evaluation.

In the implementation, all codes ran in Python, and this work conducted the experiment on the Windows 10 platform. A laptop with Intel Core i5-8400, 2.80GHz 2.81GHz, NVIDIA RTX2070 GPU, and 16GB RAM was used as the testing computing environment. Both L-net [112] and sketchKeras [113] are trained with thousands of color illustration and line drawing pairs and we adopted the official pre-trained models in the comparative experiment. In addition, this work trained Pix2Pix [111] with the same data with 200 epochs as a baseline. Note that 200 is the best epoch in the implementation because Pix2Pix contains more than 20 million parameters which lead to a more seriously over-fitting after 200 epochs.

Because the other three models adopted the U-net structure, their input sizes must be a multiple of $2 \times N$ (N is the layer number of U-net). As shown in Table 3.2, our proposed CNN model is the fastest with the same size of input images and gets a competitive RMSE result (0.16) when inputting the original size of images to our model compared with L-net (0.15) and sketchKeras (0.14). What's more, the both the recall rate and $F1$ rate of our model outperform the others. Lower precision of our CNN due to background interference which can be illustrated by the second example of Figure 3.8: Our CNN does not distinguish between foreground and background; When there is no background in ground truth but the texture is added to the illustration, the precision is reduced.

Table 3.2: Quantitative evaluation.

Model	**Input size**	**Time cost (s)**	**Precision**	**Recall**	**F1**	**RMSE**
Pix2Pix (baseline)	(512,512)	0.08	0.09	0.14	0.11	0.27
L-net	(512,512)	6.46	**0.87**	0.63	0.71	0.15
sketchKeras	(512,512)	0.22	0.83	0.74	0.76	**0.14**
Ours	(512,512)	**0.05**	0.60	**0.91**	0.70	0.20
	Origin	0.41	0.71	0.89	**0.77**	0.16

(a) Input (b) L-net (c) sketchKeras (d) Pix2Pix (e) Ours (f) Ground truth

Figure 3.8: Comparison results for line drawing style transfer with the proposed model and other models.

3.3.6.2 Visual Comparison

As the RMSE score and recall are statistic indicators, it has difficulty completely expressing the details of the results. This work shows several comparison results from the proposed framework and the other three models in Figure 3.8. As a baseline, the results of Pix2Pix are not good enough because of its over-fitting; the proposed model can remove most of the shadow from color illustrations which do not exist in the ground truth. In the last row of the results, the other results got a false contour, while ours calculated the right line, avoiding the double-edge problem. This verified that the proposed data augmentation scheme is useful for extracting bold lines.

3.3.7 Disscusion and Limitation

This section proposed a one-shot learning-based framework for line drawing style transfer from color illustrations. The evaluation experiment verified that the proposed approach can extract competitive results of line drawings with their original sizes from color illustrations if the training data is selected elaborately. Since this method solves the double-edge problem, the resulting line drawings contain more thick strokes and are more suitable as guidance for users. However, the relationship between training data and test data is not intuitive. To solve this

issue, a feasible solution is to pick up suitable training data for a given test dataset. As the lack of a criterion that can effectively evaluate the style of the pair data, the selection of valid training samples as prior knowledge is not further investigated in this section.

Due to its generalization properties, it is also suitable for line extraction of anime portraits. In the future, the proposed line drawing style transfer approach is intended to be applied in sketch-based applications with generative models [114].

3.4 Summary

This chapter introduced a sketch-art pair generation framework for artistic portraits in Section 3.2. This framework is used in subsequent chapters in the sketch-art generation tasks, providing strong data support for the subsequent chapters of this study. In particular, to overcome the data shortage and improve the visual quality of the guidance for anime style, a one-shot light-weight line drawing style transfer from color illustrations is proposed in Section 3.3 according to the basic version of the proposed paradigm in Chapter 1.

Chapter 4

AI-assisted Drawing with Explicit Conversation Strategy

This chapter attempts to verify the explicit conversation strategy in AI-assisted drawing with the low-level information extracted from sketches. The extracted shape features are used for complementary retrieval and semantic parsing of sketch inputs, which is successfully applied to a realistic style portrait drawing assistance system called "dualFace"; As these low-level features $\boldsymbol{x}$ from sketches in Equatin (1.10) are visualized as intermediate results and can be controlled by users interactively with sketching, they are called explicit expressions for user-AI conversation in this system.

4.1 Introduction

Portrait drawing is one important art genre to represent a specific human from the real world or one's imagination. Some artists, together with their famous portrait drawings, have been widely adored for hundreds of years (e.g., *Mona Lisa* and *Girl with a Pearl Earring*). However, drawing portraits is cumbersome and requires special skills and capabilities (for example, spatial imagination and essential drawing skills), which are inaccessible to novices without prior artistic training. Therefore, the present paper aims to establish a user-friendly framework to support the process of drawing freehand portraits.

Several systems have been proposed for supporting portrait drawings in the guidance-based method. For example, Portraitsketch [97] proposes a framework to display an artistic rendering sketch using tracing. However, the user must prepare a reference image in advance, which can be time-consuming. Shadow-draw [1] and Sketchhelper [98] incorporate image retrieval methods with tracing tools to dynamically search the relevant images from a database instead of manual selection and enable users to understand geometric structures of target designs (e.g., facial parts' locations and proportions). Although the approaches mentioned above can help users copy existing drawings, it is still difficult to explore "new" portrait designs. In addition, these systems are unsuitable for drawing the details

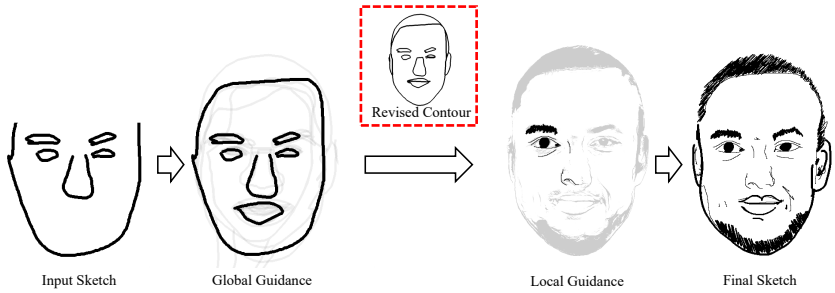


Figure 4.1: The proposed portrait drawing interface provides both global and local guidance from the input of the user sketch. The revised contour sketch in the back end is from the merged mask generated by our conversion algorithm according to the input sketch, which is the reference for local guidance generation.

of portraits (e.g., facial parts' details) because they simply blend a set of relevant images. That is, the details of each image might be lost. Conversely, to explore new drawing designs (detailed drawings), Ghosh et al. [115] and Zhu et al. [116] employ deep learning methods, especially with generative adversarial networks (GAN), to generate possible images with given color or edge constraints. However, the resulting image quality is still determined by the user's drawing skill, such as locating facial parts, so it remains difficult for novices to design high-quality portrait drawings.

To address the problems above, this work referred to the conventional portrait drawing procedures [117]. According to the conventional procedures, it is essential for novices to adopt two types of guidance; (i) global guidance, which helps users locate facial parts (geometric structures) with correct proportions, and (ii) local guidance, which helps users design facial details (e.g., eye and nose). Nonetheless, previous studies do not argue how to guide users to draw both global and local features of portraits, to our knowledge. Thus, this work first considers a method to automatically generate two types of visual guidance, called global and local guidance, from user drawings (see Figure 4.1). In the case of global guidance, as with Shadowdraw [1] and Sketchhelper [98] mechanism, when the user draws contour lines on a canvas, the system dynamically searches relevant images from a database and generates a blended image. In the case of local guidance, the system generates detailed facial portraits from the user-drawn contour lines by using a GAN-based system and displays one of them. Second, this work implements a realistic style portrait drawing assistance system, called dualFace, that incorporates the above visual guidance and is able to switch between the two stages freely.

Our principal contributions are summarized as follows.

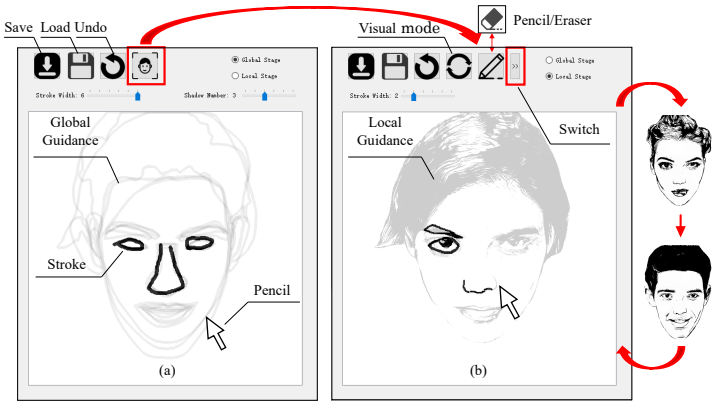


Figure 4.2: Our two-stage AI-assisted drawing system. Given a user's intermediate drawing at run-time, the system generates (a) global guidance generated by blending relevant images from the database and (b) local guidance (i.e., realistic facial portraits) generated by a generative model-based method.

- A two-stage guidance system that helps users design portrait drawings with data-driven global guidance and GAN-based local guidance.
- An optimization method to automatically generate detailed facial portraits with semantic constraints from user-drawn strokes. By using the generated portraits as drawing guidance, the user can explore the desired details without prior artistic training.
- A user study to demonstrate the benefits of our proposed system.

4.2 User Interface

A sketching system's guidance has been thoroughly investigated [91–93]. Especially, displaying visual guidance that can be extracted from reference images (e.g., geometric structures [94, 95]) on the canvas enables one to support the process of the freeform drawing of objects by tracing over the guidance [96, 97]. However, the user must select reference images, which can be time-consuming. Lee et al. [1] and Choi et al. [98] dynamically search relevant images from a large-scale database based on intermediate drawing results at drawing time and generate shadow guidance that suggests a sketch completion to users. A similar drawing interface was designed for calligraphy practice [99]. With these retrieval-based approaches, visual guidance may limit in the predefined database. To overcome this issue, image generation approaches can increase the variations from simple strokes, such as Drawfromdrawings [100] and MaskGAN [101]. Our AI-assisted drawing system combines both sketch-based retrieval and generation with

optimization conversion from sketch-mask mapping.

This section describes how users interact with the proposed two-stage AI-assisted drawing system (see Figure 4.2) to draw portraits with global and local guidance.

4.2.1 Drawing Tool

As with commercial drawing tools, the system enables the user to draw black strokes, in which the stroke width is manually determined using a slider, on canvas with a mouse-drag operation. Then, the system automatically records all the vertices of the strokes and the stroke order for the mask generation step. In contrast, with the eraser tool, the user clicks on a stroke, and the system deletes the selected stroke. Moreover, the undo tool can delete the last stroke from the stroke list. Note that our system can also load (or export) the user-drawn strokes by clicking the "Load" (or "Save") buttons.

4.2.2 Visual Guidance

Given user-drawn strokes, the system generates two types of visual guidance (i.e., global and local guidance) to use tracing. First, in the step of global guidance, the system dynamically searches several relevant images from a database based on the user's intermediate drawing and generates a "blended" image (global guidance) rather than a single image. With the global guidance, users can roughly understand the locations and shapes of facial parts with correct proportions, as shown in Figure 4.2(a). Second, in the step of local guidance, the system generates several detailed facial portraits (guidance candidates) based on the user's intermediate drawings, and displays "one" of them instead of a blended image. The system has a switching function to change the generated images, so the user can search for the most reasonable local guidance. By using the local guidance, users can easily design local details such as eyes and nose; see Figure 4.2(b). Note that the system allows the user to freely switch global and local guidance modes by clicking the global/local radio button or the face icon button.

4.2.3 Rewind Tool

In order to help users to draw the desired portraits, this work provides the rewind tool in our user interface. If users thought the local guidance does not meet their vision, the drawing process can return to the global stage by selecting the corresponding radio button, as shown in Figure 4.2. Our drawing interface can automatically save the sketches while switching between global and local stages

so that users can revise their drawn contour sketches by reloading the recorded data.

4.3 Two-Stage Drawing Guidance

A sketching system's guidance has been thoroughly investigated [91–93]. Especially, displaying visual guidance that can be extracted from reference images (e.g., geometric structures [94, 95]) on the canvas enables one to support the process of the freeform drawing of objects by tracing over the guidance [96, 97]. However, the user must select reference images, which can be time-consuming. Lee et al. [1] and Choi et al. [98] dynamically search relevant images from a large-scale database based on intermediate drawing results at drawing time and generate shadow guidance that suggests a sketch completion to users. A similar drawing interface was designed for calligraphy practice [99]. With these retrieval-based approaches, visual guidance may limit in the predefined database. To overcome this issue, image generation approaches can increase the variations from simple strokes, such as Drawfromdrawings [100] and MaskGAN [101]. Our framework combines both sketch-based retrieval and generation with optimization conversion from sketch-mask mapping.

Inspired by conventional portrait drawing processes, this work proposes dualFace, a two-stage framework for portrait drawing with both a global stage and a local stage for drawing guidance. For the global stage of user guidance, this work provides interactive drawing guidance for each facial part. To help users achieve balanced facial contour drawing, this work adopts the data-driven facial feature query by matching the Gabor Local Line-based Feature (GALIF) [118]. For the local stage of user guidance, this work adopts a GAN-based neural network to generate corresponding fine-grained sketches from a user's rough contour sketch on the global stage. Since this work provides photo-realistic facial details in the local guidance, dualFace can help users concentrate on detailed drawing for facial features and improve their drawing skills. this work believes the two-stage framework of dualFace may narrow the gap between novices and artists in portrait sketching due to the separation of the global contour information from local facial details.

4.3.1 Solution Formulation

According to our paradigm in Figure 1.2(b), the framework proposed for dualFace with explicit conversation strategy is shown in Figure 4.3. Assume that the user always draws the facial contours first when drawing a portrait. The key idea is using function decomposition for guidance generation operation $\mathcal{G}(\cdot)$ of AI in

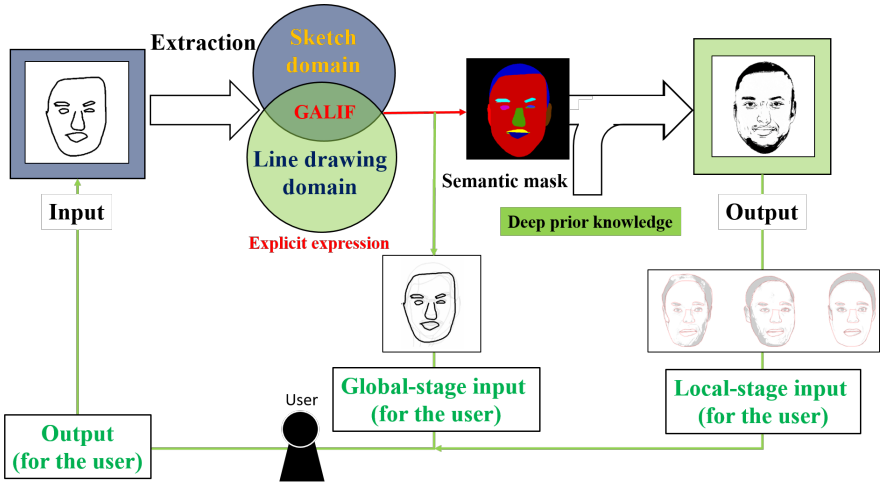


Figure 4.3: The proposed framework with explicit conversation strategy. GALIF extract from sketches is visualized as global guidance explicitly before the user obtains the local guidance with details.

Equation (1.6) to obtain the guidance $\mathbf{Im}$ with details. $\mathcal{G}(\cdot)$ is decomposed into two generation function $\mathcal{G}_{\mathbf{global}}(\cdot)$ and $\mathcal{G}_{\mathbf{local}}(\cdot)$ corresponding to global and local stage. Then, there is

$$\mathbf{Im}_g^t = \mathcal{G}_{\mathbf{local}}(\mathcal{G}_{\mathbf{global}}(\boldsymbol{x}_t|\mathbf{PK_g})|\mathbf{PK_l}) \tag{4.1}$$

where $\mathbf{PK_g}$ and $\mathbf{PK_l}$ denote prior knowledge used in the global and the local stages, respectively. In detail, $\mathbf{PK_g}$ corresponds to the facial contour dataset generated in Section 4.3.2.1 while $\mathbf{PK_l}$ corresponds to pre-trained GAN for the facial mask-realistic drawing conversion. The intermediate output of $\mathcal{G}_{\mathbf{global}}(\boldsymbol{x}_t|\mathbf{PK_g})$ is a user-defined semantic mask M^* generated from the user's sketch automatically with our sketch-mask mapping algorithm described in Section 4.3.3.1.

Similarly, the system objective function f in Equation (1.10) is then decomposed into objective function f_g in the global stage and f_l in the local stage. As time t increases, there is

$$f = f_g + f_l \to 0 \tag{4.2}$$

where f_g is

$$f_g = L_m(R|_{\boldsymbol{\varepsilon}(\boldsymbol{x}_t)}, \mathcal{G}_{\mathbf{global}}(\boldsymbol{x_t}|\mathbf{PK_g})) \to 0 \tag{4.3}$$

while f_l

$$f_l = L_m(R|_{\boldsymbol{\varepsilon}(\boldsymbol{x}_t)}, \mathcal{G}_{\mathbf{local}}(\boldsymbol{M}^*|\mathbf{PK_l})) \to 0 \tag{4.4}$$

Note that loss function L_m has the same meaning as the one in Equation (1.9).

In the AI view, the f_g is the contour matching process in 4.3.2.2, which always finds out the most similar face contours as global guidance from $\mathbf{PK_g}$.

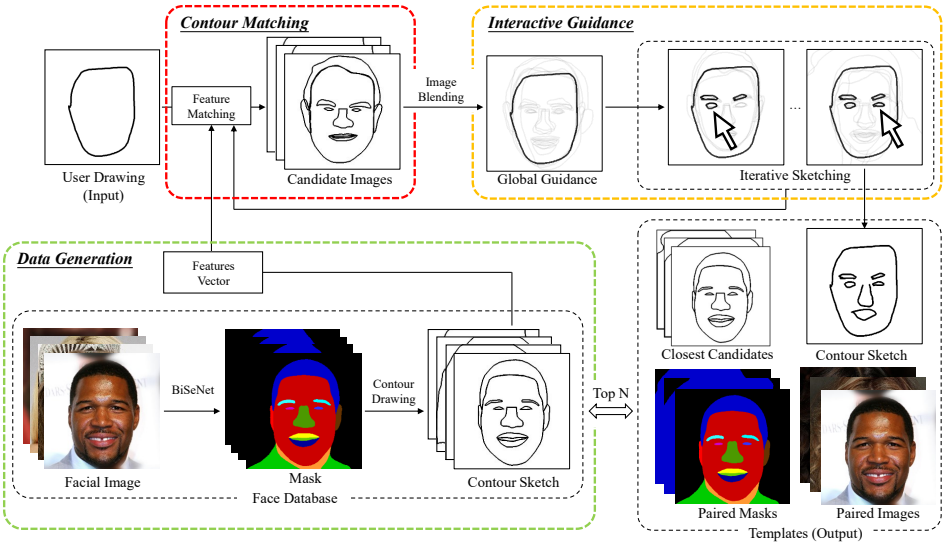


Figure 4.4: The stage of global guidance consists of three steps: data generation, contour matching, and interactive guidance. The contour sketches in our database are extracted from masks as source images which are more meaningful for feature matching to achieve better drawing guidance than previous work [1].

Since the user has already obtained a desired contour sketch with the interactive guidance on the global stage, the next step is simply to select the expected local guidance as a drawing reference from multiple candidates all of which meet the user's rough contour sketch. This selection can make f_g tend to 0 and the user can finally obtain their expected drawing. The following sections will introduce the implementations of Equation (4.1).

4.3.2 Global Guidance

It is difficult to draw recognizable portraits with correct locations and portions of facial features, especially, for novices. To solve this issue, dualFace first aims to help users to draw balanced facial contours by minimizing the global-stage objective function f_g of Equation (4.3). Figure 4.4 shows the workflow of global guidance, including data generation, contour matching, and interactive guidance. For the data generation step, face images are converted to contour images from a face database. For the contour matching step, the local facial features are calculated and stored as feature vectors indexed in the database. For the interactive guidance step, the most similar candidates are retrieved as shadow guidance in real time. In contrast to the previous work of Shadowdraw [1] with edge maps, this

work adopted the labelled contour sketches for feature matching with the semantic sketch information. Therefore, each stroke of users' drawing input can be matched with the meaningful facial features for the next stage of local guidance.

4.3.2.1 Data Generation

It is challenging to collect an enormous number of artist-designed portraits for face retrieval. Instead of the artistic portraits, this work generated semantic label masks [119] by utilizing a Bilateral Segmentation Network (BiSeNet) pre-trained on the CelebAMask-HQ dataset [101]. Each pixel in the masks has a facial label ID from facial images (e.g., eyes, nose, and mouth). This work follows our sketch generation part of the framework described in Figure 3.2 and adopted the contour function of OpenCV library for the line drawing functions. The contours of facial components are extracted from the semantic label masks with balanced facial features. Note that the contour images are stored with the corresponding original face images, which are used for sketch retrieval on the global stage and for system input on the local stage.

4.3.2.2 Contour Matching

To explore the closest contour sketches from the database as the guidance according to a user's incomplete freehand sketch in real time, this work used GALIF features for sketch retrieval and local shape matching [118]. For the online query method, the user sketch is encoded as a histogram. This work calculated the similarity with the stored contour images in our database to obtain the closest contour images.

4.3.2.3 Interactive Guidance

Similar to the shadow drawing interface [1], the top N relevant retrieval results in the face database are merged as a shadow image by image blending ($N = 3$ in our implementation). Benefiting from the interactive global guidance for portrait drawing, users could realize the locations and shapes of each facial part. The global guidance is updated in real-time for each drawing stroke. With the help of global guidance for portrait sketching, the user can complete the contour sketch to express the rough shape and the location of facial parts meeting their drawing intentions.

4.3.3 Local Guidance

In the field of non-photorealistic rendering (NPR) of portraits [45], existing approaches typically take one of two approaches. One approach is to extract contour

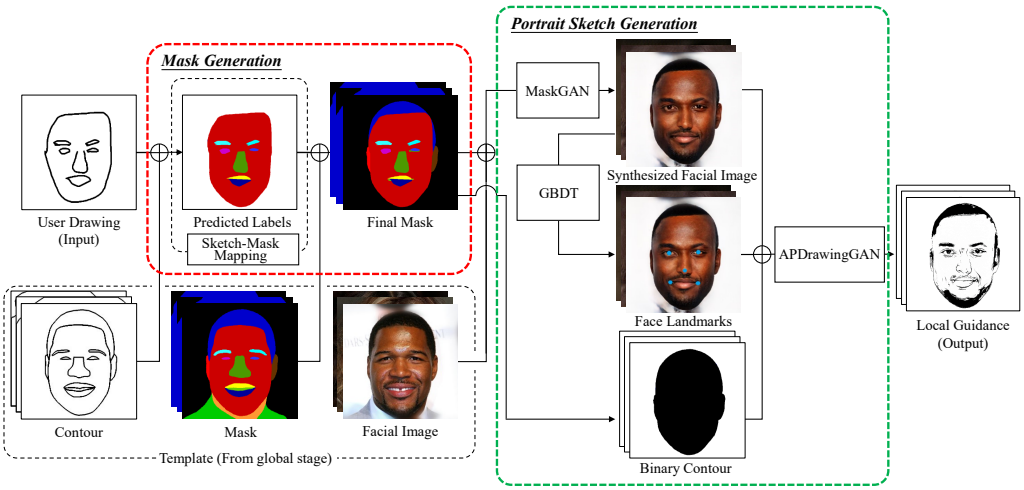


Figure 4.5: The stage of local guidance consists of two steps: mask generation and portrait sketch generation.

lines from images [46–48]. While these can be useful for visual abstractions (e.g., preserving and enhancing local shapes), it is difficult to consider semantic constraints and capture specific styles. The other approach is to train a network that automatically generates artistic-like drawings from facial images [49–52]. In these problem settings, training a network requires pairs of facial images and portraits. However, it is challenging to construct pixel-based (dense) correspondence because facial components (e.g., eye and nose) in portraits are manually located by artists. Lie et al. [53] combine a global network (for images as a whole) and a local network (for each facial component recognition) and transform high-quality portraits while preserving facial components. This work adopted a similar portrait rendering model to generate portrait drawings, and use them as local guidance.

In order to guide users to draw details of facial components (e.g., black irises and eyelashes), dualFace provides local guidance using relevant templates extracted from our database on the global stage. Local guidance for portrait sketching includes mask generation and portrait sketch generation (Figure 4.5). For the mask generation step, user strokes in the global stage are recorded and converted to face masks based on the top N relevant templates ($N = 3$ in our implementation). For the portrait sketch generation step, all templates can generate fine-grained portrait sketches, and the user can select the most desirable one as the reference for further drawing. Note that the input contour sketch is not required to contain all facial parts, and the missing parts can be completed automatically with our stroke-mask mapping optimization.

GAN-based neural networks are used in local guidance for mask and portrait

sketch generation. In our implementation, this work adopted MaskGAN [101] to generate portrait images matching the facial contour sketch and APdrawing-GAN [53] to transfer the portrait images into artistic portrait sketching. Note that two generative models are trained independently. To connect the two models, the facial landmarks are calculated with Gradient Boosting Decision Tree (GBDT) [120], and the binary background mask is converted from the merged mask.

4.3.3.1 Mask Generation

For portrait image generation, the conventional approaches adopted the facial mask with manually defined label information as shown in Figure 4.5 (red dash line of mask generation, and different colors denote facial labels). However, it is a boring and time-consuming task of manual labeling for portrait drawing in our work. To alleviate the manual labor and adapt to freehand sketching, this work proposed automatic sketch-mask mapping with an optimization algorithm to generate a facial mask according to the contour sketch from the users' drawing.

This work first calculates the shape similarity F between user-drawn strokes S and regions of face template mask M. Any single stroke $\boldsymbol{s} \in S$ can be regarded as in-sequence vertices, where $\boldsymbol{s} = \{\boldsymbol{p}_i \,|\, i = 1, \cdots, N\}$. Then, this work obtains the correspondence between two regions using the following equation.

$$\begin{aligned} F(S, M) &= \min_{\boldsymbol{s}} \sum_{\boldsymbol{s} \in S} Dis(\boldsymbol{s}, m_k) \\ &= \min_{\boldsymbol{p}} \sum_{\boldsymbol{s} \in S} (\frac{1}{N} \sum_{\boldsymbol{p} \in \boldsymbol{s}} L_2(\boldsymbol{p}, m_k)) \\ s.t.\ label(\boldsymbol{s}) &= k\ and\ m_k \in M \end{aligned} \tag{4.5}$$

where $Dis(\boldsymbol{s}, m_k)$ denotes the distance between a single stroke $\boldsymbol{s}$ with m_k (region of M with the label ID is k). $Dis(\boldsymbol{s}, m_k)$ consists of $dis(\boldsymbol{p}, M)$, which denotes the average of L_2 distance from all vertices $\boldsymbol{p} \in s$ to m_k. $label(\boldsymbol{s})$ is the discriminant function to calculate the label ID of $\boldsymbol{s}$ decided by the majority vote algorithm of vertex $\boldsymbol{p} \in \boldsymbol{s}$, as calculated by the following equations:

$$\begin{cases} label(s) = \arg\max_{\boldsymbol{p}} C_{\boldsymbol{p} \in s}(V(\boldsymbol{p}, M)) \\ V(\boldsymbol{p}, M) = k^* = \arg\min_{\boldsymbol{p}}\ dis(\boldsymbol{p}, m_k) \end{cases} \tag{4.6}$$

where $C_{\boldsymbol{p} \in \boldsymbol{s}}(\cdot)$ is the aggregate function for stroke s to count the number of its vertices with the same label ID. Discriminant function $V(\boldsymbol{p}, M)$ can determine the label ID k^* for a single vertex $\boldsymbol{p}$ in M by searching the minimum distance of $\boldsymbol{p}$ in each region of M.

Algorithm 4.1 Sketch-Mask Mapping

Input: Strokes S, Matched mask M
Output: User-defined mask M^*
$M^* = zeros(M.\text{shape})$
Number of mask M $m \leftarrow len(M)$
for *k=1:m* **do**
  Merged stroke with same label ms
  Mask region in same label $mask = M[k]$
  if *mask is None* **then**
    continue;
  end
  ms= MergeStrokes($s \in S$ s.t. $label(s) == k$)
  if *ms is None* **then**
    $M^*[k] = mask$;
  end
  $M^*[k]$=ConcaveHull(ms)
end
return M^*

The sketch-mask mapping algorithm is described in Algorithm 4.1. User's strokes are classified to the labels in the matching mask respectively, and strokes with the same labels are merged as a new stroke. Then, a contour (concave hull) of each new stroke is calculated as a new mask to replace the old one in the matching mask.

In terms of the correspondence between the user sketch and face template, this work transfers semantic labels of facial components in the facial template to each region of the user-drawn stroke (e.g., hair, mouth, eyes). Then, this work replaces the corresponding template regions with ones of user-drawn regions if existed and merges the user's stroke feature into the mask. Note that the contour sketch can be auto-completed even if the user input sketch is partial. Finally, this work replaces user-drawn regions (partial sketch) and the corresponding template regions and generates a complete label mask.

4.3.3.2 Portrait Sketch Generation

Generating facial images with details from rough sketches is an under-determined problem. An end-to-end GAN-based model requires extensive artistic drawing with similar styles for training, which is expensive and time-consuming. To solve this issue, this work divides this problem into sketch-to-portrait image generation and artistic rendering for simplification as shown in Figure 4.5. This work first generates a realistic facial image using the MaskGAN network based on the

complete label mask, corresponding face image, and the face template from the global stage. Then, this work converts the face image to a portrait sketch using the APDrawingGAN network for artistic rendering. This work obtains the locations of facial components based on GBDT and the binary contour of the background from the final mask to connect the two generative networks of mask and portrait sketch generation. Note that the global features of the generated local references have been restricted by users' contour sketches.

4.3.4 Implementation

In the implementation of this section, dualFace was programmed in Python as a real-time drawing application on the Windows 10 platform. A workstation with Intel Core i9 10900KF, 3.7GHz 5.10GHz, NVIDIA RTX2080ti GPU $\times$ 2, and 64GB RAM was used as the testing computing environment. In addition, 518 images with a size of 512 $\times$ 512 were picked up from the CelebAMask-HQ dataset and converted to contour sketches. GALIF features were extracted for sketch retrieval on the stage of global guidance. For the implementation of local guidance, this work used MaskGAN for mask generation consisting of the Dense Mapping Network (DMN) for image generation and U-Net like MaskVAE for mask editing, which is pre-trained on CelebAMask-HQ with more than 200 thousand images. This work used APDrawingGAN for portrait sketch generation with a hierarchical GAN structure using U-Net with skip connections for each facial feature (i.e., left eye, right eye, nose, and mouth). This work utilized the pretrained models with 300 epoch training on the APDrawing dataset (140 face images and corresponding portrait drawings by an artist).

Our prototype system requires, on average, 0.36s for image retrieval in global guidance after mouse release every time and 2.78s for each portrait image generation in local guidance. Note that the image generation was conducted only once, meaning dualFace can provide effective feedback for portrait drawing. Because dualFace generates facial images for local guidance, there are no reference images or labels available as ground truth for quantitative evaluation. Therefore, this work conducted a user study to verify the proposed approach in a qualitative way.

4.4 User Study

Due to the difficulty of objectively quantifying the response function $R(\cdot)$ in Equation (4.3) and Equation (4.4), this work adopts user studies to indirectly demonstrate its validity, with visual results, usefulness, and satisfaction. To evaluate the usefulness of the proposed AI-assisted drawing system dualFace, this work compared dualFace with two conventional drawing interfaces: suggestive

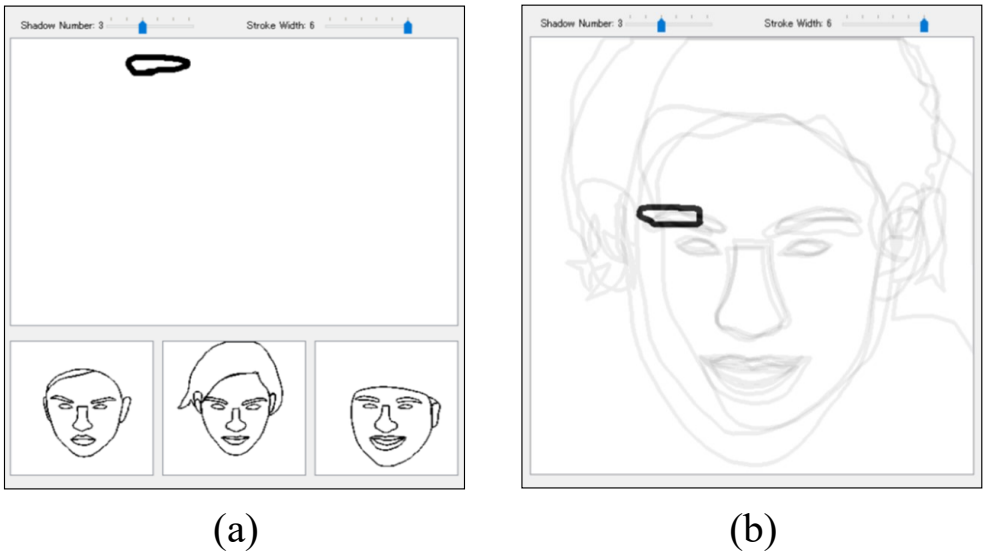


Figure 4.6: Drawing interfaces used in the user study: (a) suggestive drawing UI and (b) shadow drawing UI.

drawing UI (Figure 4.6(a)) and shadow drawing UI (Figure 4.6(b)). The implemented suggestive drawing UI provides the three most related contours in subwindows below the main canvas from the face database. The shadow drawing UI provides the blended shadow image from dualFace's global stage, similar to Shadowdraw [1].

4.4.1 Evaluation Procedure

This work invited 14 participants in the comparison study (graduate students, nine males, and five females). All participants were asked to draw realistic style portraits with a pen tablet (WACOM with 22.4 cm $\times$ 14.0 cm drawing area) and an LCD monitor (126.2 cm$\times$83.7 cm). All participants were asked to draw portraits freely and aimlessly and try to draw more details as possible as they can with all three drawing interfaces: suggestive UI, shadow UI, and ours in random order. They first drew freely on the tablet until they felt comfortable using the devices before the user study. This work instructed all participants on how to use dualFace with a user manual. Considering the usage of facial masks, this work asked the participants to draw each facial mask in a well-closed curve. All participants were required to draw carefully and choose the most anticipated references for local guidance from multiple generated candidates after they completed the global stage. Finally, this work administered the questionnaire to all participants after

Table 4.1: Questionnaire results in the user study. SD is short for Standard Deviation.

#	Question	Score	Mean	SD
Global	Clear and easy to understand?		4.50	0.60
	Feedback is meaningful and helpful?		4.10	1.00
	Easy to follow and use?		4.10	0.80
Local	Clear and easy to understand?		4.60	0.80
	Feedback is meaningful and helpful?		4.00	1.10
	Easy to follow and use?		4.00	1.00
Overall	Helped me learn how to draw faces?		4.20	0.90
	Useful for helping learn how to draw faces?		4.10	0.60
	Useful for helping improve face drawing skill?		3.90	1.20

they finished the user study.

The questions in the questionnaire were designed to confirm the effectiveness of global and local guidance, and the overall evaluation using dualFace, as shown in Table 4.1. All questions adopted a five-point Likert scale (1 for strongly disagree, 5 for strongly agree).

4.4.2 Drawing Evaluation

After all participants completed the comparison study, the other 25 participants joined the online questionnaire for drawing quality evaluation. All participants were asked to score up 12 portrait sketches (four for each drawing UI). This work confirmed two questions about the qualities of the spatial relationship and facial details for all portrait sketches. This work adopted five-point Likert scales for all questions (1 for very poor, 5 for very good). A good spatial relationship of portrait sketches means well-balanced facial parts, and good facial details mean that each facial part has a finely detailed drawing, such as eyes and mouth. This work explained the meanings of the two qualities to all participants before the online questionnaire.

4.5 Results

This section discusses the implementation results of dualFace, evaluation results, user feedback, and the observations from the user study in this section.

4.5.1 Visual Guidance

Figure 4.7 shows some examples of our implementation results sketched with dualFace. Users can achieve the desirable local guidance according to their free-

Figure 4.7: Some examples of our implementation results. *User sketch* denotes results drawn under the global guidance. *Revised contour* denotes the matched facial masks in local guidance. *Local guidance* denotes the generated portrait sketch image for reference to the user. Note that the portrait images in local guidance were selected by users as the closest alternatives to user drawing expectations. *Final result* denotes the final outcome from the users' drawing.

hand contour sketches from the global guidance. If a user's sketch is incomplete, it can be completed automatically and revised with our sketch-mask matching optimization. The last column of Figure 4.7 shows an example of a partial sketch. Although the user only has drawn the left eye and eyebrow contour sketch on the global stage, the proposed system can still work well. The matching of sketches and the corresponding revised contour (combining of input sketch and global guidance) reflects the reduction in f_g of Equation (4.3).

Compared with previous work of the drawing interface ShadowDraw [1], dualFace has no limitation on facial details in drawing guidance. If this work blends the relevant templates (face images with details), it is difficult to distinguish the facial references with the loss of facial details, as shown in Figure 4.8. Therefore, ShadowDraw can only support the drawing guidance of simple subjects without photo-realistic details.

In local guidance, mask generation plays an important role in meeting the user's intention in freehand drawing. To verify this issue, this work compared the system results with and without mask generation, as shown in Figure 4.9. In the case without the mask generation process, the feature lines in the user's contour sketch did not conform to the generated portrait drawing, as shown in Figure 4.9 (left). Meanwhile, more plausible results were achieved by the

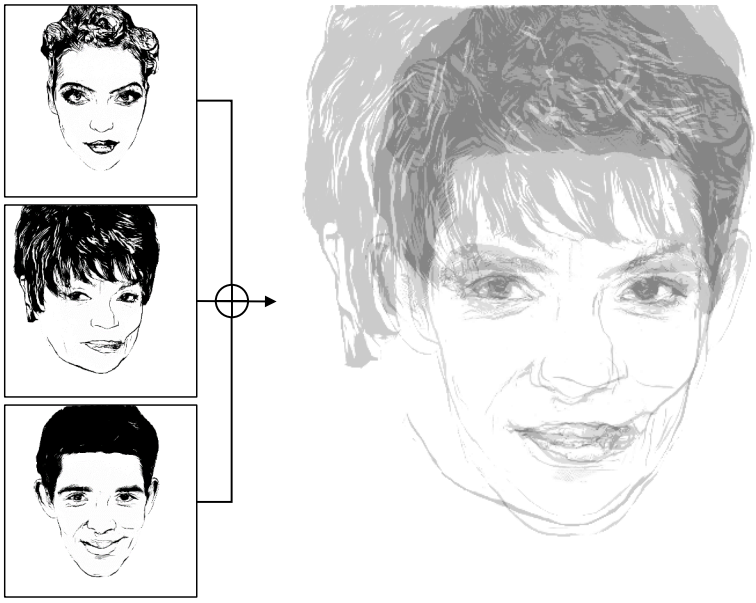


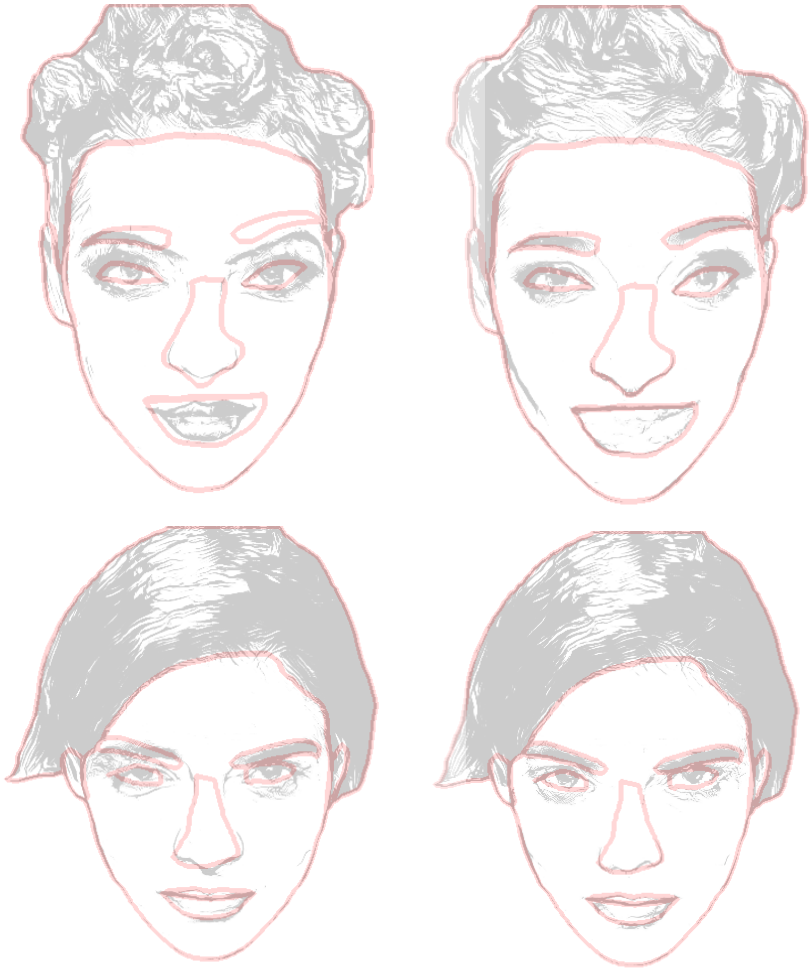
Figure 4.8: As a limitation of ShadowDraw, the blended image (right) has difficulty preserving details of facial images (left).

proposed framework with mask generation. The matching of sketches and the corresponding local guidance reflects the reduction in f_l of Equation (4.4).

4.5.2 User Evaluation

The results of the questionnaire are illustrated in Table 4.1. Participants were asked to score dualFace by answering nine questions in total (three for global guidance, three for local guidance, and three for overall evaluation). The mean scores of all questions are above 3.9, verifying that the proposed drawing interface dualFace is easy to understand and follows at a high level. For overall user experiences, all participants thought our system can help them to draw portraits well and improve their drawing skills. Because dualFace provides guidance on a whole portrait sketch to the participants, users may want to practice basic drawing skills such as arrangements of straight lines or curves. This work plans to improve the current drawing interface to help users practice basic drawing skills in the near future.

Figure 4.10 shows the results of an evaluation study of portrait sketches from our online questionnaire. The proposed drawing interface achieved comparatively high scores in drawing evaluations of both spatial relationships and facial details, and the average scores are 4.5 and 4.32, respectively. Therefore, dualFace can guide users to achieve better portrait drawings with correct facial spatial



Without mask generation With mask generation (ours)

Figure 4.9: Comparison results with and without the mask generation process. Mismatches are obvious between the user's contour sketch (red lines) and the generated local guidance without mask generation (left).

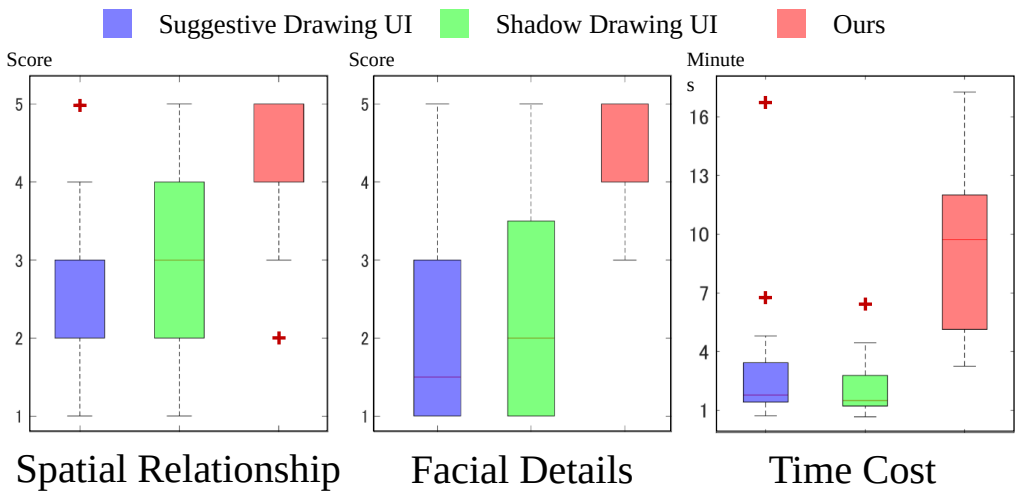


Figure 4.10: Evaluation results of spatial relationship and facial details in portrait drawings (left and middle). Time cost to complete portrait sketching for each drawing interface (right).

relationships and detailed facial features, whereas the other drawing interfaces may fail to provide them.

Figure 4.11 shows the portrait sketches from our comparison study among suggestive drawing UI, shadow drawing UI, and our dualFace UI in our comparison study. This work found that dualFace can not only help users with weak or middle drawing skills to achieve much better portrait sketches but also help the high-skilled users to complete high-quality portrait sketches different from their customary styles of drawing. Note that participants were asked to score their drawing skills using a five-point Likert scale.

This work has received the participants' comments about system usage, such as, "I think dualFace is useful, for helping the freehand drawing especially." This work also received comments about our guidance system, such as, "Local guidance with mask generation fit my stroke more than the one without it" and "Local guidance was surely based on my own, but it looked like a creature." All these feedback indicated that mask generation can increase the variation of sketches but sometimes generate unnatural facial images. This issue can be solved with other neural rendering approaches or a larger face database. This work would improve the current prototype to help users draw from different viewpoints and have high matching rates with users' drawn strokes.

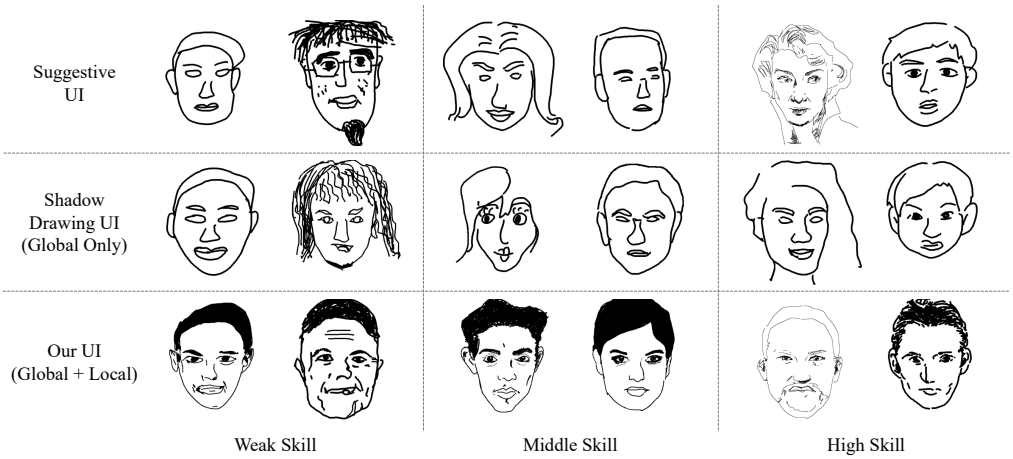


Figure 4.11: Drawing results from six participants. Each column corresponds to the same participant's drawing.

4.5.3 User Satisfaction

To verify user satisfaction and whether or not the proposed drawing interface helped users match their objectives, this work conducted the user evaluation among three aforementioned interfaces: suggestive drawing UI, shadow drawing UI, and ours (Figure 4.6). This work recruited 10 graduate students to join this evaluation and a questionnaire was conducted afterward.

In this section, two questions are confirmed in the questionnaire. The average score for the question "Do you think your rough sketch is matching with the detail guidance to your expectation with dualFace?" is 4.33 (1, not matched at all; 5, well matched). consistent with the decrease of overall system objective function f of Equation (4.2). The average score for the question "How would you rate your satisfaction of drawing with dualFace comparing with other two interfaces?" is 4.44. Therefore, dualFace is verified to enable users draw portraits that match their visions. This work also interviewed the participants for further feedback on user experiences. The comments on the final drawings include: "My drawing was better than I thought.", and "There were plenty of the details of my drawing which makes it look better.". For the usability of dualFace, the users thought that "It is interesting that it generated the details accordingly." and "It can automatically generate details, but also beautify the face (drawing).", which are consistent with our findings in Table 4.1.

4.5.4 Discussion

4.5.4.1 Computation Cost

This work measured the time cost of portrait drawing for each drawing interface, as shown in Figure 4.10. The minimum time among all sketches using dualFace is 4m15s, and the maximum is 17m15s. Although the users' drawing skills may differ from each other, the drawing results with more time cost lead to better drawing results. The average time cost is around 10min, and the portrait drawings, which cost longer than the average time cost, had more facial details and comparatively better quality than the shorter ones. it is believed that our local guidance can not only provide enough detailed features for users to follow but also stimulate users' creativity if they intended to spend more time using dualFace.

4.5.4.2 System Interactivity

Compared with the related sketch to facial image generation approaches [75, 121, 122], the main contribution of dualFace is providing interactive feedback to users for improving their drawing skills. For these works, users cannot get any help from the systems until the drawing is completed, where users' essential drawing skills are usually required. Although DeepFaceDrawing can generate high-quality facial images from rough sketches with shadow guidance [123], it is difficult to improve user drawing skills because they used edge maps extracted from images as guidance without separated local-global facial information. In contrast, dualFace can provide interactive sketch support with two-stage guidance for both global features and local facial details. Our system can provide balanced facial information in real-time, so that users can concentrate on learning how to sketch balanced facial contours, especially for novices.

4.5.4.3 Generation Diversity

To meet users' drawing expectations, it is necessary to ensure the generation diversity of the facial image database. In this work, facial diversity could be influenced by database size and mask generation. However, the best size of the image database for retrieval on the global stage is a hyper-parameter because it is difficult to find out a desirable criterion to evaluate whether generative guidance of dualFace matches users' vision automatically for size optimization. In our implementation, thus, this work selected around 500 typical facial images manually covering different facial types and shapes of facial parts. Our selection strategy is to ensure completed facial parts with clear contours in front view and avoid overlapped parts with hair or glasses. For mask generation, this work can improve the diversity of the generation results for local guidance with multiple

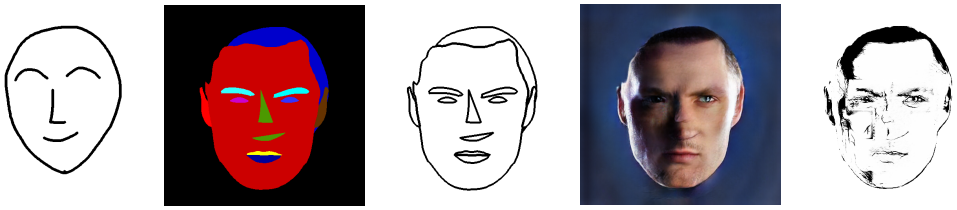
Figure 4.12: Multiple reference candidates (right) are generated from the user sketch (left) for local drawing guidance.

references. Figure 4.12 shows the facial references to users from the global stage so that users can select the most satisfying generated image as local guidance for facial details drawing. All references maintained the shape restrictions (red lines) from sketch input.

4.6 Limitation and future work

In this work, a portrait drawing assistance system with two-stage global and local guidance is proposed. First, this work generates a shadow image to provide locations of facial parts when drawing strokes as global guidance. After specifying the locations of facial parts as a contour sketch, this work then generates detailed facial images from user contour sketches with face masks and portrait drawing generation networks in local guidance. The proposed AI-assisted drawing system, dualFace, was verified to be useful and satisfactory in portrait drawing for users with different levels of drawing skills. This work is believed of contributing to accelerate freehand drawing interfaces.

Because the proposed system converts users' sketches to masks by matching the strokes with the example mask, the contour sketch must contain the exact shape information. dualFace can only support drawing portraits with realistic style due to real photos in the face database. It is difficult to achieve high-level semantic sketches such as emotional faces and exaggerated cartoon-style drawings because it is challenging to determine the shapes of facial parts currently. If the strokes for facial parts are not closed curves, this may lead to indeterminate contours of facial parts. Figure 4.13 shows an input sketch with a smiling face may generate a strange mask with two separated parts of the nose. This work plans to improve the representation of facial sketches and increase the robustness of dualFace, and weigh users' intention and portrait quality.



(a) User sketch (b) Generated mask (c) Revised contour (d) Facial image (e) Local guidance

Figure 4.13: Limitations of our work. An abstract sketch may fail to be converted to a reasonable mask (b), where the mouth in the user's contour sketch is wrongly regarded as a part of the nose. This caused the degeneration of generative image (d) and local guidance (e).

4.7 Summary

In this chapter, an AI-assisted drawing system with an explicit conversation strategy was implemented as a global-local drawing process according to the proposed paradigm in Chapter 1. Low-level feature matching for rough sketches has been experimentally shown to be useful for user drawing, although it fails when misidentifying high-level semantics. With the help of the hand-crafted feature GALIF [118], an incomplete rough sketch during sketching is successfully converted into an input format acceptable to the deep network based on the proposed rough portrait database as prior knowledge and the desired guidance is finally obtained by style transfer with deep prior knowledge.

Chapter 5

AI-assisted Drawing with Implicit Conversation Strategy

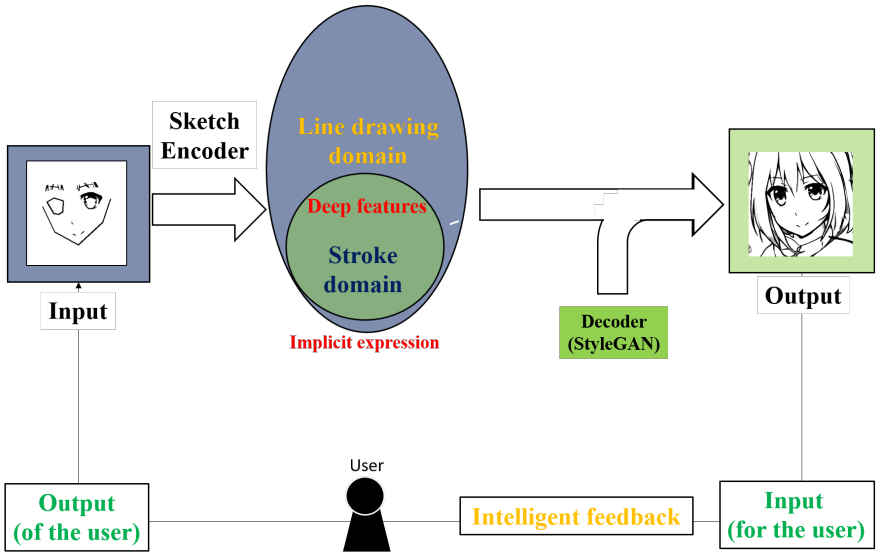


Figure 5.1: The overall framework with implicit conversation strategy. In this strategy, the input sketch influences the output guidance directly, and the user cannot explicitly observes and manipulates depth features.

This chapter introduces a comprehension-based drawing support system with an implicit conversation strategy compatible with both realistic style and anime-style portrait drawing. As stated in Chapter 3, deep features extracted from CNN are implicit expressions that can not be understood and controlled directly.

According to Equation 1.9, the implicit strategy requires a reasonable guidance $\mathbf{Im}_g^t$ generated end-to-end based on incomplete sketches $\boldsymbol{S}_t$ for any time t. For this reason, we propose a portrait generation method based on stroke-level disentanglement. If sketches can be disentangled by strokes in the AI to generate portrait guidance, it means that the AI contains information about the semantics of the portrait face implicitly. Extraction of the above semantic information allows

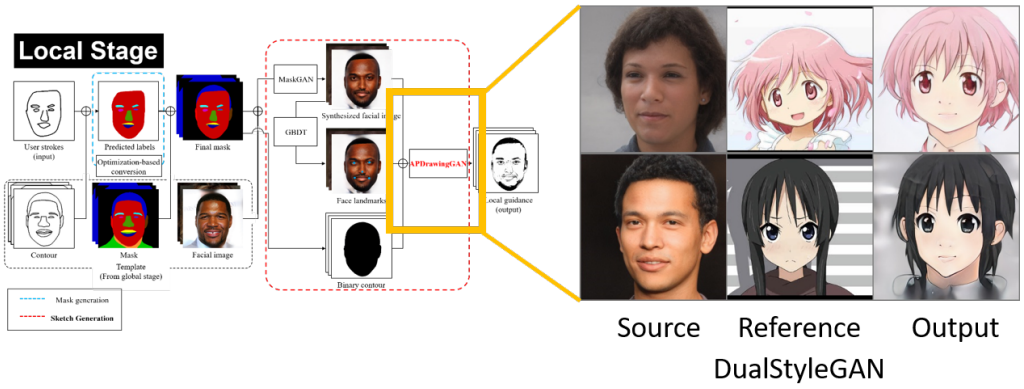


Figure 5.2: Another limitation on the local stage of dualFace caused by prior knowledge when generating anime style drawing is mode collapse.

the AI to further understand user intent and provide more intelligent feedback. Therefore, the implicit conversational drawing assistance framework shown in Figure 5.1 attempts to solve two major issues:

- 1. Sketch-based portrait generation with stroke-level disentanglement in Section 5.3;
- 2. Intelligent feedback from AI itself without additional auxiliary information for anime style is described in Section 5.4.

Note that the deep prior knowledge of both issues is the same pre-trained Sytle-GAN generator. What's more, the final drawing assistance system can support both anime-style and realistic-style.

5.1 Motivation

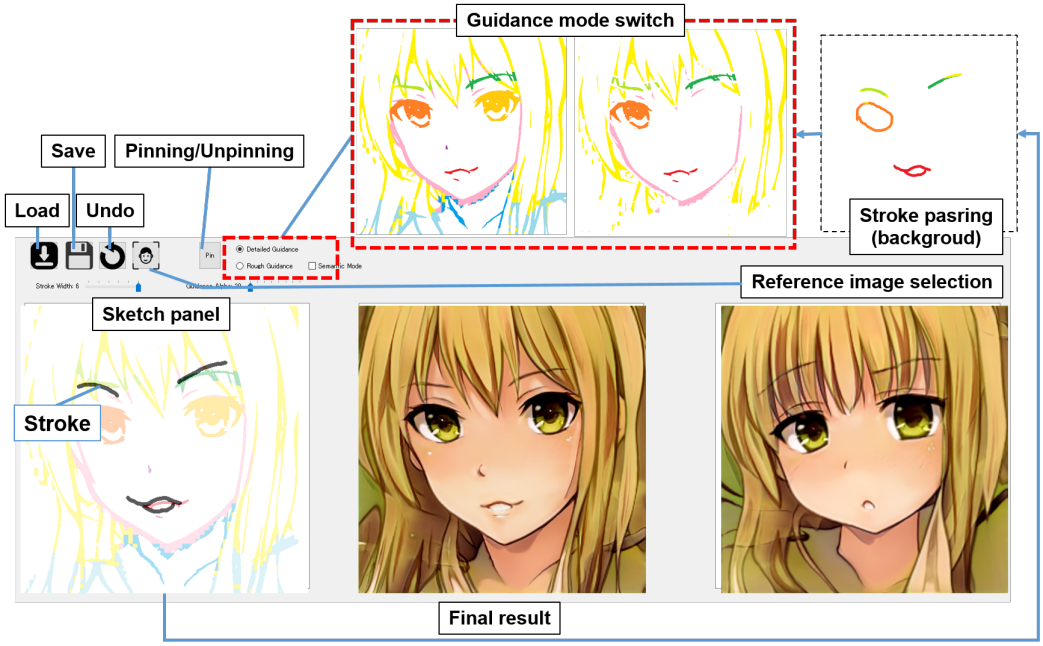
dualFace uses an explicit strategy to generate monochromatic portraits of real faces from sketches through a two-stage process. But both stages require different prior knowledge as data support: the global stage requires a preset database for profile retrieval, while the local stage requires real faces as intermediate results, generating real faces from masks before generating monochrome portraits. In addition to the limitations discussed in Section 4.6, when the difference between the style painting and the real face is relatively large, the results generated by the method using the real face as the intermediate result tend to fall into over-fitting. Figure 5.2 shows an example to show this limitation: when the prior knowledge of APDrawingGAN [53] (real face-line drawing style transfer) on the local stage of dualFace is simply replaced by face-anime style transfer with DualStyleGAN [124], the input of different source and reference images will

generate similar results in shape which is known as “mode collapse”. Thus, anime-style portrait generation is a typical but challenging issue. As a popular drawing style, anime-style portraits are simpler and more abstract than real human faces. To make our AI-assisted system support this style of creation, stroke-level disentanglement for StyleGAN is proposed as an implementation of the implicit strategy of our paradigm. In addition, since there is no off-the-shelf semantic segmentation method for anime faces available, this chapter also introduces an alternative scheme that extracts semantics directly from parameters in StyleGAN with one-shot learning. After the introduction of UI, we will describe the two parts of the work separately in detail.

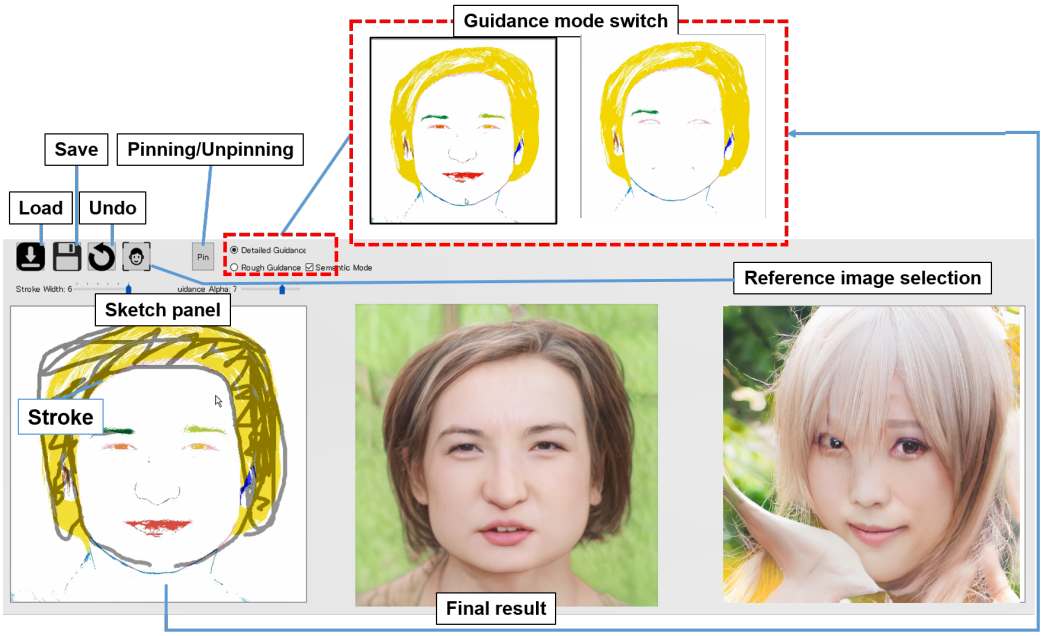
5.2 User Interface

Figure 5.3 shows the proposed user interface of the drawing support system. Both realistic style and anime style portrait drawing can be supported. When our AI-assisted system provides real face drawing support, it is called “RealFace”, and when it provides anime face drawing support, it is called “AniFace”. The system automatically records all the vertices of the strokes and the stroke order and converts strokes to a raster image and corresponding guidance display on the sketch panel in real time. Similar to ShadowDraw [1], this system provide two types of guidance, i.e. “rough guidance” and “detailed guidance” under the drawing board as semi-transparent shadows, which users can switch whenever they wish. Detailed guidance shows the full face portrait to the user as a prompt, while rough guidance shows the user a part of the face that has been drawn roughly or will be drawn soon as a prompt by predicting the user’s drawing progress. Both of them are useful and high-quality, detailed guidance allows the user to understand the overall layout of the face to draw, and rough guidance enables the user to focus on the depiction of the local facial parts. Note both the input sketch and guidance are labeled with semantics automatically and show as different colors for each part with the proposed one-shot semantic labelling approach in AniFace for the lack of semantic data of anime faces. In RealFace, a pre-trained CNN model [125] is adopted for real-face semantic parsing with a variation of BiSeNet [126] which is trained on CelebAMask-HQ dataset [101]. If the user is satisfied with the current guidance and does not want it to change any further for sketch trace, he/she can press the “Pin” button to realize this purpose. When the sketching is completed, users can generate the final color image as a result by clicking on the ‘reference image selection (face)’ button to choose the desired coloring style among reference images.

In contrast, with the eraser tool, users right-click on a stroke, and the system deletes the selected stroke. Moreover, the undo tool can delete the last stroke



(i) AniFace



(ii) RealFace

Figure 5.3: Proposed user interface of drawing assistance system. AniFace supports anime-style drawing while RealFace supports real-face style drawing.

from the stroke list. Note that our system can also load (or export) the user-drawn strokes by clicking the “Load” (or “Save”) buttons.

5.3 Features Disentanglement for Sketch

It is a challenging issue to generate high-quality images from sketches with a low degree of completion due to ill-posed problems in conditional image generation. However, it becomes urgent to convert users’ rough sketches to high-quality images interactively and progressively during the creation processes to expand their drawing skills. This section introduces a novel StyleGAN controlling approach with stroke-level disentanglement in two stages of training to tackle this issue. According to 5.1, the inputs are users’ incomplete sketches and the outputs are corresponding high-quality portrait images or line drawings as shadow guidance in the section. As the deep features extracted from sketches are fully contained by the latent space of high-quality images, the only deep prior adopted in this task is the decoder of deep features which is a pre-trained StyleGAN.

5.3.1 Introduction

With the rapid development of deep learning, image generation techniques for anime portraits have become sophisticated. Especially, the emergence of StyleGAN makes it possible to generate high-quality images. This great success in turn led to the rapid development of GAN control and editing. By linear regression on the disentangled latent space, users can control various properties of the generated image by changing the attribute parameters.

As attribute manipulation with parameters is not intuitive for shape-related attributes (e.g., pose, mouth shape, nose location), sketches become effective inputs of editing for these attributes for Sketch-to-Image (S2I) synthesis. However, most S2I approaches tend to consider only complete sketches as input for image generation – in the case of incomplete sketches and especially the ones with more abstract strokes, they can’t keep the quality of outputs. This issue is particularly evident in artwork image generation. Imagine a scenario where a user draws an artwork with the S2I synthesis system, since the user draws it stroke by stroke, the generated image should also keep matching the sketch locally as the number of strokes increases.

Taking sketch-to-anime-portrait generation with StyleGAN as an example, pSp (Pixel2Style2Pixel) [75] encoder shown in Figure 5.4(a) is an encoder for GAN inversion that can successfully reconstruct a complete line drawing into an anime portrait, which can tolerate small missing areas (first row in Figure 5.4(a)), but it got poor outputs when the input is a sketch with large missing areas (second

(a) Encoder comparison (b) Our anime portrait generation during a drawing process

Figure 5.4: A original pSp encoder working for line drawing with small areas of missing (first row in (a)) can not correctly recognize user sketches, even for a complete sketch on the third row of (a). In contrast, our proposed approach can generate high-quality images. (b) shows our approach can generate high-quality results that consistently match the input sketch throughout the sketching process. To make the matching of sketches and results of our method clear, the intermediate results disentangled most of the color information (second row in (b)) are stacked below the input strokes (blue and red strokes on the first row in (b)) once a new stroke (red stroke on the first row in (b)) is added. The final results on the third row in (b) are generated with random style-mixing techniques. Note that all generated results with "near-white" hair are intermediate results which are style mixing with fixed "near-white" color latent code.

row in Figure 5.4(a)) or a more abstract sketch with fewer details (third row in Figure 5.4(a)). Therefore the conventional S2I synthesis does not naturally maintain partial match and performs well for the stroke drawing process.

To solve these issues, this chapter proposes a stroke-level disentanglement of StyleGAN that allows the generated results to be matched with the user's sketches during the freehand drawing process. Our main contributions are summarized as follows:

- This work presents the first S2I synthesis framework which can generate high-quality anime portraits stably from freehand sketches throughout the whole drawings process;
- This work proposes an unsupervised stroke-level disentanglement training strategy for StyleGAN so that rough sketches with sparse strokes can automatically match the corresponding local facial parts in anime portraits respectively without inputting any semantic labels or strokes.

5.3.2 Stroke-level Disentanglement

This work first explains the concept of stroke-level disentanglement with a simple example in Figure 5.5. Given a generated image I from StyleGAN with fixed color latent code, the left eye and right eye of I in the green rectangle are mapped to $\boldsymbol{L}$and $\boldsymbol{R}$in disentangled latent space with GAN inversion. Stroke-level disentanglement means that there is a sketch-GAN-inversion encoder for the rough sketch which make Strokes 1 and Stroke 2 in the red rectangle can be mapped to the subset of the corresponding latent codes $\boldsymbol{L}$and $\boldsymbol{R}$respectively. Note that the percentage of the latent code of Strokes 1 to $\boldsymbol{L}$is higher than that of the latent code of Strokes 2 to $\boldsymbol{R}$because Strokes 1 contains more details. In addition, there may be a one-to-many relationship between strokes and latent code of different facial parts – for instance, if a stroke includes shape information of both left and right eyes at the same time, then it will correspond to a subset of both $\boldsymbol{L}$and $\boldsymbol{R}$after encoding.

Then, this work describes the problem formally as follows. Let $\boldsymbol{P}$ and $\boldsymbol{S}$ indicate the anime portrait domain and sketch domain respectively. $\boldsymbol{Q}$ is a subset of $\boldsymbol{P}$, which separates most of the representations of color information from structural ones and can form a one-to-one mapping with S. Our sketch encoder learns a mapping $\boldsymbol{F} : \boldsymbol{S} \rightarrow \boldsymbol{Q}$ which can find the correct correspondence with drawing strokes increased. This mapping $\boldsymbol{F}$ is called "Sketch GAN inversion" in this dissertation. The output during the drawing process should gradually converge and maintain high quality as input strokes increase. There are two main issues needed to be addressed:

- Q1. How to learn a stroke-level disentangled mapping $\boldsymbol{F}$ which allows the

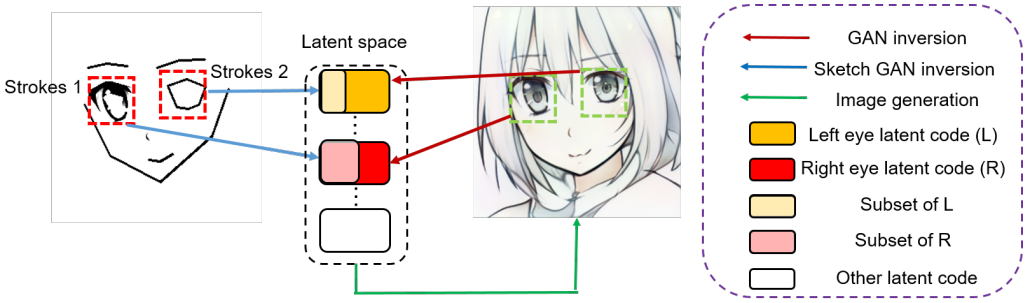


Figure 5.5: Illustration of stroke-level disentanglement. Strokes can be mapped into the subset of latent code of corresponding parts related to shape information.

strokes to be matched locally to the generated image.

- Q2. How to make the aforementioned mapping not affected by the stroke order?

Given a sketch consisting of a series of strokes $\{\boldsymbol{s_1}, \boldsymbol{s_2}, ...\boldsymbol{s_n}\}$, these two issues make it necessary for mapping $\boldsymbol{F}$ to satisfy the following two conditions respectively.

Stroke independence. Assume that an image encoder that can convert an anime portrait to completely disentangled structural latent codes $\{\boldsymbol{d_1}, \boldsymbol{d_2}, ...\boldsymbol{d_n}\}$ corresponding to the strokes, there is

$$\boldsymbol{F}(\boldsymbol{s_i}) = \boldsymbol{d_i} \tag{5.1}$$

Where i is the index of strokes and $i \leq n$.

Stroke order invariance. For any different index of strokes $i, j \leq n$, there is:

$$\begin{aligned} &\boldsymbol{F}(\boldsymbol{s_i}|\boldsymbol{s_1}, \boldsymbol{s_2}, ...\boldsymbol{s_{i-1}}, \boldsymbol{s_{i+1}}...\boldsymbol{s_n}) \\ &= \boldsymbol{F}(\boldsymbol{s_j}|\boldsymbol{s_1}, \boldsymbol{s_2}, ...\boldsymbol{s_{j-1}}, \boldsymbol{s_{j+1}}...\boldsymbol{s_n}) \end{aligned} \tag{5.2}$$

Where $\boldsymbol{s_i}|\boldsymbol{s_1}, \boldsymbol{s_2}, ...\boldsymbol{s_{i-1}}, \boldsymbol{s_{i+1}}...\boldsymbol{s_n}$ means add stroke $\boldsymbol{s_i}$ to a sketch consist of strokes $\{\boldsymbol{s_1}, \boldsymbol{s_2}, ...\boldsymbol{s_{i-1}}, \boldsymbol{s_{i+1}}...\boldsymbol{s_n}\}$. Note this work does not use any label or semantic stroke, the only inputs are monochrome sketches.

Figure 5.6 shows our core idea to simulate the drawing process and make the sketch with a higher completion degree closer to the original sketch into latent space limited in a neighbor region, which can provide the solution for the aforementioned Q1 and Q2. Given a generated image from StyleGAN, the point calculated with GAN inversion in the latent space P is p, and the point in the image fixed color latent code (first row in red dash rectangle) projected into the

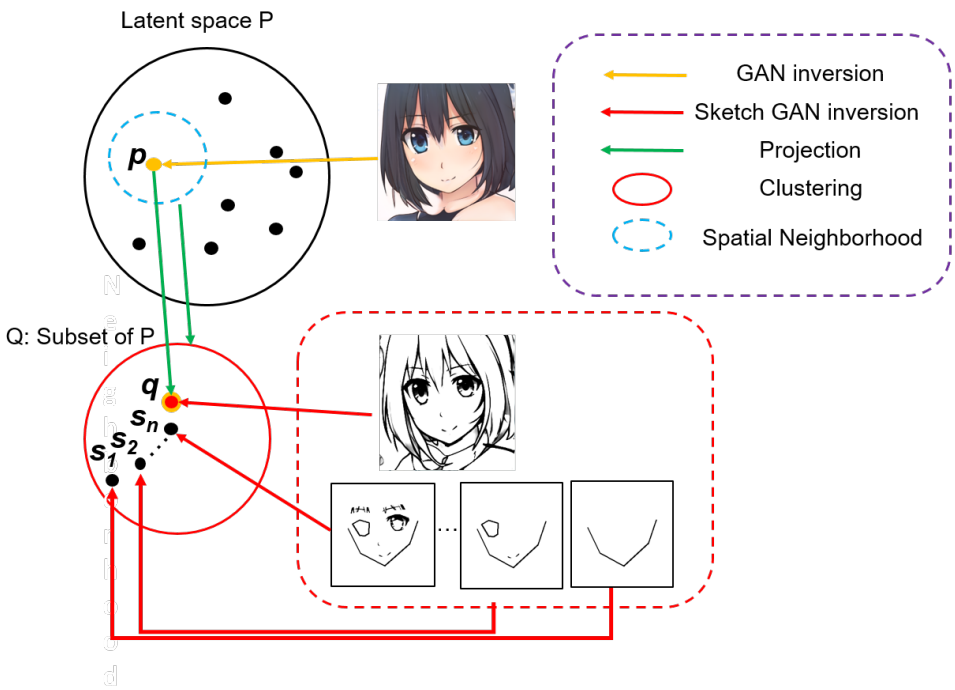


Figure 5.6: Illustration of our core idea.

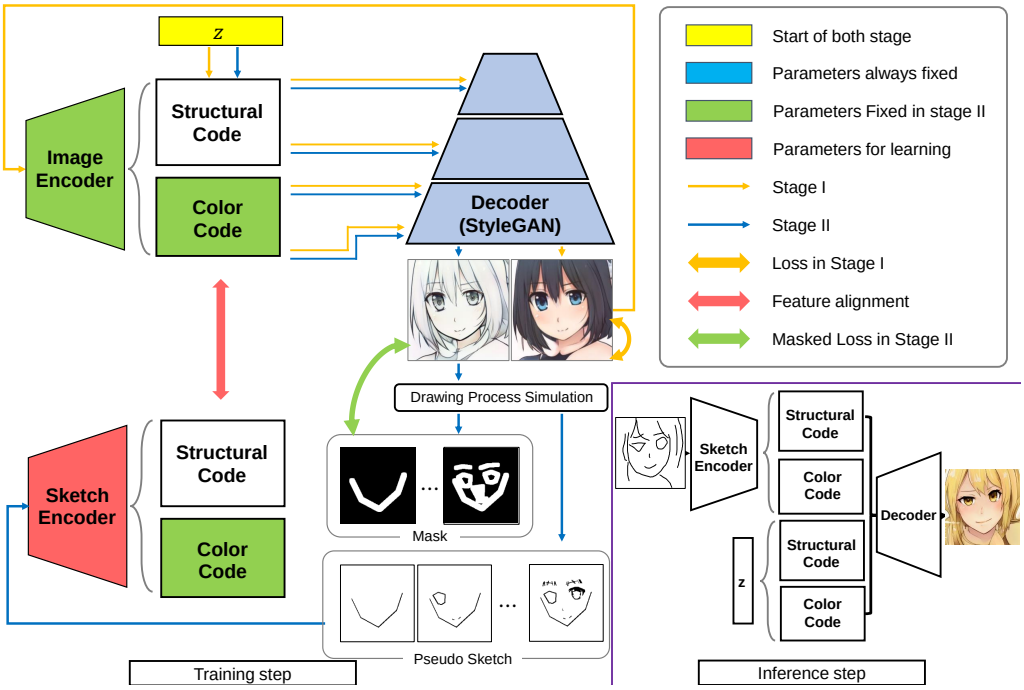


Figure 5.7: Stroke-level disentanglement based S2I Framework. This is a part of the overall framework in Figure 5.1.

latent subspace Q is q. Our drawing process simulation generates a sequence of simulated sketches (second row in red dash rectangle) from simple to complex, whose positions in Q space are denoted as S_1 to S_n. The key idea is to learn a spatial neighborhood in P whose projection in subspace Q can make the sequence of points S_1 to S_n gradually approximate the point q as shown in Figure 5.6.

5.3.3 Proposed Framework

The overview of the framework is shown in Figure 5.7. In the training step, this work first trains an image encoder using the randomly generated images from the decoder as our stage I, which projects the anime portrait correctly back into latent space. Then, in stage II, this work then rearranges the latent space vectors in this image encoder by simulating the drawing process, so that sketches with similar strokes retain more rational distribution when projected into $\boldsymbol{Q}$. In the inference step, this work concatenates the structural codes derived from the sketch encoder with the color codes from random Gaussian noise z, which is known as style-mixing. Note once the decoder has been determined, all data is derived from the

randomly generated images from this decoder, and no additional label information of sketch is required as the input or the output in the proposed approach. Thus, this is an unsupervised learning approach.

The training in stage I is similar to the previous work [75]. The difference is that this work simply adopts L2 loss between the original images from StyleGAN and the reconstructed images encoded by our image encoder in stage I.

5.3.3.1 Drawing Process Simulation

In stage II, the drawing process simulation generates sketch-image pairs automatically from StyleGAN. Before the drawing process simulation, this work should get a complete line drawing from the original anime portrait generated from StyleGAN as the simulation input first. As the first row in Figure 5.8 shown, this work conducts style-mixing between the original and reference image so that most of the color information can be removed and get a complete line drawing from style-mixing result with XDoG [127].

Then, this work uses landmark detection techniques for anime face [128] to obtain information on the contours of each part of the face. This work simulates the intermediate results of sketching starting from a single stroke using Algorithm 5.1. In Algorithm 5.1, both functions RandomProcess and RandomDrawing execute each function in their own lists with equal probability, respectively. That means, during a drawing simulation, each stroke is chosen at random from the contours of the selected facial features and the original line drawing with a random process. The second row in Figure 5.8 shows a list of pseudo sketches with a background augmentation approach which is described as follows.

Background augmentation. As the hair and other parts of the lines could not be extracted using the anime face detection algorithm, this work treats them as background. To increase stability, a random selection from the background and facial contours is conducted as augmentation data in addition to Algorithm 5.1, respectively. The effects of this method are discussed in Section 5.3.4.2. At this point, this work has a series of pseudo sketches for training in Stage II which is termed "Feature alignment".

5.3.3.2 Feature Alignment

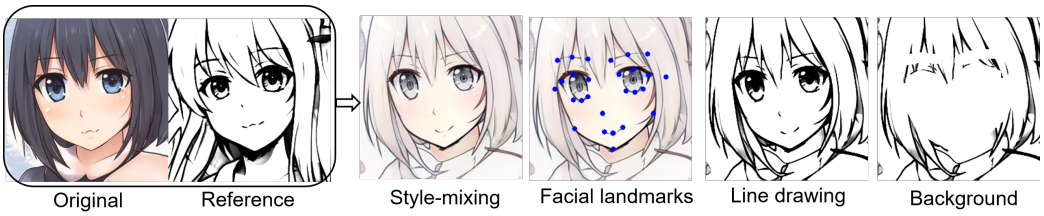
Given a Gaussian noise $\boldsymbol{z}$, the input image of our encoder is $\boldsymbol{x} = G(\boldsymbol{z})$ and the output latent code is then defined as

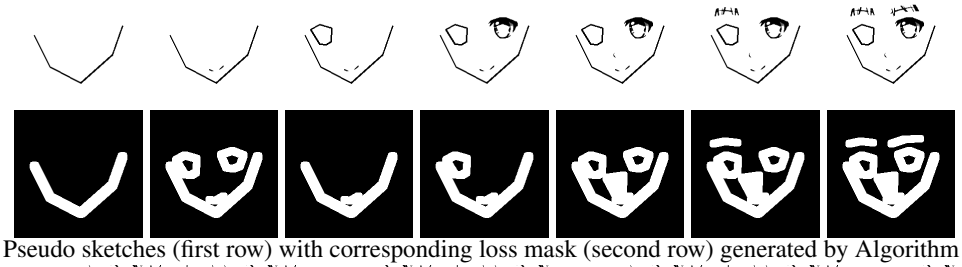
$$I(\boldsymbol{z}) := E_1(G(\boldsymbol{z})) \tag{5.3}$$

where $E_1(\cdot)$ and G(·) denote the image encoder and StyleGAN generator, respectively.

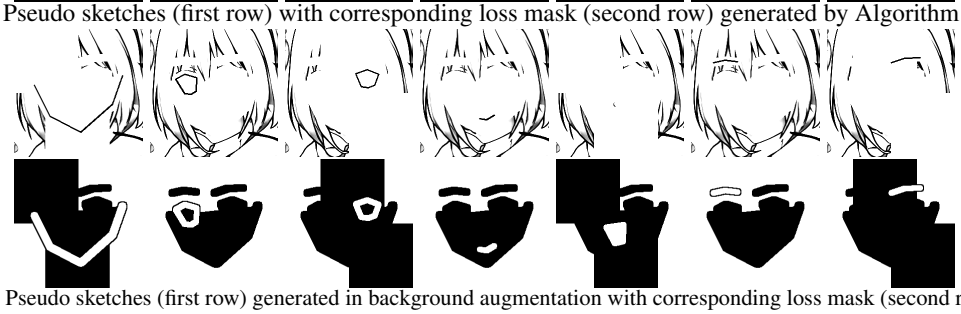
Algorithm 5.1 Drawing process simulation

Input: Portrait image P
Output: List of pseudo sketch S, List of loss mask M
Landmarks of portrait $L \leftarrow FaceDectect(P)$
Strokes of facial parts $C \leftarrow Resort(L)$
Number of $n \leftarrow len(C)$
Temporary while image $p_t \leftarrow$ ones(P.shape) $\times$ 255
Temporary loss mask $m_t \leftarrow$ zeros(P.shape)
RandomProcess=[GaussianBlur; Dilate; Erode, KeepOriginal]
RandomDrawing=[DrawOriginal, DrawContours]
$S \leftarrow \emptyset$
$M \leftarrow \emptyset$
for *k=1:n* **do**
  Index i=RandomSelectOneStroke(C)
  Part stroke s=C.pop(i)
  p_t= RandomDrawing(RandomProcess(p_t ,s))
  S.push(p_t)
  m_t= DrawNewStrokesMask(m_t ,s)
  M.push(m_t)
end
return S,M



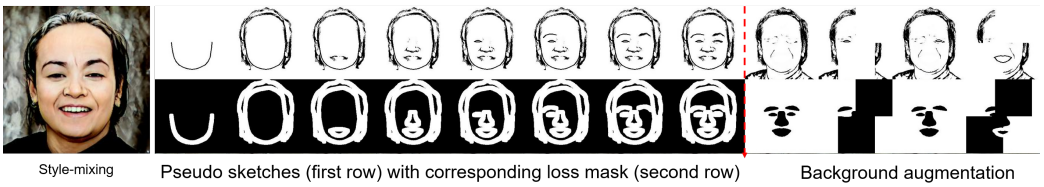


(i) Pseudo sketches (first row) with corresponding loss mask (second row) generated by Algorithm 5.1



(ii) Pseudo sketches (first row) generated in background augmentation with corresponding loss mask (second row)

(a) Drawing process simulation for anime-style.



(b) Drawing process simulation for realistic style. A generated real human face after style-mixing is input for drawing process simulation.

Figure 5.8: An example of drawing process simulation and background augmentation for anime-style and real-face style.

The method for training an image encoder in stage I follows the usual GAN inversion method. The loss function in stage I L_I this work used is as follows:

$$L_I = L_1(G(I(\boldsymbol{z})), G(\boldsymbol{z})) \tag{5.4}$$

Only by calculating the L1 distance between the input image and the reconstructed image, the image encoder can already learn the inverse mapping very well.

Similarly, this work defines the output latent code of our sketch encoder as:

$$S(\boldsymbol{z}) := E_2(Draw_i(G(\boldsymbol{z}))) \tag{5.5}$$

where $E_2(\cdot)$ and $Draw_i(\cdot)$ denote our sketch encoder and our drawing process simulation as described in Algorithm 5.1, which can convert the image $\boldsymbol{x}$ to a series of intermediate sketches of the drawing process and select the ith one from these sketches.

In each iteration of training in stage II, this work can generate sketches $\boldsymbol{S}$ and corresponding loss masks $\boldsymbol{M}$ after our drawing process simulation. Then, the loss function is:

$$L_S = L_1(G(S(\boldsymbol{z})) * M, G(\boldsymbol{z}) * M) + L_2(I(\boldsymbol{z}), S(\boldsymbol{z})) \tag{5.6}$$

Here, L2 loss function $L_2(I(\boldsymbol{z}), S(\boldsymbol{z}))$ ensures that the sketch with higher completion degree is closer to the projection of the original in the latent structure subspace, while L1 loss function $L_1(G(S(\boldsymbol{z})) * M, G(\boldsymbol{z}) * M)$ with loss mask ensures the local similarity between the original images and the generated results.

5.3.4 Experiments and Results

5.3.4.1 Implementation Details

In our implementation of AniFace, the image encoder and the sketch encoder (Figure 5.7) adopted the pSp architecture [75]. This work chooses layers 1-8 in W+ (which is described in 2) space as the structural code and layers 9-18 as the color code, respectively. This work adopts Ranger optimizer and set a learning rate to 0.0001. As a training environment, NVIDIA RTX3090 GPU was used to train our encoders which are programmed in Python on the Linux platform. Then, a workstation with Intel Core i7 8700, 3.20GHz/3.19GHz, NVIDIA RTX1070 GPU, and 64GB RAM on the windows 10 platform were used as the testing environment. Naturally, RealFace also can also be trained with this approach. The difference between AniFace and RealFace in training is that RealFace can extract the hair contour using semantic segmentation contour information and reduce its reliance on background augmentation. Thus, In each stroke simulation generation, AniFace added 8 additional background augmentation images, while RealFace

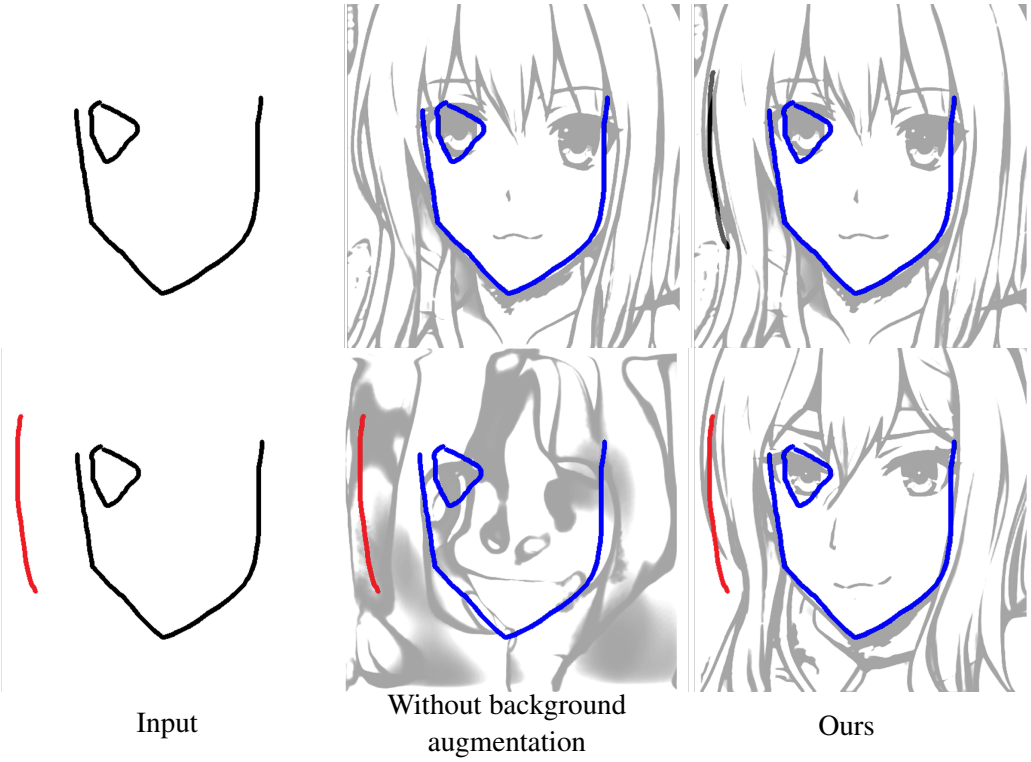


Figure 5.9: Comparison between an encoder trained without background augmentation in Stage II and the one with our approach. When a new stroke (red) is added to the input of the first row, the result from the encoder trained without background augmentation is highly degraded.

used only 4 as Figure 5.8 shown. The result of RealFace will be shown in the next section and the discussion in this section is dominated by AniFace.

5.3.4.2 Stability Testing

To test the stability of our sketch encoder during the whole freehand sketch process, this work first conducts the following experiments.

About the influence of stroke order and multiply strokes for one facial part. Figure 5.10 compares the intermediate process of the same sketch with different stroke orders. It can be seen that the final results are not very different, but the intermediate processes maintain some diversity. This figure also shows that even if only one stroke is used for each part of the face during training, the generated guidance match well when the user uses multiple strokes for the same part (e.g., left eye and mouth).

Effect on training without background augmentation. The effect of back-

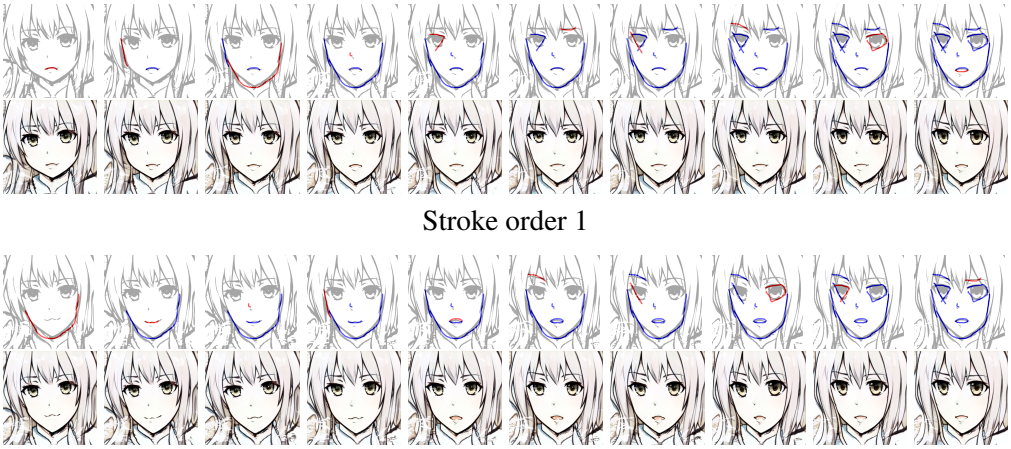


Figure 5.10: The influence of stroke order and multiply strokes for one facial part.

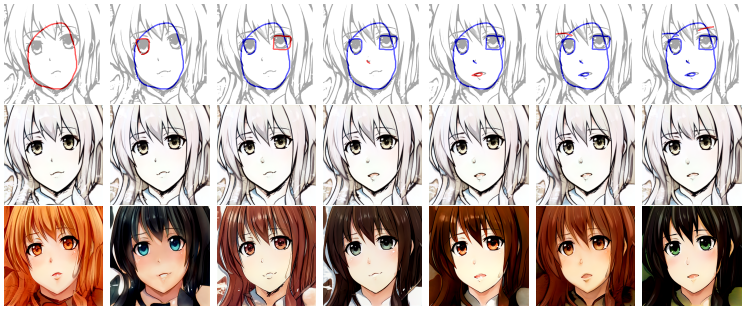
ground augmentation is shown in Figure 5.9. If only sketches in the drawing process on the second row in Figure 5.8 are trained without considering the background (on the fourth row), the sketch encoder cannot correctly understand the strokes associated with the hair (or the background) and project them near the correct position.

Results generation for "Bad" strokes. If only a partial part of a stroke has valid information, then the stroke is considered a "bad" stroke. In freehand sketching, a "bad" stroke is not uncommon. The results generated by our method provide a reasonable match to the valid part of such a "bad" stroke. For example, the strokes depicting the left eye in Figure 5.10 form a triangle, a shape that is not natural as a depiction of the eye contour, while the generated result is still reasonable. Another example is the first stroke in Figure 5.11, which only partially matches the normal face contour, but our approach still succeeds in capturing this information.

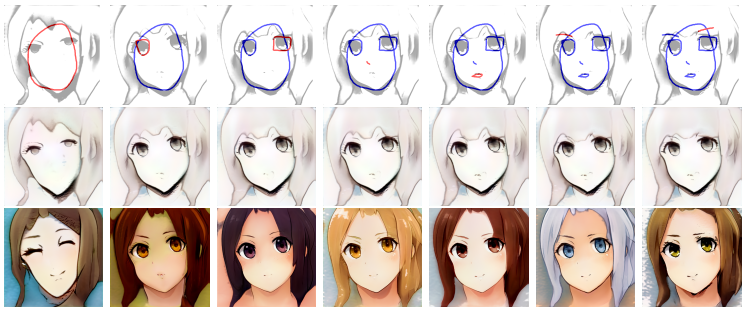
5.3.4.3 Qualitative Results

It is found that there is no S2I synthetic technique for anime portraits. Therefore, this work trains an additional sketch encoder for the complete sketch using a random cropping strategy as a baseline for a fair comparison. Except for the training strategy, the hyperparameters and the architecture of the baseline network are the same as those in the sketch encoder. The comparison results are shown in Figure 5.11. It is verified that our approach can provide consistently high-quality guidance that better match the input during the sketching process.

(a) Input



(b) Ours



(c) Baseline (pSp encoder trained with complete sketches with a random crop)

Figure 5.11: Qualitative comparison with the same input sketch sequences. A red color stroke represents the last stroke in a sketch.

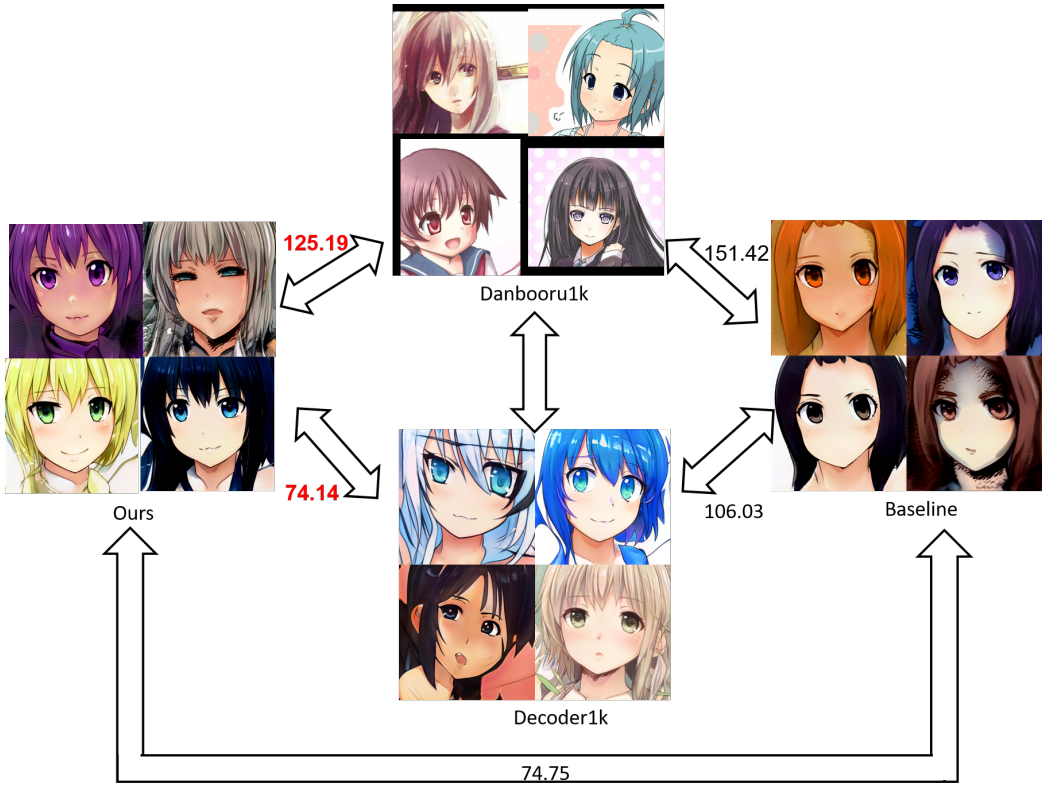


Figure 5.12: Samples from different datasets or approaches and the FID with each other.

5.3.4.4 Quantitative Results

Our approach is evaluated from two aspects: the quality of generated images, and the match between input and output. The quality of image generation affects both the quality of the guidance received by the user and the evaluation of the final generated result, so it is necessary for this indicator to be measured quantitatively in addition to the subjective evaluation of the user. For similar reasons, the match between the input sketch and the guide also needs to be measured quantitatively to ensure that the validity of our approach is subjectively and objectively consistent. To evaluate usability and satisfaction, a user study is conducted which is described in Section 5.5 for the overall system.

Quality of generated images. Unlike normal S2I synthesis, this work is dedicated to the stability of matching rough sketches and intermediate results throughout the drawing process. To evaluate the matching degree between strokes and hints during the drawing process for each stroke, this work uses FID to measure the gap between the generated images: first, users are asked to draw 10 sketches

Table 5.1: FID scores of baseline and our approaches. As a reference, the FID between Decoder1k and Danbooru1k is 70.86.

DBName	Ours	Decoder1k	Danbooru1k
Baseline	74.75	106.03	151.42
Ours	-	**74.14**	**125.19**

Table 5.2: The average of different metrics from the proposed approach and from the baseline method (ours/baseline in table). At the beginning of the drawing process, the input sketches are usually more sparse, which makes it more difficult to generate matching results. Thus, the average recall of the first k strokes is more important.

Metrics \ k	1	3	6	9	∞ (whole process)
p	0.04/0.03	0.04/0.05	0.05/0.07	0.07/0.08	0.12/0.12
r	**0.48**/0.40	**0.46**/0.39	**0.45**/0.38	**0.43**/0.37	**0.39**/0.31
$F1$	0.07/0.05	0.07/0.09	0.09/0.11	0.11/0.13	0.17/0.16

and record the total 177 images generated by our method as a database "Ours", the results generated by the baseline method with the same input as a database "Baseline", one thousand randomly generated images using StyleGAN in our decoder as a database "Decoder1k", and one thousand randomly selected images from the Danbooru database as a database "Danbooru1k". The FIDs between them are shown in Table 5.1. It can be seen that, in line with the observation in qualitative results, our method generates better-quality images – similar to the images generated by Decoder in Decoder1k as well as to the real images in Danbooru1k. Figure 5.12 shows some samples from each dataset or approach mentioned above, which makes results more intuitive.

Matching of the generated image to the input sketch. To evaluate the match between the input sketch and generated guidance, sketch-guidance matching can be thought of as a prediction problem. Although neither sketch S nor the generated line drawing L is reliable enough as ground truth, the input sketches are regarded as ground truth here because what we are concerned about is how the system will cater to the input with the guidance. Then, recall is calculated as follows. Given a sketch S and the corresponding line drawing L, then the overlapping part of the two is denoted as $S \cap L$. Regard the input sketch S as the

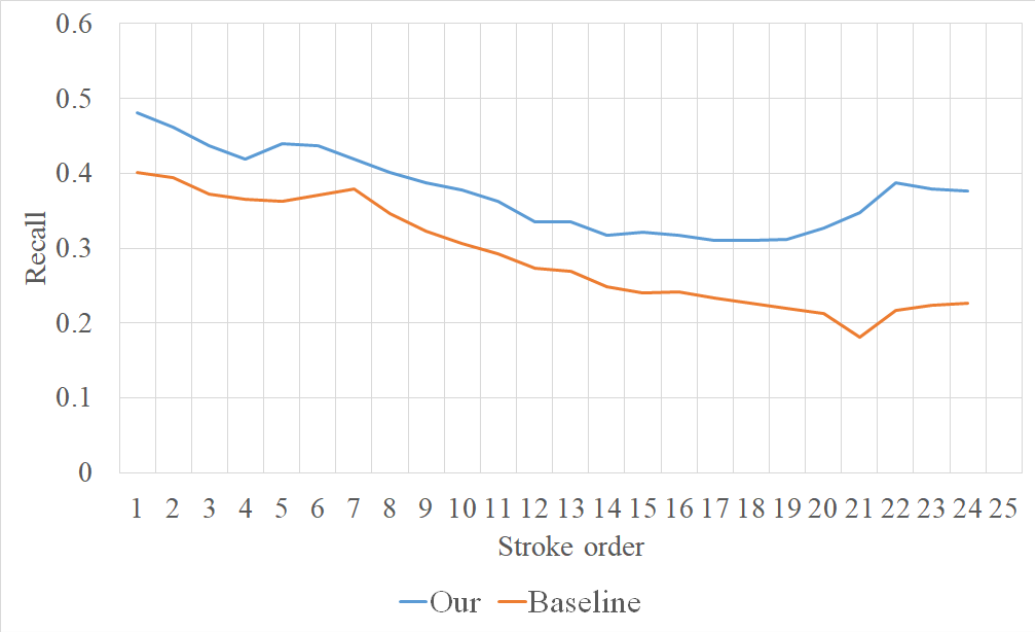


Figure 5.13: Recall comparison as strokes increase. The average recall rate of our approach was higher than the ones from the baseline method throughout.

ground truth and the recall of sketch matching r is

$$r = \frac{\text{Area}(S \cap L)}{\text{Area}(S)} \tag{5.7}$$

where function $\text{Area}[\cdot]$ counts the number of pixels of its input image. Similarly, the precision of sketch matching p is

$$p = \frac{\text{Area}(S \cap L)}{\text{Area}(L)} \tag{5.8}$$

and the $F1$ of the sketch matching $F1$

$$F1 = \frac{\text{Area}(S \cap L)}{\text{Area}(S \cup L)} \tag{5.9}$$

where $S \cup L = \text{Area}(S) + \text{Area}(L) - \text{Area}(S \cap L)$. If the output guidance is considered as the ground truth, one possibility for the low r and $F1$ in our approach may be caused by the fact that more details are generated on L according to the qualitative results in Figure 5.11.

Table 5.2 shows the results of the comparison between the proposed methods in this section and the baseline method with is described in Section 5.3, a pSp

encoder trained with random cropped sketches. It is even more important to provide high-matching guidance in the early stages of drawing when the strokes are sparse. Therefore, the average recall rate of the first 1, 3, 6, and 9 strokes and the whole sketching process is calculated. We believe that the average recall best describes the match in numerical terms because, in this evaluation metric, our results consistently outperform the results of the baseline method during the whole drawing process as shown in Figure 5.13. This result is consistent with the observation of the qualitative comparison shown in Figure 5.11, while slightly lower results of our approach in evaluation metrics p and $F1$ compared with those of baseline indicate that the method provides more details in the generation guidance. This experiment demonstrated that the guidance generated by this system can better match the input rough sketch, both at the beginning and throughout the drawing process. Based on the above results, the recall can be considered as a valid metric to measure the match between the input sketch and output guidance.

5.3.5 Discussion

This section successfully reordered the feature vectors in latent space at the stroke level by unsupervised learning with drawing process simulation. Experiments demonstrated the stability and effectiveness of the proposed approach. The experiment results show our proposed method can consistently obtain high-quality generation results during freehand sketching, independent of stroke order and "bad" strokes. As the results generated by our method are completely dependent on the decoder, i.e. pre-trained StyleGAN, the decoder in turn restricts the types of the generated images. For example, since our pre-trained model is trained on an anime portrait database extracted from Danbooru [129], the generated anime portraits are all female. Thus, how to expand the results with more styles while keeping the strokes matching will be promising for future work.

5.4 One-shot semantic lablelling in StyleGAN

As Section 5.3 has introduced portrait generation during the drawing process based on disentanglement stroke-level with the help of landmark detection technique, the last question is how to make AI automatically recognize the stroke semantics during the drawing process and give reasonable guidance for drawing support.

5.4.1 Introduction

Semantic analysis for free-hand sketching is an important research topic in visual computing and computer graphics. Especially, it has been widely applied to cross-media computing and content-based retrieval. The freehand sketch has received a lot of attention from researchers because of its high abstraction in representing scenes or objects of the real world in recent years [130]. Previous studies on freehand sketch parsing concentrated on stroke-level labelling – line segments or strokes are grouped into semantic components [131, 132]. There is a great gap between the labelling approaches of this type and the semantic segmentation approaches of real images because the previous one only requires semantic labelling on sketches according to strokes, while the latter involves a thorough annotation of all pixels from real images one by one. For this reason, those existing approaches that are effective for real image parsing can not be applied to the task of stroke-level labelling for sketches directly. This work proposed a one-shot semantic labelling method based on deep prior knowledge in StyleGAN with a single pair of image-mask which can finally generate a semantic mask for users' incomplete sketches during sketching as well as the corresponding high-quality images combining stroke-level feature manipulation from Section 5.3.

5.4.2 Framework

This section concentrate on the intelligent feedback generation part of our framework with implicit conversation in Figure 5.1. Once again, this issue is solved by the basic version of our paradigm which is introduced in Figure 1.2. The input is an anime portrait generated from StyleGAN and the output is a semantic map. The prior knowledge adopted in this section is the deep prior in StyleGAN and a manually labelled image-mask pair. Because the semantic information is fully contained by the latent space of the generated portrait image as a subset, figuring out the semantic map can be viewed as the visualization of implicit deep features.

The pipeline of the one-shot semantic labelling-based drawing support is shown in Figure 5.15. As a preparatory phase, both a GAN inversion encoder and

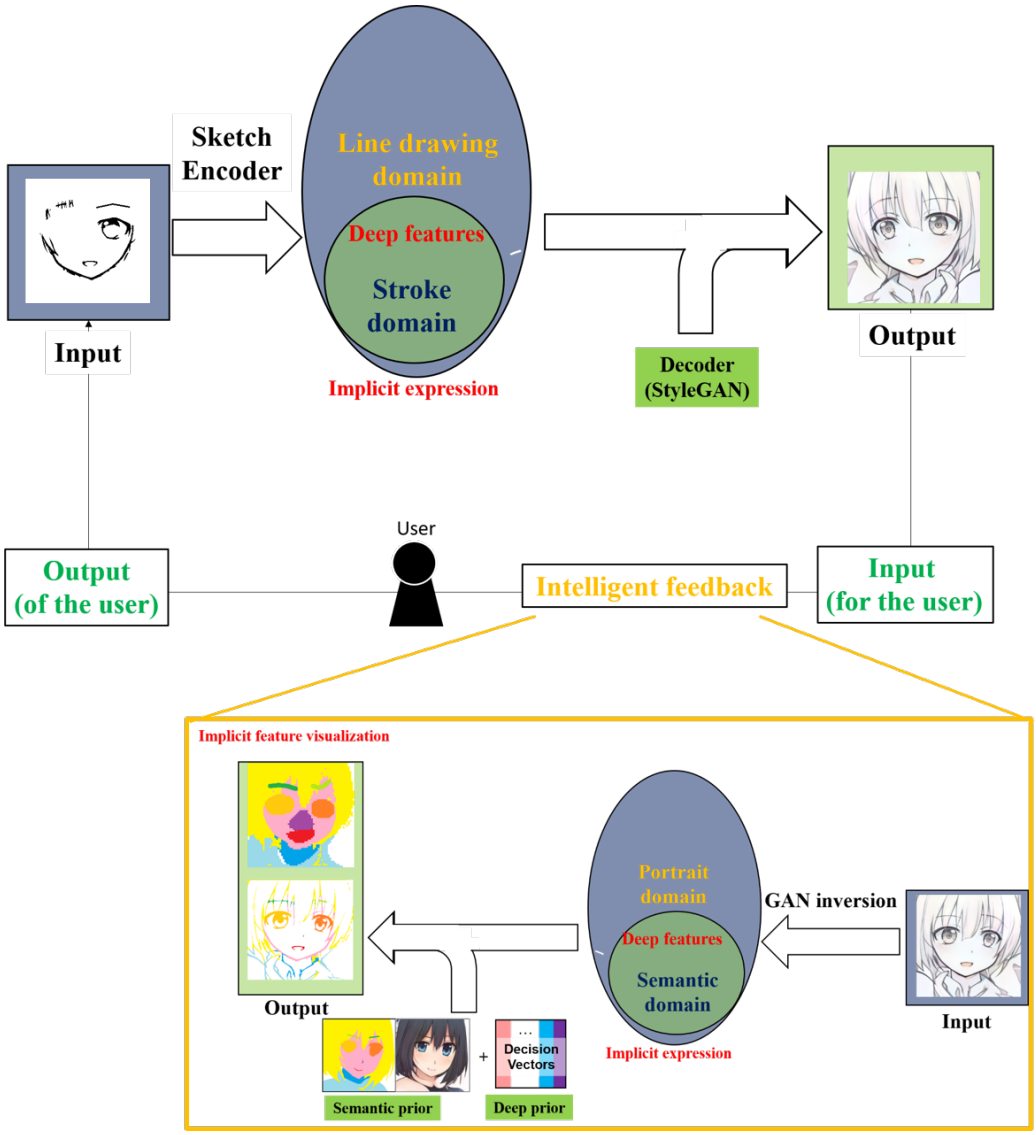


Figure 5.14: Proposed drawing support framework.

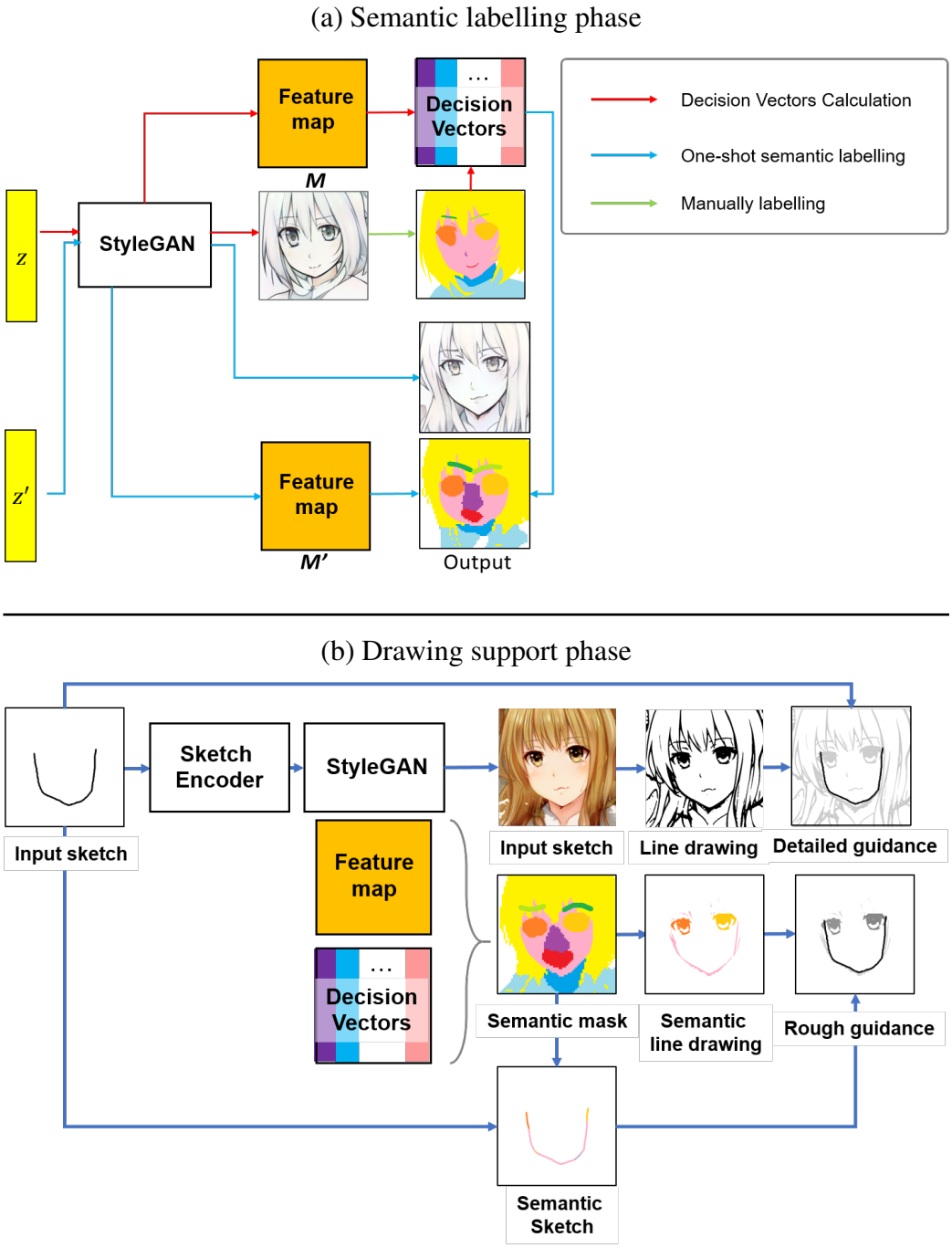


Figure 5.15: Pipeline of one-shot semantic labelling for anime drawing support. In the semantic labelling phase, a semantic mask is annotated manually for a generated image from Gaussian noise z and the decision vectors are calculated for one-shot semantic labelling. In the drawing support phase, once a stroke is added to the input sketch, users can get both rough guidance and detailed guidance at the same time and can switch between them at any time for drawing assistance.

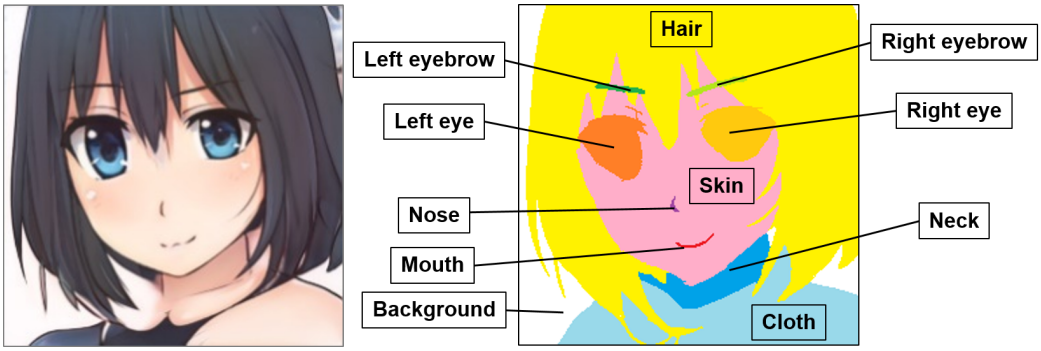


Figure 5.16: An example of input image-mask pair for one-shot semantic labelling. A generated RGB color image from StyleGAN on left is labeled with semantic annotations as a mask in right. For anime portraits, there are 12 semantic labels in total including the "unknown" label with black color which is not shown in this mask.

sketch are trained following the training steps which are described in Section 5.3.

Then, in the semantic labelling phase, generated images from StyleGAN (decoder) are annotated automatically by the proposed prior knowledge-based semantic labelling approach and these image-mask pairs. The structural codes derived from the sketch encoder are concatenated with the color codes from random Gaussian noise z, which is known as style-mixing. Note once the decoder and one image-mask pair have been determined, all data is derived from the randomly generated images from this decoder and no additional assistant database is required. Thus, this is a one-shot learning-based approach. In the drawing support phase at last, with strokes increased during users' sketching, the system consistently delivers high-quality images and converts these images to line drawings which are displayed as local guidance based on semantic mask prediction.

The training for the GAN inversion encoder is similar to the previous work [75]. The only difference is the loss function – simply L2 loss is adopted, between the original images from StyleGAN and the reconstructed images encoded in this GAN inversion encoder. The training for the sketch encoder is similar to Section 5.3. In the following Section 5.4.3, the proposed one-shot semantic labelling is introduced.

5.4.3 One-shot Semantic Labelling Correspondence based on Feature Matching

5.4.3.1 Formal Statement

Given a Gaussian noise $\boldsymbol{z}$, the generated image of StyleGAN generator is $\boldsymbol{x} = G(\boldsymbol{z})$ and the output latent code L_z is then defined as

$$L_z := E_{inv}(G(\boldsymbol{z})) \tag{5.10}$$

where $E_{inv}(\cdot)$ and $G(\cdot)$ denote the image encoder for StyleGAN inversion and StyleGAN generator, respectively.

In the forward propagation process during $G(\cdot)$ operation, $M \in \mathbb{R}^{N_c \times W \times H}$ is returned as feature maps, where N_c, W, and H are the number of channels, width, and height of M respectively. Note that M has been scaled to the same width W and height H of the input color image $\boldsymbol{x} \in \mathbb{R}^{3 \times W \times H}$ with the nearest neighbor method for the convenience of calculation. Then, for the above-mentioned generated image $\boldsymbol{x}$, labels with C classes (including "unknown" class) of user-defined semantics are annotated as a reference mask $\boldsymbol{m} \in R^{C \times W \times H}$. As a pair of image-mask $\{\boldsymbol{x}_K, \boldsymbol{M}_K\}$ sample is given as prior knowledge, the question is how to calculate the masks of another unlabelled generated image $\boldsymbol{x}' = G(z')$ from random noise $\boldsymbol{z}'$. Especially, Figure 5.16 shows an example of the generated image of an anime face used in this work and its corresponding semantic mask labeled with 12 class annotations for portrait drawing process simulation.

Inspired by few-shot semantic segmentation [133] and one-shot learning for StyleGAN controlling [134], the proposed one-shot semantic labelling consists of two types of methods – region-based semantic labelling and pixel-based semantic labelling.

5.4.3.2 Region-based Semantic Labelling

The result of the region-based semantic labelling method is one or several connected regions. Therefore this method is suitable for semantic labels that do not intersect much with other semantics, such as eyes, nose, and mouth in a portrait.

First, StyleGAN's feature maps $\boldsymbol{M}$ are extracted corresponding to the semantic masks from the given image-mask pair $\{\boldsymbol{x}_K, \boldsymbol{M}_K\}$ as prior knowledge. Then, decision vectors $\boldsymbol{v}_c = \{\boldsymbol{v}_c^1, \boldsymbol{v}_c^2, ..., \boldsymbol{v}_c^{\mathrm{len}(\boldsymbol{v}_c)}\}$ (function $\mathrm{len}(.)$ is used to calculate the length of vectors) for each semantic class c from the pairs of extracted feature maps and semantic masks, following the approach from Wang et al. [133].

$$\boldsymbol{v}_c = \frac{\sum_{x,y} \boldsymbol{M}^{x,y} \mathbb{D}\left[\boldsymbol{M}_K{}^{(c,x,y)} == 1\right]}{\sum_{x,y} \mathbb{D}\left[\boldsymbol{M}_K{}^{(c,x,y)} == 1\right]} \tag{5.11}$$

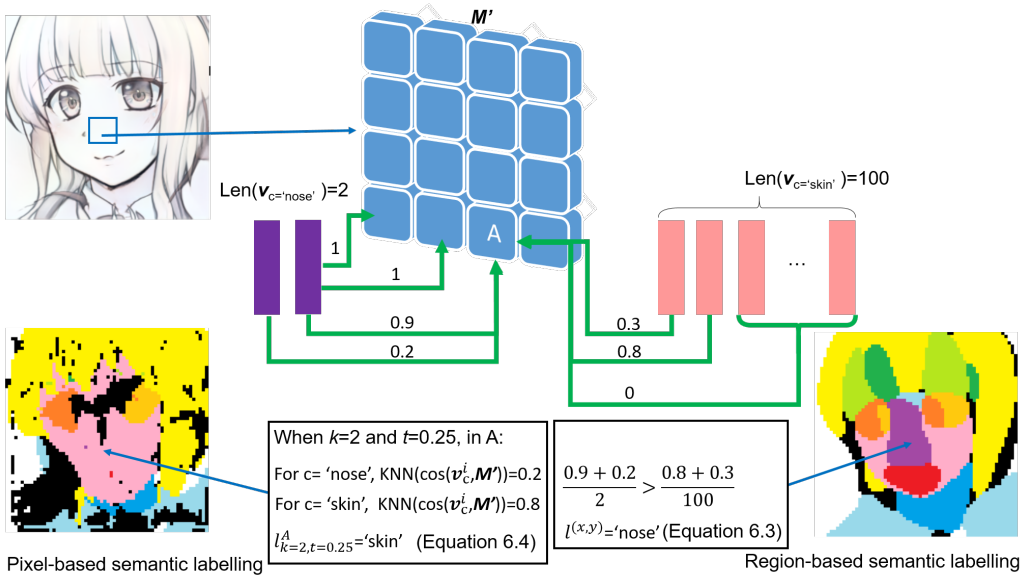


Figure 5.17: Illustrations for region-based semantic labelling method and pixel-based method. Note that in location A, the final value for $c = \text{'}nose\text{'}$ is 0 in the pixel-based method, because $t = 0.25 > 0.2$.

here (x, y) denotes pixel position in $\boldsymbol{x}$, and $\mathbb{D}[\cdot]$ is the indicator function that returns 0 if the argument is false and 1 otherwise. Once the decision vectors $\{\boldsymbol{v}_c\}_{c \in C}$ for all semantic labels are calculated, another unlabeled images $\boldsymbol{x'}$ sampled from a random Gaussian noise $\boldsymbol{z'}$ with the extracted feature maps M' in StyleGAN can be annotated via feature matching between v_c and corresponding pixel-wise feature vectors $\{\boldsymbol{M'}^{(x,y)}\}_{x \in W, y \in H}$ in M'. Semantic label $l^{(x,y)}$ of a pixel positioned in (x, y) is determined with cosine distance as follows:

$$l^{(x,y)} = \arg\max_{c \in C} \text{means}_{i \in \boldsymbol{n}_c}(\cos(\boldsymbol{v}^i{}_c, \boldsymbol{M'}^{(x,y)})) \tag{5.12}$$

where $\text{means}_{i \in \boldsymbol{n}_c}(\cdot)$ is the function to calculate the average value of metric distances in class c and $\boldsymbol{n}_c = \{1, 2, ..., \text{len}(\boldsymbol{n}_c)\}$ is the index set of decision vector in class c. The reason for choosing the cosine distance as the metric distance here is based on the finding from Collins et al. [135] that feature vectors in StyleGAN with the same semantics tend to cluster on a unit sphere.

5.4.3.3 Pixel-based Semantic Labelling

For semantic labels that intersect much with other semantics such as hair and eyebrows, and region-based semantic labelling, the results of semantic annotation

will be more sensitive – wrong recognition of semantic labels can seriously affect the subsequent stroke segmentation. Thus, pixel-based semantic labelling is necessary for these semantic parts.

The only difference between pixel-based semantic labelling and region-based one is the metric function. $\mathrm{KNN}(\cdot)$, a K-nearest-neighbors function is adopted, instead of $\mathrm{means}(\cdot)$ in Equation (5.12).After obtaining decision vectors $\{\boldsymbol{v}_c\}_{c\in\boldsymbol{C}}$ for all classes C with Equation (5.11), Semantic label $l^{(x,y)}$ of a pixel positioned in (x, y) with hype-parameters $K = k$ for K-nearest-neighbors function and $T = t$ as low confidence threshold is determined as follows:

$$l_{k,t}^{(x,y)} = \arg\max_{c\in C} \mathrm{KNN}_{K=k,i\in\boldsymbol{n}_c}(cos(\boldsymbol{v}^i{}_c, \boldsymbol{M'}^{(x,y)}) \cdot \mathbb{D}(cos(\boldsymbol{v}^i{}_c, \boldsymbol{M'}^{(x,y)}) > t)) \tag{5.13}$$

The advantage of adopting the K-nearest-neighbors function is that it increases the confidence of sparse labels and reduces overfitting instead of being smoothed by $\mathrm{means}(\cdot)$. According to Equation (5.11), the number of decision vectors $\boldsymbol{v}_c$ in class c is positively related to the area occupied by this semantics in the mask, which makes the $\mathrm{means}(\cdot)$ strategy in Equation (5.12) bias sparse semantics in semantic labelling.

This is illustrated by the comparison of the two semantic annotation methods in Figure 5.17. It is known that there is an intersection between the nose and the facial skin, i.e., the nose is on top of the skin. Assume that for the feature map $\boldsymbol{M}'$, the number of decision vectors for the nose is 2, and the number of feature vectors for the skin is 100. For a pixel with location A near the nose in $\boldsymbol{M}'$, if the cosine distance from each decision vector to A is as shown by the blue arrow in the figure, region-based semantic labelling taking the mean value will decide that this pixel belongs to the nose, while pixel-based semantic labelling with $\mathrm{KNN}(\cdot)$ will decide that this pixel belongs to the skin.

5.4.3.4 Prior Knowledge-based Semantic Mask Optimization for Portraits

The above two annotation methods have their own strengths and weaknesses and have different results for different semantics. Take Figure 5.17 as an example, in the semantics of portrait drawing, there is more crossover of eyebrows and hair, so it is easy to overfit with region-based semantic labelling method in eyebrows' labels. Besides, parts that appear in pairs such as eyes and eyebrows are also prone to recognition errors due to their owning similar features. Therefore, once a paired image-mask data $\{\boldsymbol{x}_K, \boldsymbol{M}_K\}$ is given as the prior knowledge, the annotation method applied to each part is decided and whether some of these parts require additional processing is also defined respectively.

The proposed method is described in Algorithm 5.2. Firstly, Algorithm 5.2 defines 3 types for each label i.e. "pair", "line-like" and "normal" in $\mathrm{ProcessTypes}$

as well as their corresponding process functions in $\mathrm{ProcessFuction}$. As a pixel-based semantic labelled mask, $\boldsymbol{M}_p$ and region-based semantic labelled mask $\boldsymbol{M}_r$ have been calculated with the above-mentioned one-shot method, functions TypeDetection finds out a set of types $T \subseteq \mathrm{ProcessTypes}$ for each label c with following objective function:

$$T = \underset{t \in T, c \in \boldsymbol{M}_K, q \in \{r,p\}}{\arg\min} \mathrm{SM_c}(\mathrm{ProcessFuction}_t(\boldsymbol{M}_q), \boldsymbol{M}_K) \tag{5.14}$$

where moment invariants [136] is adopted for shape match function $\mathrm{SM}(\cdot,\cdot)$. This objective function means to find out the most suitable type set T which is most similar in shape to the known labelled mask $\boldsymbol{M}_K$ for each label class $c \in \boldsymbol{M}_K$.

Then, the most related label d to a label with index c is found with the following objective function

$$d = \underset{c,d \in \boldsymbol{M}_K, \mathrm{CD}(\boldsymbol{M}_r[c], \boldsymbol{M}_r[d])=1}{\arg\max} L_2(\mathrm{CD}(\boldsymbol{M}_r[c], \boldsymbol{M}_r[d]), \mathrm{CD}(\boldsymbol{M}_K[d], \boldsymbol{M}_K[d])) \tag{5.15}$$

where, $\boldsymbol{M}_r[\cdot]$ and $\boldsymbol{M}_K[\cdot]$ are considered as a point set, and function $\mathrm{CD}(\cdot,\cdot)$ is a distance function to calculate minimal distance between two point sets. As points in $\boldsymbol{M}_K$ are pixel coordinates originally, the minimal of between two point sets $\boldsymbol{M}_K[c]$ and $\boldsymbol{M}_K[d]$ is 1 in case of a label c and another d are neighbours in $\boldsymbol{M}_K$. Then, Equation (5.15) means if the label c and the label d are neighbours in $\boldsymbol{M}_K$ while not in $\boldsymbol{M}_r$, If the label c and the label d are neighbours in $\boldsymbol{M}[c]_r$, the most differentiated adjacency represents the most likely overfit between two labels c and d. As a result, Figure 5.18 illustrates the benefits of the proposed method – the mask is calculated based on this prior knowledge to obtain relatively accurate segmentation results in stroke-level with Algorithm 5.2. On the middle row of Figure 5.18, landmarks of the input anime face of the first row are detected and overlayed on the corresponding images at the top row with the anime-face detector [128] which has been mentioned in Section 5.3. It is obvious that the number of landmarks located in the correct semantic region with the proposed method is the highest, which proves the effectiveness of this approach.

5.4.4 Experiments and Results

5.4.4.1 Implementation Details

In the implementation of this section, the sketch encoder in Figure 5.15 adopted the pSp architecture [75]. In the pixel-based semantic labelling method, $k = 3$ and $t = 0.5$ is set. These two hyper-parameters were selected manually based on the results of Figure 5.19, which can preserve the sparse semantics of the annotation very well. Ranger optimizer is adopted with a learning rate of 0.0001

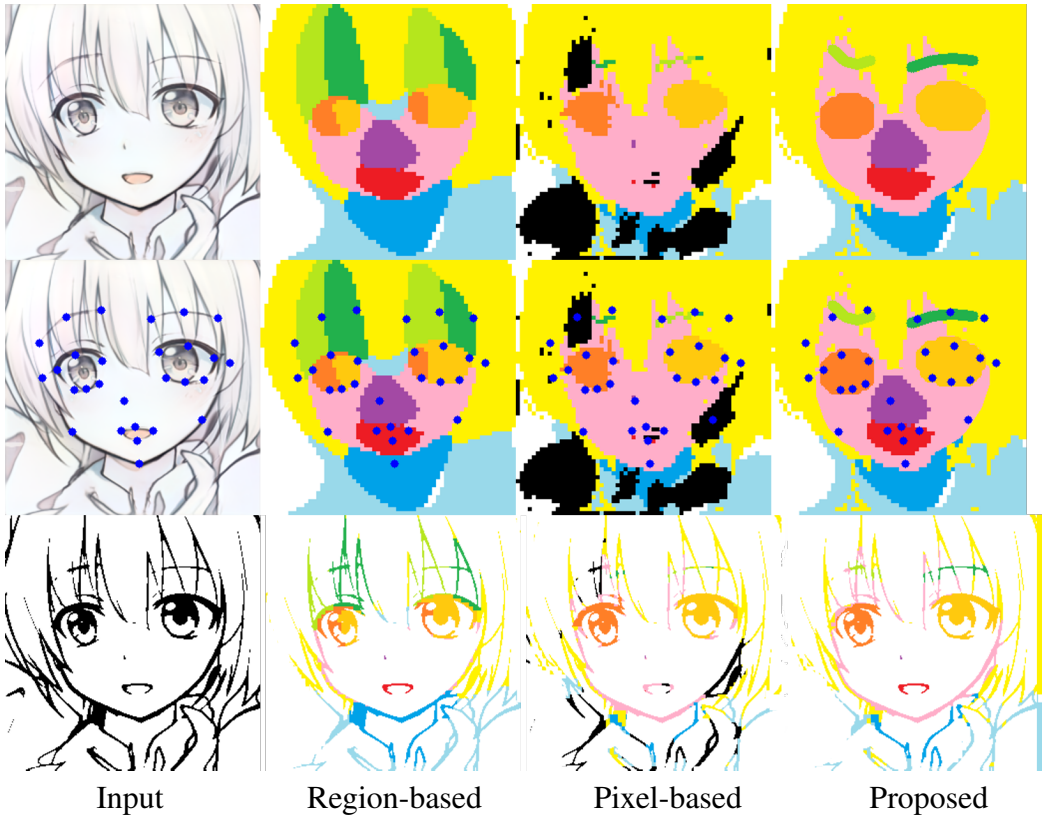


Figure 5.18: Comparison of region-based semantic labelling method, pixel-based one, and the proposed prior knowledge-based optimization. The top row is labelled semantic masks from the input image and the bottom row is a line drawing of the input image segmented by the corresponding masks on the top row. Also, semantic masks with facial landmarks are shown in the middle row to show the effectiveness of the proposed method.

Algorithm 5.2 Prior knowledge-based semantic mask optimization for portrait images

Input: Portrait image $\boldsymbol{x}$, Region-based semantic labelled mask $\boldsymbol{M}_r$, Pixel-based semantic labelled mask $\boldsymbol{M}_p$, Known semantic label mask $\boldsymbol{M}_K$ (prior knowledge)

Output: Semantic mask of portrait $\boldsymbol{M}_{out}$

Region-based semantic label masks of portrait $R \leftarrow$ Onehot($\boldsymbol{M}_r$)

Pixel-based semantic label masks of portrait $Pix \leftarrow$ Onehot($\boldsymbol{M}_p$)

ProcessTypes=[Pair; Line-like; Normal]

ProcessFuction=[$\mathrm{Kmeans}_{K=2}$; Polynomial Fitting; Contour Detection;]

$S \leftarrow \emptyset$

$\boldsymbol{M}_{out} \leftarrow \emptyset$

for *c=1:n* **do**

  Label $s \leftarrow \emptyset$

  One-hot label mask $m_r \leftarrow R[c]$

  One-hot label mask $m_p \leftarrow Pix[c]$

  Known one-hot label mask $m_k \leftarrow M_K[i]$

  The most related label d to c, $d \leftarrow -1$

  Types T, d $\leftarrow$ TypeDetection(m_k, m_r, m_p)

  foreach(Type t in T)

    if($d > -1$) **then**

      $s \leftarrow$ ProcessFuction($m_r, m_p, R[d], Pix[d]$)

    else

      $s \leftarrow$ ProcessFuction(m_r, m_p)

  $\boldsymbol{M}_{out} \leftarrow$ DrawLabel(s)

end

return $\boldsymbol{M}_{out}$

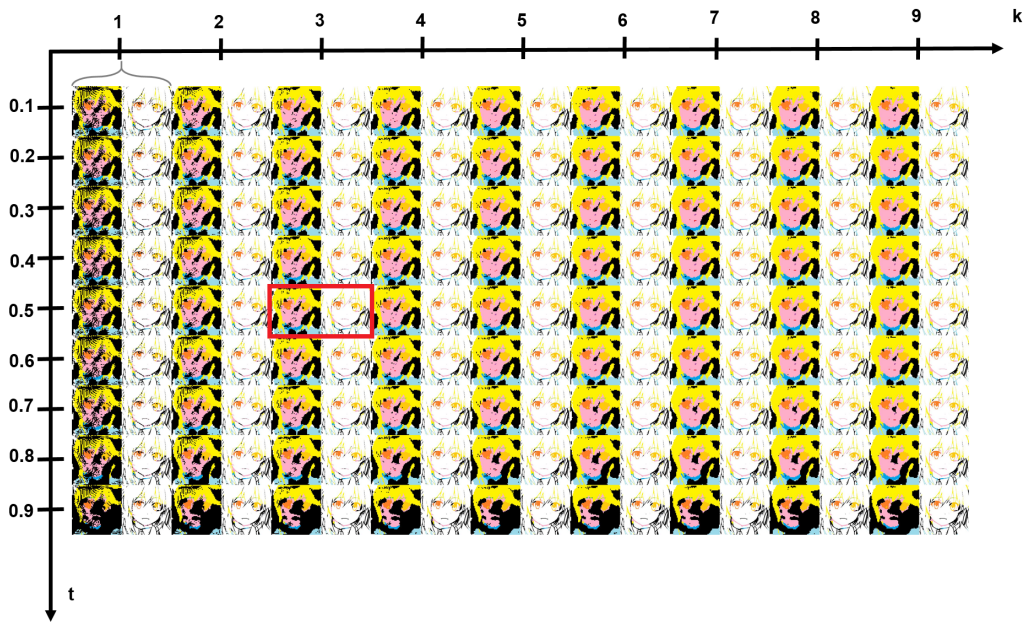


Figure 5.19: Effect of different parameters in pixel-based semantic labelling on the results. According to the degree of preservation of sparse semantics such as nose and eyebrows, this work chose $k = 3$ and $t = 0.5$ for Equation (5.13).

for training. As a training environment, NVIDIA RTX3090 GPU was used to train our encoders which are programmed in Python on the Linux platform. Then, a workstation with Intel Core i7 8700, 3.20GHz/3.19GHz, NVIDIA RTX1070 GPU, and 64GB RAM on the windows 10 platform was used as the testing environment.

5.4.4.2 Qualitative and Quantitative Results

The stability between the sketch and the generated results is described in Section 5.3. Therefore, this chapter measures the reasonableness of the system guidance from sketch. To evaluate the matching degree between strokes and guidance during the drawing process for each stroke, this work exhibits visual results for a whole drawing process. As shown in Figure 5.20, the proposed system supports the whole process of sketching well and provides high-quality drawing guidance continuously for both anime-style and realistic style drawing. At the same time, the system can accurately identify the semantics of the strokes and provide the corresponding semantic segmentation of line drawing results in the detailed guidance and show reasonable prediction results in the rough guidance. Both the rough guidance and detailed guidance provided by AniFace are visually similar to the rough guidance and detailed guidance in RealFace, which reflects the effectiveness of the one-shot learning method in this section. Also, we evaluate the usefulness quantitatively with the following user study, as colored rough guidance and detailed guidance which can provide intelligent feedback to users.

What's more, as a quantitative results, the number of landmarks located in the right semantic region, which has been shown in Figure 5.18, is calculated to evaluate the matching degree between guidance and semantic mask. Note that there are 28 facial landmarks in total. Using the recorded 10 sketch processes that we have obtained in Section 5.3.4.4, we can calculate 177 image-mask pairs for evaluation. The average of this evaluation metric is 15.29 (54.61%) with the proposed one-shot approach, 8.11 (28.96%) with the region-based semantic labelling method, and 8.97 (32.04%) with the pixel-based semantic labelling method. Thus, it can be clearly demonstrated that, compared with these two methods, our method has a great improvement in this evaluation metric and obtains a more accurate semantic mask.

5.5 User Study

To verify the effectiveness of our drawing assistance system for both anime-style and real-face style, this work invited 15 participants (graduate students)

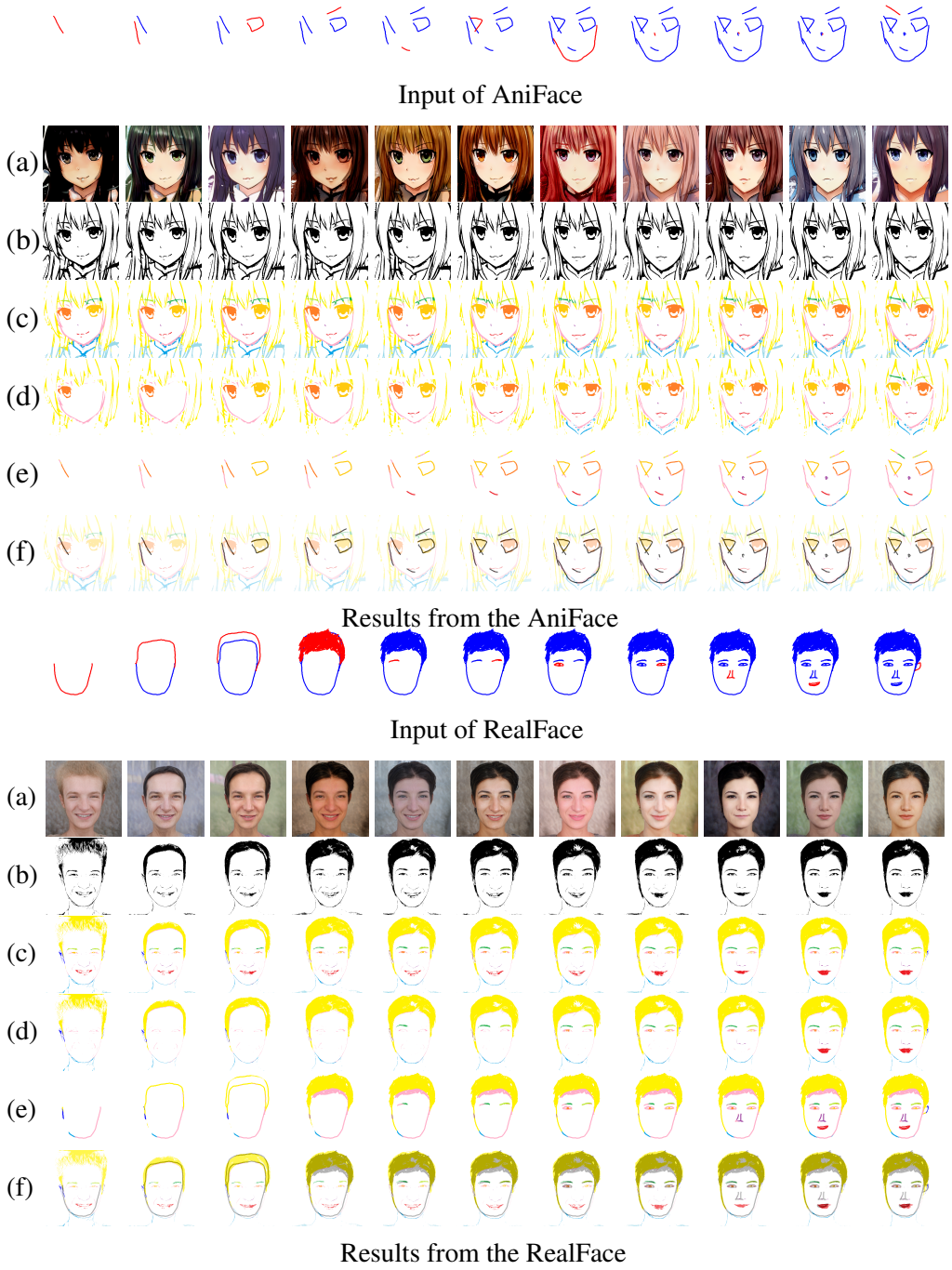


Figure 5.20: Qualitative results with input sketch sequences. A red color stroke represents the last stroke in a sketch. From the top to the bottom row, the corresponding results from the proposed system with the input sketches are (a) generated images (with random style-mixing), (b) line drawings from generated images (c) rough guidance, (d) detailed guidance, (e) semantic segmented stroke, and (f) stroke with detailed guidance.

to this user study. All participants were asked to draw realistic portraits and anime-style portraits by remote control of the mouse online. One of them, P0 only participated in the anime drawing assistance according to his/her personal preference. All participants were asked to draw portraits freely and aimlessly and try to draw more details as possible as they can. For each system, they did drawing creation twice: the first time to experience the whole process of drawing creation to familiarize themselves with the operation until they got used to this system and felt comfortable; The following time, the participants completed the whole process independently and took part in the user study. This work instructed all participants on how to use AniFace and RealFace with user manuals. Before the practical operation, they were asked to watch the corresponding tutorial video. All participants were required to draw carefully and choose the most anticipated references for local guidance from multiple generated candidates after they completed the global stage. When the generated guidance meets your wishes and expectations, participants were required to press the "Pin" button to draw carefully for refining the input sketch. Participants can select a reference image for color portrait generation at any time during their drawing until they are satisfied with the results. Finally, this work administered the questionnaire to all participants after they finished their second drawing creation for each style.

5.5.1 Design of questionnaire

Our questionnaire consists of three parts: System Usability Scale (SUS) [137], creativity-support index (CSI) [138], and a series of customized questions shown in Table 5.4 to investigate the relationship between user satisfaction and guidance matching.

In SUS, ten questionnaire items are set up to capture subjective evaluations of the system's usability. A five-point Likert scale is used in the evaluation experiment. SUS is easy to investigate from a wide range of users and the result can be reliable even with a small sample size.

Since the purpose of this work is to support user drawing creativity, CSI is used to quantitatively evaluate the effectiveness of the proposed method. The CSI score defines the creativity of the tool with six factors: Collaboration, Enjoyment, Exploration, Expressiveness, Immersion, ResultsWorthEffort, and is scored with a maximum of 100 points. Here, the factor "Collaboration" is set to 0 (not applicable) as a slash symbol because there is no collaboration with other users in our task – users should complete the art drawing independently.

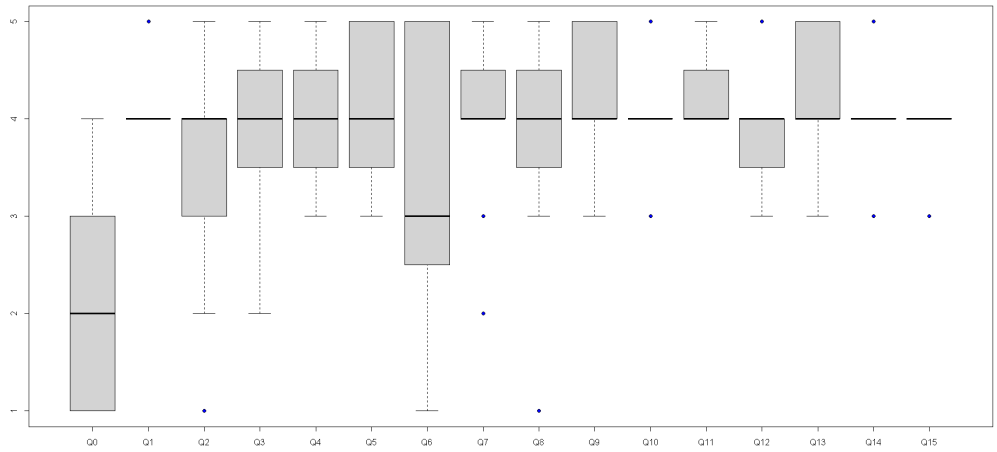
5.5.2 Results

For the similar reason as Section 4.4 mentioned, the difficulty of objectively quantifying the response function $R(\cdot)$ contained in Equation (1.9) implicitly makes this work adopt user studies to indirectly demonstrate the validity of our system, with visual results, usefulness, and satisfaction. This section discusses the implementation results of both AniFace and RealFace, evaluation results, user feedback from the user study. Here, Table 5.4 shows the mean and SD of each question in our customized questionnaire, while Figure 5.21 shows corresponding boxplots to these questions.

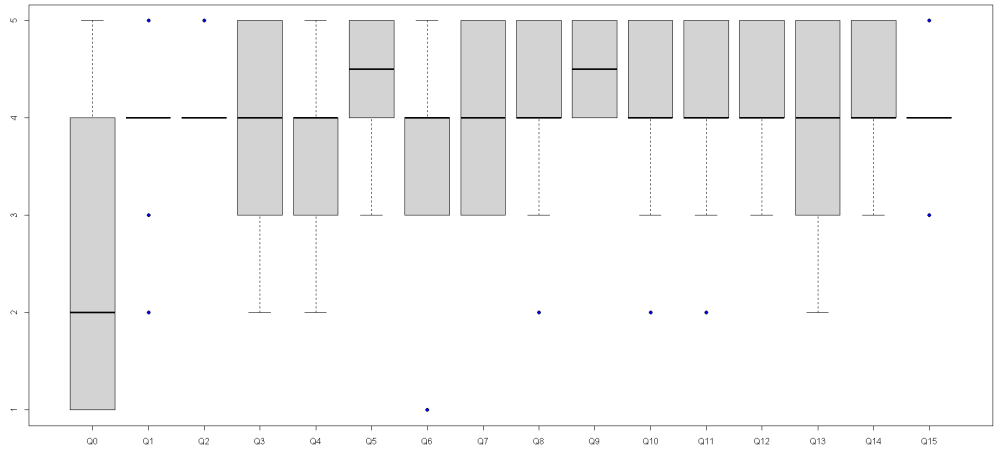
Visual Results. Figure 5.22 shows some examples of our implementation results in our user study. Our system can successfully turn the user's rough sketches into high-quality anime portraits and realistic portraits. It is worth noting that the real-face semantic segmentation in RealFace employs an off-the-shelf pre-trained CNN model. Thus, by comparing the color guidance with semantic information in Figure 5.22 provided by the system, it is obvious that our one-shot method in Section 5.4 can obtain face segmentation results for anime-style portraits comparable to those learned through a large amount of data for realistic portraits. What's more, according to Q0 in our customized questions, 86.66% participants thought their drawing skills is not high enough (less than or equal to 3) for anime style and 71.43% for realistic style. Results in Figure 5.22 come from these novices. From these results, it can be considered that even beginners can make reasonable sketches with the assistance of the system and end up with high-quality color art drawings.

System Usability. The average score on SUS drawing assistance for anime style was 73.84 (SD=20.04). The upper and lower limits of SUS were 90 and 65, respectively. For realistic style drawing assistance, the SUS score on average was 77.80 (SD=11.24), ranging from 65.00 to 97.50. The usability of our drawing assistance system could be considered as "good" for both anime style and realistic style. For the usability of our system, "Overall it's a good tool for those like me who don't have much drawing skills, and it's easy to use in terms of guidance generation and color selection" and "I was not familiar with the operation when I first experimented, but I can get an amazing generated result in the second experiment. If I were shown the second generated image, I would not be able to distinguish whether the person who is real or not" is commented.

Creative Support Capability. As Table 5.3 shown, the average scores on CSI for both anime style and realistic style are 77.69 and 79.07, which means that although there is still room for improvement in terms of immersion and expressiveness, the system possesses effective creative support capabilities for art drawing. In other words, from the user's perspective, our system is considered to

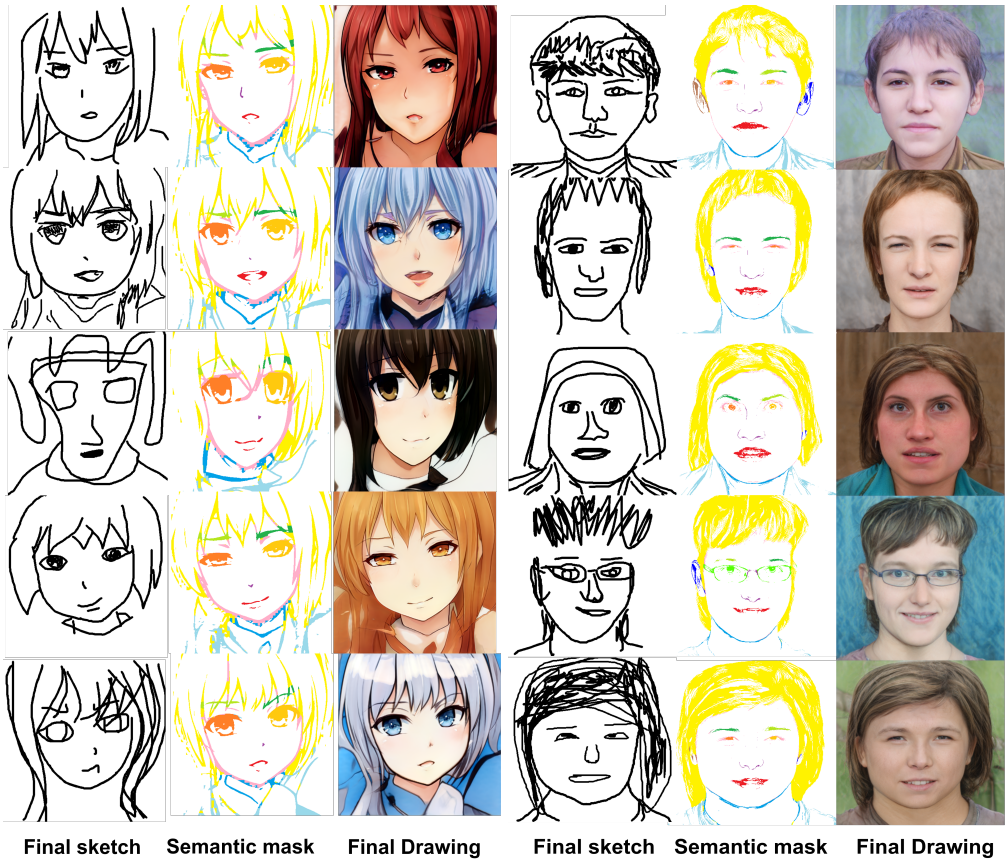


AniFace



RealFace

Figure 5.21: Boxplots of customized questions in our user study for bot AniFace and RealFace. The questions Q0 to Q15 are corresponding to those in Table. 5.4.



AniFace RealFace

Figure 5.22: Visual results from our user study for both realistic style and anime style. From the left to right column, they are the final user sketches, semantic guidance in detail mode, and the final coloring images created by users according to their free-hand sketches and user-selected reference images.

provide “good” creative support for art creation.

Time cost. After the user draws a stroke, RealFace gives an average response time of 1.71 seconds for the guidance generation while the average response time from AniFace is 1.65s. This is not short for the user. Someone said, ”One small problem is the not-so-short wait time after each stroke is completed”. This also affects the immersion score of CSI in Table 5.3 to some extent. In order to improve the user experience, the calculation time needs to be further reduced. Although there is no time limit in our creation experiments, the average time for users to complete a drawing is about 9 minutes for both AniFace and RealFace.

User-perception match degree. According to the results from Q1 to Q7 Table 5.3, the average scores on these items range from 3.40 to 4.43, which illustrates that our AI-assisted drawing system can output relatively matching guidance to the sketches input during the drawing process for both styles. In these questions, A score of five indicates “complete match” and 1 point means “complete mismatch”. Although the statistics showed that the input and output matched relatively well, Users dispute whether the hair in the input sketch and the generated image match. The proponents commented, ”(In RealFace) I think the hand-drawn drafts fit the real face very well, especially the hair and facial details, such as eye sockets and cheek creases. ” and “The drawing assistance system performs better on hair and eyes, and can match well with the drawing person’s draft to generate (anime portrait)”. Critics said, “I tried to draw a double ponytail character, but couldn’t achieve it” and “Some special hairstyles cannot be generated by RealFace”. The main reason for this phenomenon is that the stroke-level disentanglement for both anime style and realistic style is focused on facial contour features in the training step restricted by the adopted prior knowledge. Even so, most participants still tended to think that the sketch-hairs match is positive, which shows the generalization capability of the proposed system at a certain level.

User-perception quality. According to the results from Q10 to Q12 Table 5.3, users believe that the system consistently produces high-quality and reasonably balanced facial guidance throughout the drawing process in both art-style and realistic drawing assistance. This result is consistent with the results of the qualitative experiments in Figure 5.20 and Figure 5.11, and they corroborate each other.

User satisfaction for guidance mode. As a user evaluation of the intelligent feedback, the comparison between AniFace and RealFace of Q11 and Q12 illustrates that the anime-style portrait generated using our semantic labelling described in Section 3.3 achieves almost the same effect as the real human face semantic segmentation with CNN, which is a powerful proof of the effectiveness of this one-shot method. There are comments such as “In detailed mode I had to focus on the generated auxiliary lines, thus unconsciously following the auxiliary, while

Table 5.3: CSI Questionnaire results in the user study.

Terms	AniFace		RealFace	
	Mean	SD	Mean	SD
Collaboration	-	-	-	-
Enjoyment	27.93	10.09	27.96	10.25
Exploration	29.11	11.27	28.79	10.65
Expressiveness	23.50	6.30	23.79	6.72
Immersion	13.46	10.86	15.50	13.19
ResultsWorthEffort	22.54	11.65	22.57	11.40
CSI Score	77.69		79.07	

rough mode alleviated this problem to some degree" and "As a beginner, I think this system is very helpful for beginners, allowing one to get a good result and enjoy the drawing without having many professional skills.".

User satisfaction for guidance. Results from Q10 to Q12 show that users generally agree that our system provides good support for the creation of both anime style and realistic portraits, improving both the user's own sketches and generating a desirable final color image according to their expectations. Taking Q1 to Q7 into consideration, the consistency of these scores illustrates that our approach, which can be regarded as an implicit strategy for our paradigm, achieves the optimal matching between the sketch input and the guidance output so that users' satisfaction reaches the threshold of Figure 1.1 and successfully converts the unknown user-perception evaluation function optimization problem into a solvable AI-User conversation.

5.6 Summary

In this chapter, an AI-assisted drawing system with an implicit conversation strategy was implemented as a stroke-level disentanglement in StyleGAN according to the proposed paradigm in Chapter 1. this comprehension-based drawing assistance system proposed can well analyze the semantics of the sketch when the user is drawing and provide corresponding high-quality guidance. Users can switch between global and local guidance at any time according to their needs, which eventually helps them to draw the desired portraits and extend their drawing skills. The most significant aspect of this system is that the proposed one-shot semantic annotation method for anime-style portraits, relying heavily on feature information within the pre-trained StyleGAN as the deep prior knowledge, demonstrates that deep AI itself contains semantic information about image generation. This one-shot approach achieves comparable results in our

Table 5.4: Customized questionnaire results in the user study.

Question	AniFace		RealFace	
	Mean	SD	Mean	SD
Q0: How do you think your drawing skills for anime/real faces are? (with a mouse)	2.07	1.10	2.43	1.40
Q1: How well does the shadow guidance match your sketch overall?	4.07	0.26	3.93	0.73
Q2: How well does the shadow guidance match your sketch when drawing the mouth?	3.40	1.12	4.21	0.43
Q3: How well does the shadow guidance match your sketch when drawing the left eye?	3.93	0.88	3.86	0.95
Q4: How well does the shadow guidance match your sketch when drawing the right eye?	4.00	0.76	3.79	0.89
Q5: How well does the shadow guidance match your sketch when drawing the nose?	4.07	0.80	4.43	0.65
Q6: How well does the shadow guidance match your sketch when drawing the hairs?	3.47	1.51	3.43	1.16
Q7: How well does the shadow guidance match your sketch when drawing the face contour?	4.07	0.80	4.00	0.78
Q8: For your sketch and shadow guidance, which facial balance is more reasonable?	3.87	1.06	4.07	0.83
Q9: What is the quality of the guidance	4.27	0.59	4.50	0.52
Q10: Does the shadow guidance maintain high quality in your sketching process?	4.00	0.53	4.14	0.86
Q11: Is rough semantics guidance mode helpful for your drawing?	4.27	0.46	4.07	0.83
Q12: Is detailed semantics guidance mode helpful for your drawing?	3.93	0.70	4.29	0.61
Q13: Are you satisfied with the final coloring results?	4.13	0.74	3.86	0.95
Q14: Does the guidance follow your will?	3.93	0.59	4.14	0.66
Q15: Are you satisfied with the final sketch result?	3.87	0.35	4.14	0.53

user study to the semantic segmentation network in RealFace, which has been trained with large amounts of paired data. Combining with a reasonable approach, such as disentanglement learning at stroke level in Section 5.3, these promising information can be converted into a human-understandable form and used as an aid to help humans perform various tasks. What's more, user studies have proven the effectiveness and generalization capability of our intelligent system - it supports not only free sketch-based realistic portrait creation but also anime face creation which is more abstract and challenging. As a limitation, despite the multi-faceted measurement for our drawing assistance systems, a more intuitive quantitative indicator of the user-perception exception function is still lacking. In future work, a metric that can measure user exception online deserves to be explored as a valid alternative to user satisfaction that can also be used to guide AI learning.

Chapter 6

Conclusion

6.1 Summary

Unlike the traditional use of AI to generate the final results directly, this dissertation explores AI's understanding and interpretation of the generation process. Several sketch-based AI interactive creation assistance techniques are proposed which are applied to art drawing. The main idea of this dissertation is to consider the user response as an expression of the user-perception exception function and embed it as part of the assistance system for user-AI conversation and cooperation, converting the unknown and dynamic user-perception exception function into an overall optimization function of the system. According to this idea, a user-AI cooperation paradigm is proposed based on a style-transfer problem, utilizing the valid features in different conversation strategies based on various prior knowledge from sketches and translating input sketches into a language that AI can understand, and finally, generating images satisfying successfully. The greatest advantage of this paradigm is that it integrates the user's creative process into the system: starting from incomplete rough sketches, it can gradually approach the users' desired drawing targets in their minds by consistently delivering high-quality guidance. This user-AI cooperation paradigm not only improves the AI's input sketches but also expands the user's drawing skills in the drawing process, resulting in a win-win situation.

And our results are exciting – the success of the proposed drawing assistance system fully illustrates that AI not only generates high-quality results but also implies an understanding of the generation process, which in this dissertation is reflected in the AI's precise understanding of the user's sketch input and the generation of matching results. The detailed work of each chapter is as follows.

Data is an inseparable topic for deep learning. Chapter 3 proposes a framework for generating sketch-art drawing data pairs using deep prior knowledge and contributes to line drawing style transfer for anime style, which provides data support and reference for the study in the subsequent chapters.

In Chapter 4, the two-stage drawing assistance system, dualFace, verifies the effectiveness of an explicit conversation strategy in AI-assisted creation by

visualizing shape features for global stage conversation, achieves sketch-semantic level conversion using these feature descriptors, and successfully implements realistic style portrait drawing assistance.

Chapter 5 addresses a more challenging research question: sketch-based anime portrait drawing assistance. This is because the anime style is more abstract compared to the real style of human faces and lacks a semantic database. Faced with these difficulties, Section 5.3 adopts an implicit conversation strategy with prior knowledge about the key points of the face to encourage the disentanglement learning of neural networks at the level of strokes. As a further extension, this AI-assisted drawing system uses only the prior knowledge of a facial semantic mask from a single image for the anime style. This one-shot semantic labeling approach automatically finds out the correspondence between strokes from users' input sketches and line drawings generated as guidance. What's more, with this one-shot approach, our final drawing assistance system in Section 5.4 can predict users' sketching intentions and provide suitable guidance intelligently which is confirmed in the following user study in Section 5.5.

6.2 Future work

Future work can be expanded in at least the following directions.

More artistic drawing styles. In this paper, the proposed drawing-assisted AI is only for real style and anime portraits, which is mainly limited by deep prior knowledge from StyleGAN. The sketch-based feature manipulation in Chapter 5 can be considered as a reordering of the depth-based prior knowledge at stroke-level, and therefore relies heavily on the pre-trained GAN model. If more images of art drawing styles can be obtained by using the style transfer approach, and then more SytleGAN pre-trained models that can generate different styles of art drawings can be obtained as prior knowledge, the scope of drawing assistance in this dissertation can be extended.

Drawing assistance based on sketch vectorization. On the one hand, the sketch generation simulation process in Chapter 5 for feature manipulation utilizes the key point information of faces as an approximation of stroke lines, so this method is limited to portrait matching. Once a deep network can predict stroke lines similar to those used in human drawing, this method can be extended to assist in animal drawings or even landscape drawings. On the other hand, the guidance generated by the system in Chapter 5 is mainly given in raster image format, which can only serve as a global reference for users' sketching and cannot guide him/her more precisely in stroke-level training. However, the current sketch vectorization methods are often limited to simple sketches with low quality. Thus, there is still

a long way to go for complex artistic drawing assistance.

References

[1] Y. J. Lee, C. L. Zitnick, and M. F. Cohen, "Shadowdraw: real-time user guidance for freehand drawing," *ACM Transactions on Graphics (TOG)*, vol. 30, no. 4, pp. 27:1–27:10, 2011.

[2] E. Protter, "Painters on painting (dover fine art, history of art)," http://dlib.net/, 2011.

[3] T. Kaluarachchi, A. Reis, and S. Nanayakkara, "A review of recent deep learning approaches in human-centered machine learning," *Sensors*, vol. 21, no. 7, p. 2514, 2021. [Online]. Available: https://doi.org/10.3390/s21072514

[4] E. Frid, C. Gomes, and Z. Jin, "Music creation by example," in *CHI '20: CHI Conference on Human Factors in Computing Systems, Honolulu, HI, USA, April 25-30, 2020*, R. Bernhaupt, F. F. Mueller, D. Verweij, J. Andres, J. McGrenere, A. Cockburn, I. Avellino, A. Goguey, P. Bjøn, S. Zhao, B. P. Samson, and R. Kocielnik, Eds. ACM, 2020, pp. 1–13. [Online]. Available: https://doi.org/10.1145/3313831.3376514

[5] R. Louie, A. Coenen, C. Z. Huang, M. Terry, and C. J. Cai, "Novice-ai music co-creation via ai-steering tools for deep generative models," in *CHI '20: CHI Conference on Human Factors in Computing Systems, Honolulu, HI, USA, April 25-30, 2020*, R. Bernhaupt, F. F. Mueller, D. Verweij, J. Andres, J. McGrenere, A. Cockburn, I. Avellino, A. Goguey, P. Bjøn, S. Zhao, B. P. Samson, and R. Kocielnik, Eds. ACM, 2020, pp. 1–13. [Online]. Available: https://doi.org/10.1145/3313831.3376739

[6] V. Balasubramanian, S. Chakraborty, S. Krishna, and S. Panchanathan, "Human-centered machine learning in a social interaction assistant for individuals with visual impairments," in *Symposium on Assistive Machine Learning for People with Disabilities at NIPS*, 2008.

[7] H. Kacorri, K. M. Kitani, J. P. Bigham, and C. Asakawa, "People with visual impairment training personal object recognizers: Feasibility and challenges," in *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, 2017, pp. 5839–5849.

[8] S. Feiz, S. M. Billah, V. Ashok, R. Shilkrot, and I. Ramakrishnan, "Towards enabling blind people to independently write on printed forms," in *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, 2019, pp. 1–12.

[9] K. Lee and H. Kacorri, "Hands holding clues for object recognition in teachable machines," in *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, 2019, pp. 1–12.

[10] Y. Zhao, S. Wu, L. Reynolds, and S. Azenkot, "A face recognition application for people with visual impairments: Understanding use beyond the lab," in *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, 2018, pp. 1–14.

[11] N. M. Davis, C. Hsiao, K. Y. Singh, L. Li, and B. Magerko, "Empirically studying participatory sense-making in abstract drawing with a co-creative cognitive agent," in *Proceedings of the 21st International Conference on Intelligent User Interfaces, IUI 2016, Sonoma, CA, USA, March 7-10, 2016*, J. Nichols, J. Mahmud, J. O'Donovan, C. Conati, and M. Zancanaro, Eds. ACM, 2016, pp. 196–207. [Online]. Available: https://doi.org/10.1145/2856767.2856795

[12] M. Eitz, J. Hays, and M. Alexa, "How do humans sketch objects?" *ACM Trans. Graph. (Proc. SIGGRAPH)*, vol. 31, no. 4, pp. 44:1–44:10, 2012.

[13] D. Zhang and G. Lu, "Review of shape representation and description techniques," *Pattern recognition*, vol. 37, no. 1, pp. 1–19, 2004.

[14] M. Bober, "Mpeg-7 visual shape descriptors," *IEEE Transactions on circuits and systems for video technology*, vol. 11, no. 6, pp. 716–719, 2001.

[15] H. Chatbri and K. Kameyama, "Sketch-based image retrieval by shape points description in support regions," in *2013 20th International Conference on Systems, Signals and Image Processing (IWSSIP)*. IEEE, 2013, pp. 19–22.

[16] F. Mokhtarian and S. Abbasi, "Shape similarity retrieval under affine transforms," *Pattern Recognition*, vol. 35, no. 1, pp. 31–41, 2002.

[17] Y. Cao, C. Wang, L. Zhang, and L. Zhang, "Edgel index for large-scale sketch-based image search," in *CVPR 2011*. IEEE, 2011, pp. 761–768.

[18] W.-P. Choi, K.-M. Lam, and W.-C. Siu, "Maximal disk based histogram for shape retrieval," in *2003 IEEE International Conference on Acoustics,*

Speech, and Signal Processing, 2003. Proceedings.(ICASSP'03)., vol. 3. IEEE, 2003, pp. III–9.

[19] Q. Jing and W. Zengfu, "A method for freehand sketch retrieval based on affine adaptive skeleton," *Journal of University of Science and Technology of China*, vol. 40, no. 10, p. 1043, 2010.

[20] K. Atal, A. Arora, P. Purkait, B. Chanda *et al.*, "Face image retrieval based on probe sketch using sift feature descriptors," in *Indo-Japanese Conference on Perception and Machine Intelligence*. Springer, 2012, pp. 50–57.

[21] M. Eitz, K. Hildebrand, T. Boubekeur, and M. Alexa, "Sketch-based image retrieval: Benchmark and bag-of-features descriptors," *IEEE transactions on visualization and computer graphics*, vol. 17, no. 11, pp. 1624–1636, 2010.

[22] ——, "A descriptor for large scale image retrieval based on sketched feature lines." *SBIM*, vol. 9, pp. 29–36, 2009.

[23] X. Shu and X.-J. Wu, "A novel contour descriptor for 2d shape matching and its application to image retrieval," *Image and vision Computing*, vol. 29, no. 4, pp. 286–294, 2011.

[24] S. J. Belongie, G. Mori, and J. Malik, "Matching with shape contexts," in *Statistics and Analysis of Shapes*, ser. Modeling and Simulation in Science, Engineering and Technology, H. Krim and A. A. Yezzi, Eds. Birkhäuser / Springer, 2006, pp. 81–105. [Online]. Available: https://doi.org/10.1007/0-8176-4481-4_4

[25] T. Watanabe, S. Ito, and K. Yokoi, "Co-occurrence histograms of oriented gradients for pedestrian detection," in *Advances in Image and Video Technology, Third Pacific Rim Symposium, PSIVT 2009, Tokyo, Japan, January 13-16, 2009. Proceedings*, ser. Lecture Notes in Computer Science, T. Wada, F. Huang, and S. Lin, Eds., vol. 5414. Springer, 2009, pp. 37–47. [Online]. Available: https://doi.org/10.1007/978-3-540-92957-4_4

[26] H. Skibbe and M. Reisert, "Circular fourier-hog features for rotation invariant object detection in biomedical images," in *9th IEEE International Symposium on Biomedical Imaging: From Nano to Macro, ISBI 2012, May 2-5, 2012, Barcelona, Spain, Proceedings*. IEEE, 2012, pp. 450–453. [Online]. Available: https://doi.org/10.1109/ISBI.2012.6235581

[27] T. Kato, R. Relator, H. Ngouv, Y. Hirohashi, O. Takaki, T. Kakimoto, and K. Okada, "Segmental HOG: new descriptor for glomerulus detection in kidney microscopy image," *BMC Bioinform.*, vol. 16, pp. 316:1–316:16, 2015. [Online]. Available: https://doi.org/10.1186/s12859-015-0739-1

[28] F. M. Porikli, "Integral histogram: A fast way to extract histograms in cartesian spaces," in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2005), 20-26 June 2005, San Diego, CA, USA*. IEEE Computer Society, 2005, pp. 829–836. [Online]. Available: https://doi.org/10.1109/CVPR.2005.188

[29] Y. Wen, C. Zou, J. Liu, S. Du, and S. Chen, "Sketch-based 3d model retrieval via multi-feature fusion," in *22nd International Conference on Pattern Recognition, ICPR 2014, Stockholm, Sweden, August 24-28, 2014*. IEEE Computer Society, 2014, pp. 4570–4575. [Online]. Available: https://doi.org/10.1109/ICPR.2014.782

[30] P. Zhao, G. Wu, Y. Lu, X. Wu, and S. Yao, "A novel hand-drawn sketch descriptor based on the fusion of multiple features," *Neurocomputing*, vol. 213, pp. 66–74, 2016. [Online]. Available: https://doi.org/10.1016/j.neucom.2016.03.098

[31] R. Dechter, "Learning while searching in constraint-satisfaction-problems," in *Proceedings of the 5th National Conference on Artificial Intelligence. Philadelphia, PA, USA, August 11-15, 1986. Volume 1: Science*, T. Kehler, Ed. Morgan Kaufmann, 1986, pp. 178–185. [Online]. Available: http://www.aaai.org/Library/AAAI/1986/aaai86-029.php

[32] P. Xu, T. M. Hospedales, Q. Yin, Y.-Z. Song, T. Xiang, and L. Wang, "Deep learning for free-hand sketch: A survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1–1, 2022.

[33] Q. Yu, Y. Yang, Y. Song, T. Xiang, and T. M. Hospedales, "Sketch-a-net that beats humans," in *Proceedings of the British Machine Vision Conference 2015, BMVC 2015, Swansea, UK, September 7-10, 2015*, X. Xie, M. W. Jones, and G. K. L. Tam, Eds. BMVA Press, 2015, pp. 7.1–7.12. [Online]. Available: https://doi.org/10.5244/C.29.7

[34] R. Kiran Sarvadevabhatla, J. Kundu, and B. R. Venkatesh, "Enabling my robot to play pictionary: Recurrent neural networks for sketch recognition," *arXiv e-prints*, pp. arXiv–1608, 2016.

[35] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural*

Information Processing Systems 25: 26th Annual Conference on Neural Information Processing Systems 2012. Proceedings of a meeting held December 3-6, 2012, Lake Tahoe, Nevada, United States, P. L. Bartlett, F. C. N. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, Eds., 2012, pp. 1106–1114. [Online]. Available: https://proceedings.neurips.cc/paper/2012/hash/c399862d3b9d6b76c8436e924a68c45b-Abstract.html

[36] K. Cho, B. van Merrienboer, Ç. Gülçehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using RNN encoder-decoder for statistical machine translation," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL*, A. Moschitti, B. Pang, and W. Daelemans, Eds. ACL, 2014, pp. 1724–1734. [Online]. Available: https://doi.org/10.3115/v1/d14-1179

[37] Q. Jia, M. Yu, X. Fan, and H. Li, "Sequential dual deep learning with shape and texture features for sketch recognition," *CoRR*, vol. abs/1708.02716, 2017. [Online]. Available: http://arxiv.org/abs/1708.02716

[38] J. He, X. Wu, Y. Jiang, B. Zhao, and Q. Peng, "Sketch recognition with deep visual-sequential fusion model," in *Proceedings of the 2017 ACM on Multimedia Conference, MM 2017, Mountain View, CA, USA, October 23-27, 2017*, Q. Liu, R. Lienhart, H. Wang, S. K. Chen, S. Boll, Y. P. Chen, G. Friedland, J. Li, and S. Yan, Eds. ACM, 2017, pp. 448–456. [Online]. Available: https://doi.org/10.1145/3123266.3123321

[39] D. Ha and D. Eck, "A neural representation of sketch drawings," *arXiv preprint arXiv:1704.03477*, 2017.

[40] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," in *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*, Y. Bengio and Y. LeCun, Eds., 2014. [Online]. Available: http://arxiv.org/abs/1312.6114

[41] P. Xu, Y. Huang, T. Yuan, K. Pang, Y.-Z. Song, T. Xiang, T. M. Hospedales, Z. Ma, and J. Guo, "Sketchmate: Deep hashing for million-scale human sketch retrieval," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 8090–8098.

[42] P. Xu, C. K. Joshi, and X. Bresson, "Multigraph transformer for free-hand sketch recognition," *IEEE Transactions on Neural Networks and Learning Systems*, 2021.

[43] C. Zhao, "A survey on image style transfer approaches using deep learning," in *Journal of Physics: Conference Series*, vol. 1453, no. 1. IOP Publishing, 2020, p. 012129.

[44] L. A. Gatys, A. S. Ecker, and M. Bethge, "Image style transfer using convolutional neural networks," in *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*. IEEE Computer Society, 2016, pp. 2414–2423. [Online]. Available: https://doi.org/10.1109/CVPR.2016.265

[45] P. L. Rosin, D. Mould, I. Berger, J. P. Collomosse, Y. Lai, C. Li, H. Li, A. Shamir, M. Wand, T. Wang, and H. Winnemöller, "Benchmarking non-photorealistic rendering of portraits," in *Proceedings of 15th International Symposium on Non-Photorealistic Animation and Rendering (NPAR@Expressive 2017)*. ACM, 2017, pp. 11:1–11:12.

[46] Z. Chen, J. Zhou, X. Gao, L. Li, and J. Liu, "A novel method for pencil drawing generation in non-photo-realistic rendering," in *Proceedings of 9th Pacific Rim Conference on Multimedia (PCM)*, ser. Lecture Notes in Computer Science, vol. 5353. Springer, 2008, pp. 931–934.

[47] D. Xie, Y. Zhao, and D. Xu, "An efficient approach for generating pencil filter and its implementation on GPU," in *Proceedings of 10th International Conference on Computer-Aided Design and Computer Graphics (CAD/-Graphics)*. Beijing, China: IEEE, 2007, pp. 185–190.

[48] J. Zhang, R. Wang, and D. Xu, "Automatic generation of sketch-like pencil drawing from image," in *Proceedings of IEEE International Conference on Multimedia & Expo Workshops (ICMEW)*. Hong Kong, China: IEEE Computer Society, 2017, pp. 261–266.

[49] C. Li and M. Wand, "Combining markov random fields and convolutional neural networks for image synthesis," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Las Vegas, NV, USA: IEEE, 2016, pp. 2479–2486.

[50] J. Liao, Y. Yao, L. Yuan, G. Hua, and S. B. Kang, "Visual attribute transfer through deep image analogy," *ACM Transactions on Graphics (TOG)*, vol. 36, no. 4, pp. 120:1–120:15, 2017.

[51] F. C. Silva, P. A. L. de Castro, H. R. Júnior, and E. C. Marujo, "Mangan: Assisting colorization of manga characters concept art using conditional GAN," in *Proceedings of IEEE International Conference on Image Processing (ICIP)*. Taipei, Taiwan: IEEE, 2019, pp. 3257–3261.

[52] J. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *Proceedings of IEEE International Conference on Computer Vision (ICCV)*. Venice, Italy: IEEE, 2017, pp. 2223–2232.

[53] R. Yi, Y. Liu, Y. Lai, and P. L. Rosin, "Apdrawinggan: Generating artistic portrait drawings from face photos with hierarchical gans," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Long Beach, CA, USA: CVF / IEEE, 2019, pp. 10 743–10 752.

[54] J. F. Canny, "A computational approach to edge detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 8, no. 6, pp. 679–698, 1986.

[55] H. Kang, S. Lee, and C. K. Chui, "Coherent line drawing," in *Proceedings of 5th International Symposium on Non-Photorealistic Animation and Rendering (NPAR)*. New York, NY, USA: ACM, 2007, pp. 43–50.

[56] H. Winnemöller, J. E. Kyprianidis, and S. C. Olsen, "Xdog: An extended difference-of-gaussians compendium including advanced image stylization," *Comput. Graph.*, vol. 36, no. 6, pp. 740–753, 2012.

[57] E. Simo-Serra, S. Iizuka, K. Sasaki, and H. Ishikawa, "Learning to simplify: fully convolutional networks for rough sketch cleanup," *ACM Trans. Graph.*, vol. 35, no. 4, pp. 121:1–121:11, 2016.

[58] C. Li, X. Liu, and T. Wong, "Deep extraction of manga structural lines," *ACM Trans. Graph.*, vol. 36, no. 4, pp. 117:1–117:12, 2017.

[59] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. C. Courville, and Y. Bengio, "Generative adversarial networks," *CoRR*, vol. abs/1406.2661, 2014. [Online]. Available: http://arxiv.org/abs/1406.2661

[60] A. Brock, J. Donahue, and K. Simonyan, "Large scale gan training for high fidelity natural image synthesis," *arXiv preprint arXiv:1809.11096*, 2018.

[61] T. Karras, S. Laine, and T. Aila, "A style-based generator architecture for generative adversarial networks," in *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*. Computer Vision Foundation / IEEE, 2019, pp. 4401–4410. [Online]. Available: http://openaccess.thecvf.com/content_CVPR_2019/html/Karras_A_Style-Based_Generator_Architecture_for_Generative_Adversarial_Networks_CVPR_2019_paper.html

[62] B. Liu, Y. Zhu, Z. Fu, G. De Melo, and A. Elgammal, "Oogan: Disentangling gan with one-hot sampling and orthogonal regularization," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 04, 2020, pp. 4836–4843.

[63] X. Chen, Y. Duan, R. Houthooft, J. Schulman, I. Sutskever, and P. Abbeel, "Infogan: Interpretable representation learning by information maximizing generative adversarial nets," *Advances in neural information processing systems*, vol. 29, 2016.

[64] Y. Shen and B. Zhou, "Closed-form factorization of latent semantics in gans," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 1532–1540.

[65] S. Mukherjee, H. Asnani, E. Lin, and S. Kannan, "Clustergan: Latent space clustering in generative adversarial networks," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 33, no. 01, 2019, pp. 4610–4617.

[66] X. Huang and S. Belongie, "Arbitrary style transfer in real-time with adaptive instance normalization," in *ICCV*, 2017.

[67] R. Abdal, Y. Qin, and P. Wonka, "Image2stylegan: How to embed images into the stylegan latent space?" in *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*. IEEE, 2019, pp. 4431–4440. [Online]. Available: https://doi.org/10.1109/ICCV.2019.00453

[68] T. Karras, S. Laine, M. Aittala, J. Hellsten, J. Lehtinen, and T. Aila, "Analyzing and improving the image quality of stylegan," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*. Computer Vision Foundation / IEEE, 2020, pp. 8107–8116. [Online]. Available: https://openaccess.thecvf.com/content_CVPR_2020/html/Karras_Analyzing_and_Improving_the_Image_Quality_of_StyleGAN_CVPR_2020_paper.html

[69] W. Xia, Y. Zhang, Y. Yang, J. Xue, B. Zhou, and M. Yang, "GAN inversion: A survey," *CoRR*, vol. abs/2101.05278, 2021. [Online]. Available: https://arxiv.org/abs/2101.05278

[70] T. Karras, S. Laine, and T. Aila, "A style-based generator architecture for generative adversarial networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 12, pp. 4217–4228, 2021.

[71] C. Chiu, Y. Koyama, Y. Lai, T. Igarashi, and Y. Yue, "Human-in-the-loop differential subspace search in high-dimensional latent space," *ACM Trans. Graph.*, vol. 39, no. 4, p. 85, 2020.

[72] E. Härkönen, A. Hertzmann, J. Lehtinen, and S. Paris, "Ganspace: Discovering interpretable gan controls," in *Proc. NeurIPS*, 2020.

[73] Y. Shen, C. Yang, X. Tang, and B. Zhou, "Interfacegan: Interpreting the disentangled face representation learned by gans," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 4, pp. 2004–2018, 2022. [Online]. Available: https://doi.org/10.1109/TPAMI.2020.3034267

[74] J. Y. Yuxuan Han and Y. Fu, "Disentangled face attribute editing via instance-aware latent space search," in *International Joint Conference on Artificial Intelligence*, 2021.

[75] 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 *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021*. Computer Vision Foundation / IEEE, 2021, pp. 2287–2296.

[76] 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. 133:1–133:14, 2021. [Online]. Available: https://doi.org/10.1145/3450626.3459838

[77] Y. Alaluf, O. Patashnik, and D. Cohen-Or, "Restyle: A residual-based stylegan encoder via iterative refinement," in *2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021*. IEEE, 2021, pp. 6691–6700. [Online]. Available: https://doi.org/10.1109/ICCV48922.2021.00664

[78] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen, "Improved techniques for training gans," in *Proceedings of the 30th International Conference on Neural Information Processing Systems*, ser. NIPS'16. Red Hook, NY, USA: Curran Associates Inc., 2016, p. 2234–2242.

[79] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter, "Gans trained by a two time-scale update rule converge to a local nash equilibrium," in *Proceedings of the 31st International Conference on Neural Information Processing Systems*, ser. NIPS'17. Red Hook, NY, USA: Curran Associates Inc., 2017, p. 6629–6640.

[80] L. Liu, F. Shen, Y. Shen, X. Liu, and L. Shao, "Deep sketch hashing: Fast free-hand sketch-based image retrieval," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Honolulu, HI, USA: IEEE, 2017, pp. 2298–2307.

[81] Q. Yu, F. Liu, Y. Song, T. Xiang, T. M. Hospedales, and C. C. Loy, "Sketch me that shoe," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Las Vegas, NV, USA: IEEE, 2016, pp. 799–807.

[82] T. Dekel, C. Gan, D. Krishnan, C. Liu, and W. T. Freeman, "Sparse, smart contours to represent and edit images," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Salt Lake City, UT, USA: IEEE, 2018, pp. 3511–3520.

[83] T. Portenier, Q. Hu, A. Szabó, S. A. Bigdeli, P. Favaro, and M. Zwicker, "Faceshop: Deep sketch-based face image editing," *ACM Transactions on Graphics (TOG)*, vol. 37, no. 4, 2018.

[84] H. Tseng, M. Fisher, J. Lu, Y. Li, V. G. Kim, and M. Yang, "Modeling artistic workflows for image generation and editing," in *Proceedings of 16th European Conference on Computer Vision (ECCV)*. Springer, Cham: Springer, 2020, pp. 158–174.

[85] S. Yang, Z. Wang, J. Liu, and Z. Guo, "Deep plastic surgery: Robust and controllable image editing with human-drawn sketches," in *Proceedings of 16th European Conference on Computer Vision (ECCV)*. Springer, Cham: Springer, 2020, pp. 601–617.

[86] Z. Hu, H. Xie, T. Fukusato, T. Sato, and T. Igarashi, "Sketch2vf: Sketch-based flow design with conditional generative adversarial network," *Computer Animation and Virtual Worlds*, vol. 30, no. 3-4, pp. e1889:1–e1889:11, 2019.

[87] Y. Peng, Y. Mishima, Y. Igarashi, R. Miyauchi, M. Okawa, H. Xie, and K. Miyata, "Sketch2domino: Interactive chain reaction design and guidance," in *2020 Nicograph International (NicoInt)*. Tokyo, Japan: IEEE, 2020, pp. 32–38.

[88] T. Fukusato, S.-T. Noh, T. Igarashi, and D. Ito, "Interactive meshing of user-defined point sets," *Journal of Computer Graphics Techniques (JCGT)*, vol. 9, no. 3, pp. 39–58, 2020. [Online]. Available: http://jcgt.org/published/0009/03/03/

[89] T. Igarashi and J. F. Hughes, "A suggestive interface for 3d drawing," in *Proceedings of the 14th Annual ACM Symposium on User Interface Software and Technology (UIST)*. New York, NY, USA: ACM, 2001, pp. 173–181.

[90] Y. He, H. Xie, C. Zhang, X. Yang, and K. Miyata, "Sketch-based normal map generation with geometric sampling," in *International Workshop on Advanced Image Technology (IWAIT 2021)*. Kagoshima, Japan: SPIE, 2021, pp. 1–6.

[91] M. Flagg and J. M. Rehg, "Projector-guided painting," in *Proceedings of the 19th Annual ACM Symposium on User Interface Software and Technology (UIST)*. New York, USA: ACM, 2006, pp. 235–244.

[92] T. Igarashi, S. Matsuoka, S. Kawachiya, and H. Tanaka, "Interactive beautification: A technique for rapid geometric design," in *Proceedings of the 10th Annual ACM Symposium on User Interface Software and Technology (UIST)*. New York, NY, USA: ACM, 1997, pp. 105–114.

[93] J. Laviole and M. Hachet, "Papart: Interactive 3d graphics and multi-touch augmented paper for artistic creation," in *Proceedings of IEEE Symposium on 3D User Interfaces (3DUI)*. Costa Mesa, CA, USA: IEEE, 2012, pp. 3–6.

[94] D. Dixon, M. Prasad, and T. Hammond, "icandraw: using sketch recognition and corrective feedback to assist a user in drawing human faces," in *Proceedings of the 28th International Conference on Human Factors in Computing Systems (CHI)*. New York, USA: ACM, 2010, pp. 897–906.

[95] E. Iarussi, A. Bousseau, and T. Tsandilas, "The drawing assistant: Automated drawing guidance and feedback from photographs," in *Proceedings of the 26th Annual ACM Symposium on User Interface Software and Technology*. New York, USA: ACM, 2013, pp. 183–192.

[96] Q. Su, W. H. A. Li, J. Wang, and H. Fu, "Ez-sketching: three-level optimization for error-tolerant image tracing." *ACM Transactions on Graphics (TOG)*, vol. 33, no. 4, pp. 54:1–54:9, 2014.

[97] J. Xie, A. Hertzmann, W. Li, and H. Winnemöller, "Portraitsketch: face sketching assistance for novices," in *Proceedings of the 27th Annual ACM Symposium on User Interface Software and Technology (UIST)*. New York, USA: ACM, 2014, pp. 407–417.

[98] J. Choi, H. Cho, J. Song, and S. M. Yoon, "Sketchhelper: Real-time stroke guidance for freehand sketch retrieval," *IEEE Transactions on Multimedia*, vol. 21, no. 8, pp. 2083–2092, 2019.

[99] Z. He, H. Xie, and K. Miyata, "Interactive projection system for calligraphy practice," in *2020 Nicograph International (NicoInt)*. Tokyo, Japan: IEEE, 2020, pp. 55–61.

[100] Y. Matsui, T. Shiratori, and K. Aizawa, "Drawfromdrawings: 2d drawing assistance via stroke interpolation with a sketch database," *IEEE Transactions on Visualization and Computer Graphics (TVCG)*, vol. 23, no. 7, pp. 1852–1862, 2017.

[101] C. Lee, Z. Liu, L. Wu, and P. Luo, "Maskgan: Towards diverse and interactive facial image manipulation," in *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. Seattle, WA, USA: IEEE, 2020, pp. 5548–5557.

[102] P. Jongejan, H. Rowley, T. Kawashima, J. Kim, and N. Fox-Gieg, "The quick, draw! - a.i. experiment," in *btitle*, https://quickdraw.withgoogle.com/, 2016, pp. 70–77.

[103] P. Sangkloy, N. Burnell, C. Ham, and J. Hays, "The sketchy database: Learning to retrieve badly drawn bunnies," *ACM Transactions on Graphics (proceedings of SIGGRAPH)*, 2016.

[104] M. Li, Z. L. Lin, R. Mech, E. Yumer, and D. Ramanan, "Photo-sketching: Inferring contour drawings from images," in *IEEE Winter Conference on Applications of Computer Vision, WACV 2019, Waikoloa Village, HI, USA, January 7-11, 2019*. IEEE, 2019, pp. 1403–1412. [Online]. Available: https://doi.org/10.1109/WACV.2019.00154

[105] Adobe, "Adobe stock," 2018. [Online]. Available: https://stock.adobe.com/

[106] Z. Wang, S. Qiu, N. Feng, H. Rushmeier, L. McMillan, and J. Dorsey, "Tracing versus freehand for evaluating computer-generated drawings," *ACM Trans. Graph.*, vol. 40, no. 4, Aug. 2021. [Online]. Available: https://doi.org/10.1145/3450626.3459819

[107] K. Sasaki, S. Iizuka, E. Simo-Serra, and H. Ishikawa, "Learning to Restore Deteriorated Line Drawing," *The Visual Computer (Proc. of Computer Graphics International 2018)*, vol. 34, no. 6-8, pp. 1077–1085, 2018.

[108] C. Yan, D. Vanderhaeghe, and Y. Gingold, "A benchmark for rough sketch cleanup," *ACM Transactions on Graphics (TOG)*, vol. 39, no. 6, Nov. 2020. [Online]. Available: https://doi.org/10.1145/3414685.3417784

[109] W. Shen, X. Wang, Y. Wang, X. Bai, and Z. Zhang, "Deepcontour: A deep convolutional feature learned by positive-sharing loss for contour detection," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Boston, MA, USA: IEEE, 2015, pp. 3982–3991.

[110] K. He, X. Zhang, S. Ren, and J. Sun, "Identity mappings in deep residual networks," in *the 14th European Conference on Computer Vision (ECCV)*, ser. Lecture Notes in Computer Science, vol. 9908. Springer, 2016, pp. 630–645.

[111] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *IEEE conference on computer vision and pattern recognition*. IEEE, 2017, pp. 1125–1134.

[112] T. Zan, "Paintschainer." [Online]. Available: https://github.com/pfnet/PaintsChainer

[113] L. Zhang, "sketchkeras." [Online]. Available: https://github.com/lllyasviel/sketchKeras

[114] Z. Huang, Y. Peng, T. Hibino, C. Zhao, H. Xie, T. Fukusato, and K. Miyata, "Dualface: Two-stage drawing guidance for freehand portrait sketching," *Computational Visual Media (CVMJ)*, vol. 8, no. 1, pp. 63–77, 2022.

[115] A. Ghosh, R. Zhang, P. K. Dokania, O. Wang, A. A. Efros, P. H. Torr, and E. Shechtman, "Interactive sketch & fill: Multiclass sketch-to-image translation," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*. Long Beach, CA, USA: IEEE, 2019, pp. 1171–1180.

[116] J.-Y. Zhu, P. Krähenbühl, E. Shechtman, and A. A. Efros, "Generative visual manipulation on the natural image manifold," in *Proceedings of the 14th European Conference on Computer Vision (ECCV)*. Cham, Switzerland: Springer, 2016, pp. 597–613.

[117] B. Bradley, *Drawing People, How to Portray the Clothed Figure*. North Light / Writers Digest, 2003.

[118] M. Eitz, R. Richter, T. Boubekeur, K. Hildebrand, and M. Alexa, "Sketch-based shape retrieval," *ACM Transactions on Graphics (TOG)*, vol. 31, no. 4, pp. 31:1–31:10, 2012. [Online]. Available: https://doi.org/10.1145/2185520.2185527

[119] C. Yu, J. Wang, C. Peng, C. Gao, G. Yu, and N. Sang, "Bisenet: Bilateral segmentation network for real-time semantic segmentation," in *Proceedings of the 15th European Conference on Computer Vision (ECCV)*, ser. Lecture Notes in Computer Science, vol. 11217. Cham, Switzerland: Springer, 2018, pp. 334–349.

[120] V. Kazemi and J. Sullivan, "One millisecond face alignment with an ensemble of regression trees," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Columbus, OH, USA: IEEE, 2014, pp. 1867–1874. [Online]. Available: https://doi.org/10.1109/CVPR.2014.241

[121] Y. Li, X. Chen, F. Wu, and Z.-J. Zha, "Linestofacephoto: Face photo generation from lines with conditional self-attention generative adversarial networks," in *Proceedings of the 27th ACM International Conference on Multimedia (MM' 19)*. New York, NY, USA: ACM, 2019, pp. 2323–2331.

[122] Y. Li, X. Chen, B. Yang, Z. Chen, Z. Cheng, and Z. Zha, "Deepfacepencil: Creating face images from freehand sketches," in *Proceedings of the 28th ACM International Conference on Multimedia (MM' 20)*, C. W. Chen, R. Cucchiara, X. Hua, G. Qi, E. Ricci, Z. Zhang, and R. Zimmermann, Eds. New York, NY, USA: ACM, 2020, pp. 991–999.

[123] S. Chen, W. Su, L. Gao, S. Xia, and H. Fu, "Deepfacedrawing: deep generation of face images from sketches," *ACM Transactions on Graphics (TOG)*, vol. 39, no. 4, p. 72, 2020.

[124] S. Yang, L. Jiang, Z. Liu, and C. C. Loy, "Pastiche master: Exemplar-based high-resolution portrait style transfer," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022*. IEEE, 2022, pp. 7683–7692. [Online]. Available: https://doi.org/10.1109/CVPR52688.2022.00754

[125] zllrunning, "Using modified bisenet for face parsing in pytorch," https://github.com/zllrunning/face-parsing.PyTorch, 2019.

[126] C. Yu, J. Wang, C. Peng, C. Gao, G. Yu, and N. Sang, "Bisenet: Bilateral segmentation network for real-time semantic segmentation," in *Computer Vision - ECCV 2018 - 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part XIII*, ser. Lecture Notes in Computer Science, V. Ferrari, M. Hebert, C. Sminchisescu, and Y. Weiss, Eds., vol. 11217. Springer, 2018, pp. 334–349. [Online]. Available: https://doi.org/10.1007/978-3-030-01261-8_20

[127] H. Winnemöller, "Xdog: advanced image stylization with extended difference-of-gaussians," in *9th International Symposium on Non-Photorealistic Animation and Rendering, NPAR 2011, Vancouver, BC, Canada, August 5-7, 2011, Proceedings*, J. P. Collomosse, P. Asente, and S. N. Spencer, Eds. ACM, 2011, pp. 147–156. [Online]. Available: https://doi.org/10.1145/2024676.2024700

[128] hysts, "Anime face detector," https://github.com/hysts/anime-face-detector, 2021.

[129] G. Branwen, Anonymous, and D. Community, "Danbooru2019 portraits: A large-scale anime head illustration dataset," https://www.gwern.net/Crops#danbooru2019-portraits, March 2019, accessed: DATE. [Online]. Available: https://www.gwern.net/Crops#danbooru2019-portraits

[130] Y. Zheng, H. Yao, and X. Sun, "Deep semantic parsing of freehand sketches with homogeneous transformation, soft-weighted loss, and staged learning," *IEEE Trans. Multim.*, vol. 23, pp. 3590–3602, 2021. [Online]. Available: https://doi.org/10.1109/TMM.2020.3028466

[131] K. Li, K. Pang, J. Song, Y. Song, T. Xiang, T. M. Hospedales, and H. Zhang, "Universal sketch perceptual grouping," in *Computer Vision - ECCV 2018 - 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part VIII*, ser. Lecture Notes in Computer Science, V. Ferrari, M. Hebert, C. Sminchisescu, and Y. Weiss, Eds., vol. 11212. Springer, 2018, pp. 593–609. [Online]. Available: https://doi.org/10.1007/978-3-030-01237-3_36

[132] R. G. Schneider and T. Tuytelaars, "Example-based sketch segmentation and labeling using crfs," *ACM Trans. Graph.*, vol. 35, no. 5, pp. 151:1–151:9, 2016. [Online]. Available: https://doi.org/10.1145/2898351

[133] K. Wang, J. H. Liew, Y. Zou, D. Zhou, and J. Feng, "Panet: Few-shot image semantic segmentation with prototype alignment," in *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*. IEEE, 2019, pp. 9196–9205. [Online]. Available: https://doi.org/10.1109/ICCV.2019.00929

[134] Y. Endo and Y. Kanamori, "Controlling stylegans using rough scribbles via one-shot learning," *Comput. Animat. Virtual Worlds*, vol. 33, no. 5, 2022. [Online]. Available: https://doi.org/10.1002/cav.2102

[135] E. Collins, R. Bala, B. Price, and S. Süsstrunk, "Editing in style: Uncovering the local semantics of gans," in *2020 IEEE/CVF Conference*

on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020. Computer Vision Foundation / IEEE, 2020, pp. 5770–5779. [Online]. Available: https://openaccess.thecvf.com/content_CVPR_2020/html/Collins_Editing_in_Style_Uncovering_the_Local_Semantics_of_GANs_CVPR_2020_paper.html

[136] M. Hu, "Visual pattern recognition by moment invariants," *IRE Trans. Inf. Theory*, vol. 8, no. 2, pp. 179–187, 1962. [Online]. Available: https://doi.org/10.1109/TIT.1962.1057692

[137] A. Bangor, P. T. Kortum, and J. T. Miller, "An empirical evaluation of the system usability scale," *Int. J. Hum. Comput. Interact.*, vol. 24, no. 6, pp. 574–594, 2008. [Online]. Available: https://doi.org/10.1080/10447310802205776

[138] E. A. Carroll, C. Latulipe, R. Y. K. Fung, and M. A. Terry, "Creativity factor evaluation: towards a standardized survey metric for creativity support," in *Proceedings of the 7th Conference on Creativity & Cognition, Berkeley, California, USA, October 26-30, 2009*, N. Bryan-Kinns, M. D. Gross, H. Johnson, J. Ox, and R. Wakkary, Eds. ACM, 2009, pp. 127–136. [Online]. Available: https://doi.org/10.1145/1640233.1640255

Publications

[1] Z. Huang, Y. Peng, T. Hibino, C. Z. Zhao, H. Xie, T. Fukusato, and K. Miyata, *dualface: Two-stage drawing guidance for freehand portrait sketching*, Computational Visual Media, 2021.

[2] Z. Huang, H. Xie, K. Miyata, *Manifold Learning for Hand Drawn Sketches*, NICOGRAPH International 2020, poster, Tokyo, 2020.06

[3] Z. Huang, Y. Peng, H. Xie, T. Fukusato, and K. Miyata, *One-shot Line Extraction from Color Illustrations*, NICOGRAPH International 2021, 2021.06

[4] Z. Huang, H. Xie, T. Fukusato, *Interactive 3D Character Modeling from 2D Orthogonal Drawings with Annotations*, Proceedings of International Workshop on Advanced Image Technology 2022 (IWAIT 2022), full paper, 2022.01