

Wenyan Yang

# THE EFFECTS OF USER-AI CO- CREATION ON UI DESIGN TASKS

# ABSTRACT

Wenyan Yang: The effects of user-AI co-creation on UI design tasks

M.Sc. Thesis

Tampere University

Master's Degree Programme in Master's degree program in Human-technology Interaction

May 2019

---

With the boost of GPU computation power and the developments of neural networks in the recent decade, a lot of AI technique are invented and show bright potential of improving human tasks. GAN (generative adversarial network) as one of recent AI technique has powerful ability to perform image generation tasks. Besides, many researchers are working on exploring the potentials and understand user-AI collaboration by developing prototype with the help of neural networks (such as GAN). Unlike previous works focus on simple sketch task, this work studied the user experience with UI design task to understand how AI could improve or harm the user experience within practical and complex design tasks. The findings are as follows: multiple-hint AI turned out to be more user-friendly, and it is important to study and understand how AI's presentation should be designed for user-AI collaboration. Based on these findings and previous works, this research discussed about what factors should be taken into consideration when designing user-AI collaboration tool.

Key words and terms: HCI, AI, user experience, user-AI cooperation, creative work, UI design, deep learning, GAN.

Acknowledgements: For this research, I would like to thank my supervisor Joni-Kristian Kämäräinen for helping me with the machine learning studying and providing the hardware support implement the demo. I would also like to thank my thesis supervisor Jaakko Hakulinen and Markku Turunen for helping me with my HTI studying and thesis writing.

# Contents

<b>1. INTRODUCTION</b> .....	<b>1</b>
<b>2. LITERATURE REVIEW: AI, HCI AND INTELLIGENT INTERACTION SYSTEM</b> .....	<b>4</b>
2.1. WHAT IS ARTIFICIAL INTELLIGENCE .....	4
2.2. HCI AND AI.....	7
2.3. INTELLIGENT INTERACTION SYSTEM (IIS) .....	9
2.4. AUTONOMY OF INTELLIGENT INTERACTION SYSTEMS .....	11
2.5. IIS APPLICATIONS.....	12
<b>3. DESIGNING CHALLENGES FOR INTELLIGENT INTERACTION SYSTEMS</b> .....	<b>18</b>
3.1. CHALLENGES FROM INTELLIGENT INTERACTION SYSTEMS' FEATURES .....	18
3.2. CHALLENGES FROM INTELLIGENT INTERACTION SYSTEMS' GOALS.....	21
3.3. CHALLENGES FOR DESIGNERS.....	24
3.4. USER STUDY OF USER-AI CO-CREATIVE INTERFACE .....	25
3.5. SUMMARY .....	32
<b>4. GENERATIVE MODELS: GAN, VAE AND BICYCLEGAN</b> .....	<b>33</b>
4.1. VAE - VARIATIONAL AUTO-ENCODER.....	34
4.2. GAN (GENERATIVE ADVERSARIAL NETWORK) .....	37
4.3. BICYCLEGAN .....	39
4.4. SUMMARY .....	43
<b>5. DESIGN</b> .....	<b>44</b>
5.1. MOTIVATION AND GOALS .....	44
5.2. PROTOTYPE DESIGN.....	46
5.3. STYLES .....	48
<b>6. IMPLEMENTATION</b> .....	<b>50</b>
6.1. GENERATIVE MODEL STRUCTURE.....	50
6.3. WIREFRAME DESIGN TOOL .....	56
6.4. FINAL DESIGN AND IMPLEMENTATION .....	57
<b>7. EXPERIMENTS</b> .....	<b>63</b>
7.1. PROCEDURE .....	63
7.2. PARTICIPANTS .....	64
7.3. TASKS.....	65
7.4. DATA COLLECTION .....	67
7.5. SEMI-STRUCTURED INTERVIEW .....	67
<b>8. RESULTS</b> .....	<b>69</b>
8.1. QUANTITATIVE ANALYSIS.....	69
8.2. QUALITATIVE ANALYSIS .....	75
<b>9. DISCUSSION</b> .....	<b>79</b>
9.1. DESIGN RECOMMENDATIONS.....	79
9.2. LIMITATIONS OF THIS WORK.....	80
9.3. FUTURE WORK.....	83
<b>10. CONCLUSION</b> .....	<b>84</b>
References .....	85

## **1. Introduction**

With the development of artificial intelligence, the AI technology is not only changing the way of human-machine interaction, it is also changing artists' ways of creating contents, making the communication between human and machine more intelligent. One typical feature of artificial intelligence algorithms is that they can be trained based on a large number of data samples (Goodefellow, 2016). Data such as texts, pictures, videos, even live broadcasts, could be the training materials for AI. By exploring the data and information, AI could learn the patterns and rules behind the data. On the other hand, people can also help the machine to learn better rules by providing more samples to the machine during the interaction process, and the machine could become more intelligent through learning. The interaction between human and AI not only helps AI learn better rules, but more importantly this human-AI interaction could augment designer and artist's creativity. By working with artificial intelligence, artists can find unexpected signals and inspirations from a different perspective (Oh et al., 2018).

AI based creative works has been applied in various fields such as music composing, visual design, and script writing. For the music composing task, there are many mature applications of music generation based on AI models. Even in jazz that is quite demanding for impromptu performances, there have been attempts to jointly improvise performances. For example, Professor AI Biles's GenJam project has performed dozens of concerts since 2005 (Huang, 2016); In the field of NLP (natural language processing), AI has also shown its possibility of story generation. One famous example is OpenAI's GPT-2 algorithm, which is a large transformer-based language model that is able to generate synthetic text samples (Radford et al. 2019). Based on the given topic, the GPT-2 model is able to accomplish the story. For visual design, many tools have been developed to help artist to colorize photos or accomplish painting tasks.

From a pragmatic perspective, automatically generating rich and personalized contents is becoming a common requirement (Ha and Eck, 2017). Therefore, it is important to find ways to help creative workers do creative tasks more effectively. However, traditional tools such as video editing, audio effects production or image synthesis, the tool itself can contribute little to augmenting creative abilities, everything needs to start from scratch.

Although the current artificial intelligence can hardly accomplish creative design jobs by itself, it could provide a novel way for designers to observe the objects and tasks. As the AI has shown its ability to generate creative works, it has provided a possibility that designers and creativity support tool can interact in a more dynamic, flexible, and human

way. For example, to reduce the workload of repetitive design works, the AI creative-support tool (Shneiderman, 2007) can automatically generate several drafts or templates so that designers do not have to start from scratch. To create such a tool or workflow that better assists human in creative tasks, it is important to study the creative process of human beings.

In ColorAIze (Matulic, 2018), the researchers designed an intelligent interaction system which could automatically finish users' paintings. Users can freely draw line-sketches and describe the color preferences to AI, and the system will automatically finish the painting. By observing the users' interaction, their work provided an overview of the usability and users' reaction to human-AI creative works cooperation.

Rather than simply observing and analyzing the qualitative data, DuetDraw (Oh et al. 2018) provided more a detailed user study and discussion of user-AI cooperation on creative works. By focusing on communication and leadership between user and AI, the researchers conducted a detailed user study to analyze how communication and initiative affected user experience. Based on these findings, they discussed the design implications for user interfaces with which users and AI can closely cooperate on creative work.

Drawing Apprentice (Davis, 2016) showed a drawing agent which can simulate user's creativity, so that it can improvise and collaborate on abstract sketches with users. Instead of analyzing cooperation creative works from user experience aspect, their cognitive science theory of enaction and its conceptual framework called participatory sensemaking to model and understand creative collaboration. Their work provided a discussion about how user makes sense of the intelligent agent's generations and how it is related to the creative collaboration.

From Drawing Apprentice (Davis, 2016) to DuetDraw (Oh et al., 2018), the creative design task in the above experiments and evaluation is a simplified painting task, which only requires a user to draw sketches with simple combinations of lines and colors. However, compared to some creative design tasks in real working environments, these creative design tasks are relatively simple and aimless. Design tasks such as UI design are quite complicated and have many known and unknown restrictions. Especially the UI design task should not only meet the requirements for aesthetics but also the functionality. In such case, the creative design task is becoming more challenging for both designers and AI.

Besides, most of the AI techniques implemented in previous research only generate a single output based on user's input. In the community of AI research, representing multimodality is an interesting topic. The multi-model representation allows the algorithm to represent aspects of the possible outputs not contained in the given input. This means that AI algorithm is able to generate multiple outputs based on the given input image.

Since AI algorithms have shown great power in art designing and many studies have also proved its contributions to user-AI creative cooperation, many intelligent interaction systems have been developed to help designers in creative work. Previous research such as DuetDraw provides good examples and general guidelines for user-AI interface design. Therefore, it becomes interesting and meaningful to extend the prototypes to real design tasks. In this thesis, the UI design task was chosen as the extension of the creative designing prototypes, and the goal is to see if the previous research results apply to more complicated design tasks.

To discover how user-AI creative cooperation performs in the practical works, an intelligent interaction tool was designed and implemented for UI visual design task. By conducting a user study through qualitative and quantitative analysis approaches, this work explored the difficulties and challenges when user-AI creative cooperation meets the real UI design tasks. To explore the user experience of user-AI cooperation in UI visual design tasks, this thesis mainly focused on the following aspects:

- How does AI affect user experience of UI design?
- How do different AI models (single-model representation and multi-model representation) affect the user experiences of UI design?
- Which factors are important when implementing an intelligent interaction system for UI design?

## **2. Literature review: AI, HCI and Intelligent Interaction System**

A literature review is presented in this chapter, to provide a comprehensive overview of the definition of AI (Artificial Intelligence), the intersection research fields between AI and HCI (Human-computer Interaction), and related implementation. This chapter first introduces the definition of artificial intelligence and its developing history. The four commonly followed approaches are introduced along with the discussion about strong-AI and weak-AI. As an intersection research area, the related terms computational creativity, co-creativity and CAIS (Collaborative/Creative Artificial Intelligence System) are explained. Additionally, an overview of recent human-AI collaboration creative works is given to present the examples. The principles of the related algorithms are given as well for a better understanding of the implementations.

Since the birth of the Dartmouth Conference in 1956, Artificial Intelligence (AI) has experienced many stagnant stages. Due to advances in precision equipment manufacturing technology in recent years, computer hardware devices such as GPUs and heterogeneous computing have made great progress in all aspects. Such developments have provided promising hardware foundation for the rejuvenation of artificial intelligence. Besides, the establishment of large-scale datasets in recent years has also made a great contribution to the research of artificial intelligence. In 2006, with the deep learning neural network proposed by Hinton (Hinton et al., 2006), AI research flourished again. At the same time, artificial intelligence has been successfully applied in many fields, such as computer vision (image recognition, image understanding, video recognition), speech engineering (speech recognition, semantic understanding, speech synthesis), natural language processing (machine translation, sentiment analysis, semantic understanding), decision-making systems and big data statistical analysis,.

Human-Computer Interaction (HCI) is an ever-changing field that responds to technological innovations to meet the demands that follows. Since the technology innovation of AI shows potential to change the use of conventional tools and the way of problem solving, the traditional definitions and rules may not be able to meet the needs of such development. In order to discover the new design patterns and issues nowadays, many researchers, artists and developers are trying to explore the potential how users and intelligent agents can cooperate to do creative works.

### **2.1. What is artificial intelligence**

The first concept of artificial intelligence was proposed by Alan Turing: can machines really think? If a machine can talk to humans without being able to be identified as a machine, then this machine has intelligent features. As a research discipline, artificial intelligence was first formally established by scientists in different fields (mathematics, psychology, engineering, economics, and political science) in 1956, at a conference held

at Dartmouth College (Russell, 2016). A large number of successful AI programs and new research directions have been emerging ever since.

The definitions of artificial intelligence and research questions have undergone many changes (Russell, 2016). Even today different definitions are still widely accepted. Which definition is used depends on the context in which we discuss the issue and the focus of attention. Historically, early in the 50s, the famous machine intelligence test, the Turing test, was developed to evaluate whether an artificial entity has the features of being intelligent. The Turing test is an operational experiment, to pass the test, the machine should be able to communicate with a human interrogator without being distinguished as a machine.

In the 60's, artificial intelligence was considered as a general-purpose robot, which has the characteristics of imitating intelligence, the ability to extract abstract concepts and the ability to reason its own behaviour and be able to solve realistic problems. Due to the limitations of theoretical study, computation power and amount of data, AI technology was developing slowly in the different areas such as pattern recognition (Russell, 2016).

In the 80's, AI researchers proposed that artificial intelligence's inference ability should be more important than its abstract ability. To acquire the genuine intelligence, it should have a physical entity with perception, movement and interaction ability within the real world to collect data. The activity cognition ability is vital for commonsense reasoning and other high-level cognitive abilities. The expert system invented at this period could answer or solve problems in a particular area based on a set of logical rules derived from expertise. The research results at this time also promoted the development of natural language and machine vision in the future. (Russell, 2016).

Although the definitions and research focus change over time, four approaches are commonly followed to define artificial intelligence: 1) **humanly thinking**, 2) **rationally thinking**, 3) **humanly acting** and 4) **rationally acting** (Russell and Norvig, 2016).

**Human-like acting** is defined as the famous Turing test approach. When a human interrogator asking questions from the machine cannot tell whether the responses are from a person or a machine, the machine could be considered as artificial intelligence. More specifically, to be identified as AI, the machine should be able to satisfy the following four abilities: a) NLP (natural language processing) ability, b) ability of knowledge representation, c) automatical reasoning ability, d) adaptive learning abilities for new patterns, aka machine learning. However, the so-called total Turing test requires two more criteria to measure intelligent agent's perceptual abilities: vision ability to perceive objects and robotic ability to interact with physical worlds. These six criteria build the majority of artificial intelligence.

**Humanly thinking** approach (aka the cognitive modeling approach), focuses on how the human mind works with cognitive science methods. Three ways of observation are



proposed in this approach: a) observing through introspections, b) observing a person in action by psychological experiments, c) observing the brain in action by imaging. The idea is to observe and compare the machine's input-output behavior with the human's behavior. If the machine behaves similar to human, then it could be considered to have intelligent features.

**Rationally thinking** approach is also called the laws of thought approach. It refers to study mental abilities by using computational models (Charniak and McDermott, 1985), and making computations possible to perceive, reason and act. This approach stresses the importance of logic. The “logician” tradition behind this approach is to create intelligent systems that could solve any solvable problem described in logical notation. The intelligent agent should make correct inference. However, to make AI follow this approach is not easy. Stating the informal knowledge in the formal term by logical symbols is difficult. Besides, it is hard for AI to solve problems in practice without guidance for first reasoning steps.

**Rationally acting** mainly emphasizes the agent part. The rational agent is defined to “operate autonomously, perceive their environment, persist over a prolonged time period, adapt to change, and create and pursue goals” (Stuart, 2010, p178) and “AI is concerned with intelligent behavior in artifacts.” (Nilsson, 1998, p52). Although rational thinking (reasoning ability) is vital, the rationality, however, it has certain situations when there is no proper action but reaction is still needed. Besides, some human behavior does not involve inference such as reflex action. Compared to laws of thought approach, rationally acting approach is more general as in this case, AI contains most of the possible mechanism for rational strategy; AI based on such approach is also more suitable for development.

However, not all AI can match all four characteristics mentioned above. Two hypotheses are given by philosophers to categorize the type of AI: 1) if the machine acts as if it has intelligence, or simulating thinking, it should be **weak AI**; 2) if the machine is actually thinking, it is called **strong-AI** (Stuart, 2010).

The question for **strong-AI** is "can machines really think" rather than simulating thinking (Stuart, 2010). The most popular and widely accepted standard is the Turing test. However, even the Turing test itself focuses on the indistinguishability between the behavior of the computer and the human behavior, from the perspective of the observer. It does not mention the specific traits or capabilities that a computer needs to have in order to achieve this indistinguishability. Stuart introduced six features that AI needs to be a strong-AI in his book: using uncertain factors to reason, using strategies, solving problems, and decision-making capabilities; the ability to express knowledge, including the ability to express common sense knowledge; planning capabilities; learning ability;

the ability to communicate in natural language; the ability to integrate these capabilities to achieve the stated goals.

There is one controversial argument brought by the definition of strong-AI: whether it is necessary for it to have human "consciousness". Some researchers believe that only the AI with human consciousness can be considered as strong-AI. While some researchers consider that Strong-AI only needs to have the ability complete the tasks like of human beings, and we do not need to if it has human-like consciousness.

In contrast to the definition strong-AI is weak-AI, which is also known as Narrow AI or Applied AI. It refers to artificial intelligence technologies that only focus on solving problems in specific areas. Compared to strong-AI, the philosophical question for weak AI is "can machines act intelligently?" (Russell and Norvig, 2016). However, current AI techniques' performance can hardly be qualified as strong-AI, thus all the artificial intelligence algorithms and applications deployed nowadays belong to weak artificial intelligence.

With the boost of GPU's computation power, the machine learning based artificial algorithm have achieved remarkable progress in many research fields such as computer vision (CV), natural language processing (NLP) and RL (reinforcement learning). Alpha Go is one of the best examples of weak artificial intelligence. This weak AI surpassed the top players in the world of Go. However, its ability is only limited to Go (or similar game field).

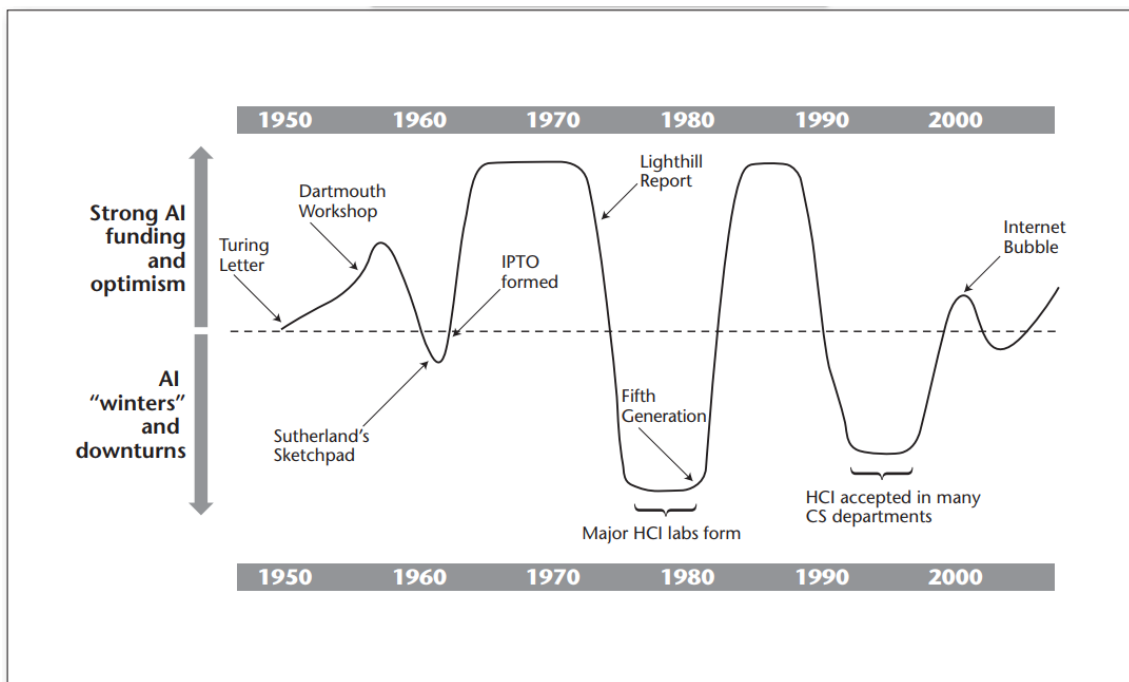
## **2.2. HCI and AI**

It is easy to see the connection between AI and HCI as description of artificial intelligence: as what is defined in Turing test, the human investigator is involved to identify if the intelligent agent is intelligent.

Human-computer interaction began with the intersection of computer science and human factors. With the continuous development of technology, cognitive psychology, sociology and design science were gradually introduced into human-computer interaction. Nowadays, human-computer interaction involves many research and application fields such as computer science, psychology, sociology and ergonomics, and has become a cross-disciplinary subject of great concern (Hewett et al. 1992). The overall trend of human-computer interaction development is toward a user-centric, more intuitive interactive approach (Goert and Reinhart, 2015). In the development history, the first thing is to emphasize in HCI is the "interaction", and then to human-centered computing. (HCC), eventually decentralized human-computer symbiosis system (CHS). As early as 1960, Licklider (Licklider, 1960) proposed the concept of "human-machine symbiosis", pointing out that computers can help humans to solve problems.

The relationship of AI and HCI have been converging in recent decades (Grudin, 2009). Throughout the history, there are some differences between HCI and AI that made

them to a direct tension (Grudin, 2009). Grudin's article specifically explained three differences: 1) HCI usually is more practical while AI strongly focuses on the future possibilities and tolerated slow process; 2) the goal of AI research is to devise an intelligent agent to compete human intelligence while HCI is to improve applications; 3) in the past, AI research required expensive mainframe and workstation platforms while HCI research explores availability. Such differences made an interesting situation shown in Figure 1: "When AI was ascendant, HCI languished; during AI winters, HCI thrived." (Grudin, 2009).



**Figure 1: the Changing Seasons of AI and HCI.**

**Funding climate and public perception with three HCI high points. (Grudin, 2009)**

Despite these differences throughout the history, HCI and AI are converging nowadays. Especially with the development of AI research, the new AI applications create demands for innovative interfaces (Grudin, 2009). AI researchers and HCI researchers are contributing to each other's field. As is concluded by Lieberman, HCI and AI have the same purpose: "- making user interfaces more effective and easier for people to use and that together, the community can make user interfaces smarter and less frustrating to use " (Lieberman, 2009).

From the perspective of artificial intelligence, human-computer interaction is a research approach of artificial intelligence. Michael Jordan, the pioneer of machine learning, proposed that the first breakthrough in artificial intelligence is human-machine dialogue (Michael, 2018). Further achievements can help humans handle daily affairs and even home robot making decisions.

From the perspective of human-computer interaction, artificial intelligence brings breakthroughs for human-computer interaction (Grudin, 2009). Traditional human-computer interaction technologies such as mouse and keyboard and touch screen make it difficult for people and computers to achieve efficient and natural interactions. Artificial intelligence techniques such as image analysis, gesture recognition, semantic understanding, and big data analysis can help computers better perceive human intentions, accomplish tasks that humans cannot accomplish, and drive the development of human-computer interaction.

### 2.3. Intelligent Interaction System (IIS)

The typical intersection area between AI research and HCI research is studying the cooperation between human and AI. However, there is not a unified term that could generally describe the collaboration works between human and intelligent systems yet. Different terms have been proposed according to different priorities. Each of them has its own specific research question, even though there is a general intersection between these studies.

One typical research area is the **creative systems**. Creative intelligent systems are the systems able to perform creative works with or without human participants. The tasks conducted by creative systems are various as well. There are three types of creative system generally: **fully autonomous systems**, **creativity support tools**, and **co-creative systems**. They are categorized based on how human is involved in the cooperation. Derived from these sub-research areas, multiple research problems are proposed, such as Creativity Support Tools (Shneiderman, 2007), Computational Creativity (Colton and Wiggins, 2012), Co-creativity systems (Karimi et al. 2018), Collaborative/Creativity artificial intelligence (Wikström 2018, Feldman 2017). These different research problems and definitions are introduced as follows.

**Fully autonomous system** is a system that is able to generate creative artifacts independently. The creativity of the outputs usually is judged by users or evaluation metrics. The tasks of fully autonomous system vary as well as the implemented algorithms. One typical example is the art generative model GAN which was developed by Elgammal in 2017 (Elgammalet al. 2017); It is a Generative Adversarial Network based creative agent that can simulate image art creations.

**Creativity support tools** rely on user's operation, generally refers to the systems or tools built to help user to do creative works. As is described in Shneiderman 's research (Shneiderman, 2007), creativity support tools are developed because innovative designers and user interface visionaries are looking for tools for discovery and innovation. Thus creativity support tool is proposed to transfer traditional, relatively safe field of productivity support tool to support creativity. As introduced by Shneiderman, such tools should extend users' capability to make discoveries or inventions from early stages of

gathering information, hypothesis generation, and initial production, through the later stages of refinement, validation, and dissemination.

**Co-creative system** is defined by Karimi as "a system in which users and computers interact with each other to make creative artifacts". It is related to the definition of **co-creativity** defined by Davis (Davis et al. 2015). Davis introduced the term **co-creativity** as a collaboration process, in which the contributions of participants from different parties are synthesized during the interaction (Karimi, 2018). Users and the machines will collaborate together to create artifacts. By establishing synchronous collaboration as a requirement, Dave defines co-creation as a process in which users and machine can collaboratively create and share artifacts in the creative process (Davis et al. 2015).

**Computational creativity**, as a subfield of Artificial Intelligence research, was recently defined by Colton as "the philosophy, science and engineering of computational systems which, by taking on particular responsibilities, exhibit behaviours that unbiased observers would deem to be creative" (Colton and Wiggins, 2012). Colton & Wiggins addressed two considerations in this definition. The first is "*responsibility*". The creativity responsibility: 1) access the aesthetic value of the system generations; 2) invent innovative materials or "derivation of motivations, justifications and commentaries with which to frame their output". The second emphasis is "*evaluation with unbiased observers*". The problem is that people allow their beliefs that machines can't possibly be creative to bias their judgement on such issues, thus system's behaviors should be fairly judged (Colton and Wiggins, 2012; Eigenfeldt et al., 2012; Moffat and Kelly, 2006).

**IUI (Intelligent User Interfaces)** was initially introduced as an example of ICAI (Intelligent Computer Assisted Instruction). It is usually considered as user interfaces involving AI features (Wikipedia contributors, 2018). To study the usability of such systems, in Hartmann's work IUI is defined as follows: "*Intelligent User Interfaces are human-machine interfaces that aim to improve the efficiency, effectiveness and naturalness of human machine interaction by representing, reasoning and acting on models of the user, domain, task, discourse, context, and device*" (Hartmann, 2009).

**Interface agent** is defined by Maes as "computer programs that employ Artificial Intelligence techniques to provide active assistance to a user with computer-based tasks" (Maes, 1995). In Maes' study cases, "interface agent" is mainly defined as an assistant that collaborates with users in the same environment. The assistant does not act as an interface or layers between user and application. It focuses on the cooperation with user to solve tasks. The user could ignore the assistant if necessary. **IUI** could be considered as a subtype of **interface agents**. However, as described in the definition, the study of **IUI** focuses more on the presentation of the system and the design strategy of interaction; While **interface agents** research is focusing on the approaches to build the agents.

**CAIS (Creative/Collaborative Artificial Intelligence System)** is mentioned in Feldman's work (Feldman, 2017). Feldman developed a Human-computer cooperative drawing tool called EVOLVER. The system uses a genetic algorithm to produce generative visual design artifacts based on several constraints controlled by the designer/artist. As introduced by Feldman, the CAIS *"focuses on this notion of understanding and bringing that cognitive experience into computational systems to support that artistic expression and the magic that happens between the artist and their work."* In Wikström's case studies, CAIS is perceived to foster creativity, but not truly collaboratively (Wikström, 2018).

As introduced above, there is not a unified framework or definition about intelligent interaction systems for creative works. Each definition is related to others but focuses on own specific issues. In following discussions, we will use the term "IIS (Intelligent interaction system)" to refer the interaction system where AI features are integrated.

#### **2.4. Autonomy of Intelligent Interaction Systems**

To properly choose the AI algorithm and integrate AI features into IIS, it is important to design the cooperation. Since AI could be regarded as a cooperation agent, the autonomy of the AI features should be considered. Rajiv T. Maheswaran et al. presented in total three ways of autonomy: 1) permission requirements; 2) consultation requirements; 3) MDP (Markov Decision Process) driven transfer-of-control strategy. These autonomies are analyzed and proposed from two perspectives: **user-based** and **agent-based** (Dorais et al. 1998; Maheswaran 2003).

**User-based autonomy** is aimed to improve the controllability of the system. As a supervised strategy, it is proposed since the AI may generate undesirable results, user should have the ability to take the control of task to ensure the system's performance. In this case, there are mainly two issues to be solved: AI's capability and system's personalization. The problem for AI's capability is widely observed in applications, which is difficult to develop agents could conduct all the problem-solving tasks as capable as humans. Another problem is there may have different solution to one task, or users would have their own preferences of the strategy. Thus such problems requires a mechanism that user can dynamically modify or adjust the autonomy of the system especially the AI agent. As Maheswaran suggest, such autonomy should be "natural, easy to use, sufficiently expressive to enable fine-grained specifications of autonomy levels". For the context where AI could to operate independently, the autonomy strategy is applied to limit the scope of actions which AI takes.

Maheswaran proposed two strategies: 1) **Permission requirements**, which means the AI should get authority from users before performing the tasks; 2) **Consultation requirements**, refers to certain tasks that are controlled by users. It is necessary to be aware that these policies are based on a premise, that AI system is a Belief-Desire-

Intention (BDI) model whose parameterized plans for tasks are predefined (Maheswaran 2003).

The setting in **user-based autonomy** is that users don't know the domains of authority and all the decision transferring are specified to users. In this case, the system needs to consider the trade-off of transferring decision-making: although transferring the control to human user achieves highest quality decision makings, it interrupts user's operations and user cannot communicate for decision making. From this perspective, Maheswaran thought the transferring should be minimized.

**Agent-based autonomy** is proposed to balance the conflicts (Maheswaran 2003; Dorais 1998; Ferguson 1996; Horvitz 1999), Previous works investigated various methods focused on individual agent-human interactions to solve such problems, such as using uncertainty score to determine if transferring of control should be conducted (Horvitz 1999), or if the expected utility of doing so is higher than the expected utility of making an autonomous decision (Gunderson 1996). Besides single agent situation, Maheswaran designed an autonomy strategy that considers the multi-agent situation. The strategy is operationalized using Markov decision processes (MDPs), which is a conditional sequence of two types of actions: 1) actions of transferring decision making and 2) actions to change the pre-defined cooperation with team agents, which aims at minimizing miscoordination costs. Such strategy helps minimize the disruption to team coordination with high individual decision making.

Derived from the autonomy strategy proposed by Maheswaran, Myers defined the autonomy strategy as five types to make the autonomy strategies more specific and practical for usability evaluation (Myers 2007):

- **Completely autonomous:** AI in the system perform all the tasks;
- **Conformate assistant's actions:** AI's actions need to be approved by user;
- **User gives assistant strong guidance:** AI perform the tasks based the given instructions;
- **Assistant's weak guidance:** in the contrary to the previous strategy, AI in system will give user instructions about what actions might be done;
- **Directly manipulation:** the conventional systems that all actions are manipulated by the user.

## 2.5. IIS applications

Although current AI implementations are all weak AI that could accomplish certain tasks in specific fields, there are wide explorations of the possibility to integrate AI into design works. With the rapid development of AI research in recent decades, the AI algorithms now have the ability to create artifacts with decent results. Especially the machine learning based AI nowadays could play a collaborator in creating music, drawing and other creative tasks. Among these experimental implementations and machine learning

algorithms, generative models are widely used as intelligent agent in user-AI co-creations. The details of generative model will be introduced in chapter 4.

The following sections will introduce current state-of-the-art user-AI co-creation examples. The examples will be presented separately in musical composition area and drawing area, which are two typical creative areas considered unique to human.

### 2.5.1. User-AI co-creative works for music

Music is a field full of attempts combining composing with AI techniques. One example is *A.I. Duet* (Google AI Lab, 2016), an experiment designed by Google to explore the cooperation between human and AI. It is built on the Tensorflow framework, Tone.js and the open source tools of the Magenta project.

*A.I. Duet* is an online interactive piano, when the user plays a small number of notes, it automatically generates chords to accomplish user's creation and keeps the consistency of the music. As is shown in Figure 2, the user input through virtual keys and the notes are presented as yellow blocks. Correspondingly, the *AI* will generate the melody as a chord to user's notes, it is visually presented as blue blocks. It shows one example how machine learning algorithm can inspire people in creative works.

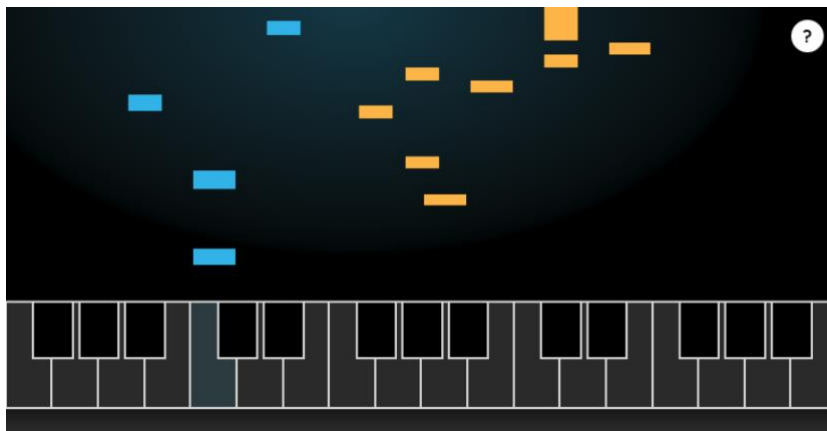


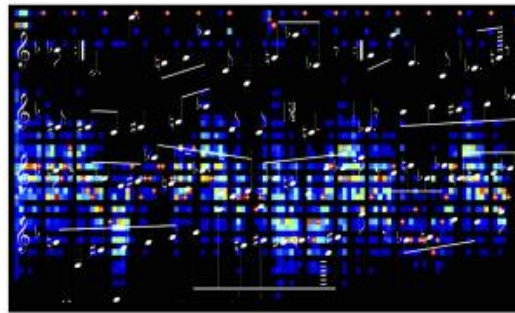
Figure 2. A.I. Duet webpage view

From the aspects of machine learning algorithms, there are a lot of research works on improving the music composing performance as well. Huang and Wu created a model of learning long-term music and capable of generating music with complex structure and rhythms (Huang and Wu, 2016). As an outcome of the overview about machine learning algorithms for music-composing, Sturm et al made a concert with these music generation algorithms, billed as “the first concert ever in which all of the music played has been written by a computer” (Sturm et al., 2019).



# PARTNERSHIPS

Music composed by and co-composed with computers having "musical intelligence"



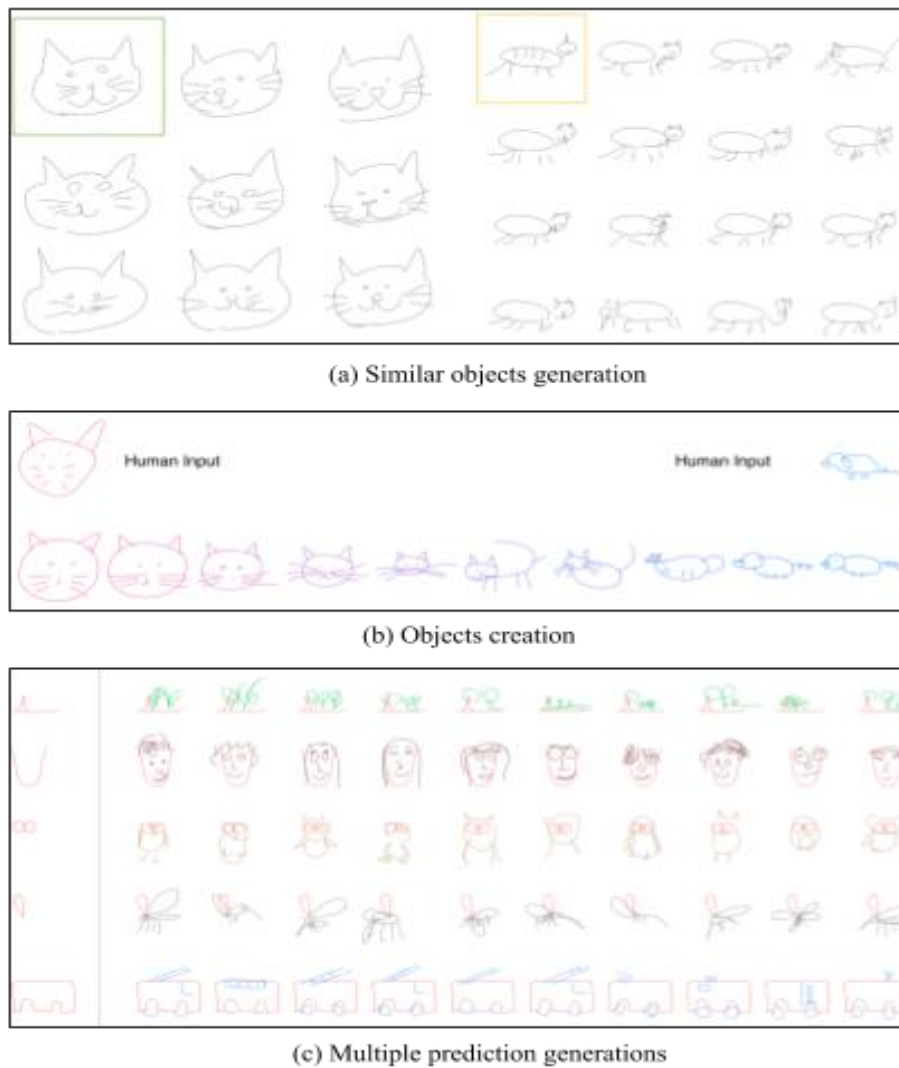
<i>Pieces for organ: The Glas Herry Comment &amp; X:7153</i> by folk-rnn + DeepBach (2017)	Richard Salmon organ
<b>Traditional Irish Sets</b> (with folk-rnn tunes in italics) <ul style="list-style-type: none"> <li>• <b>Jigs</b> (The Cull Aodha, The Dusty Windowsill, <i>The Glas Herry Comment</i>)</li> <li>• <b>Slow Reels</b> (Maghera Mountain, X:2897)</li> <li>• <b>Fast Reels</b> (The Rookery, X:1068, Toss The Feathers II)</li> </ul>	Daren Banarsë and Musicians
<i>March to the Mainframe, Interlude, The Humours of Time Pigeon</i> by Bob L. Sturm + folk-rnn (2017)	Ensemble x.y
<i>Ed SheerAI vs XenAkis vs Aldele</i> by Nick Collins (2017)	Ensemble x.y
<b>3 morphed pieces from "A Little Notebook for Anna Magdalena"</b> by J. S. Bach (1722) + MorpheuS (2017) <b>3 morphed pieces from "30 and 24 Pieces for Children"</b> by Kabalevsky (1937) + MorpheuS (2017)	Elaine Chew piano
<i>Safe Houses</i> by Úna Monaghan + folk-rnn (2017) <i>The Choice</i> by Úna Monaghan (2015) <i>The Chinwag</i> by Úna Monaghan (2015)	Úna Monaghan Irish harp, concertina, electronics
<i>Pieces for organ: X:633 &amp; The Drunken Pint</i> by folk-rnn + DeepBach (2017)	Richard Salmon organ
<i>Chicken Bits and Bits and Bobs</i> by Bob L. Sturm + folk-rnn (2017)	Ensemble x.y
<i>Bastard Tunes</i> by Oded Ben-Tal + folk-rnn (2017)	Ensemble x.y

Figure 3: Playbill for the AI concert. From Sturm's work (Sturm et al., 2019)

## 2.5.2. User-AI co-creative works for drawing

For drawing tasks, the algorithms applied are mostly generative models. Such algorithms and applications are mostly derived from computer vision field. CNN (convolutional neural network) (LeCun, 1998) and RNN (recurrent neural networks) (Pearlmutter, 1989) are widely used to solve computer vision problems nowadays. Based on such neural network models, many new interfaces have been developed for creative works. The typical examples of user-AI co-creation drawing examples are *Sketch-RNN* and its related applications (Ha and Eck, 2017), *Paintschainer* (Yonetsuji, 2016) and *Photo colorization*.

*Sketch-RNN* is an AI algorithm proposed by Ho and Eck in 2017 (Ha and Eck, 2017). It is a generative recurrent neural network (RNN), which can draw sketches of ordinary objects in a human-like way and summarize abstract concept. Derived from this work, many innovative interactions were developed to explore the possibility of co-creation between human and AI. There are mainly 3 types of applications derived from *Sketch-RNN*: 1) Reconstructing similar objects; 2) Creating new objects; 3) Predict and complete unfinished sketches.

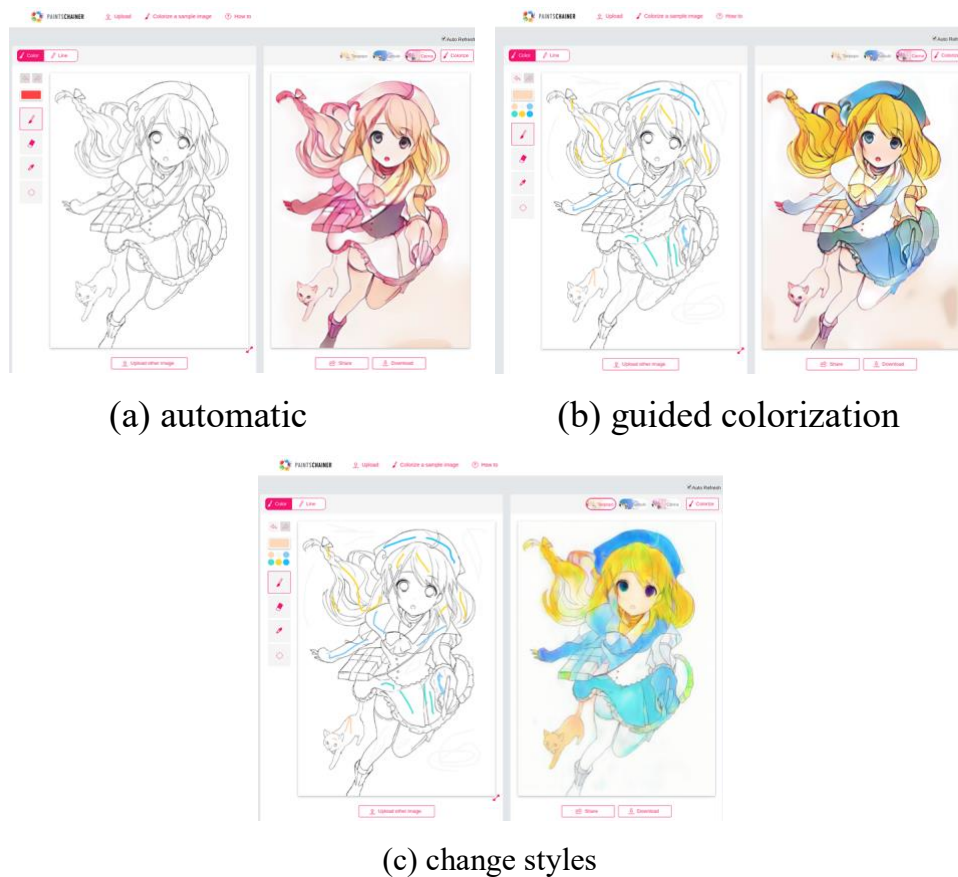


**Figure 4: The examples generated by Sketch-RNN (Ha and Eck, 2017)**

The reconstruction of similar object is a function provided for pattern designers. One example given in *Sketch-RNN* work is that for textile or wallpaper pattern designing, designers need to create multiple patterns. With the help of *Sketch-RNN*, a large number of similar, but unique pattern designs can be generated based on one example given by pattern designer. Figure 3.4.a shows the example, the sketches inside the green and yellow frames are human-made sketches. Based on these inputs, *Sketch-RNN* generated several unique but similar sketch patterns.

The sketch information learned by Sketch-RNN is encoded as latent vectors, which makes model learn representations of multiple objects. By interpolating the latent vectors, Sketch-RNN is able to morph from one drawing to another drawing. As is shown in Figure 4 (b), *Sketch-RNN* generates "cat-pig" sketches from interpolated latent vectors combined with a cat and a pig. In some of the sketches, *Sketch-RNN* successfully made some creations by attaching the cat head with a pig body. In this condition, user could cooperate with AI to make novel creations.

Users can also draw pictures with *Sketch-RNN*. The decoder module of *Sketch-RNN* could be trained to predict different possible endings of incomplete sketches. In such case, it could work as an assistant to help designer finish their works by suggesting alternative ways and inspirations. Figure 4 (c) shows the example that *Sketch-RNN* finished the uncompleted drawings of grass, face, bird, mosquito and bus.



**Figure 5: Paintschainer examples (Yonetsuji, 2017)**

*Painstchainer* is an online comic colorizer with a CNN-based algorithm as backend. It offers two interaction methods for user to support their creative works: 1) fully automatic generation. In this condition, *Painstchainer* is only given a single line sketch as its input, the CNN-based algorithm could automatically finish the colorization of the comic draft. 2) guided generation, in which condition user could draw indicates on specific area, which tells AI what color would be preferred to be used on the painting. The figure 5 (b) shows the guided generation result. Besides, *Painstchainer* offers three painting styles for user to choose. Figure 5 (b) and (c) show the different painting results are generated based on different painting style with the same input.

Another example of such application is automatic photo colorization. Zhang et al. developed a real-time user guided image colorization model in 2017 (Zhang et al., 2017). It is a deep learning approach that "directly maps a grayscale image, along with sparse,

local user 'hints' to an output colorization with a Convolutional Neural Network (CNN)" (Zhang et al., 2017). As is shown in Figure 6, the CNN model colorizes a grayscale image (left), which is guided by sparse user inputs (second). Multiple plausible photo colorizations could be generated in real-time (middle to right).



**Figure 6: Real-time user-guided image colorization, proposed method Photograph of Migrant Mother by Dorothea Lange, 1936 (Public Domain). (Zhang et al., 2017)**

### 3. Designing challenges for Intelligent Interaction Systems

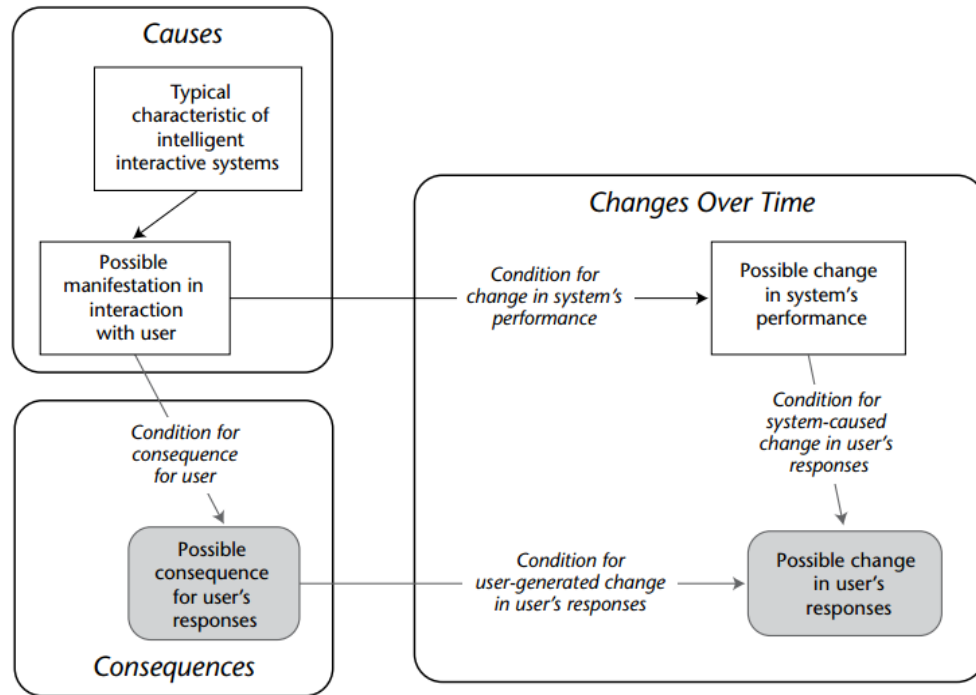
As is introduced above, although these IIS examples and implementation are still experimental, these works explored the potential about how to improve the design works with AI technique. To understand what criteria should be used for user experience evaluation in intelligent interaction system, an overview of design challenges for intelligent interaction system will be provided in this section.

As a subset of user experience study, the evaluation metrics of user interfaces could be applied to evaluate intelligent user interfaces as well. However, there are two features that make evaluation of intelligent interaction system different from the conventional interaction system: 1) From the aspects of HCI, besides studying how to make user perform appropriate actions, the evaluation should also focus on “to incorporate knowledge to be able to assist the user in performing actions” (Hartmann, 2009); 2) From the AI study perspective, the developed AI algorithm should also contribute to user-computer interaction, rather than just make intelligent agent smarter (Hartmann, 2009). Thus related research on challenges of intelligent interaction systems are: 1) analysing from the perspective of **AI' features** (James, 2009) and 2) analysing the challenges from the perspective of intelligent interaction systems' **goals** (Hartmann, 2009).

By going through the challenges, this chapter will provide the understanding of the intelligent interactive system design. In the end of this chapter, a list of design considerations are concluded based on the discussions.

#### 3.1. Challenges from Intelligent Interaction Systems' features

From the **AI features' perspective**, the challenges were studied by James as “side effects” in his work (James, 2009). James proposed a general schema that analyzes usability side effects as is shown in Figure 7. In total, the side effects were defined based on 4 aspects: 1) The **causes** of the side-effects, which highly depend on the features of the intelligent systems; 2) The possible **consequences** of how those features influence users' behaviors and experiences; 3) How side effects **change over time** with more experience acquired by users and adapted by intelligent systems; 4) The **prevention** strategy to reduce such side effects.



**Figure 7. Schema for Analyzing Usability Side Effects. Rectangles: properties of system; rounded boxes: responses of user. (James, 2009)**

In total 9 specific types of side effects were defined: Switching between applications or devices; teaching the AI agent; narrowing experience; unsatisfactory aesthetics or timing; learning process is required for users; inadequate control over interaction styles; threats to privacy; inadequate predictability and comprehensibility; imperfect system performance

**Switching between applications or device** is the situation where is hard to integrate AI technology into the application. In this case, users have to switch between AI and application. This challenge obviously reduces the efficiency and users spend more time and effort.

**Unsatisfactory Aesthetics or Timing.** As integration of AI and interaction systems could be problem, it may make the systems less visually satisfying and familiar compared to conventional systems. It is also a problem that exists in the traditional interaction systems. However as discussed above, combining AI and conventional interaction systems could be irreconcilable. Unsatisfactory aesthetics could be more common than traditional systems. Besides, the AI algorithms' timing performance is another factor to be consider. Jameson pointed out AI brings additional phenomena: 1) The system will automatically generate hints for users which are needed by them, which Jameson introduced as "proactivity"; 2) To give users certain level of control, the system would need extra inputs.

**Need to Teach the System** and **Threats to privacy** are highly connected. **Need to Teach the System** is caused by implemented algorithms. To perform the tasks, the AI algorithm needs to be trained based on the knowledge acquired from the users or from the tasks performed. This will also be time-consuming. Jameson mentioned that after initial interaction, the AI would need less information from users and users will become skillful. However, in certain cases if the algorithm fails in learning, it will make system non-intelligent and reduce the usability.

**Threats to Privacy.** Although privacy issues have been widely discussed in AI areas, it still is a challenge that brings side-effects in IUIs. Many AI algorithms need training to perform certain tasks, which requires training data. The intelligent systems infer users' behavior based on collected data without users knowing how the data is used. Besides, the customized system's behavior may reveal users' information such as preferences or interaction history. One example is the recommendation system (Liu et al., 2016), where the systems expect to acquire users' information. Based on such data the IUIs appearance and contents will be customized for users. From the perspective of user experience, it can make users feel uncomfortable and unwilling to trust the system.

**Need for Learning by the User** and **Inadequate Predictability and Comprehensibility** are related. They are different aspects of the challenge concerning AI's incorporation. **Need for Learning by the User** is the situation where users are not familiar with the AI features, they also need to be familiar with the context of the usage. Same as previous side effects, learning is also time-consuming, as the learning process is a long-term process.

**Inadequate Predictability and Comprehensibility**, as is discussed above, integrated AI brings new feature to interaction systems. It brings uncertainty to the system. Users may be less able to predict or understand systems' behavior, as AI algorithms especially machine learning are considered as black box. (Samek et al. 2017; Liu et al. 2017).

Since user lacks understanding, it would become harder for users to assess the system's performance, explain and understand the systems actions. As the result, intelligent systems will lack predictability, comprehensibility and understandability. The unexpected outputs of AI also make users wonder the reason of unmatching and try to fix them. However, one positive aspect is that some unexpected but task-relevant results bring users surprises, thus increase satisfactory.

**Narrowing of Experience**, refers AI will restrict or narrow users' abilities of accomplishing the task without AI. The situation is that AI tends to take over the works or give instructions on normal behaviors that users could accomplish by themselves. Besides, it may also make user less skilled.

**Inadequate Control over Interaction Style.** This could be caused by various aspects. The main reason is that the integrated AI limits the customization. Besides, users may also want to avoid privacy violation and take control to achieve better performance. These issues are related to **Threats to Privacy** and **Imperfect System Performance** as well.

As the interaction with IUIs is somehow like collaboration, getting AI interaction correct could be a challenge. It would be more difficult to design multiple interactions. Besides, it may be more difficult to design a method that allows the user to explicitly control the interaction style of the system. If the interaction cannot be changed, the consequence is that users may feel frustrated and get unsatisfied with the systems. Eventually, users may get used to the interaction, while in certain cases, users may still fail with tasks due to the frustration.

**Imperfect System Performance** is the case when IUIs generate errors or suboptimal results that needs to be corrected or refined by the users. As discussed in Chapter 1, currently most of the AI technologies are generally considered as weak AI. Depends on the task, used algorithm, training sets and other factors, AI's performance varies as well. Jameson pointed out that as AI's performance is unstable, it cannot guarantee the accurate outputs. If the system frequently attempts to generate behaviors which make users consider the system is not intelligent enough, the frustration will be caused. In different tasks, the imperfect system behavior has different level of influence as well. Correspondingly, imperfect performance will make system less convincing and confusing. The other consequences are similar as discussed in **Inadequate Predictability and Comprehensibility**.

In this thesis, to understand current machine learning algorithms performances, the evaluation overview will be introduced in Chapter 5.

### **3.2. Challenges from Intelligent Interaction Systems' goals**

The second way of analyzing challenges in intelligent interaction systems is from the **goals** of intelligent interaction system. Early in the 1995, Maes mentioned that AI in interaction systems should be able to "be used to implement a complementary style of interaction", which means that AI works as an assistant in this cooperative process should be effective (Maes, 1995). To build such intelligent agent system, Maes introduced three main problems that need to be considered: 1) **Competences**, the capability of the AI: it is able to assist users, how to help and when to help; 2) **Trust**, how to make users comfortable and willing to use the intelligent interaction system. 3) **Interface** issues. As Maes mentioned, it is an open question about how the interface agents should be presented and integrated in the system (Maes, 1995).



In Mayers' research, the goal was defined that intelligent agents should be helpful. From this perspective, 3 sub-goals are defined to better describe the goal of intelligent interaction systems: **usable**, **useful** and **trustable** (Hartmann 2009, Myers 2007).

In Hartmann's work, he defines that the goal of such system is to provide more effective and efficient interaction, as well as the presentation of information to support users' needs (Hartmann 2009). Hartmann considered the intelligent interaction systems' goals to be matched with the three challenges proposed by Maes. By giving more details based on the Maes' discussion (Maes 1995), the overview of the 3 challenges were given by Hartmann as shown in Figure 8.

<b>Presentation</b>	Interaction design
	Unobtrusiveness
	Adaptivity
<b>Competence</b>	Few usage data
	Changing user behavior
	Accuracy
<b>Trust</b>	Controllable behavior
	Intelligibility
	Privacy

**Figure 8: Challenges in developing user-adaptive UIs. (Hartmann, 2009)**

**Presentation** refers to the challenges in designing interaction part of intelligent systems: The first sub challenge is **designing the interaction**, which means that the AI should be naturally integrated into the system and should not hinder the normal usage of the application (Hartmann 2009, Apple 2008). It is also related to how user should control the system and how their expectation should be raised (Hartmann 2009). The second sub-challenge is **unobtrusiveness**, that AI in the system should not be distractive (Hartmann 2009; Jameson, 2007; Langley and Fehling, 1996). The last sub challenge is that the intelligent interaction system should be user-adaptive. **Adaptivity**, means it should adjust its presentation to different users and situations. It will not only influence the visual presentation, but also the users' trust and competence of the tasks.

**Competence** of the intelligent interaction systems usually depends on the implemented AI algorithms. From this aspect, it is important to be aware of the algorithms

that require large amount of data for training to perform appropriately. Thus for adaptive intelligent interaction system it should have the ability to work functionally with a few training data. Besides, users' behaviors change over time (Hartmann 2009; Höök, 2000). Höök proposed that such algorithm or system should be able to adjust the importance weights of recent action patterns for AI. Related to both competence and trust, the system should also be **accurate** (Hartmann 2009 ; Leetiernanet al., 2001).

**Trust** is determined by many factors. First it is related to **presentation**. To make the system trustable, users need to have the feeling of **control**. The system should offer methods for users to manipulate the actions and autonomy (Hartmann 2009; Höök, 2000; Bellotti and Edwards, 2001; Glass et al., 2008; Dey and Newberger, 2009). Although controllability is important, too much control all the time may also be distractive or time consuming, thus reduce the usability (Hartmann 2009; Jameson and Schwarzkopf, 2002; Kay, 2001). Besides, the AI should have **intelligibility**, at least the users should understand the systems' actions. To make the intelligent system trustworthy, it should meet following features: 1) **transparency**, users are able to understand the system's action; 2) user could have the **access to the knowledge** of the system's model (e.g: the principle how it works); 3) the system's actions should be **predictable** so that it could match user's expectation; 4) the system needs to concern user's **Privacy**.

To address the importance of adaptivity of the intelligent system, Hartmann proposed extra sub factors based on user-adaptive intelligent interaction system (Hartmann 2009). As is discussed by Hartmann, user usually measure the adaptivity with the usability of the interface, such as efficiency, effectiveness and satisfaction. From aspect of adaptivity, the factors to improve the usability, Hartmann summarized a table of features to improve the usability as is shown in table 9 (Hartmann 2009).

<b>Presentation</b>	Spatial Stability Locality
<b>Competence</b>	Accuracy Predictability
<b>Further factors</b>	Interaction frequency Task Complexity Average interaction costs

**Figure 9: Factors influencing the value of an adaptation for a user (Hartmann**

2009)

*Spatial stability* and *locality* were proposed to contribute usability from **presentation** perspective. *Spatial stability* is required to increase user satisfaction, as user could maintain the mental model of the system (Hartmann 2009). *High locality* is related to *spatial stability*, which means the presentation of the system is similar as the conventional system without AI integration. As Hartmann concluded, it improves the discoverability of the adaption.

*Accuracy* and *predictability* are proposed from the **competence** perspective. Considering the AI algorithm's performance, increased accuracy contributes the user satisfaction and user's efficiency (Hartmann, 2009; Gajos et al., 2008; Tsandilas and Schraefel, 2005). As a consequence, the accuracy also influences the user's perception of the system's predictability (Findlater and McGrenere, 2008; Hartmann, 2009). The better accuracy increases the predictability and consistency.

### 3.3. Challenges for Designers

Generally for UX design, combining the UX and ML in design could help designers make better decisions. For example, ML algorithms help the designer to better predict users' behavior and unique preferences (Carmona 2018; Lepp 2014), which leverage the work. It also helps designer collect and analyze data in real time and create more reliable user pictures (Wikström, 2018).

However, machine learning algorithms not only bring positive enhancement for UX design, there are still usage difficulties for UX designers. Three challenges of integrating ML and UX designing are raised in Dove's study:

- Understanding the principles and capabilities of ML;
- Appropriately integrating ML algorithms.
- Challenges with the purposeful use of ML (Dove et al. 2017)
- Ethical issues of ML (Carmona 2018).

**Understanding the principles and capabilities of ML:** designers may feel it is hard to understand the capabilities and limitations of ML. As the performance of ML algorithms usually depends on the big data provided for model training, lack of data or dirty data may mislead the design strategies. As is shown in the studies conducted by Yang (Yang et al, 2018), the participants can hardly tell the principles of the algorithm (Dove et al, 2017; Carmona 2018). In Dove's study, the AI algorithms are considered as a black box to users. Their participants state that "We designers do not understand the limits of machine learning and what it can/can't do. Machine learning experts often complain to me that designers act like you can just sprinkle some data science onto a design and it will become automatically magical" (Dove et al, 2017).

**Appropriately integrating ML algorithms:** Due to lack of understanding, the designers may not be able to appropriately implement ML algorithms. It may make designers overlook the advantages, underestimate the potential usages or even limit the innovation (Dove et al, 2017; Carmona 2018). Firstly, ML prototyping is hard without actually having the ML model and data. In Dove's studies one response is that "Machine learning is hard to prototype. Machine learning requires highly skilled collaborators" and "...making interactive prototypes that incorporate machine learning is hard (haven't found a way to do that yet in an easy fashion)" (Dove et al, 2017). Besides, the performances highly rely on the training procedures and data. If the system acts unstably, designers and users may consider the system non-intelligent, unreliable, and unintuitive (Yang et al.,2018; Yang and Newman, 2013; Wikström, 2018).

**Challenges with the purposeful use of ML:** AI should be appropriately used in the system with certain purpose (Dove et al. 2017). Designers should consider if the AI's integration should be user-oriented or task-oriented. Dove's study emphasizes human-centered perspective in ML. Their participants responded that the design may be more engineered-led, rather than design-led or equally-led.

**Ethical issues of ML:** The ethical issue is the last challenge for utilizing UX and ML, which is widely discussed throughout the history. One ethical issue is who should be responsible for the intelligent-system's error, designers or ML algorithms. As shown in Dove's study, one response is that "...can it be trusted to make decisions or take actions on its own?" (Dove et al. 2017). Thus how to utilize the human factor and ML is an important challenge to be considered.

As concluded in Lovejoy's work, the relationship between UX and ML should be "human-centered machine learning". The designers should have correct understanding of ML's principles and capabilities. The relationship between developer, designer and algorithms should be balanced.

### **3.4. User study of user-AI co-creative interface**

Evaluating user-AI collaboration design tool is one of the tasks in this work. Thus it is necessary to understand what evaluation metrics should be applied and which aspects should be evaluated. Although the user experience evaluation metrics are mature and have been widely used in many research works, there is not a general evaluation metric for intelligent interaction system yet.

Before designing the user-AI cooperation interface for UI design, it is important to understand what one should be aware of when integrating an intelligent agent as a collaboration partner.

This section will introduce three related studies about user-AI co-creative interaction systems. By developing or integrating the state-of-the-art algorithms as the co-creative partner, these works attempted to analyze and understand users' perceptions of new user-

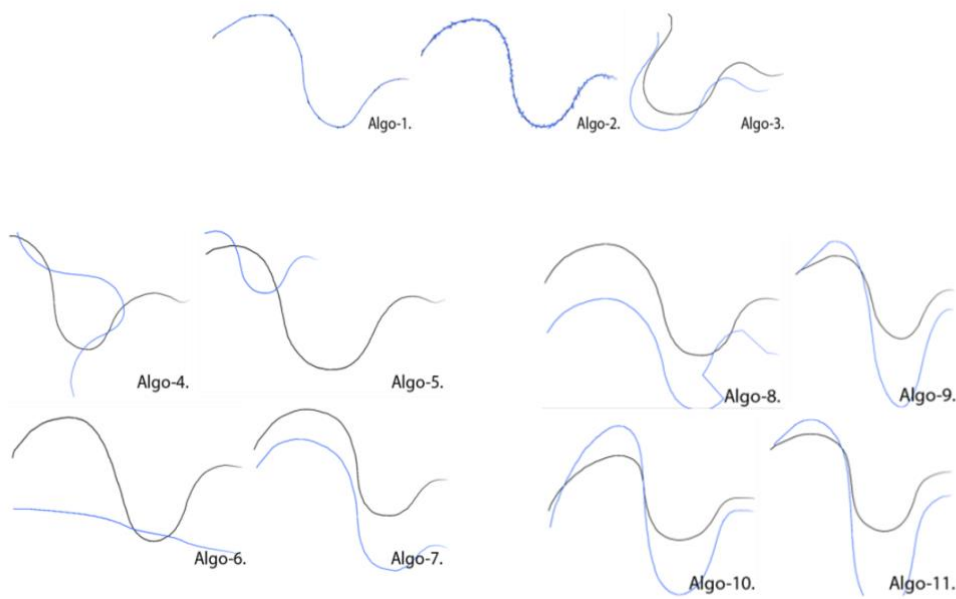
AI cooperation interfaces. These studies mainly analyzed the user-AI creative cooperation from these aspects:

- 1) The work of Drawing Apprentice analyzed the users' perceptions towards user-AI creative cooperation from cognitive aspect. The researchers attempted to understand how users make sense of the system's behaviour and their contributions together with AI. By comparing human-human collaboration and human-machine collaboration, design recommendations were concluded (Davis et al. 2015, 2016).
- 2) The work of DuetDraw provided detailed study from usability aspect. It not only generally studied how AI techniques affected the user experience, but also analysed and discussed the communication and leadership between human and machine (Oh et al. 2018).
- 3) General behaviour observation. In the work of ColorAize, the researchers simply observed users' behaviours and reactions (Matulic 2018).

The work of Drawing Apprentice and DuetDraw provided design recommendations and guidelines for user-AI creative cooperation. The following sections will introduce the related research and conclusions. By going through the studies, the evaluation criteria will be concluded in the final of this chapter.

### **3.4.1. Drawing apprentice**

As one of the first applications combining machine learning and drawing in recent decades, Davis et al created an Enactive Co-Creative Agent for artistic collaboration. User draw lines as inputs, the agent in the system will transform the lines based on pre-encoded line transformation techniques, and outputs the transformed line on the drawing panel. The features of the input lines are sampled by clustering the data points and the collected data, which are post-processed by the neural network. Based on the data, the neural network will generate the classification schema, which will be used for the user-AI cooperation task. In total there are 12 experimental line transformation styles implemented as is shown in Figure 10 (Davis et al. 2016). The AI is able to choose proper sketch schema and accomplish the sketch drawing with the user.



**Figure 10: The 11 types of drawing results from intelligent agent (blue lines: agent; black lines: human) (Davis et al. 2016)**

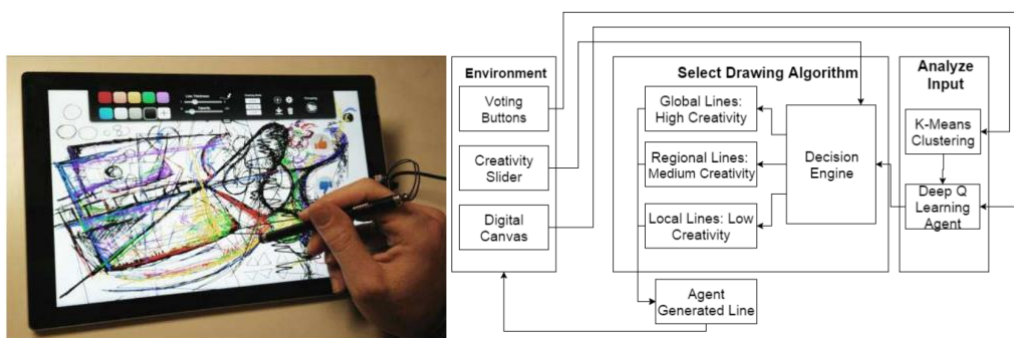
Derived from this work, Davis et al. did a detailed study on the co-creation evaluation and user experience evaluation, *Drawing Apprentice*. In an early study of the system evaluation (Davis et al. 2016), users reported that the *Drawing Apprentice* motivates them continue the creative tasks to explore intelligent agent's ability. The results generated by AI are both impressive and confusing, as the system sometimes understands user's intention while sometimes the system could be unpredictable. Besides, the user's mental models seemed to attribute a greater degree of 'intentionality' and 'creativity' to the *Drawing Apprentice* system than researchers predicted (Davis et al. 2016).

In their later studies, they mainly focused on users' participatory sense-making of the co-creative cognitive agent. The questions were proposed as follows: "1) To what degree was participatory sense-making present during the collaboration; 2) What metrics and features did users employ to determine whether contributions 'made sense'? 3) How did users try to define shared meaning structures with the agent, i.e. how did they attempt to teach the system?". By conducting a Wizard of Oz comparison study between human-human collaboration and human-machine collaboration, they found that *Drawing Apprentice* can engage users in participatory sense-making, thus resulted in discovering novel visual concepts and emerging meanings.

By analyzing the qualitative data and evaluating user's behaviours, the results of their study indicated the following:

- 1) Spatial awareness, visual similarity determination and perceptual logic are critical for user to make sense of the AI generations.

- 2) Spatial awareness refers that AI generated visual elements should be close to the previous visual elements created by user, so that user's awareness would not get distracted. It is a foundational skill for a co-creative drawing AI.
- 3) Visual similarity means that AI's output should retain some visual similarity to the user's contribution. Users should be aware that there are connections between their interactions and AI's contributions, so that AI's generation is understandable.
- 4) In Davis's case, perceptual logic refers the case that when the user and AI collaborated to complete the drawing, the structure of the sketches should be logically related. In the target region, the visual elements should be relevant and logical. Perceptual logic is also subject to change as regions change and interact with other regions. It is important to be aware this dynamic.
- 5) Users should be able to fully understand the mechanism of operation of the system.



**Figure 11: The user-AI collaboration process of Drawing apprentice (Davis et al. 2016)**

### 3.4.2. ColorAIze

In Matulic's work, the researchers implemented an AI-Driven Colourisation of Paper Drawings with Interactive Projection System. Based on the algorithm provided in *PaintsChainer*, a physical system was developed. *PaintsChainer* allows the user to draw pictures on physical paper, and system will project the colored picture on it (Matulic 2018). As is shown in Figure 11, the user interface is composed of a projector, a color palette (on which user can select color and drawing functions, and 3 different drawing styles) and a drawing panel. By pressing the start button, a webcam will capture current user's drawing frame, the sketch will be transferred by the algorithm to generate results. One to two seconds later the result will be projected and displayed on the drawing panel.

In this work, the goal was to get as many visitors as possible to experience this tool. In total, more than a thousand visitors tried this tool. Instead of interviews, the researchers made observations about users' behaviors and spontaneous comments for analysis.

Based on the observations, they found most of the visitors thought ColourAIze brought a novel experience and was pleasant to use. Their conclusions are as follows:

- 1) The proper results generated by AI impressed users. Users were amazed as it could generate professional colorizations and finish the task almost in real time. 35% of the users were impressed as this tool brought up potential inspiration they did not imagine during the painting.
- 2) 16% of the users modified or created their own sketches before colourizing: 15% of the users drew their own comic sketches and then cooperated with the system to colourize. The users thought cooperating with AI to colourize the comic was entertaining and was possible to contribute to artists' works.
- 3) Although the AI generated best results with well-drawn sketches, users tend to explore the potential about how AI could colorize their own handworks.
- 4) Inconsistencies were observed in this work as well. This problem arised when users noticed that AI generated different coloring plan for the same input. Since the system took the real-time video as input, sometimes there will be fluctuations between frames. The AI algorithm did not establish the relationship between the frames, thus the colorization results could have huge changes. Thus some limitations of the AI influenced the usability as it reduced the consistency.







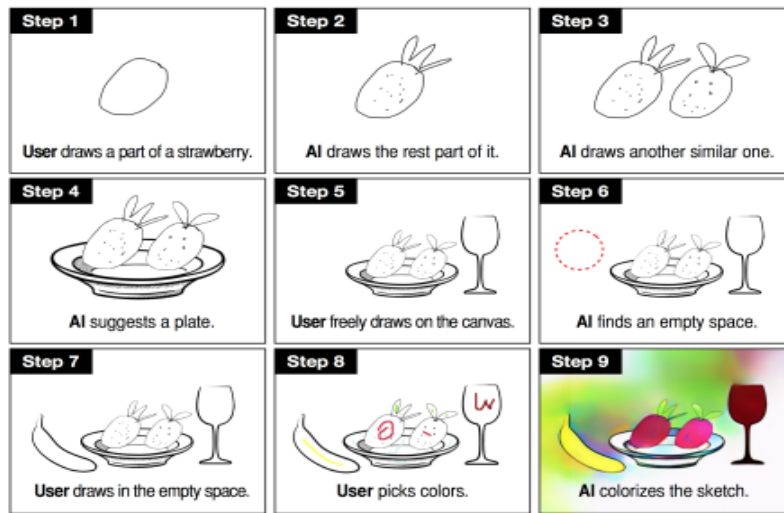
**Figure 11: The user-AI collaboration process of ColorAIze. From left to right: initial colourisation; user adding local color hint; result generated by AI (Matulic, 2018)**

### 3.4.3. Duet-Draw

To understand what factors influence different aspects of usability, and how cooperation should be designed in the user-AI collaboration in creative works, Oh et al. implemented a user-AI cooperating drawing system, *DuetDraw* (Oh et al. 2018). The experiments conducted were more specific than Matulic's observation experiments. Rather than cooperating colorization with AI, they integrated *Sketch-RNN* into their works. In this case, the whole painting process, sketching and colorizing could be operated in a collaborative way. As is shown in Figure 12, the basic process is user draws part of the sketch, meanwhile AI may take control or give hints about the rest of the objects. Once the sketching part is finished, the user and AI will cooperate to colorize the image. The AI algorithm is same used in *PaintsChainer* and *Sketch-RNN*.

In their work, the questions are mainly focused on these aspects: 1) How do users and AI communicate in creative contexts? 2) Would users like to take the initiative or let AI take it when they cooperate? 3) What factors are associated with the various experiences in this process?

When designing the co-creative tool and the experiments, they mainly took two factors into consideration: **initiative** and **communication**. Similar to what is described in Chapter 3.2, for these two aspects were considered from user-based and agent-based points of view. For **initiative**, they designed two initiative styles: 1) *Lead* style, in this mode users take the initiative and draw the major part of the painting, while AI finishes the secondary tasks; 2) *Assist* style, in contrast, AI will carry out the major work while users finish the rest. For **communication** styles, they designed a *detailed instruction* style, where intelligent agent gives detailed instructions for each operation, and *basic instruction*, AI will automatically proceed to the next step with basic notifications. For experiments, they also provided *no-AI* style, which has the same interface but no interactions with AI. Thus in total the participants were asked to do 5 interaction conditions during the test: (a) Lead-Detailed, (b) Lead-Basic, (c) Assist-Detailed, (d) Assist-Basic) and (e) no-AI. To reduce the bias they randomized the condition order when conducting the usability study.



<i>Step</i>	<i>Description</i>
1	The leader starts to draw a part of an object.
2	The assistant completes the rest of the object.
3	The assistant draws the same object in a different style.
4	The assistant draws another object that matches the objects.
5	The leader freely draws on the canvas.
6	The assistant finds an empty space to draw a new object.
7	The leader draws an appropriate object in the empty space.
8	The leader chooses colors and marks them on each object.
9	The assistant colorizes the sketch with the chosen colors.

**Figure 12: The user-AI cooperation process of DuetDraw and the descriptions. (Oh et al. 2018)**

To evaluate the user experience, in total 15 criteria were used : 12 commonly used UX evaluation criteria, 1) useful, 2) easy to use, 3) easy to learn, 4) effective, 5) efficient, 6) comfortable, 7) communicative, 8) friendly, 9) consistent, 10) fulfilling, 11) fun, and 12) satisfying) and 3 criteria for AI interface evaluation ( 13) predictability, 14) comprehensibility, and 15) controllability).

Based on the results, they concluded the following points as the design guidelines for user-AI cooperation tools:

- 1) User should take the initiative during user-AI creative cooperation. When creating contents, it is better that the user makes most of the decisions. AI partner should perform as an assistant.
- 2) Cordial and detailed communication is necessary. During the interaction, enough instructions can improve predictability, comprehensibility, and controllability. It is also important that instructions should be given at proper moment.
- 3) For creativity support tool, one of the goals is to motivate users' creative action. Thus interesting elements should be embed in system. Interesting elements contribute to user's creativity, meanwhile this feature also enhances the user experience and the interface's usability.

- 4) AI should present stable outputs. The unstable and inconsistent AI generations will make user feel frustrated.

### **3.5. Summary**

This chapter introduced the challenges of designing an intelligent interaction system from three aspects: the goal of intelligent interaction system, the feature of intelligent interaction system and the challenges faced by designers.

The challenges from the intelligent interaction system's features and goals provided good design guidelines for this thesis work. The discussion about how to apply these guidelines into the design and implementation are presented in Chapter 6.4 and 7.1.

From the user study aspect, although these three works analyzed the user's perspective toward user-AI cooperation from different aspects, there are several mutual findings and guidelines can be concluded as follows:

- 1) The interaction of user-AI creative cooperation makes the design task fun and interesting.
- 2) The AI could generate some unexpected elements, users could be impressed and inspired by the AI generations. However, this also lower the predictability and controllability.
- 3) The instructions are necessary to interpret the AI's behaviour or help user better understand the cooperation interaction.
- 4) AI can contribute to user experience in the aspects of usefulness, effectiveness, efficiency and fun.
- 5) AI would perform badly in predictability, comprehensibility, and controllability of user experience. However, lower predictability could improve user's enjoyment.

For designers, as is introduced above, the main challenge is that designer could be unfamiliar with the algorithms. This usually result in inappropriate integration of the algorithms and designs. Besides, too much focus on the AI algorithms could also bring up some usability problems.

On one hand, designers should be able to explain and understand the principle of the implemented AI algorithms, on the other hand, the designers should also find a balance point between the algorithm-driven design and design-led ideas. For this consideration, the next chapter will introduce the AI algorithm implemented in this work.

#### 4. Generative models: GAN, VAE and BicycleGAN

The machine learning based AI algorithms shows very promising opportunities to merge ML into services. The advantages brought by ML is that such techniques “will cause us to rethink, restructure, and reconsider what’s possible in virtually every experience we build.” (Lovejoy, 2018). Chapter 2.5 also introduced several machine learning based user-AI applications (*e.g Sketch-RNN, Paintschainer, etc*). Although current neural network based content generative models are considered as a black box, understanding the algorithm's principle is still necessary for designers and developers.

In the field of machine learning, the models of machine learning can usually be divided into two types, discriminative model and the generative model (Ng and Jordan, 2002).

Discriminative model refers to the model that directly learns the decision function  $Y=f(X)$  or the conditional probability distribution  $P(Y|X)$  from the data as the prediction model. The basic idea is to establish a discriminant function with finite sample conditions, and acquire the prediction model without considering the sample’s generative model (Srihari, 2010). Typical discriminant models include k-nearest neighbors, perceptrons, decision trees, support vector machines, etc.

For generative models, the data is learned from the joint probability distribution  $P(X, Y)$ , and then the conditional probability distribution  $P(Y|X)$  is obtained as the predicted model. In such case, the generative model is represented as :  $P(Y|X)= P(X,Y)/ P(X)$ . The basic idea is to first establish the joint probability density model  $P(X,Y)$  of the sample, and then get the posterior probability  $P(Y|X)$ . With such model, it is possible to do discriminative tasks or generative tasks of sampled data Model (Ng and Jordan, 2002).

The generative model can be roughly divided into three categories according to the algorithms: autoregressive models, Auto-encoder models (AE), and Generative Adversarial Nets (GANs) (Goodfellow, 2016). The generative model has been well-applied in many fields. Among those, image synthesis is the most typical field that uses generative models. However, the quality of generated synthesized images still needs improvement. Besides, there is not a uniformed image generation quality evaluation metric yet. In this work, one of our goals is to use the state-of-the-art generative models for UI creation tasks. To better design and implement the intelligent interaction system, it is necessary to understand the principle of the generative algorithm.

This section will introduce the algorithms that were used in this work. The AI algorithms introduced in this section are mainly Neural Networks based approaches. Several generative models will be introduced as the foundation to better understand the final implemented algorithm: from VAE (Variational Auto-encoder) to GAN (generative

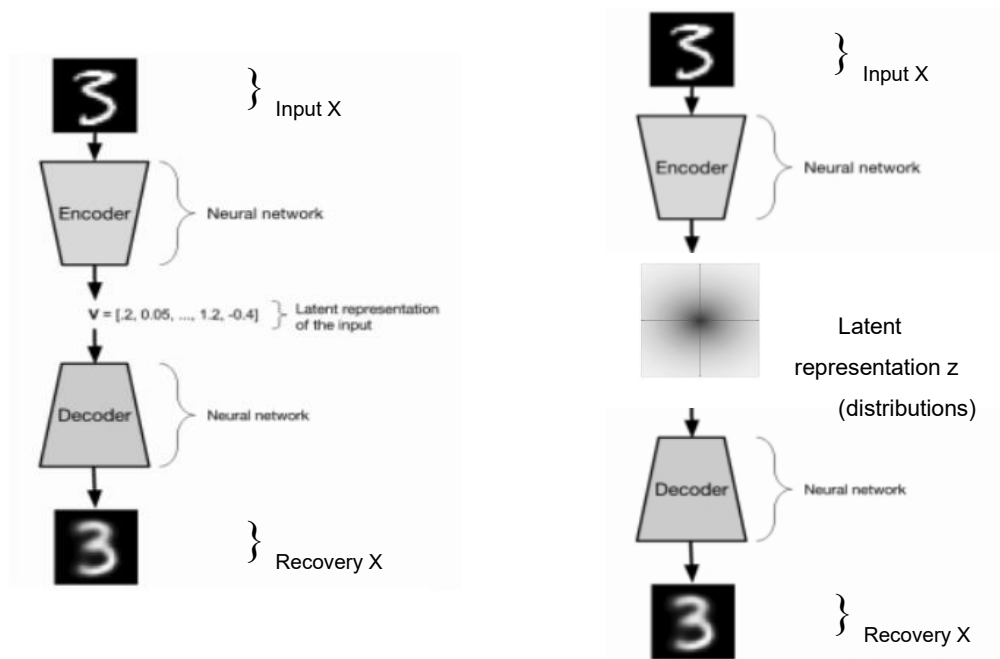
adversarial network) and the variant of this algorithm, BicycleGAN. Based on these introductions, the detailed explanation of BicycleGAN is given, which was chosen as the AI partner for UI design tool.

#### 4.1. VAE - Variational Auto-encoder

To understand what variational auto-encoder (VAE) is, the first concept to be introduced is auto-encoder. The auto-encoder (Schmidhuber, 2015) was originally used as a data compression method, and now it is mainly used in following aspects: 1) data denoising, 2) visually dimensionality reduction and 3) generating artificial data.

Auto-encoder usually contains two parts: the first part is the encoder and the second part is the decoder. Nowadays the neural network models are commonly used as the encoder and decoder (Kingma, 2013). It first compresses the observed vector  $X$  from the high-dimensional space into a low-dimensional vector  $V$ , which is regarded as the latent representation of the input data. Then the decoder decompresses the low-dimensional vector to reconstruct  $X$  through the decoding layer. The reconstructed data and real data should be close to each other.

As shown in the figure below, the encoder first compresses a sample handwritten font 3, encodes it into a latent vector  $V$ , and then reconstructs the original sample through network decoding. The latent vector  $V$  is the low-dimensional representation of the sample data. Thus the goal of auto-encoder is to train a model to compress the input data into a low-dimensional feature representation, and reconstruct the original data from the encoded features. To measure how well the recovery data is generated, L1 norm or L2 norm are usually used to measure the element-wise similarity between original data and recovery data. Thus the autoencoders are trained to minimise reconstruction errors (loss):  
$$L_{reconstruction} = \| x_{input} - x_{recovery} \|_2 .$$



**Figure 13: Auto-encoder (left) and Variational auto-encoder(right)  
(Isaac, 2016)**

However, the auto-encoder model has following problems (Schmidhuber, 2015):

- 1) The generated data is highly correlated with the training data, which means that the auto-encoder can only compress data similar to the training data. This is actually quite obvious, because the features extracted using neural networks are generally highly related to the original training set;
- 2) The data after compression usually loses information. It is inevitable due to the dimensionality reduction;
- 3) It is not able to generate new samples. The standard auto-encoder's target is only dimension reduction (feature extraction) and simply reconstruction

As a variant of auto-encoder, variational auto-encoder (VAE) is an important generative model proposed by Diederik P.Kingma and Max Welling in 2013 (Kingma, 2013). Its structure is similar to that of the auto-encoder that it is composed of an encoder and a decoder. The difference is that each latent attribute for a given input is represented as a probability distribution (such as normal Gaussian distribution). In this way, by inputting the original dataset  $X$ , the encoder can output a latent representation  $z$  in a distribution space  $Z$ ; By sampling the vector  $z$  from the encoded latent state distributions  $Z$ , the decoder model will be able to reconstruct the related original encoder's input  $x$ .

As is shown in Figure 13, to make the VAE able to encode input data and recover them, the sampling process is divided into two steps: (1) sampling the sample  $z$  from the latent distribution  $Z$  based on the probability function  $P(z)$ ; (2) Recovery the input data  $x$  according to the conditional probability distribution function  $P(x|z)$ .

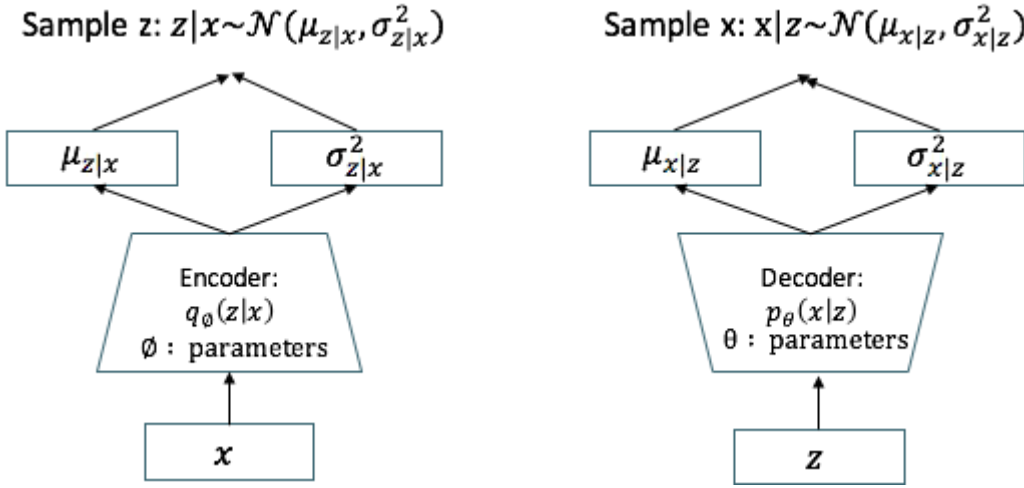
$$z^{(i)} \sim P_{\theta}(z)$$

$$x^{(i)} \sim P_{\theta}(x|z)$$

A set of input data samples  $X = \{X_1, X_2, X_3, \dots, X_k\}$  is given. Ideally the distribution  $p_{\theta}(X)$  can be acquired. For the decoder model, it is assumed that recovery  $x$  is generated from the latent representation  $Z$ , thus we have the data likelihood:

$$p_{\theta}(x) = \int p_{\theta}(x|z)p_{\theta}(z)dz$$

The goal of VAE model training is to estimate true parameters  $\theta$  for the generative model. However, computing  $p(x)$  is quite difficult as  $p(x)$  is an intractable distribution. To solve this problem, distribution  $q_{\phi}(z|x)$  is defined to approximate distribution  $p_{\theta}(x|z)$  so that it has a tractable distribution.



**Figure 14: A training-time variational autoencoder implemented as a feedforward neural network, where  $q_{\phi}(z|x)$  and  $p_{\theta}(x|z)$  are Gaussian. Left is the encoder model, right is the decoder model**

To ensure these two distributions are similar, a measurement of difference between two probability distributions is needed. For this purpose, KL divergence is chosen. Thus this equation needs to be minimized:  $argmin(KL(q_{\phi}(z|x)||p_{\theta}(z|x))$ ; On the other hand, the generative model also has the goal to get the distribution of  $X$ . Thus the data likelihood of  $p_{\theta}(x)$  could be defined as:  $log p_{\theta}(x) = E_{z \sim q_{\phi}(z|x)}[log p_{\theta}(x)]$ . Extending this equation we will have:

$$\begin{aligned}
 \log p_{\theta}(x) &= E_z \left[ \log \frac{p_{\theta}(x|z)p_{\theta}(z)}{p_{\theta}(z|x)} \right] \\
 &= E_z \left[ \log \frac{p_{\theta}(x|z)p_{\theta}(z)}{p_{\theta}(z|x)} * \frac{q_{\Phi}(z|x)}{q_{\Phi}(z|x)} \right] \\
 &= E_z [\log p_{\theta}(x|z)] - E_z \left[ \log \frac{q_{\Phi}(z|x)}{p_{\theta}(z)} \right] - E_z \left[ \log \frac{q_{\Phi}(z|x)}{p_{\theta}(z|x)} \right] \\
 &= \underbrace{E_z [\log p_{\theta}(x|z)]}_a - \underbrace{D_{KL}[q_{\Phi}(z|x) \parallel p_{\theta}(z)]}_b + \underbrace{D_{KL}[q_{\Phi}(z|x) \parallel p_{\theta}(z|x)]}_c
 \end{aligned}$$

In this equation, term a represents the decoder network that gives distribution  $p_{\theta}(x|z)$ , which is able to reconstruct the data from latent vector  $z$ . Since it measures samples' similarity element-wisely, it is also introduced as the element-wise reconstruction error in other works (Larsen, 2015); term b represents the KL term between Gaussians  $q_{\Phi}(z|x)$  for encoder and  $z$  prior, which makes the approximate distribution closer to prior distribution; c represents the KL term between the real distribution  $p_{\theta}(x|z)$ (which is intractable) and approximate distribution  $q_{\Phi}(z|x)$ . Although term c is intractable, based on the definition of KL (that the value of KL divergence is always positive),  $c \geq 0$ . Finally, variational lower bound is defined as follows:

$$\begin{aligned}
 \log p_{\theta}(x) &\geq L(x, \theta, \Phi) \\
 L(x, \theta, \Phi) &= E_z [\log p_{\theta}(x|z)] - D_{KL}[q_{\Phi}(z|x) \parallel p_{\theta}(z)]
 \end{aligned}$$

By maximizing the lower bound, the encoder parameter  $\theta$  and decoder parameter  $\phi$  can be found.

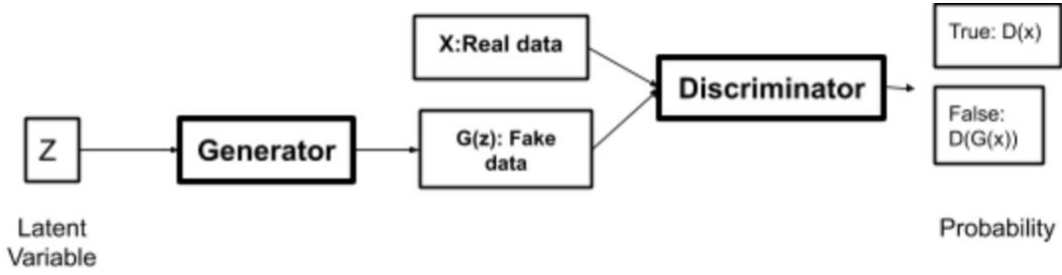
## 4.2. GAN (Generative Adversarial Network)

GAN (generative adversarial network) was first proposed by Ian J. Goodfellow in 2014 (Goodfellow, 2014). This is an innovative way to learn the basic distribution of data, allowing the generated artificial objects to achieve striking similarities with real objects. The idea behind GAN is very straightforward: the two networks of generator and discriminator play against each other. The goal of the generator is to generate an object (such as a person's photo) and make it look similar as the real data. The goal of the discriminator is to find the difference between the generated result and the real image. The training process will continue till the discriminator cannot distinguish if the image is fake or not.

Similar to the VAE model, the GAN model also contains a pair of submodels: generator G, which generates target data; discriminator D, which distinguishes if the data



is fake or real. Discriminator D is designed to help the generator to better learn the conditional distribution of the observed data. Its input is any image x in the data space. The output of D is a probability value indicating the probability that the image is from real data. For generating model G, its input is a random variable vector Z, and the output is an image G(z) generated from Z. D will evaluate the output G(z). If the probability value output from the model D is very high, it means that the generation model G has learned the distribution pattern of the real data, and can produce image samples with given vector Z.



**Figure 15: The structure of GAN (Goodeffellow, 2014)**

The target of GAN is to make discriminator D correctly classify the images as much as possible, meanwhile make generator G generate images that can cheat discriminator D. The goal of two networks could be summarized as follows:

- 1) For discriminator D, it should be maximize objective so that  $D(x)$  is as close to 1 as possible, that the real data are correctly classified; and  $D(G(z))$  is as close to 0 as possible, that the generated fake data are correctly rejected. Thus for D,the goal is to get:

$$\max [E_{x \sim p_{data}(x)} [\log D(x)] + E_{z \sim p_z(z)} [\log D(1 - D(G(z)))]]$$

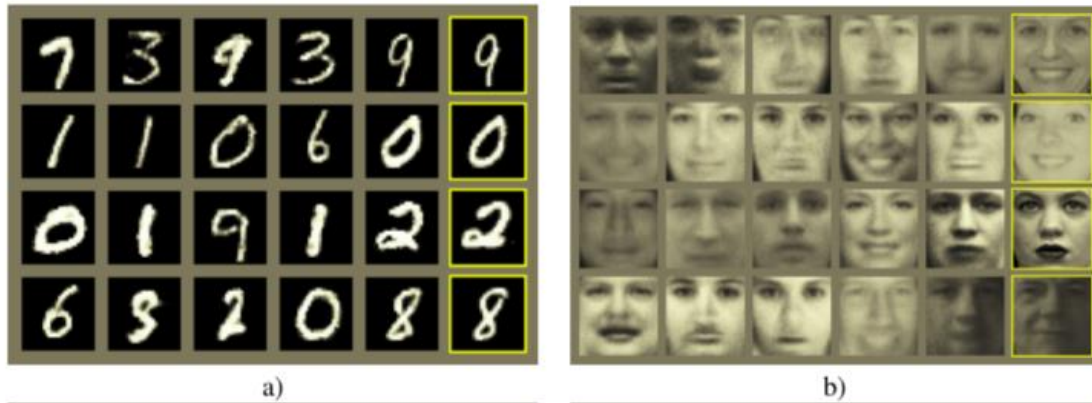
- 2) For generator G, it should make  $D(G(z))$  as close to 1 as possible, so that discriminator D cannot distinguish the fake data generated by G.

$$\min [E_{z \sim p_z(z)} [\log D(1 - D(G(x)))]]$$

- 3) Together, D and G play the following two-player minimax game with value function V (G, D):

$$\min_G \max_D V(D, G) = \min_G \max_D E_{x \sim p_{data}(x)} [\log D(x)] + E_{z \sim p_z(z)} [\log(1 - D(G(z)))]$$

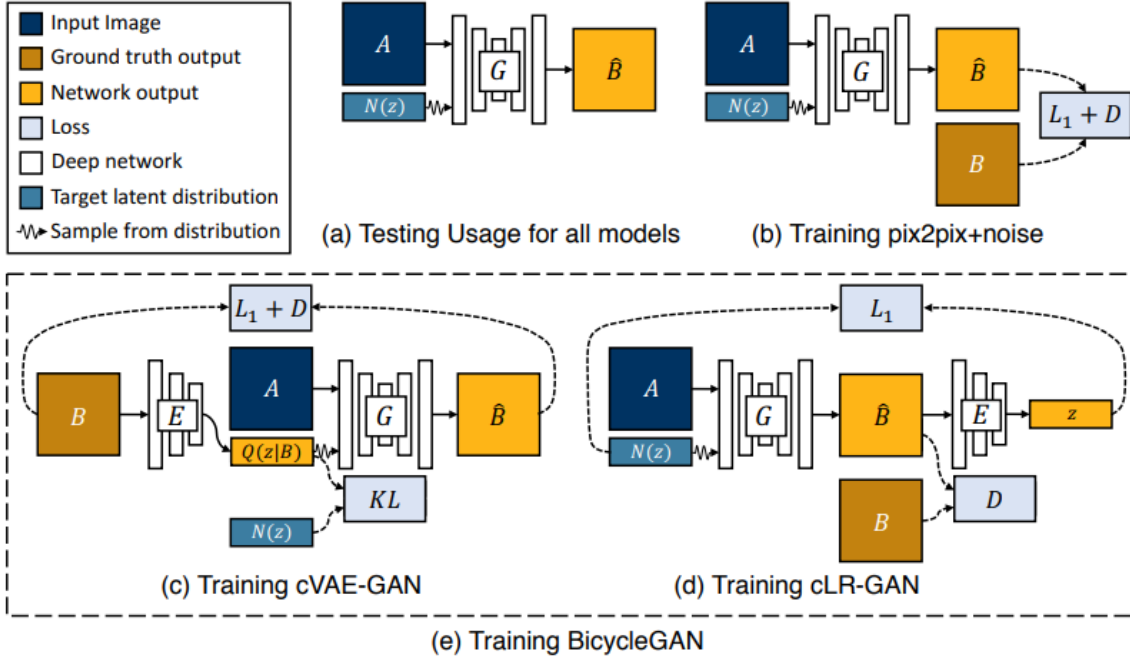
In the end, the discriminator  $D$  should not be able to distinguish if the data is real or generated by  $G$ . Define  $P_{data}(x)$  is the generated data distribution,  $P_g(x)$  is the real data distribution, then  $D(x) = P_{data}(x)/(P_{data}(x) + P_g(x)) = 1/2$ , which means that the data generated by  $G$  has the same distribution as the real data.



**Figure 16: The generation examples of GAN. (a) shows the examples of MNIST dataset, (b) are the examples of TFD dataset. The images in yellow frame are the original dataset images. (Goodefellow, 2014)**

### 4.3. BicycleGAN

In order to make the generative model controllable when creating multiple ambiguous generations, Zhu et al proposed the method BicycleGAN to model a possible generation distribution in a conditional setting (Zhu et al. 2017). BicycleGAN's goal is to learn a multi-model mapping from input image domain  $A$  to another image domain  $B$ . As is shown in Figure 18, such multi-model generation can change the style of the photo from spring to different seasons.



**Figure 17: The overview of BicycleGAN structure. (Zhu et al. 2017)**

Traditionally, to train such image translation model, the pairs of the domains are needed (e.g. spring photo and winter photo, red shoes and black shoes, etc.). While for BicycleGAN, such pairs are not strictly needed. It is able to generate diverse outputs, corresponding to different modes in the distribution  $p(B|A)$ .

As is shown in Figure 17, BicycleGAN consists of two models, cVAE-GAN and cLR-GAN, which are two types of generative models sharing the parameters during training.

VAE-GAN (Larsen, 2015) was proposed by in 2015. The idea is jointly training GAN and VAE to improve the performance of generative model. As an improved version of VAE-GAN, cVAE-GAN (Bao et al. 2017) was chosen to be implemented in BicycleGAN. For cVAE-GAN part, the encoder encoded the image samples into a latent space. Based on the latent code  $z$ , the generator will be able to recover the image data. As is presented in Figure 17, the encoder takes the original shoe image  $b$  as input to encode latent code  $z$ . The generator takes a sketch image  $a$  and  $z$  sampled from the distribution  $p(z)$  to output generated image by generator  $G(a, z)$ . Same as VAE, the latent distribution is assumed as a standard normal distribution, which is regularized using KL-divergence. However, the cVAE-GAN model implemented in BicycleGAN is a bit different from the original cVAE-GAN. In this case, the generator uses both encoded latent code  $z$  and image sample  $a$  ( $a \in A$ ) to reconstruct image  $b$  ( $b \in B$ ). The encoded latent distribution  $E(B) = q(z|B)$  should be close to a random Gaussian distribution. In such case the objective function for encoder is defined as:

$$L_{KL}(E) = E_{B \sim p(B)} [D_{KL}(E(B) \parallel \mathcal{N}(0, I))]$$

Besides, the cVAE-GAN also learns mapping between generated image and original image. Thus  $L_1$  loss is defined to ensure the generated image content can match the original input image, which is defined as:

$$L_1(G) = E_{A, B \sim p(A, B), z \sim q(z|B)} \| B - G(A, z) \|_1$$

For the GAN model (which contains generator G and discriminator D), similar as introduced in section 4.2, the objective function formulation is defined as:

$$L_{VAE-GAN}(G, D, E) = [E_{A, B \sim p(A, B)} [\log D(A, B)] + E_{A, B \sim p(A, B), z \sim q(z|B)} [\log D(1 - D(A, G(A, z)))]]$$

Finally, the objective function of cVAE-GAN could be formulated as:

$$\min_{G, E} \max_D V(D, G, E) = L_{VAE-GAN}(G, D, E) + \lambda L_1(G) + \lambda_{KL} L_1(E)$$

In which  $\lambda$  and  $\lambda_{KL}$  are the balance parameter to control the importance of related terms. Overall, the processing pipeline is shown in Figure 17.c.

For cLR-GAN (conditional latent regressor GAN) part, as is shown in Figure 17, the goal is to make sure the latent code  $\hat{z}$  encoded from the generated image  $\hat{b}$  is close to generator's input latent code  $z$ , which is sampled from the latent distribution  $p(z)$ . In such case, the process of encoding latent code  $\hat{z}$  from the generated image  $\hat{b}$  could be seen as a reconstruction process of  $z$ , with  $\hat{z} = E(G(A, z))$ . Here to measure the difference between  $z$  and  $\hat{z}$  is defined as  $L_1$  distance:

$$L_1^{latent} = E_{A \sim p(A), z \sim q(z)} \| z - G(E(A, z)) \|_1$$

Different from the GAN model of cVAE-GAN, the latent code  $z$  input into cLR-GAN is not based on encoder  $E$ . Thus the objective function of GAN in this case is:

$$L_{GAN}(G, D) = [E_{A, B \sim p(A, B)} [\log D(A, B)] + E_{A, B \sim p(A, B), z \sim p(z)} [\log D(1 - D(A, G(A, z)))]]$$

Combining it with GAN’s loss function, the final objective function of cLR-GAN is:

$$\min_{G,E} \max_D V(D, G, E) = L_{GAN}(G, D) + \lambda_{l1} L_1^{latent}$$

In which  $\lambda_{l1}$  is the balance parameter to control the importance of related terms. Overall, the processing pipeline is shown in Figure 17.c.

Finally, as is shown in Figure 17, the training process of cVAE-GAN is  $B \rightarrow z \rightarrow \hat{B}$ , which aims to enforce the latent code, meanwhile the reconstructed image  $\hat{B}$  should be as close to the original image  $B$  as possible. While the training process of cLR-GAN,  $z \rightarrow \hat{B} \rightarrow \hat{z}$ , is to ensure the reconstructed image  $\hat{B}$ ’s encoded latent code  $\hat{z}$  as similar as the original image  $B$ ’s latent code  $z$ . During the training stage, cLR-GAN and cVAE-GAN share the same parameters of discriminator  $D$ , generator  $G$  and encoder  $E$ . By combining the objective functions of cVAE-GAN and cLR-GAN, the final objective function for BicycleGAN is:

$$\min_{G,E} \max_D V(D, G, E) = L_{GAN}(G, D) + \lambda_{l1} L_1^{latent} + L_{VAE-GAN}(G, D, E) + \lambda_{L1}(G) + \lambda_{KL} L_1(E)$$

In Zhu’s work, the models were trained on images with  $256 \times 256$  resolutions. Dataset edges  $\rightarrow$  photos, Google maps  $\rightarrow$  satellite, labels  $\rightarrow$  images and outdoor night  $\rightarrow$  day images (Isola et al., 2017) are used for training. Figure 18 shows the final generated samples produced by generator  $G$ .



Figure 18: The example generations of BicycleGAN (Zhu et al. 2017)

#### **4.4. Summary**

For intelligent interaction system design, it is important for designers to understand the principle of the algorithm so that AI could be properly integrated into the interaction systems. In this chapter, to understand principle of generative models, the overviews of two typical algorithms (VAE and GAN) and their variants are introduced.

BicycleGAN is chosen as the implementation in this work. As is introduced above, it provides a generative model for multi-model image-to-image generation. Giving a fixed input image and multiple random sampled latent vector  $z$ , it will be able to generate multiple different image styles. The code of BicycleGAN was publicly released by the author (Zhu et al. 2017) for research. The details of the implementation will be introduced in chapter 6.

## 5. Design

### 5.1. Motivation and goals

As was introduced previously, the prototypes evaluated in the previous research are mainly simple sketch drawings. Compared to some design tasks in real working environment, these prototypes are relatively simple and don't have specific design purpose. Although DuetDraw (Oh et al. 2018) showed a general representation of user-AI creative work cooperation and provided a design guideline, it is still meaningful to extend this work to more complicated design tasks. In such case, the generality of previous conclusions is still unknown, as well as how much current AI techniques could contribute to the complicated design tasks.

Unlike the creative design in sketch drawing, UI design is more like software engineering. It could be an iterative design process that has multiple steps: the first step is structure design, which is to design the concept of the application structure by requirements extraction, analyzing task and understanding users and functionality. The next step is interaction design, which specifies the interaction and communication between human and computer. The final step is visual design, based on the structural design to design the appearance. It includes the designing of color, process of graphics and image, font design, page layout, etc (Garrett, 2010).

In this work, the UI design is chosen as a representation of complicated creative design task. Compared to the simple sketch drawing, the UI visual design should have several principles to follow. From the visual design aspects, the UI designers usually follow these principles (Garrett, 2010):

- Proper color patterns to enhance visual stimulations and give user right feelings;
- Proper color patterns to maintain the consistency and aesthetic.
- The functions should be properly designed to reduce user's burden of short-term memory;
- Visual elements should be easy to understand and identify.

Comparing UI visual design and sketch drawing, there are some differences:

- From the perspective of the purpose, the sketch drawing tasks, as studied in DuetDraw and other research work, have no limitations about creativity. However, there are more restrictions exist in UI design. UI design is part of the software engineering and the goal is to design the interfaces for operating devices. It should not only consider the visual looks, styles and usability, but also the functionality.
- UI visual design usually contains multiple steps, from sketch to wireframe and final visual design, each step could be considered as a sub design task. Compared to UI visual design, sketch drawing is quite simple.

Beside the complexity of the tasks, the AI models used in previous intelligent interfaces are mostly single-output models. Taking the user's interaction and drawing as input, the collaboration AI will only generate single output as feedback. However, there are many AI algorithms that are able to generate several possible outputs based on single input. It is meaningful to study if such multi-model generation AI algorithms will improve or deteriorate the user experience.

As is discussed in Chapter 3.4, user-AI creative cooperation can give user inspirations, but the unstable performance will bring problems as well. As a cooperation partner, the unstable performance of AI will lower the predictability and controllability of the interaction system. Besides, the users will also feel confused and have trouble in understanding the results generated by AI. As is described in the early work of Drawing Apprentice, sometimes the AI drawing agent seems to understand user's intention, but its output does not exactly match the intention. In the work of ColorAIze (Matulic, 2018), some participants noticed the inconsistencies of the AI outputs, when user modified the structure of the image, the AI's painting generation could make huge changes. In such case, user-AI creative cooperation faces some problems of predictability, comprehensibility, and controllability.

Based on previous conclusions, the following hypothesis are concluded:

- 1) Since the UI visual design has more constraints compared to sketch drawing tasks, user-AI creative cooperation may have different impact on the user experience.
- 2) By providing multiple outputs, the user-AI creative cooperation could be improved in the aspects of predictability, controllability, or other unexpected aspects.
- 3) Compared to single-output AI, the multiple-output AI may deteriorate the user experience in comprehensibility and learnability or brings unexpected negatives.

The UI design contains multiple stages, such as visual structure design, interaction design and visual design. In this work, the user-AI creative cooperation prototype developed mainly focuses on visual design. This tool is aiming to provide a better connection between sketch design (such as sketches and wireframes) and final visual design. The outputs of the UI prototyping tool are supposed to provide a better mockup for the final UI design.

This work could be seen as extension work of Matulic's research (Matulic 2018). In the end, we expect to improve the previous conclusions about user-AI interface guidelines and discover the new aspects of user-AI creative cooperation in UI design tasks. In this implementation, the AI will generate fuzzy colorized user interfaces from the wireframes



which are designed by users. The goal of the tool is to help the user generate useful wireframes and information for visual design.

## 5.2. Prototype design

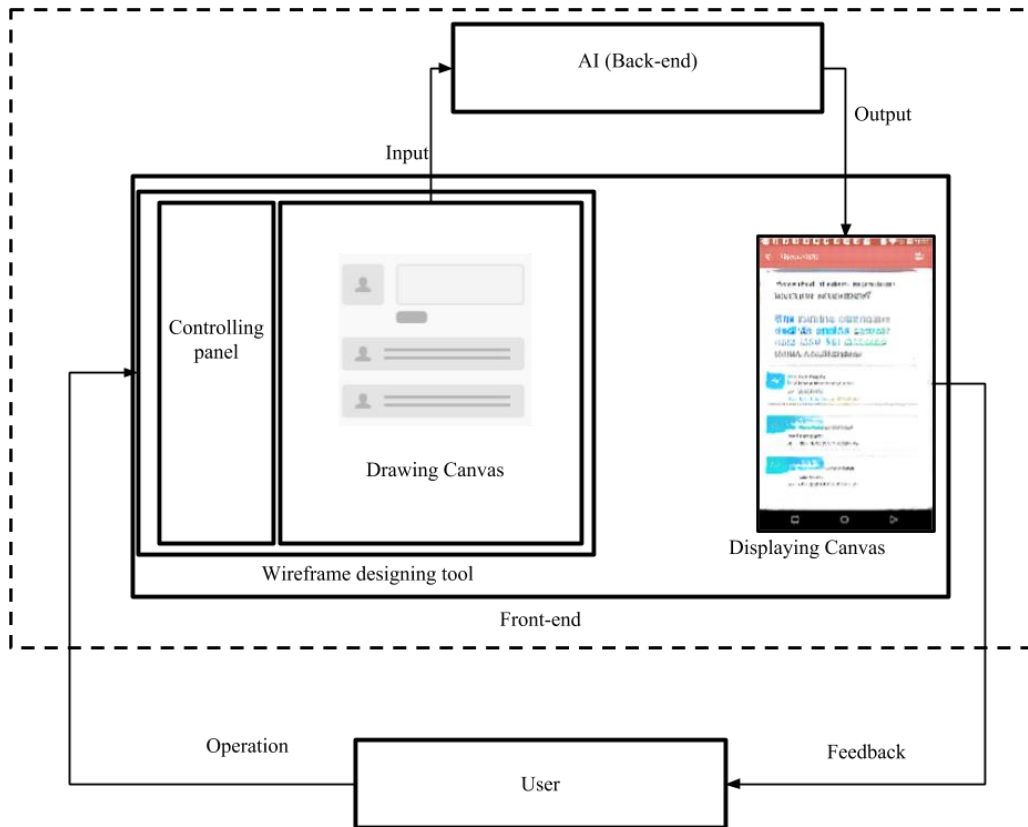
The idea in this work is to develop a prototype, with which user and AI can cooperate to design user interfaces. As is discussed previously, user-AI cooperation brings two significant advantages to the creative design works: 1) As researchers found in related works, although AI could be less predictable, it could generate something user did not image but inspiring to support user's work (Matulic, 2018). 2) The AI could provide alternatives outside of the human cognitive capacity as the interview presented in Feldman's work states: "*Human brain is sometimes limited, I find Evolver to have this unlimited capacity for creativity*" (Feldman, 2017).

As is discussed in Chapter 3.4, DuetDraw (Oh et al. 2018), ColorAIze (Matulic, 2018) and related research works have the same conclusion that AI can augment user's creativity. Especially Oh et al. provided a detailed user experience studies about user-AI co-creation (Oh et al. 2018), which mainly focused on the communication and initiative between user and AI. However, one limitation of the implemented AI algorithms in these works is that they only provide single output during the user-AI cooperation. This feature may provide limited inspirations. Besides, the user-AI creative cooperation design guidelines may not suit the complicated design tasks. Some design tasks have complex design process and constraints.

As the hypothesis proposed in previous section, the design of user-AI cooperation prototype will mainly focus on two aspects: 1) how user-AI cooperation influences the UI visual design user experience and 2) if multiple-out AI improve the user experience compared to single-output AI.

The prototype of user-AI cooperating UI design tool in this work could be considered as an extension work of Oh's research (Oh et al. 2018). Similar to previous user-AI co-creation implementations, this work includes design and implementation of a prototype to assist designer do UI design. To evaluate how user-AI cooperation impacts on user interface visual design, the first task is to develop the prototype.

The idea is to implement an AI agent that could provide multiple generations to users. BicycleGAN, as an improved GAN for multi-modal Image-to-Image Translation which was proposed in 2017 (Zhu et al. 2017), was chosen as the AI agent in this work. Two AI styles of user-AI collaboration tool were designed: (a) single hint, (b) multiple hints. This work evaluates and compares two AI styles' effects on user-AI user interface prototyping.



**Figure 19: Overview of the user-AI cooperation prototype**

Figure 19 shows an overview of the user-AI cooperation UI design tool. The prototype was designed as improved version of wireframe design tool, which mainly consists two modules: front-end part and backend part.

The front-end part is the interaction module: the wireframe design tool provides a canvas for user to design the UI wireframe, as well as the control panel that provides the necessary functions. Beside the wireframe tool, one extra canvas is provided to show the user-AI cooperation results, which is the AI modified UI visual design. With the visual design as the feedback, the user can adjust the wireframe structures to modify UI components. Meanwhile the display canvas will display the visual feedbacks in real time.

The backend part is the AI processing module. The wireframe created by the user will be sent to the AI module. Based on the input UI wireframe, the AI will modify the wireframe, generate corresponding visual components and send the result back to front-end.

When designing the user-AI collaboration prototype, the following design guidelines from previous research were taken into consideration:

- The visual elements generated by AI should be spatially close to the elements created by users. As is presented in the work of Pix2Pix and other generation models, the output generated by AI depends on the inputs of users, the structure of visual elements will not be changed after AI makes the

modifications. Thus the feature of spatial similarity is guaranteed. Besides, this consistency of spatial structure also maintains the visual similarity. Such features make it easier for users to make sense of the results of cooperation.

- Maintaining the perceptual logic is a tricky problem for GAN and other CNN based generative models. The reason is that neural networks are usually considered black box, how and why it generates certain visual elements are unknown for human. It is hard for the developers and users to understand why the decisions are made. However, in this work, the AI plays as an assistant to give the user the preview of the potential UI visual designs. Users can determine whether use the final results or not.
- Users should take the initiative. In this user-AI cooperation UI design prototype, user's role is playing the main ruler and the AI is to assist the user to accomplish the UI. The user makes all the decisions in this work.

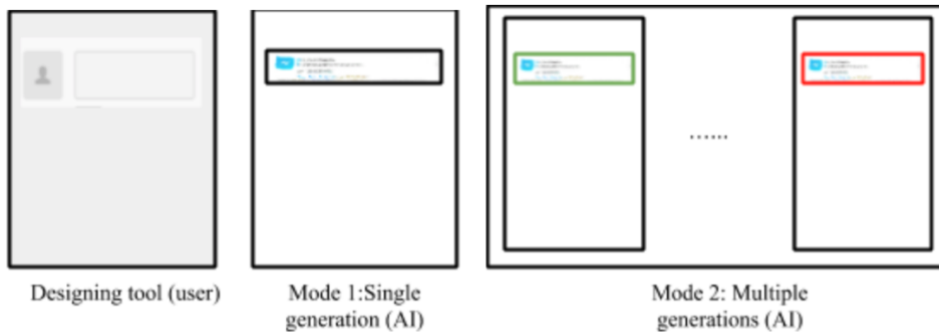
### 5.3. Styles

At the highest level, the system is a co-creative agent that gets UI wireframe designed by the user, generates color and textures with pre-trained generative models, and outputs the created new UI onto the display canvas (Figure 19).

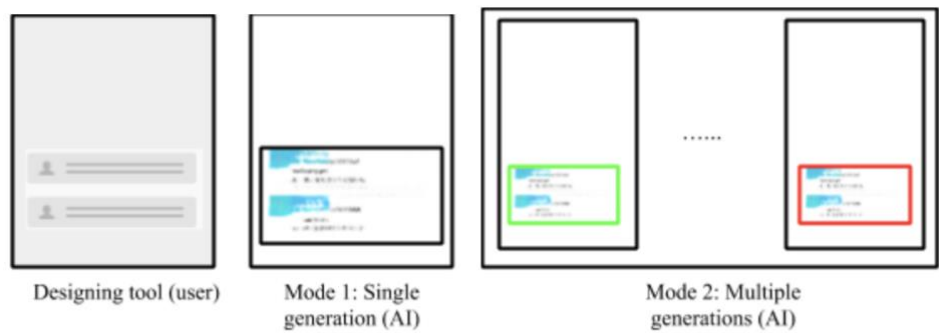
Following the hypothesis proposed in section 6.1, this work not only needs to evaluate how user-AI cooperation influences user experience of UI design, but also measure how multiple-output AI can influence the user-experience of user-AI collaboration. Thus two mode were designed for the prototype: single-hint mode and multiple-hint mode.

*Single-hint* mode is similar to the interaction defined in ColorAIze (Matulic, 2018). During the interaction, AI only generates one potential colorization schema based the wireframes created by the user. The user can create the UI wireframes in the front-end. The current view of the wireframe will be caught and sent to the backend to process. The AI module in the backend will automatically generate the fuzzy visual design for the wireframe. In most cases, the AI module is able to decide the specific color patterns, create visual elements and make modifications on the UI components. The processing of the UI generation is real time.

*Multiple-hints mode* has the almost same interaction as the single-hint mode. The only difference is that the AI agent will generate multiple potential UI visual designs in real time. The AI module is designed to generate 5 to 8 outputs from user's wireframe.



Step 1: (User) : Draw the wireframe  
(AI) : Generate the visual designing based on user's inputs



Step 2: (User) : Modify and refine the components  
(AI) : Refine the possible UI based on user's wireframe



Step final: (User) : Finish wireframe designing  
(AI) : Finish generating UI based on user's wireframe

**Figure 20. The basic user-AI cooperation process**

## 6. Implementation

Although related works provided the reference idea for implementing the user-AI cooperation UI design tool, there are still some differences and difficulties in developing such tool. The first is that to implement an AI model that is capable of drawing UI from sketches, a large training dataset should be provided. The second difficulty is that unlike sketch drawing task, the UI design tool has a lot of complicated functions. Implementing the UI design tool from scratch is unpractical. Besides, not only the *Paintschainer* (Yonetsuji, 2017) like AI model should be implemented, but also the multi-model image generation AI model should be developed as well.

Thus this chapter will introduce the work from two aspects, implementing AI algorithm and processing pipeline. First it describes how the dataset was chosen and modified for UI generation's needs. For AI model implementation, this chapter will introduce the implementation details of the generative algorithm. The last part of this chapter presents the final pipeline and the interface of the system.

### 6.1. Generative model structure

As is described above, two AI modes should be implemented to evaluate if multiple-output AI can improve the user experience of user-AI cooperation. As is described in Chapter 5.3, for multiple-hint style AI, BicycleGAN is capable of generating multiple possible images with one given input, it was chosen to be implemented as the AI agent in this work.

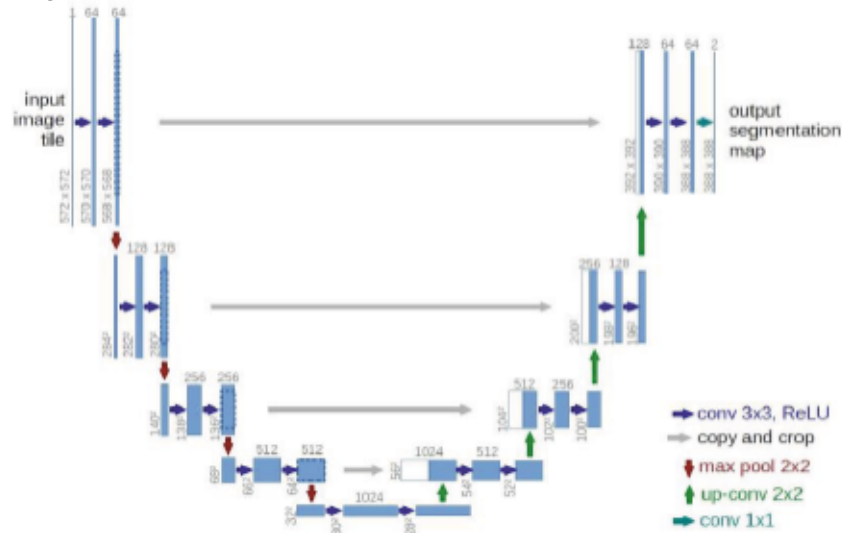
Since GAN focused on generating a single result, for single-hint mode AI, the GAN algorithm was chosen. For GAN, it consists of a generator network  $G$  and a discriminator network  $D$ . In BicycleGAN, besides a generator network  $G$  and a discriminator network  $D$ , it has an encoder network  $E$  to encode the input information.

For the implementation of GAN and BicycleGAN, once the AI model is well-trained, only the generator network  $G$  is used as the AI cooperation partner for UI design. The overview of BicycleGAN and GAN's structures are presented in Chapter 5.3. The construction of the networks will be introduced in this section.

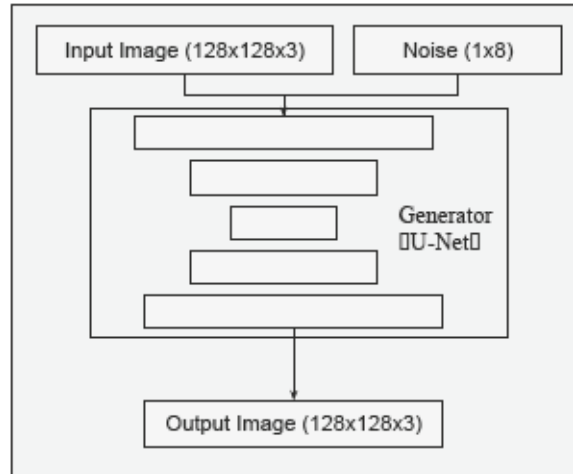
The structure used for generator  $G$  in this implementation is U-Net, which was proposed by Ronneberger, et al. in 2015 (Ronneberger,2015). U-net was originally proposed as an improved version of FCN (fully convolutional layer) for image segmentation tasks. It contains a downsampling path and an expansive path.

The downsampling path is composed of 4 blocks. Each block contains 2 convolutional layers and the output is downsampled by a  $2 \times 2$  max pooling layer. The number of feature maps doubles at each block. After the first block, it generates 64 feature maps; after the second block it generates 128 feature maps, and so on. The downsampling path extracts the contextual information and high-level features for specific task.

The expansive path symmetrically contains 4 blocks. Each block firstly deconvolutes the feature maps with stride 2, then concatenates feature maps with the corresponding cropped feature map from the contracting path. Two convolutional layers are used to extract features. The expansive path merges high-level features and low-level features and makes the network able to precisely perform the task on pixel-level. The outputs from the expansive path will be the AI results.



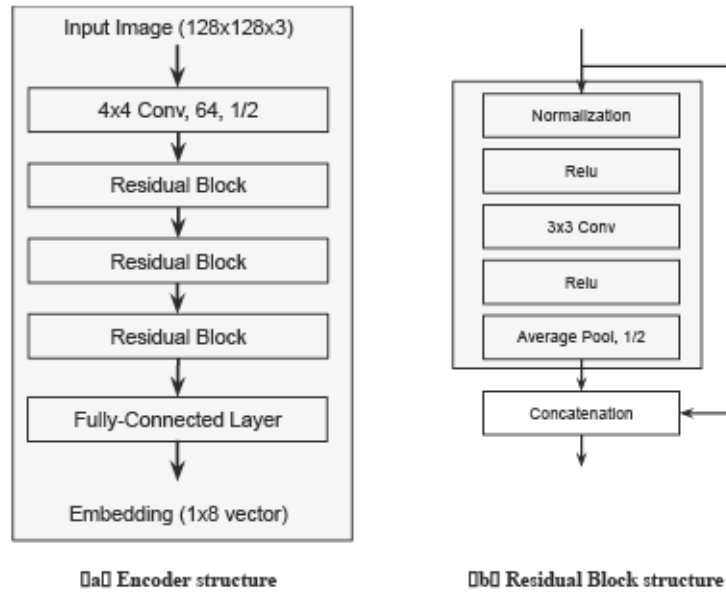
(a) Original U-Net structure



(b) Implemented generator

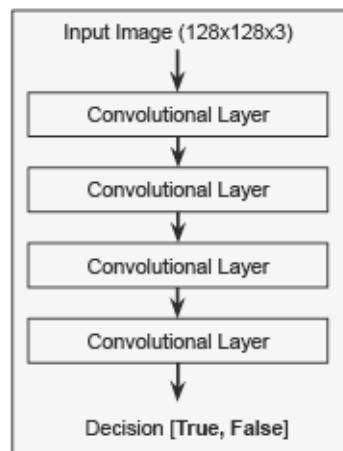
**Figure 21: The U-Net structure**

The structure used for encoder used Resnet-like structure (He et al. 2016) to extract feature. The encoder implemented in this work contains 4 residual blocks. The input of the encoder is an image, the size is  $128 \times 128 \times 3$ . The output is a 1-D embedding vector, and its size is  $1 \times 8$ . The structure of the encoder and its residual block structure are shown in Figure 21.



**Figure 22: The Encoder structure**

For discriminator, a 4-layer convolutional network was used to distinguish if the generated image is similar to real image or not. The structure is shown in Figure 22.



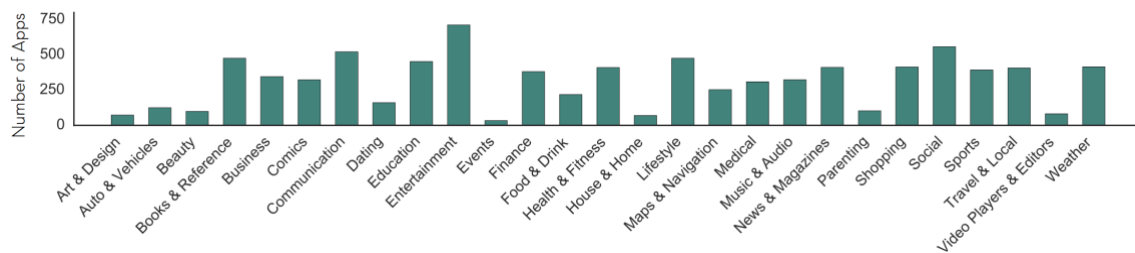
**Figure 23: The Discriminator structure**

The BicycleGAN network and CGAN network were implemented using PyTorch. PyTorch is an open source deep learning platform, which provides two high-level features: Tensor computation (like NumPy) with strong GPU acceleration, and Deep neural networks built on a tape-based autograd system. The PyTorch version used for BicycleGAN is version 1.0. After 32 epochs training on GTX1080ti GPU, the loss became converged and the generator outputs stable results.

## 6.2. Dataset preparation

Before choosing the AI algorithms, an appropriate training dataset for AI algorithms should be provided. In the related works, a large amount of data was needed to train the AI models. In the work of Paintschainer (Yonetsuji, 2017), the developer trained the AI model with the PixivDataset (Li, 2017), which contains 268116 image pairs (sketch and final colorized painting). In ColorAIze (Matulic, 2017), the developer directly used Paintschainer (Yonetsuji, 2017) as the AI collaboration agent. DuetDraw’s AI contains two generative models (Oh et al. 2018), Paintschainer and Sketch-RNN (Ha and Eck, 2018). Sketch-RNN model was trained using QuickDraw dataset which contains of hundreds of classes of common objects, and 70K training samples for each class.

Since the AI model needs large amounts of training data, the first challenge for implementation is to find or generate a proper dataset to train the models. In this implementation, RICO dataset (Deka et al. 2017) was chosen as the training dataset. RICO is a dataset collected for data-driven design. In total about 9,772 Android apps’ information were collected, spanning 27 Google Play categories.



**Figure 24: Categories of RICO collected UIs (Deka et al. 2017)**

The data analyzed and provided in RICO could mainly be categorized into 4 types of presentation: **visual**, **textual**, **structural** and **interactive**. **Visual** aspect explores the visual properties such as screen position, dimensionality and visibility; **Textual** aspect focuses on class name, id and displayed contents; **Structural** aspect mainly consists layouts and hierarchies; **Interactive** aspect analyzes how user interacts with the app. Based on these aspects, 6 processed data types are provided in RICO dataset: UI screenshots and view hierarchies, Hierarchies with semantic annotations, UI layout embedding vectors, Interaction traces, UI metadata, Play store metadata, and Animations.

In this work, only UI screenshots and view hierarchies and hierarchies with semantic annotations are used:

- **UI screenshots and view hierarchies**, which contains 72219 unique UI screens from the 9772 apps. The size of screenshots is 1440×2560. Besides the visual views, the detailed UI hierarchies are provided as well. The UI



hierarchies are JSON files containing simplified structure of UI's elements and layouts information.

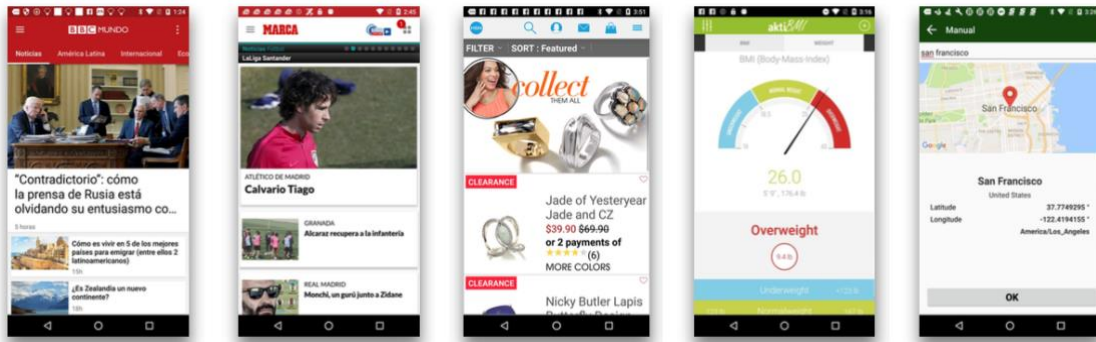


Figure 25: UI screenshots samples from RICO (Deka et al. 2017)

- **Hierarchies with semantic annotations.** To better present the UI's structure and categorize the UI elements, RICO also provides augmented visual presentations of UI elements. These semantic annotations visually describe what are the categories of UI elements displayed on the screen and how they are used. The UI components are split into 24 categories, 197 text button concepts and 97 icons. The semantic annotations encode each component, button, and icon class with a unique color. Beside the visual annotations, the corresponding JSON files are provided. Similar to the UI hierarchies, semantic hierarchy files represent the semantic portion of the original view hierarchies. The category JSON file specifies the mappings between semantic concepts and their colors are given in three separate files, corresponding to component categories.

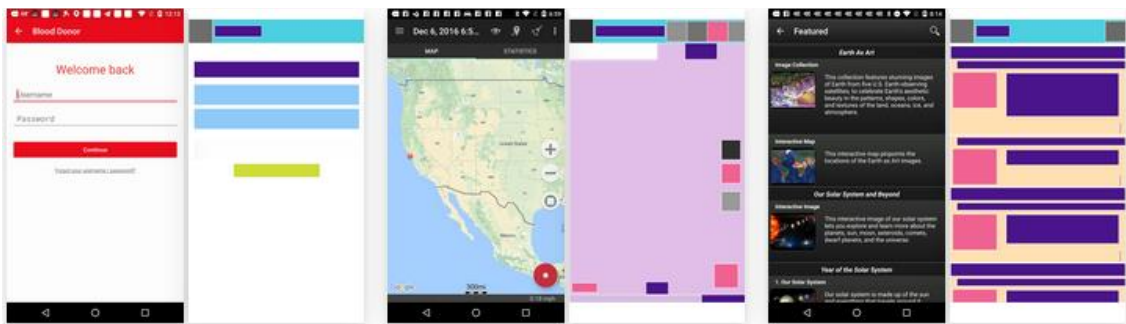
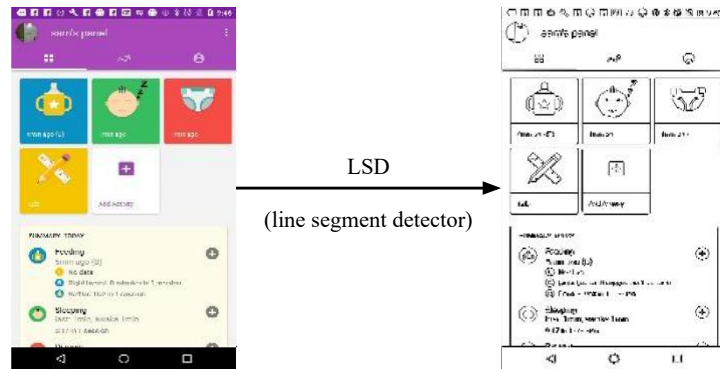


Figure 26: UI semantic annotation samples from RICO (Deka et al. 2017)

To make generative models (BicycleGAN and GAN) able to generate UI based on UI wireframes or UI sketches, the models need the image pairs as training data: the wireframe layouts as the input of generative model, and the real UI as the generation ground truths.

However, in RICO dataset, there are no wireframe or UI sketch data provided. Thus to create the training image pairs, wireframe-like UI images should be created. The LSD (line segment detector) algorithm (Von, 2008) was used to generate wireframe-like UI sketches. LSD is an algorithm aiming at detecting locally straight contours on images, which are the lines. Figure 27 shows the example about how a wireframe-like sketch is created from the UI.



**Figure 27. Using LSD algorithm to generate UI sketches**

With this method, 72219 UI pairs were generated as training samples. Figure 28 a and Figure 28 b show one pair of the training data. The LSD processed sketches are used as the inputs of the generative model. After 32 epochs of training, the generative model is able to create multiple UI visual designs based on the sketches. Figure 28 (c) shows the generated results based on the input sketch.



**Figure 28. The UI results generated by BicycleGAN**

### 6.3. Wireframe Design Tool

One difference between this and previous works is that UI design is more complicated and goal-oriented. The UI created by designers should not only be subjectively satisfying but also should meet the needs of the software developments, while the design works in previous research are relatively simple and subjective. This difference not only make the focus of research questions different but also make it difficult for implementation.

Although *PaintsChainer* and *Sketch-RNN* proposed quite intuitive interaction methods, it can hardly be directly used in UI design task. The reason is that human sketch drawing is different from the UI sketching. Since the AI model was trained with machine-generated samples, the wireframe extracted by image processing algorithms is different from human hand-sketches. Sketch lines drawn by human may be coarse and curved, while the sketch lines generated by LSD algorithm are sharp and straight. Therefore, AI algorithm can hardly generate appropriate results from hand drawn sketches. Besides UI design and sketch drawing have different interaction methods: sketches can be naturally drawn with pens, while UI wireframes usually are designed with computers. Thus a UI wireframe design software is needed as a base interaction tool.

In this work, rather than developing a new UI prototyping software, using an existing UI design tool would be a better choice. It not only makes the implementation more flexible, it also reduces the learning cost when conducting the experiments.

The open source UI prototyping tool *Pencil* (Evolus, 2008) was chosen as the front-end platform for users to create UI wireframes. *Pencil* is made public under the terms of the GNU Public License version 2, which is aimed for providing the community with most freedom for using and re-distributing the application. It runs in Ubuntu OS. Besides, the developer also provides source code under a commercial license in which licensees can obtain the source code, modify it and integrate it into their own commercial applications without the need to re-publish any of the code in the terms of the GPL. Such features of *Pencil* project made it an appropriate choice for UI prototyping platform in this work.

As an open source project, it offers similar functionality compared to other UI design tools. It provides various built-in shapes and UI components from desktop software to mobile applications, designers can freely use such components to accomplish UI design.

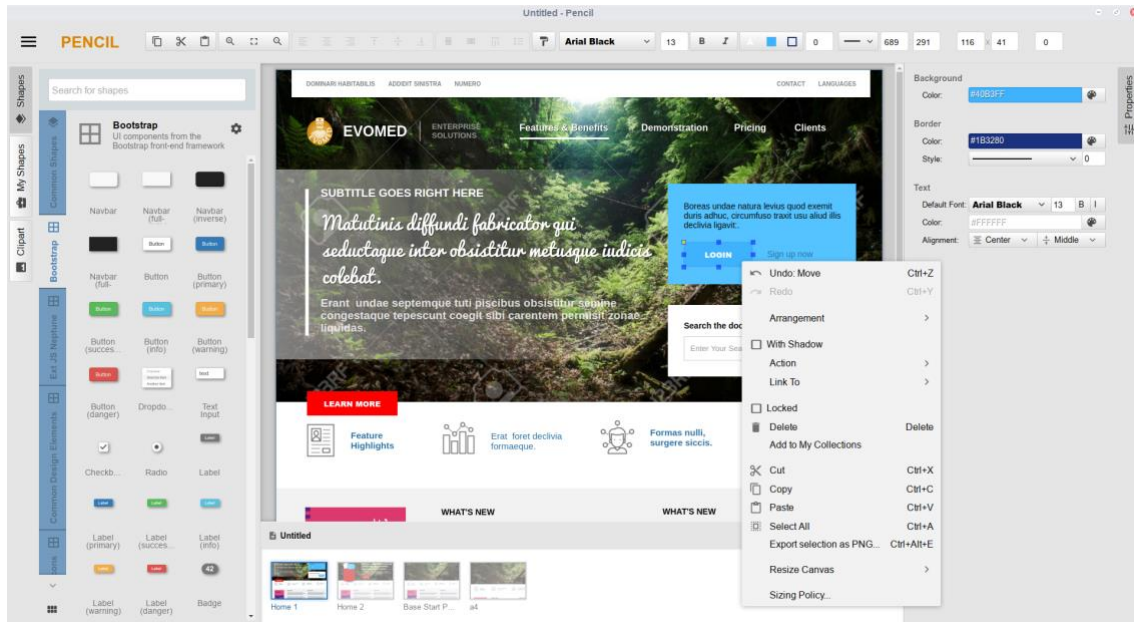
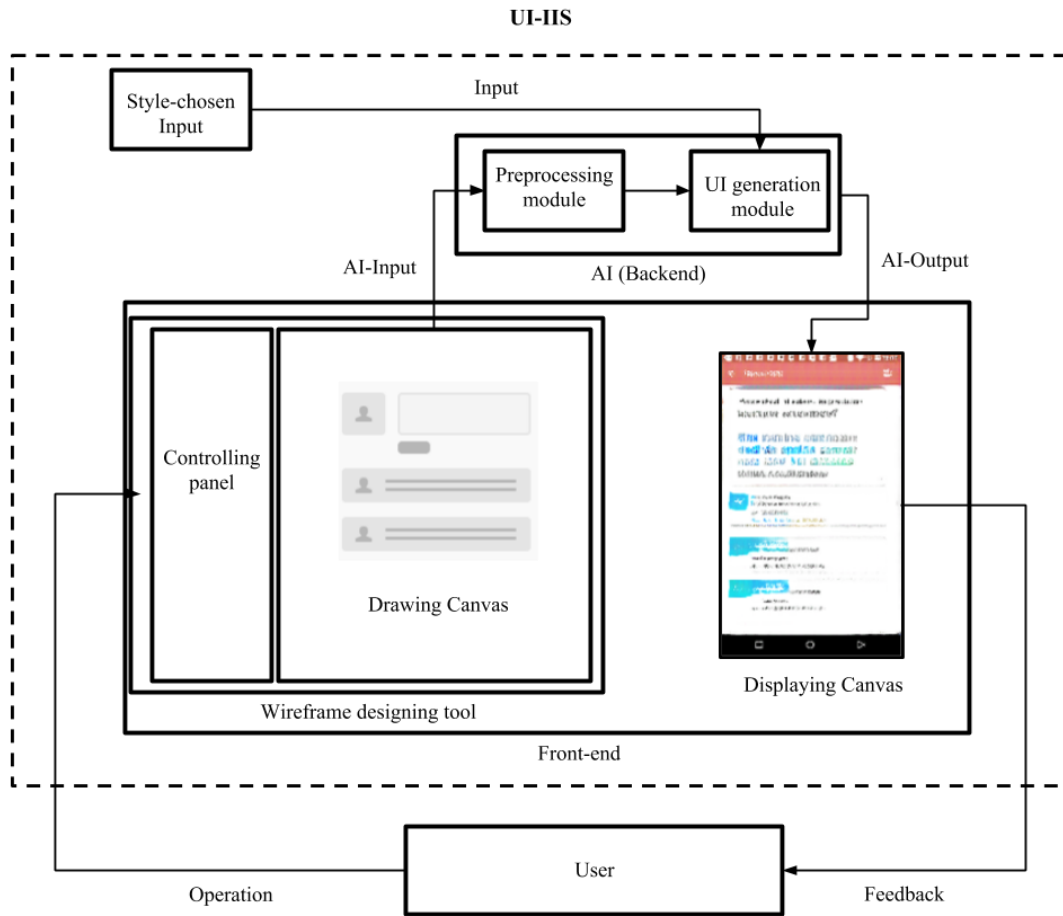


Figure 29: The UI wireframe design tool *Pen* (Evolus, 2008)

#### 6.4. Final design and implementation

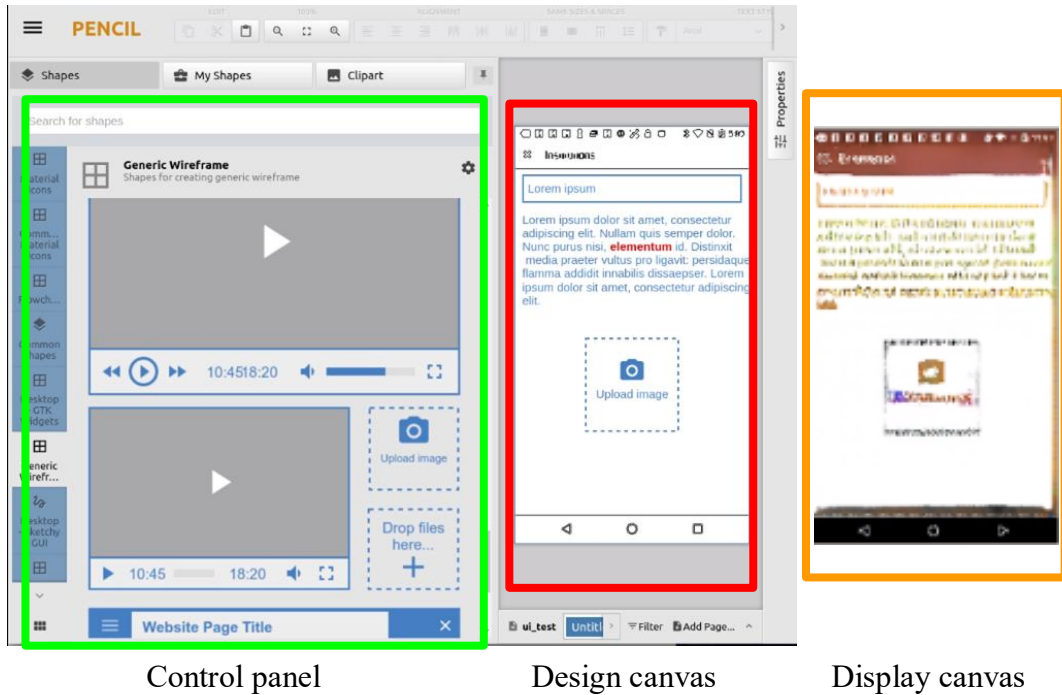
As is shown in Figure 30, the final implemented user-AI cooperation UI design tool consists of two modules: front-end module and back-end module. The front-end module includes the wireframe design tool and the AI generation display; The backend module in this implementation, could be considered as the intelligent agent in IIS. However, the back-end module consists of multiple processing modules: a pre-processing module and a UI-generating module. Pre-processing module converts the input wireframe into a line-sketch, so that inputs received by AI could be similar to its training samples, and AI will have better performance. After AI generates corresponding UI designs, it will output the results to front-end, and displays them on the side view.

The style-chosen input for the back-end module is invisible to users in this work. It is provided to the experimenter to control the style of AI generations: no-AI style, single-hint style and multi-hint style. By specifying the style, the software will provide different ways of user-AI cooperation interactions.



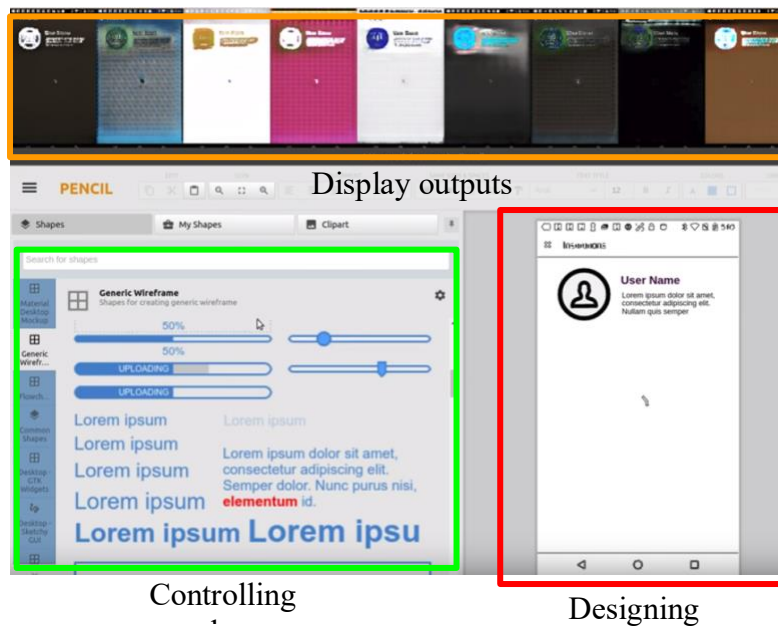
**Figure 30. The final overview of user-AI UI design tool**

Finally, the user-AI UI design tool was implemented on a local PC in Tampere university computer vision laboratory. Following figures present the two AI generation styles for final user-AI cooperative UI design tool: single-hint mode and multi-hint mode.



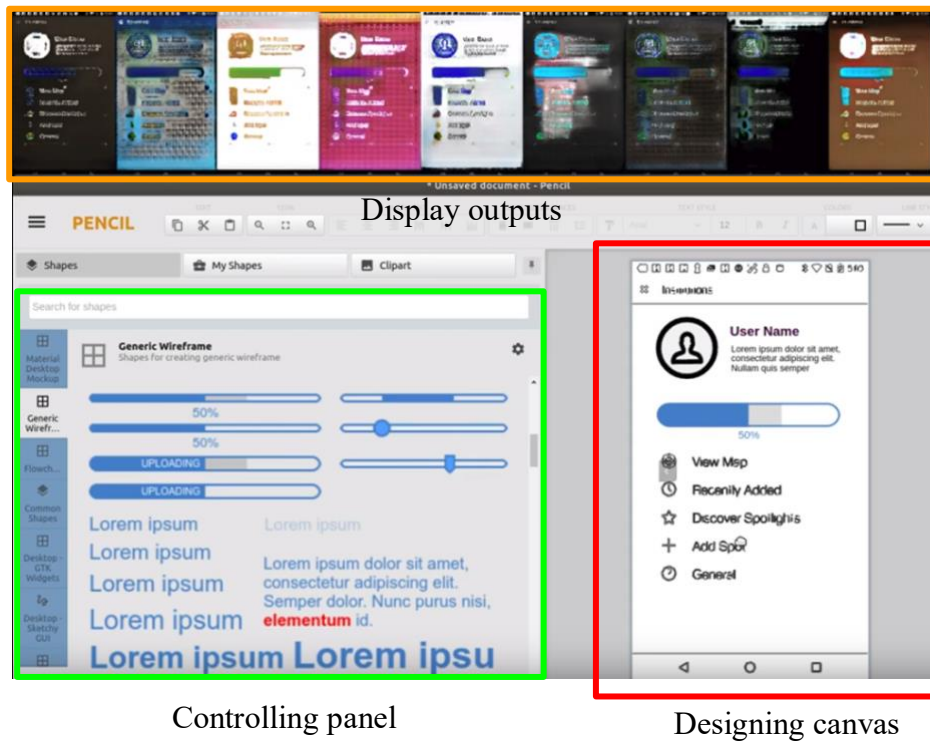
**Figure 31. The final implementation: single-hint mode**

As is shown in Figure 32, the left part is the wireframe design tool *Pencil*, users can design the UI wireframe in conventional way. The corresponding AI generated design will be displayed on the right-side view. In this single-hint mode, only one potential UI will be generated.



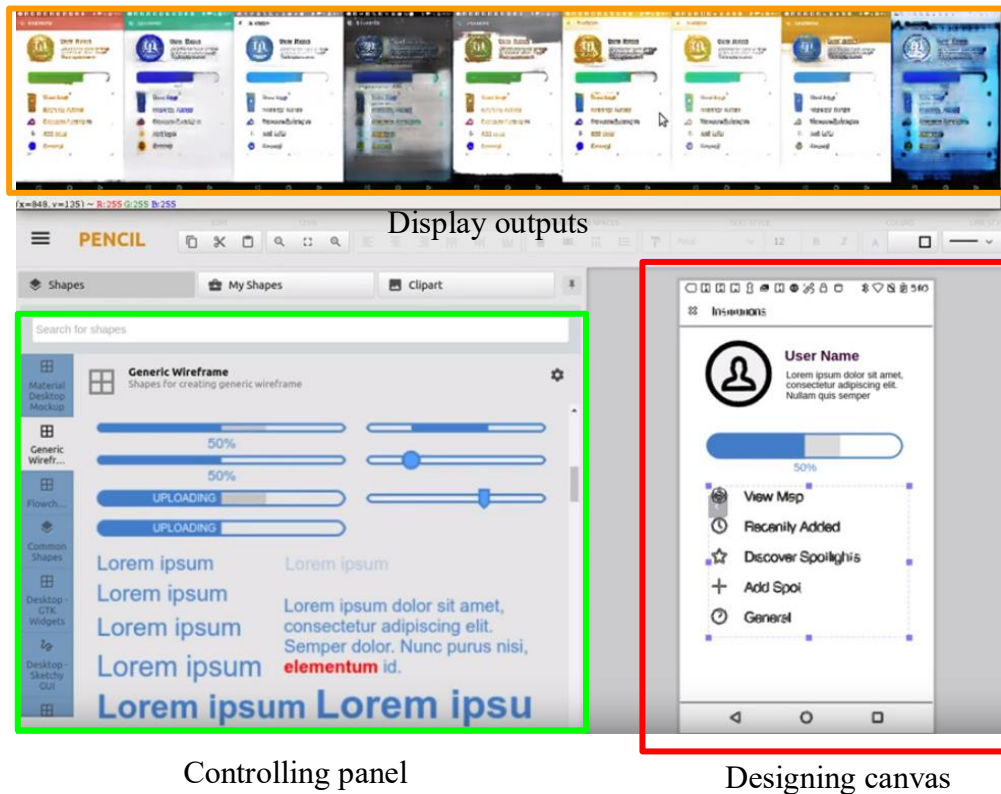
**Figure 32. The final implementation: multi-hint mode, multiple AI generations on the top**

Figure 33 shows the interaction with multi-hint AI. User designs the UI wireframe on the designing canvas. Rather than generating a single UI design, the AI partner will generate and display 9 possible designs and display them on the top of the wireframe design tool.



**Figure 33. The final implementation: multi-hint mode: AI generations changed responsively**

Every single modification made in the wireframe design tool will trigger AI's action. The 9 UI designs will be re-generated in real time. Figure 34 shows the example: once the user added some list items on the UI wireframe, the AI generation also added the UI components. Besides, each generation maintained its unique style.



**Figure 34. Multi-hint mode: Refreshing the AI generations**

Theoretically the AI model is able to generate infinite number of outputs. However, due to the AI algorithm limitations, it is not possible for human to understand how AI generated related styles, which makes it hard to control the output process. Thus the 9 UI styles are randomly generated and presented. To make AI generate as many as possible potential designs without ruining the user experience, a refresh function is provided. The user can press the “s” key on the keyboard to refresh AI’s generation if the AI outputs are not satisfying. Figure 34 shows the example, on the top of the design tool, the refreshed AI generations are presented.

To check if the user-AI cooperation tool is properly designed, it is necessary to review the design challenges of intelligent interaction system as introduced in Chapter 3.3. The features of intelligent interaction system brought up challenges for users and designers in 9 ways (*Inappropriate integration caused switches between applications or devices; user need to teach the AI; Narrowing users’ abilities; Unsatisfactory Aesthetics and latency; Need for Learning by the User; Inadequate Control over Interaction Style; Threats to Privacy; Inadequate Predictability and Comprehensibility; Imperfect System Performance*). With these challenges as references, the user-AI cooperation UI design tool includes the following considerations:



- To avoid switching between interfaces, the AI was designed as a backend module to generate the UI designs. Thus there are no extra interactions required to switch between AI function and wireframe design tool.
- The wireframe designed by user will be observed by AI in real time. Once it gets the user input, the AI model, BicycleGAN and GAN, it can generate 1 UI design in around 0.1 seconds and 8 UI designs in around 1 second. The results will be shown to the user once they have been generated. The latency of user-AI cooperation has been reduced as much as possible.
- As was introduced previously, user's main operation is almost the same as in conventional wireframe design, thus the tool theoretically would not limit user's ability of wireframe design.
- The AI model was trained before integrating into the system, thus this user-AI cooperation prototype does not require user to teach the AI. The principle of the algorithm and the task of the user-AI cooperation do not pose the privacy threats to users.
- The previous work introduced in Chapter 2.5 has proved that such user-AI creative cooperation will face the problems of inadequate predictability and comprehensibility, depending on how AI algorithm is implemented. The AI algorithms' performance also decides the tool's performance. In this work, how AI affects predictability and comprehensibility will be measured in the experiments.

With intelligent interactive systems, the challenge is to provide users a useful and helpful tool to accomplish the task. In this work, we have the hypothesis that user-AI creative cooperation could be helpful for designers to do UI design tasks.

Overall, the UI-IIS design and implementation mainly considered from following perspectives: the design guidelines provided by previous research works; designing challenges of IIS; implementation difficulties of the AI algorithms.

## 7. Experiments

As is introduced in Chapter 6, 3 hypotheses are proposed in this work: 1) user-AI creative cooperation will have different impacts on the complicated design tasks compared to sketch drawing; 2) multiple-output AI can improve the predictability and controllability of intelligent interaction system; 3) multiple-output AI may deteriorate the user experience in comprehensibility and learnability.

To assess the user experience of user-AI cooperative UI design, a user study was conducted to verify the hypotheses and to find potential discoveries. The conducted user study consists of a series of UI design tasks, post-hoc surveys and semi-structured interviews.

### 7.1. Procedure

A group of ten participants with background of UI design was invited to the lab. On average a 45-minute study was conducted in three phases. The procedures of the experiments were designed as follows:

**Introduction phase.** In the first phase, to help participants get familiar with the concept of user-AI cooperation, a uniform introduction of intelligent agent and related demos was given. Participants were asked to play with user-AI co-creative demos as well. The demos are *Sketch-RNN* (Ha and Eck, 2017), *pix2pix* (Isola et al. 2016) and *PaintsChainer* (Yonetsuji, 2017). After the participants finished playing with demos, general questions were asked to understand participants' preference and opinions towards user-AI cooperation.

**UI Design.** In this phase, the participants were given a demonstration of the user-AI collaborate UI design tool. Before the experiments, participants were oriented with the basic operations and features of user-AI UI design tool, such as how to use single-hint style AI and multiple-hint style AI. Then several minutes were given to users to explore the basic functions. A 5-minutes task was assigned for the participant to freely create a simple UI wireframe without the AI generation. Next, the experiment was conducted, it contained three design conditions: collaborating with the single-hint style AI, collaborating with multiple-hint style AI and making UI wireframe without AI. The experiment was designed as within-subjects so that all participants were asked to do 3 designs. The interfaces were the same in all conditions and the experimental conditions were randomly ordered to minimize learning effects, the details will be introduced in section 7.3.

For each participant, during the test, not only the order the three AI conditions (single-hint style, multiple-hint style and no-AI style) was shuffled, the order of the tasks was

randomized as well. At the beginning of the test, the participant was assigned with a random AI condition and a random UI design task. For example, the participant was first assigned to perform task 2 in the single-hint style AI condition; for the second round, task 1 and no-AI condition was assigned to the participant; for the last round, the participant would conduct task 3 in multiple-hint AI condition.

After each design task, the participant was asked to fill-in questionnaires. The collected data were used to study how intelligent agent affects the user experience on task-driven designing works, and if multiple AI potential hints could contribute to the user experience. The questions were focused on these aspects of user experience: 9 general criteria for user experience evaluation (useful, easy to adapt, effective, comfortable, consistent, fulfilling, fun, controllability and communicative) and 3 criteria for AI interface evaluation (predictability, comprehensibility and inspiration). These criteria were chosen based on the work of Duet-Draw (Oh et al., 2018). The participants evaluated each task on the questionnaires with a 7-point Likert scale ranging from highly disagree to highly agree.

**Feedback.** The third phase was designed for qualitative study, it was conducted as semi-structured interview after the participant finished all three design tasks. Since the Likerts questions are not able to extract deeper and detailed information, the experimenter prompted the participants to describe their thoughts of the design, how they consider the cooperation and how cooperation affected their strategies. The thoughts about different conditions were separately described by participants. All interviews were audio recorded.

To help participants better recall their ideas and impressions during the test, photo projective technique (John, 1957) was used: the final designed works were presented to participants. Based on each wireframe (and its corresponding AI output), participants described their thoughts of designing and how they would apply such generations into the final visual design. This stage is helpful for discovering the detailed reasons.

**Results analysis.** Two types of data were collected: questionnaire results provided quantitative data to see if there was any difference between three conditions; interviews provide qualitative data to understand why such feedback was given.

## 7.2. Participants

12 participants were recruited in total, including 2 pilot test participants. There were 5 males and 7 females. All of the participants in this user test claimed they had experience of software developing and UI prototyping.

Table 1 summarizes the general information about the participants. After the AI co-creation demo was introduced, 9 of the participants showed positive attitude toward the AI co-creation; 2 participants gave neutral feedbacks and 1 participant gave negative feedback.

Participant Number	Age	Gender	UI Design Experience	Opinion about AI
1	25-30	Female	Yes	Positive
2	25-30	Female	Yes	Neutral
3	30-35	Male	Yes	Neutral
4	30-35	Male	Yes	Positive
5	20-25	Female	Yes	Positive
6	20-25	Female	Yes	Positive
7	25-30	Female	Yes	Positive
8	25-30	Male	Yes	Negative
9	25-30	Male	Yes	Positive
10	25-30	Female	Yes	Positive
1 (pilot test)	25-30	Female	Yes	Positive
2 (pilot test)	-	Male	Yes	Positive

**Table 1. The participants' information**

### 7.3. Tasks

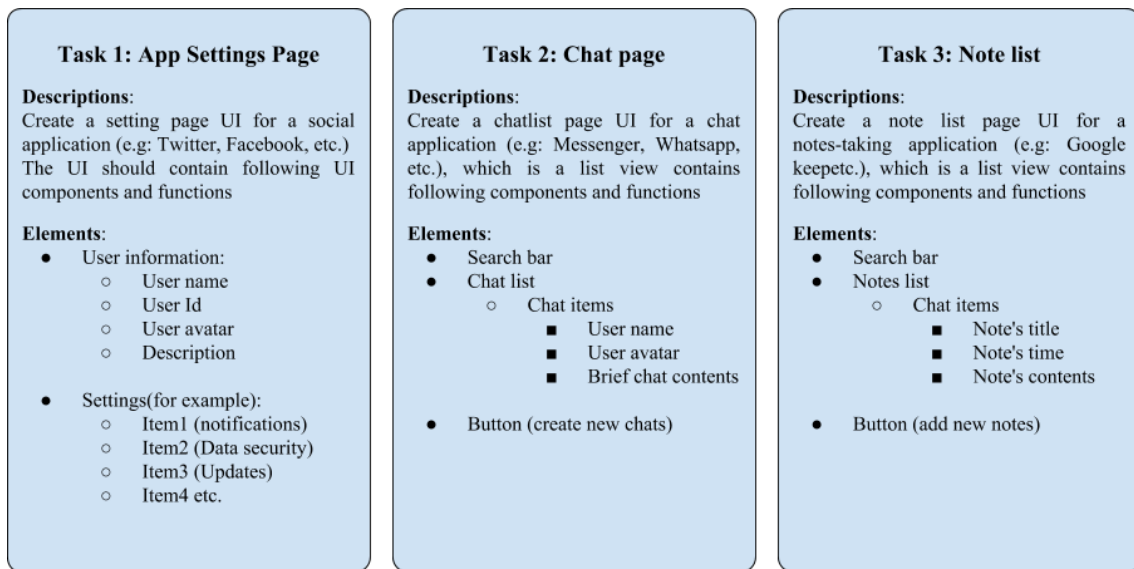
As is mentioned in Oh's work (Oh et al. 2018), to properly evaluate user experience, it is better to give specific tasks. Even though the participants could freely design any UI within user-AI UI design tool, the tasks were semi-assigned to the participants rather than letting participants create too many different UIs. Semi-assigning tasks to participants could also minimize unknown factors that affect the results.

A task pool was created for assigning tasks. It contains three simple UI design tasks: app settings page UI design, chat app UI design and note taking app UI design.

Two considerations were taken into account when creating the task pool. The first consideration is that AI's performance is unstable for different UI types. Although the AI generative models are able to generate several designs based on input UI sketches, the

quality still varies. For example, it may generate perfect styles and textures for list view or text view, but it performs badly if the UI wireframe contains complicated images or icons. Another reason is that the given tasks should be simple. Based on the pilot feedbacks, reducing the complexity of tasks is necessary, otherwise the participants would get bored and impatient. Such phenomenon will affect the survey results.

Figure 35 shows the tasks' details. On each task card, the usage context, the components and functions of the UI were specified. First, the user randomly picked one of the design tasks from the task pool, and randomly chose the AI mode. The Participants were asked to follow the task card's description to perform UI wireframe design task. The orders of tasks and AI hint conditions are shown in Table 2.



**Figure 35: Task pool**

Participant Number	AI Style Order	Task order
1	C,B,A	1,2,3
2	C,A,B	2,1,3
3	A,C,B	3,2,1
4	B,A,C	2,3,1
5	A,C,B	3,1,2

6	C,B,A	1,2,3
7	C,A,B	2,3,1
8	A,B,C	3,2,1
9	A,B,C	1,2,3
10	C,B,A	2,1,3

<b>Style(S)</b>	<b>Taks(T)</b>
A: Single-hint	1: App Setting Page
B: Multiple-hint	2: Chat List Page
C: No-hint	3: Note List Page

**Table 2. The task order and AI style order of each participant**

#### 7.4. Data collection

To quantitatively evaluate the user experience of the user-AI UI design tool. After the participants finished each task, they were asked to fill out the questionnaires based on related AI styles.

For multiple-hint AI and single-hint AI, the questionnaires contained two parts. As was introduced in section 7.1, the first part is the questions about 9 general items commonly used for user interface usability and user experience evaluations: *useful, easy to adapt, effective, comfortable, consistent, fulfilling, fun, controllability* and *communicative*. The second part is about 4 criteria for AI interface evaluation: *predictability, comprehensibility, understandable* and *inspiring*. For No-AI condition, only the general user experience evaluation questionnaires were provided for participants to fill out. The details of the questionnaires will be provided in the appendix.

#### 7.5. Semi-structured interview

As was introduced previously, the semi-structured interviews were conducted after all three different AI style tasks were accomplished. 10 interviews were recorded as audio recordings and were transcript, 3 of the recordings were in English and rest were translated into English. Besides the interview results, some feedback (such as think-aloud feedback) during the task was recorded as well. The questions are mainly focused on how the participants made sense of the AI generations and how AI affected their interactions and designing ideas.

Generally, the questions were more about the participants' overall feelings. The questions were:

- Did you take the single-hint style AI generation into consideration during the test? If yes, can you describe how?
- Did you take the multiple-hint style AI generation into consideration during the test? If yes, can you describe how?
- What is the most difficult part of user-AI co-creation in this experiment?

Additional questions were asked based on the participants' answers to the questionnaires to discover the reason why they had positive or negative user experience of the user-AI co-creation.

## **8. Results**

### **8.1. Quantitative Analysis**

Through the user study, questionnaire responses from the survey and transcriptions from the interview sessions were collected. From the design tasks, 30 wireframes and 60 blurry AI generations were collected in total.

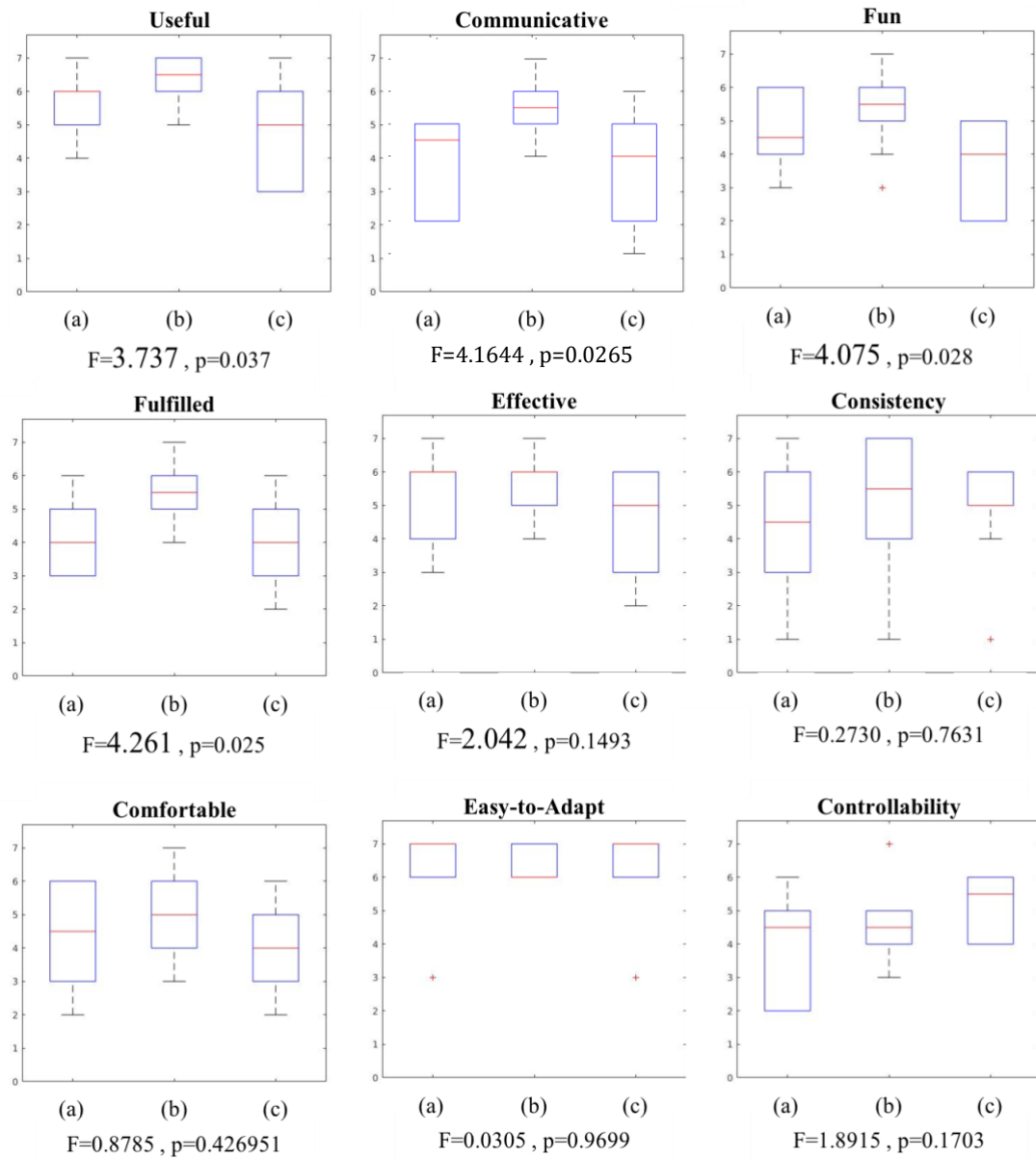
As was proposed in the previous section, we also want to verify the following hypothesis: 1) user-AI creative cooperation will have different impact on the more complicated design tasks; 2) multiple output AI can improve the predictability and controllability of intelligent interaction system; 3) multiple-output AI may deteriorate the user experience in comprehensibility and learnability. The results follow these hypotheses.

In quantitative analysis, as is described in the chapter 6, we hypothesized that AI could improve user experience of UI design and multiple-hint style AI could minimize the negative effects brought by single-hint AI. Thus one goal in this thesis work is to check if the hypothesis was correct; if not, what differences are revealed between different AI styles. The data was analyzed using a one-way repeated-measure ANOVA, comparing the effect of each condition on the user experience of the interface. Tukey's HSD test was chosen as a post-hoc test for pairwise comparisons.

In qualitative analysis, the data were mainly collected from the semi-structured interviews. From qualitative analysis this thesis work aimed to find the missing information from the quantitative analysis and the reason for the results revealed in the quantitative analysis.



### 8.1.1. Overall comparison



**Figure 36. Box plots of user ratings of each item according to each condition and result of one-way repeated-measures ANOVA. ((a) Single-hint, (b) Multiple-hint, (c) No-hint. Statistically significant**

results are reported as  $p < 0.05^*$ ). Results are collected from Matlab ANOVA function.

comparison 1		
(a) Single-hint	(b) No-hint	
<i>item</i>	<i>difference</i>	<i>p-value</i>
useful	0.7000	0.3719
easy-to-adapt	0.0000	1.0000
effective	0.6000	0.5768
comfortable	0.3000	0.8757
communicative	0.1000	0.9875
consistent	-0.3000	0.9277
fulfilling	0.0000	1.0000
fun	0.8000	0.3414

comparison 2		
(a) Multiple-hint	(b) No-hint	
<i>item</i>	<i>difference</i>	<i>p-value</i>
useful	1.4000	0.0285
easy-to-adapt	0.1111	0.9751
effective	1.2000	0.1265
comfortable	0.8000	0.4010
communicative	1.7000	0.0409
consistent	0.3000	0.9277
fulfilling	1.3000	0.0450
fun	1.6000	0.0216

comparison 3		
(a) Single-hint	(b) Multiple-hint	
<i>item</i>	<i>difference</i>	<i>p-value</i>
useful	-0.7000	0.3719
easy-to-adapt	-0.1111	0.9751
effective	-0.6000	0.5768
comfortable	-0.5000	0.6941
communicative	-1.6000	0.0567
consistent	-0.6000	0.7427
fulfilling	-1.3000	0.0450
fun	-0.8000	0.3414

**Table 3. Results of Tukey’s HSD test. Comparison 1 is between single-hint AI and No-AI; Comparison 2 is between multi-hint AI and No-AI; Comparison 2 is between multi-hint AI and single-hint AI. Results are collected from Matlab multiple comparison functions;**

For the aspects of controllability, comfortable, effective, and consistency, AI does not contribute much to the user experience: one of the hypotheses is that AI could contribute to design task performance. As is shown in Oh's qualitative research (Oh et al. 2018), for the basic usability of the interface, AI generally shows advances in the aspects of *Fun*, *useful*, *effective* and *efficient*. However, one of the survey analysis results indicated that simply integrating the AI (single-hint style AI) into UI design did not improve user experience significantly. The pairwise comparison analysis result is shown in table 3 (Comparison 1, the *difference* and *p-value* are shown in the table). The p-values of controllability (p-value = 0.1703), comfortable (p-value = 0.4269), effective (p-value = 0.1493), and consistency (p-value = 0.7631) are larger than 0.05, which indicates that for UI design task, AI did not bring much improvement to the user experience.

**Participants can easily get used to the AI features:** Another result revealed in the figures is that it was easy for the participants to get used to the AI. The comparison box plot in Figure 36 shows that the *easy-to-adapt* score in three conditions are pretty similar. The p-value is 0.9699, which is larger than 0.05. Besides, from the pairwise comparison (Table 3, comparison 1 and comparison 2), the *p-value* and *difference* of item *easy-to-adapt* also shows integrating AI features into design task does not increase learning costs. Compared to the traditional design tool, participants did not feel any difficulty for adapting AI features

### 8.1.2. Comparison of single-hint AI and multi-hint AI

**Users prefer multi-hint AI.** As is shown in table 4, multi-hint AI produced higher scores than single-hint AI. Results of comparison between single-hint AI and multi-hint AI show that the participants preferred the multi-hint AI. However, the p-values of each criteria indicated there is no significant difference among the three conditions. **Multiple-hint AI brought significant improvements on user experience:** In general, from the multiple comparison box plots, it is observed that the participants preferred the design tool with AI feature, although the difference is not significant. Among all the evaluation items, 4 out of 9 items' p-values indicate there is significant difference: *useful* (p-value = 0.037 < 0.05), *communicative* (p-value = 0.027 < 0.05), *fun* (p-value = 0.028 < 0.05) and *fulfilling* (p-value = 0.025 < 0.05). Especially in multiple-hint style AI condition, the box plots indicate the score of 8 evaluation criteria is higher than in the No-AI condition (*useful*, *communicative*, *fun*, *fulfilling*, *effective*, *consistency*, *comfortable* and *easy-to-adapt*).

Besides, from the comparison table, the *p-values* of *useful*, *communicative*, *fulfilling* and *fun* indicate multiple-hint style AI improved the task performance. It not only enhances usability but also brings joy. However, for *effective* criteria, multiple-hint style AI did not bring much improvement.

**Multi-hint AI is more fulfilling.** The p-value of *fulfilling* criteria between multi-hint AI and single-hint AI is 0.045, which is smaller than the threshold 0.05. This indicates participants felt more fulfilled when using multi-hint AI for UI design.

Besides the general UX evaluation criteria comparison, user-AI cooperation UX evaluation criteria were compared as well. The result is shown in Figure 37. Four user-AI cooperation criteria are evaluated: predictability, controllability, inspiring and understandability.

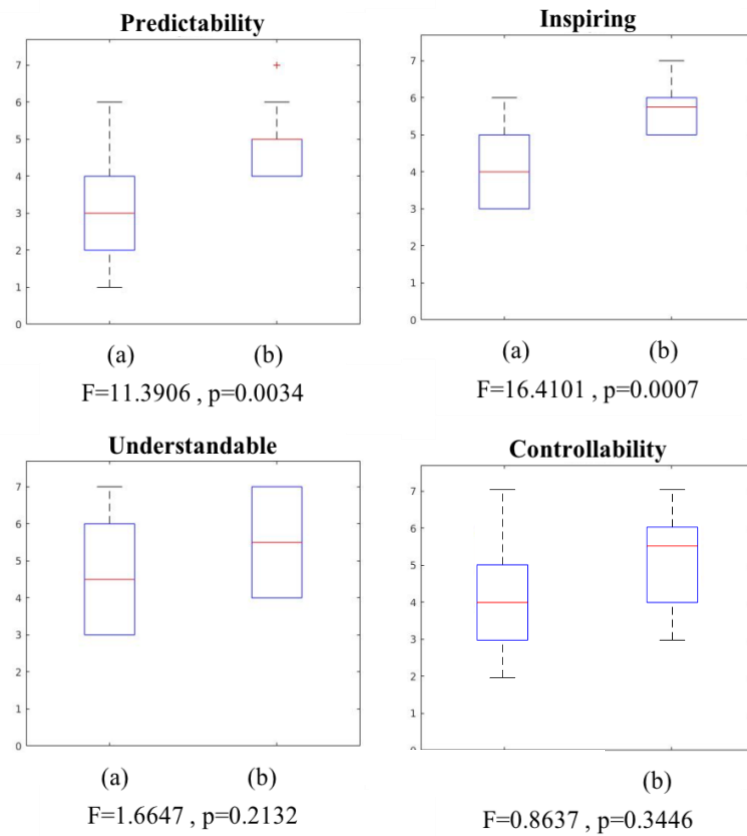
**Multiple-hint is more predictable.** The p-value 0.0034 suggests that there is significant difference in predictability between single-hint AI and multi-hint AI.

**Multiple-hint is more inspiring.** The p-value of inspiration score comparison is 0.0007, which indicates that there is significant difference between single-hint AI and multi-hint AI. The box plot comparison shows that multi-hint AI is more inspiring for UI design

**Multiple-hint does not help too much in Comprehensive and Controllability.** The p-values for comprehensive evaluation is 0.1170 and p-value for controllability is 0.2132. Both of them indicated that for these two evaluation criteria, single-hint AI and multi-hint AI have no significant difference.

Overall, for user-AI cooperation UX evaluation, the results of Tukey's HSD also indicated that overall multi-hint AI produced higher scores than single-hint AI.

As introduced in previous chapters, for user-AI cooperation system implementation, controllability, comprehensive and predictability are the common challenges for user-AI creative cooperation. The comparison results show that multi-hint AI can improve the predictability of such interaction. However, comprehensive and controllability did not improve by using multi-hint AI. However, for UI design task, multi-hint AI is more inspiring.



**Figure 37. Box plots of user ratings of each item according to each condition and result of one-way repeated-measures ANOVA. ((a) Single-hint, (b) Multiple-hint). Statistically significant results are reported as  $p < 0.05^*$ ). Results are collected from Matlab ANOVA function.**

comparison 4		
(a) Single-hint	(b) Multiple-hint	
<i>item</i>	<i>difference</i>	<i>p-value</i>
predictability	-1.8000	0.0034
controllability	-1.0000	0.1170
comprehensive	-0.8000	0.2133

**Table 4. Results of Tukey's HSD test. User-AI cooperation UX evaluation comparison between multi-hint AI and Single-hint AI. Results are collected from Matlab multiple comparison functions.**

## **8.2. Qualitative Analysis**

The goal of qualitative analysis is to investigate the users' thoughts in more depth and derive hidden characteristics behind the survey results. Specifically, we sought to identify the participants' perceptions of initiative and communication methods, the features they showed, and the factors they valued in interacting with the AI. We identified that the participants wanted the AI to provide detailed instructions but only when they wanted it to do so. In addition, they wanted to make every decision during the tasks. They sometimes anthropomorphized the AI and demonstrated a clear distinction between human and nonhuman characteristics. Finally, they reported that drawing with AI was a positive experience that they had never had before.

### **8.2.1. Expect more details**

When conducting the UI design tasks, most of the participants described they expected to see more details in the outputs. The participants expected that AI can create details such as icons, buttons and image views. 4 participants expected the AI can finish the rest of UI design task and provide usable UI designs.

After playing with other AI demos such as pix2pix and PaintsChainer, the participants found the AI implemented in this work performed worse than those examples (the AI results did not match their expectation). Participant 1 said the better AI performance was expected after the AI demo presented, while the AI's performance on UI design made participant 1 feel frustrated. Participant 8 said, "I cannot give very positive comments based on this AI's generations compared to the other AI demos you showed me"

Even though most participants described they preferred AI with multiple-hint, two participants preferred AI with single-hint. As they thought that AI just provided the general potential results, a user should consider it as a reference rather than producing fine-grained results. Participant 4 said, "...I felt more satisfied with single-hint AI, because I just need to see how the final UI might look like if it is going to be transformed from this wireframe. Too many outputs are redundant". Participant 8 described it as, "the results designed by AI did not match my intention, but it shows the final visual look which wireframes do not".

### **8.2.2. Inspiration**

Inspiration of this user-AI cooperation system was evaluated as one UX criteria. As is proven in the quantitative evaluation results, multi-hint AI is more inspiring for UI design task. Although there is no quantitative evaluation between no-AI mode and single-hint

AI mode, in the interviews most participants thought the results generated by AI (single-hint or multi-hint) were inspiring. However, the way that AI inspired participants varies.

However, some participants thought AI is not very helpful for inspiration. The reason is that participants considered wireframe design and visual design should be separate stages, they do not want to think about visual design until they are satisfied with the wireframe. They preferred not to take the AI generation into consideration when designed the wireframe. Participant 8 said, "the single hint is totally useless, so I did not take it into consideration at all...". Participant 6 also had the same feeling that when designing the wireframe, all the attention was put on the wireframe design thus the single-hint was completely ignored. Such feedback was mostly given when the participants used the system in single-hint AI mode.

Besides, the AI also seems to limit user's interaction. Users may consider the goal is to make AI generate more perfect and usable results rather than take the initiative themselves. As participant 3 said, "I just treated the AI's visual output as the target, so every action I made is to make the visual output look better".

Although some negative feedback was given by the participants, most participants saw the positive possibilities. As is both proven in quantitative and qualitative analysis, the participants preferred working with multi-hint mode AI. Participant 1 said, "I like the results provided by AI, because I can directly see different color schemas and where to draw the correct color. Traditionally even if the color schema was provided for, it still would take me some time to decide which color fits the UI component. AI eases the workload".

Multiple presentations increase AI's usability. One of the reasons is that when multiple results are shown at the same time, the participants thought they tended to compare the generated UI within the group and made better decisions. Participant 10 said, "When I saw these multiple generations, I was pleased to see so many color schema provided. I can compare them to choose the best for later visual design".

Although AI's outputs are blurry, it is still able to help user to refine their work. Many participants indicated that no matter single-hint AI or multiple-hint AI, they can make sense of the blurry UI generations. The participants thought the generations provide an overview of the final UI, which traditional wireframe design tools are not capable of. The reason is that AI would generate extra textures and details beside modifying the UI components, and these changes made the participants feel the UI drafts they designed become closer to the final usable UI. Especially the participants claimed that unexpected details from AI results made the drafts more reliable. 4 of them pointed out that even though the details of components and textures may not match their intentions, UI generations actually look like potential blurry versions of the real UI. Since the participants could make sense of the generations, the AI hint helped them to refine the

wireframe designs. As participant 2 said, "... even though I don't like the generation (single-hint), I can adjust the UI components by checking the result. Directly observing from the wireframe canvas, it may look fine, but AI added some textures and details making it more like real UI. From the (AI-generated) UIs I can tell that these components may be too sparse so I will adjust to make it look better". Participant 8 also said, "although the generations look rough and less aesthetic (compared to human designing), but it refined the layouts so that I can adjust and change my designs". Participant 9 said, "The blank space on the wireframe canvas may look pretty OK, but in the visual design stage these blank spaces may be filled with some images or color. In such case, the wireframe's layout may not be suitable for visual design. But AI helped me to imagine the final visual designs and see which part could look ugly if I continue to do the visual design... It also helped me to see which components should be on the upper layer." Participant 10 said, "after AI colorized my wireframe, I found there might be some flaws in my original layouts. Those parts looked unbalanced on the AI generations help me discover the improvements."

On the other hand, the blurry generation also caused unsatisfactory reactions. Participant 9 described, "... the AI generated some weird noises on the edge of the user's head image component, which made the avatar look like it is corroded and made me think I did wrong designs". Participant 3 said, "I did every modification based on the AI's generation, so AI's generation becomes the evaluation metric. Every step I took was to please the AI so that it can generate proper results".

### **8.2.3. Co-creations**

Participants tended to take controls or give instructions to the intelligent agent. One of the reasons is that most of the participants thought that for UI design, the visuals and functions should match the software requirements. Thus simply providing the UI wireframe is not enough for actual UI design work. Besides, AI generation is uncontrollable without extra restriction. It will generate something unexpected, which is a double-edged sword. Another reason is that, communication is important for co-creation (Oh et al., 2018). Users would wish to give instructions to the intelligent agent. However, during the testing, 6 participants gave negative comments about the co-creation. Participant 4 and participant 5 felt there was no cooperation during the test. Participant 4 said, "I did not feel any cooperation in this test, because it seems the intelligent agent did not follow my thoughts. The generated results would not be directly used in this actual work". Participant 5 said, "I wish I could give some extra information to it so that the interaction could be more cooperative".

AI function is still regarded as a function or a tool rather than a cooperator. The participants mentioned that AI actually did not work as they expected, as the results generated are rough. Participant 1 said, "...Compared to the AI examples shown in the



beginning (*Sketch-RNN*, and *pix2pix*), this user-AI UI design tool is less satisfying and not that smart. I would consider it as an integrated function". Participant 2 said, "This function is good and helpful for me to check if my current operation is appropriate". Participant 3, participant 4 and participant 5 thought that AI algorithm implemented in this work is not smart enough to be qualified as AI.

AI could lighten the workload. 8 of the participants mentioned that AI's generation could make them productive and their work easier, which matches the discussion in previous inspiration part. Since AI's UI generations gave the participant better UI structure views, the participants thought it will reduce the iteration workload. Participant 1, 2, 3, 8, 9, 10 mentioned that with this AI integration, there would be fewer refining steps and the generations could be used for checking UI elements adjustments.

## 9. Discussion

### 9.1. Design recommendations

**AI still needs the human guidance:** Based on the feedback from the interviews, many participants indicated they preferred to give AI some guidance so that it could be more useful. For example, the goal of simple works like sketch drawing in *Sketch-RNN* and the colorization in *Paintschainer* are quite general and subjective. In this study, the user is assigned to a task-oriented creative work, in which the user needs to make the final work usable and meet software requirements for UI. From the quantitative analysis, the participants think the AI generates some meaningful results, but those results more or less mismatch their intention or task's (software development) requirements. Even though they were inspired and surprised by AI generations, they would like to refine the generated UI results by taking control of the whole design interaction.

Another reason mentioned by the participants, is that AI may not be able to understand or meet the needs of high-level requirements. For example, the designer thinks of business software's UI, certain specific color schema should not be chosen based on the context. AI may generate the possible visuals which contain inappropriate color schema. The user could refine the visual design but meanwhile the user also expects the AI fixes refining post-process automatically. In this case, AI should be given extra information as a constraint during the task so that it could generate UI that matches designer's expectations.

This conclusion is similar to the conclusion given by Oh et al. in their work: *Let user take the initiative*. "Repetitive and arduous tasks should be assigned to the AI and creative and major tasks should be assigned to the user", because the user wants to make decisions and take initiative in collaboration (Oh et al., 2018).

Besides, the implemented algorithm should be interaction based, or at least should consider extra information as input rather than a single input. One possible way to give AI human constraint is *PaintsChainer*, beside inputting only sketches, the wanted colors were input as well so that the colorization could follow the constraints given by the human.

**Making AI Generate multiple possible choices is helpful:** As was revealed in the quantitative analysis, the participants preferred the multiple-hint style, even though both generations (single-hint and multiple-hint) are blurry and could be unsatisfactory. Multiple generations provided by AI show more possibility to increase the usability, which is shown in the quantitative analysis and interviews.

From the aspect of predictability, users have higher possibility to find that AI generate their desired results, and those low-quality results tend to be ignored by users.

From the perspective of inspiration, multiple generations offer users a chance to weigh the pros and cons of different color schema. As is mentioned in qualitative analysis, multiple outputs inspired the participants in color choosing and structure modification.

Besides, by choosing the AI generations, the participants felt they were taking the control, or taking the initiative. This is similar to the conclusion above, that users always want to take the initiative in the collaboration process (Oh et al., 2018). Thus providing multiple results also contribute to giving user the initiative during user-AI communication.

**AI's presentation should attract user's attention properly:** Both survey and interviews' results showed that single-hint does not affect much user experience. One reason is that in single-hint style, the AI result was generating and showing on the side of the tool's canvas. While the participants were designing the UI layouts, they can hardly transfer their attention to the AI generation canvas from the wireframe design canvas. This could also be the reason why multiple-hint style AI shows significant contributions to the user experience.

In the single-hint style mode, if a participant did a minor modification on the design canvas, there would not be significant change on the AI display canvas, participants tend to ignore such changes. However, in multiple-hint style mode, a minor modification on the wireframe canvas will cause multiple changes on AI generation canvas. In this case, the participants could notice the changes more easily and AI becomes more attractive. However, the question aroused by multi-hint AI is how to properly present the AI generations in the creative cooperation. As the AI agents implemented in related works (DuetDraw, ColorAIze) only generate single output, it is easy to directly project AI's creations onto the drawing canvas. Users can easily notice the changes and be aware that they are collaborating with AI. For multi-hint AI, it is impossible to project the results in the same way. Although Drawing Apprentice showed the example, that multiple AI drawn sketch lines are projected on the canvas, it can be hardly applied to this work: in the Drawing Apprentice situation, the lines drawn by AI only take a little space on the canvas, thus multiple results could be shown simultaneously. While UI generation is more complicated, the generated components not only take most areas of the canvas space, they also contain different textures and details. Thus when designing and implementing a user-AI cooperation system, it is necessary to consider how AI's generation will be presented.

## 9.2. Limitations of this work

**Software** is one of the issues during the test. During the pilot test, the feedback was that the participants were not familiar with the design tool *Pencil*, and the UI components were insufficient in the software. This caused problems in consistency and satisfaction evaluation. Even though before the evaluation, more UI components were offered as extra choices for the participants to use, based on the interview feedbacks the user-AI UI design

tool still had the similar issues. Participant 8 said, "... this design tool looks a bit out-of-date to me, I am not satisfied also because of the tool itself regardless of the AI part". Participant 4 said, "I feel I was constantly searching for the UI components I wanted, but so hard to find what I expected". The interaction of the chosen wireframe design tool also made participants frustrated. Participant 9 said, "the main difficulty in this test was actually the software. Like when I tried to make the corner of the rectangle round, I felt so annoying that it didn't work. I actually prefer hand drawing if possible".

**Complicated tasks** Another factor that affected the result in the experiment could be task complexity. Although each task was designed to be finished within 10 minutes, in the actual test it took 25 minutes for each participant to finish single task on average. Based on the observation, some participants became a bit frustrated and impatient when trying to finish the final task. Besides software's issue, the complicated design tasks could be another factor that causes the experiment to take longer than expected.

Compared to related works, the task for user-AI UI design tool in this work is more complicated. *Sketch-RNN* and *DuetDraw* simply provided a canvas that allows users to draw images with no restrictions. *Drawing apprentice* is relatively more complex, as it is like a combination of *Sketch-RNN* and *Paintschainer*. Although users can still freely draw objects and the structure of the painting is quite simple, as is shown in section 5.1. Compared to these works, user-AI UI design tool 's task is more complicated: for UI wireframe design, the participants would choose the UI components and do multiple refinements; while for sketching tasks, it just needs simple drawing which is more intuitive for many people.

Besides, each task also included multiple steps (designing wireframes, refining wireframe based on AI hint, describing the final visual design), which undoubtedly increased the complexity of the experiment. Many steps made the task's goal unclear. For example, some participants felt the goal for them was to finish the usable visual design rather than a rough prototype. However, with the tool offered in the test, producing usable visual design can hardly be achieved, thus they tended to keep refining the sketch and got frustrated. One of the participants also replied that the hint is useless since the wireframe design and visual design should be individual stages, thus AI's hint was not expected to be seen while building the wireframes.

**Presentation of the AI:** As is shown in the quantitative analysis and discussed in previous sections, one of the reasons that participants tend to think AI's single-hint is useless because it is easy to be ignored. As described by the participants, during the wireframe design, they wanted to focus on current work and canvas. However, although the hint was drawn by AI in real time, the participants hardly notice it; unless they chose to see the AI hint on purpose. Thus the presentation in this work somehow makes the interaction system less intelligent and less natural.

Compared to the single-hint mode, the multiple-hint mode is more attention attracting. The multiple results could generate simultaneously, which will cause obviously changes on the screen. Based on the observations, participants tend to switch their attention to the AI's results.

In the previous works, the presentation styles of the AI hint were various. Generally, there were 2 way to present AI's work: 1) on the side and 2) on the canvas. For example, *Paintschainer* shows the AI painted on the left of user's drawing canvas, which is similar as what is implemented in this work; *Sketch-RNN*'s multiple prediction version also shows different AI drawn sketches on the left of user's canvas. In Oh et al.'s work, their AI-user co-creation drawing tool directly shows the AI's work on the same canvas, on which user drew the sketches. Besides, in Matulic's and Davis' works, AI generated results were also directly shown on the same canvas. As one challenge for designing IIS, the presentation of intelligent agent in IIS could affect user's satisfaction and adaption. In conclusion, although user-AI cooperation's effects on design tasks have been studied, how AI's hint should be presented to participants is still a problem to be solved.

**Algorithm** is the most unpredictable factor both in evaluation and actual usage. In this work, the implemented algorithm mainly has the following problems: 1) the performance depends on the task and training data; 2) unstable outputs and 3) lack of constraints.

The performance of the algorithm varies from task to task. As is described in section 5.4, there was a pilot test conducted before the final implementation to observe user's initial impression about user-AI co-creation. The generative model showed impressive results for shoe design task and desert drawing task, in which participant drew the object's sketch on the left canvas and generative model output AI-drawn image on the right. However, compared to these tasks, AI generation is blurrier. The feedback from the participants was that UI generation's quality is worse than expected.

Unstable output is another issue affects user experience. As is discussed previously in section 5, for the CGAN and the BicycleGAN implemented in this work, some extra noise (latent) were used as conditional information so that the model could generate multiple possible outputs. The latent vector (which guides AI generate different styles) in this work are randomized, which made the AI results unstable and cannot be controlled by the participants. From the perspective of user, these AI outputs may not be able to match user's design intent or user cannot make sense of the outputs.

Lack of the human guide is another issue, which makes the user-AI UI design tool in this work less comparable to other works. Different algorithms provide different way of interaction. For example, *Paintschainer* not only could finish painting based on the user's sketches, it also could be constraint to generate the results based on colorization information given by user. For user-AI UI design tool, the wireframe sketch is the only

input for AI (BicycleGAN). Although user-AI UI design tool could generate several possible UI suggestions, the results still do not match designer's expectations and ideas. As one comment from the participants described, designer would prefer to give rough colorization instructions to AI so that it can generate appropriate results rather than random paintings.

### **9.3. Future work**

As was discussed above, the limitations in this work are mainly four aspects: 1) the tool chosen for UI design is not very usable, which affects the user experience itself; 2) relatively complicated tasks also bring negative user experience; 3) algorithms did not perform ideally and 4) the presentation of different AI styles are limited. For future work, the improvements would be focused on each of these downsides. Considering the participants felt frustrated about the wireframe tool used in this work, the design tool should be selected carefully.

In the future study, how AI should be presented to user during user-AI cooperation should be well-explored. The comparison experiments of different AI presentations should be conducted. For example, for single AI hint, it is necessary need to study what is the difference between displaying AI result within the canvas and displaying AI result on the side; for multiple AI hints, it is necessary to explore more possible presentation styles about how these hints should be shown to users. Besides, the design tasks should be simplified as well.

Since for IIS, the interaction is highly related to algorithms. The AI algorithm's performance should be improved as well. A comparison study of the AI algorithms is needed to discover which AI model can provide proper UI design performance.

## 10. Conclusion

To explore the potential how current AI technique could be integrated into practical design works, this research implemented an AI-based UI design tool. The previous user-AI cooperative design tasks such as DuetDraw (Oh et al. 2018) and Drawing Apprentice (Davis et al, 2016), concluded general user-AI principles by studying the user experience of user-AI cooperation from the simple sketch drawing. However, the sketch task is relatively simple and aimless compared to more complex design tasks. Derived from the research of Oh et al. (2018), Davis et al. (2016) and Matulic et al. (2018), three questions (how AI affects user experience of UI design; how different AI models affect the user experiences of UI design; what factors matters when implementing IIS) and three hypothesis (the results and conclusions for UI design task would be different from the previous research; multiple-outputs AI can at least improve predictability, controllability; multiple-output AI may deteriorate the user experience in comprehensibility, learnability and brings unexpected negatives) were proposed to study the user experience of such user-AI cooperation for UI design.

By conducting the experiments, the results of user experience turned out to be similar as previous research but with following different findings:

- For UI design task, the current state-of-the-art image generation algorithm does not perform ideally, which makes the user-AI cooperation less usable.
- AI features for UI design are useful, fun and fulfilling. But unlike previous research (Oh et al. 2018), it does not improve effectiveness.
- AI features are easy to adapt in this case, same as previous research (Oh et al. 2018), it is uncontrollable.

For the three hypotheses, the conclusion is that:

- Multiple-hint AI in general is preferred by users.
- Compared to single-hint AI, multiple-hint AI did not introduce any negative affects to the UI design task.
- Multiple-hint AI improves predictability and is more inspiring. However, it is still uncontrollable.

Beside these findings, this research also found that previous research on user-AI cooperation ignored or did not study the presentation of AI when designing the user-AI cooperation interaction. Based on the interviews, users tended to address the importance of the presentation of AI generated images, which has huge impacts on the user experience.

As discussed in Chapter 9.3, for the future work, the AI algorithm could be designed to be more controllable, and the research questions should be focus on how to design the AI presentations for different AI models.

## References

- Bao, J., Chen, D., Wen, F., Li, H., & Hua, G. (2017). CVAE-GAN: fine-grained image generation through asymmetric training. In *Proceedings of the IEEE International Conference on Computer Vision* (pp. 2745-2754).
- Carmona, K., Finley, E., & Li, M. (2018). The Relationship Between User Experience and Machine Learning.
- Chao, G. (2009, March). Human-computer interaction: process and principles of human-computer interface design. In *2009 International Conference on Computer and Automation Engineering* (pp. 230-233). IEEE.
- Colton, S., & Wiggins, G. A. (2012, August). Computational creativity: The final frontier?. In *Ecai* (Vol. 12, pp. 21-26).
- Davis, N. M., Hsiao, C. P., Singh, K. Y., Li, L., Moningi, S., & Magerko, B. (2015, June). Drawing Apprentice: An Enactive Co-Creative Agent for Artistic Collaboration. In *Creativity & Cognition* (pp. 185-186).
- Davis, N., Hsiao, C. P., Yashraj Singh, K., Li, L., & Magerko, B. (2016, March). 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* (pp. 196-207). ACM.
- Deka, B., Huang, Z., Franzen, C., Hibschan, J., Afergan, D., Li, Y., ... & Kumar, R. (2017, October). RICO: A mobile app dataset for building data-driven design applications. In *Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology* (pp. 845-854). ACM.
- Deka, B., Huang, Z., Franzen, C., Hibschan, J., Afergan, D., Li, Y., ... & Kumar, R. (2017, October). RICO: A mobile app dataset for building data-driven design applications. In *Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology* (pp. 845-854). ACM.
- Doersch, C. (2016). Tutorial on variational autoencoders. *arXiv preprint arXiv:1606.05908*.
- Dove, G., Halskov, K., Forlizzi, J., & Zimmerman, J. (2017, May). UX design innovation: Challenges for working with machine learning as a design material. In *Proceedings of the 2017 chi conference on human factors in computing systems*(pp. 278-288). ACM.Retrieved from
- E. Horvitz, A. Jacobs, and D. Hovel. Attention-sensitive alerting. In proceedings of Conference on Uncertainty and Artificial Intelligence, pages 305–313, Stockholm, Sweden, 1999.
- Ebner, M., Stickel, C., & Kolbitsch, J. (2010, November). iPhone/iPad human interface design. In *Symposium of the Austrian HCI and Usability Engineering Group* (pp. 489-492). Springer, Berlin, Heidelberg.



Eigenfeldt, A., Burnett, A., & Pasquier, P. (2012, May). Evaluating musical metacreation in a live performance context. In *Proceedings of the Third International Conference on Computational Creativity* (pp. 140-144).

Elgammal, A., Liu, B., Elhoseiny, M., & Mazzone, M. (2017). Can: Creative adversarial networks, generating "art" by learning about styles and deviating from style norms. *arXiv preprint arXiv:1706.07068*.

Evolus, Pencil project: <https://pencil.evolus.vn/>

Feldman, S. S. (2017, July). Co-creation: human and AI collaboration in creative expression. In *Proceedings of the conference on Electronic Visualisation and the Arts* (pp. 422-429). BCS Learning & Development Ltd.

Findlater, L., & McGrenere, J. (2004, April). A comparison of static, adaptive, and adaptable menus. In *Proceedings of the SIGCHI conference on Human factors in computing systems* (pp. 89-96). ACM.

G. Dorais, R. Kortenkamp, B. Pell, and D. Schreckenghost. Adjustable autonomy for human-centered autonomous systems on mars. In *proceedings of the First International Conference of the Mars Society*, pages 397–420, 1998.

G. Ferguson, J. Allen, and B. Miller. Towards a mixed-initiative planning assistant. In *proceedings of the Third conference on Artificial Intelligence Planning Systems*, pages 70–77, 1996.

Gajos, K. Z., Everitt, K., Tan, D. S., Czerwinski, M., & Weld, D. S. (2008, April). Predictability and accuracy in adaptive user interfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (pp. 1271-1274). ACM.

Garrett, J. J. (2010). *Elements of user experience, the: user-centered design for the web and beyond*. Pearson Education. Pages 132-148

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., ... & Bengio, Y. (2014). Generative adversarial nets. In *Advances in neural information processing systems* (pp. 2672-2680).

Goodfellow, Ian, Yoshua Bengio, and Aaron Courville. *Deep learning*. MIT press, 2016. Pages 654-714

Grudin, J. (2009). AI and HCI: Two fields divided by a common focus. *Ai Magazine*, 30(4), 48-48.

Ha, D., & Eck, D. (2017). A neural representation of sketch drawings. *arXiv preprint arXiv:1704.03477*.

Hartmann, M. (2009, January). Challenges in Developing User-Adaptive Intelligent User Interfaces. In *LWA* (pp. ABIS-6).

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 770-778).

- He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 770-778).
- Hewett, T. T., Baecker, R., Card, S., Carey, T., Gasen, J., Mantei, M., ... & Verplank, W. (1992). *ACM SIGCHI curricula for human-computer interaction*. ACM.
- Hinton, G. E., Osindero, S., and Teh, Y.-W. (2006). A fast learning algorithm for deep belief nets. *Neural Computation*, 18(7):1527–1554.
- Höök, K. (2000). Steps to take before intelligent user interfaces become real. *Interacting with computers*, 12(4), 409-426.
- <http://doi.acm.org.ezproxy.lib.utexas.edu/10.1145/3025453.3025739> doi: 10.1145/3025453.3025739
- Huang, A., & Wu, R. (2016). Deep learning for music. *arXiv preprint arXiv:1606.04930*.
- Isaac, D. (2016, December 21). *Conditional Variational Autoencoders*. Retrieved from <http://ijdykeman.github.io/ml/2016/12/21/cvae.html>
- Isola, P., Zhu, J. Y., Zhou, T., & Efros, A. A. (2017). Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 1125-1134).
- J. Gunderson and W. Martin. Effects of uncertainty on variable autonomy in maintenance robots. In *Workshop on Autonomy Control Software*, pages 26–34, 1999.
- Jameson, A. D. (2009). Understanding and dealing with usability side effects of intelligent processing. *AI Magazine*, 30(4), 23-23.
- Jameson, A., & Schwarzkopf, E. (2002, May). Pros and cons of controllability: An empirical study. In *International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems* (pp. 193-202). Springer, Berlin, Heidelberg.
- John Collier. 1957. Photography in anthropology: a report on two experiments. *American anthropologist* 59, 5 (1957), 843–859.
- Jordan, M. (2018). Artificial intelligence—The revolution hasn't happened yet. Medium. Apr, 19. Retrieved from: <https://corped.berkeley.edu/artificial-intelligence%E2%80%8A-%E2%80%8Athe-revolution-hasnt-happened-yet/>
- Karimi, P., Grace, K., Maher, M. L., & Davis, N. (2018). Evaluating Creativity in Computational Co-Creative Systems. *arXiv preprint arXiv:1807.09886*.
- Karimi, P., Grace, K., Maher, M. L., & Davis, N. (2018). Evaluating Creativity in Computational Co-Creative Systems. *arXiv preprint arXiv:1807.09886*.
- Kay, J. (2001). Learner control. *User modeling and user-adapted interaction*, 11(1-2), 111-127.

Kingma, D. P., & Welling, M. (2013). Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*.

Larsen, A. B. L., Sønderby, S. K., Larochelle, H., & Winther, O. (2015). Autoencoding beyond pixels using a learned similarity metric. *arXiv preprint arXiv:1512.09300*.

LeCun, Y., Haffner, P., Bottou, L., & Bengio, Y. (1999). Object recognition with gradient-based learning. In *Shape, contour and grouping in computer vision* (pp. 319-345). Springer, Berlin, Heidelberg.

LeeTiernan, S., Cutrell, E., Czerwinski, M., & Hoffman, H. G. (2001). Effective Notification Systems Depend on User Trust. In *INTERACT* (pp. 684-685).

Li, J. (2017). PixivDataset. [https://github.com/jerryli27/pixiv\\_dataset](https://github.com/jerryli27/pixiv_dataset)

Licklider, J. C. R. (1960). Man-computer symbiosis. *IRE transactions on human factors in electronics*, (1), 4-11.

Lieberman, H. (1984). Seeing what your programs are doing. *International Journal of Man-Machine Studies*, 21(4), 311-331.

Lieberman, H. (2009). User interface goals, AI opportunities. *AI Magazine*, 30(4), 16-16.

Lieberman, H. (2009). User interface goals, AI opportunities. *AI Magazine*, 30(4), 16-16.

Lipton, Z. C., Berkowitz, J., & Elkan, C. (2015). A critical review of recurrent neural networks for sequence learning. *arXiv preprint arXiv:1506.00019*.

Liu, R., Cao, J., Zhang, K., Gao, W., Liang, J., & Yang, L. (2016). When privacy meets usability: unobtrusive privacy permission recommendation system for mobile apps based on crowdsourcing. *IEEE Transactions on Services Computing*.

Liu, S., Wang, X., Liu, M., & Zhu, J. (2017). Towards better analysis of machine learning models: A visual analytics perspective. *Visual Informatics*, 1(1), 48-56.

Maes, P. (1995). Agents that reduce work and information overload. In *Readings in human-computer interaction* (pp. 811-821). Morgan Kaufmann.

Maheswaran, R. T., Tam, M., Varakantham, P., & Myers, K. (2003, July). Adjustable autonomy challenges in personal assistant agents: A position paper. In *International Workshop on Computational Autonomy* (pp. 187-194). Springer, Berlin, Heidelberg.

Maheswaran, R. T., Tambe, M., Varakantham, P., & Myers, K. (2003, July). Adjustable autonomy challenges in personal assistant agents: A position paper. In *International Workshop on Computational Autonomy* (pp. 187-194). Springer, Berlin, Heidelberg.

Mann, Y. (2016). Ai duet. *Experiments with Google*. See, <https://experiments.withgoogle.com/ai/ai-duet>.

- Mann, Y. (2016). Ai duet. *Experiments with Google*. See, <https://experiments.withgoogle.com/ai/ai-due t>.
- Matulic, F. (2018, November). ColourAIze: AI-Driven Colourisation of Paper Drawings with Interactive Projection System. In *Proceedings of the 2018 ACM International Conference on Interactive Surfaces and Spaces* (pp. 273-278). ACM.
- Moffat, D. C., & Kelly, M. (2006). An investigation into people's bias against computational creativity in music composition. *Assessment*, 13(11).
- Myers, B. A. (2007). A user acceptance equation for intelligent assistants. In *AAAI 2007 Spring Symposium on Interaction Challenges for Intelligent Assistants*.
- Ng, A. Y., & Jordan, M. I. (2002). On discriminative vs. generative classifiers: A comparison of logistic regression and naive bayes. In *Advances in neural information processing systems* (pp. 841-848).
- Oh, C., Song, J., Choi, J., Kim, S., Lee, S., & Suh, B. (2018, April). I lead, you help but only with enough details: Understanding user experience of co-creation with artificial intelligence. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems* (p. 649). ACM.
- Pearlmutter, B. A. (1989). Learning state space trajectories in recurrent neural networks. *Neural Computation*, 1(2), 263-269.
- Pytorch Project. <https://pytorch.org/>
- Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., & Sutskever, I. (2019). Language models are unsupervised multitask learners. *OpenAI Blog*, 1, 8.
- Ronneberger, O., Fischer, P., & Brox, T. (2015, October). U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention* (pp. 234-241). Springer, Cham.
- Russell, S. J., & Norvig, P. (2016). *Artificial intelligence: a modern approach*. Malaysia; Pearson Education Limited,. 1-18
- Samek, W., Wiegand, T., & Müller, K. R. (2017). Explainable artificial intelligence: Understanding, visualizing and interpreting deep learning models. *arXiv preprint arXiv:1708.08296*.
- Sangkloy, P., Lu, J., Fang, C., Yu, F., & Hays, J. (2017). Scribbler: Controlling deep image synthesis with sketch and color. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 5400-5409).
- Schmidhuber, J. (2015). Deep learning in neural networks: An overview. *Neural networks*, 61, 85-117.
- Shneiderman, B. (2007). Creativity support tools: Accelerating discovery and innovation. *Communications of the ACM*, 50(12), 20-32.

Srihari, S. (2010). Machine learning: Generative and discriminative models. *Machine Learning Course*: <http://www.cedar.buffalo.edu/srihari/CSE574/index.html>. Available at WWW: < <http://www.cedar.buffalo.edu/~srihari/CSE574/Discriminative-Generative.Pdf>.

Sturm, B. L., Ben-Tal, O., Monaghan, U., Collins, N., Herremans, D., Chew, E., ... & Pachet, F. (2019). Machine learning research that matters for music creation: A case study. *Journal of New Music Research*, 48(1), 36-55.

Von Gioi, R. G., Jakubowicz, J., Morel, J. M., & Randall, G. (2008). LSD: A fast line segment detector with a false detection control. *IEEE transactions on pattern analysis and machine intelligence*, 32(4), 722-732.

Wikipedia contributors. (2018, April 29). Intelligent user interface. In *Wikipedia, The Free Encyclopedia*. Retrieved 15:45, May 26, 2019, from [https://en.wikipedia.org/w/index.php?title=Intelligent\\_user\\_interface&oldid=838789781](https://en.wikipedia.org/w/index.php?title=Intelligent_user_interface&oldid=838789781)

Wikström, D. (2018). Me, Myself, and AI: Case study: human-machine co-creation explored in design. Retrieved from: <http://www.diva-portal.org/smash/get/diva2:1223277/FULLTEXT01.pdf>.

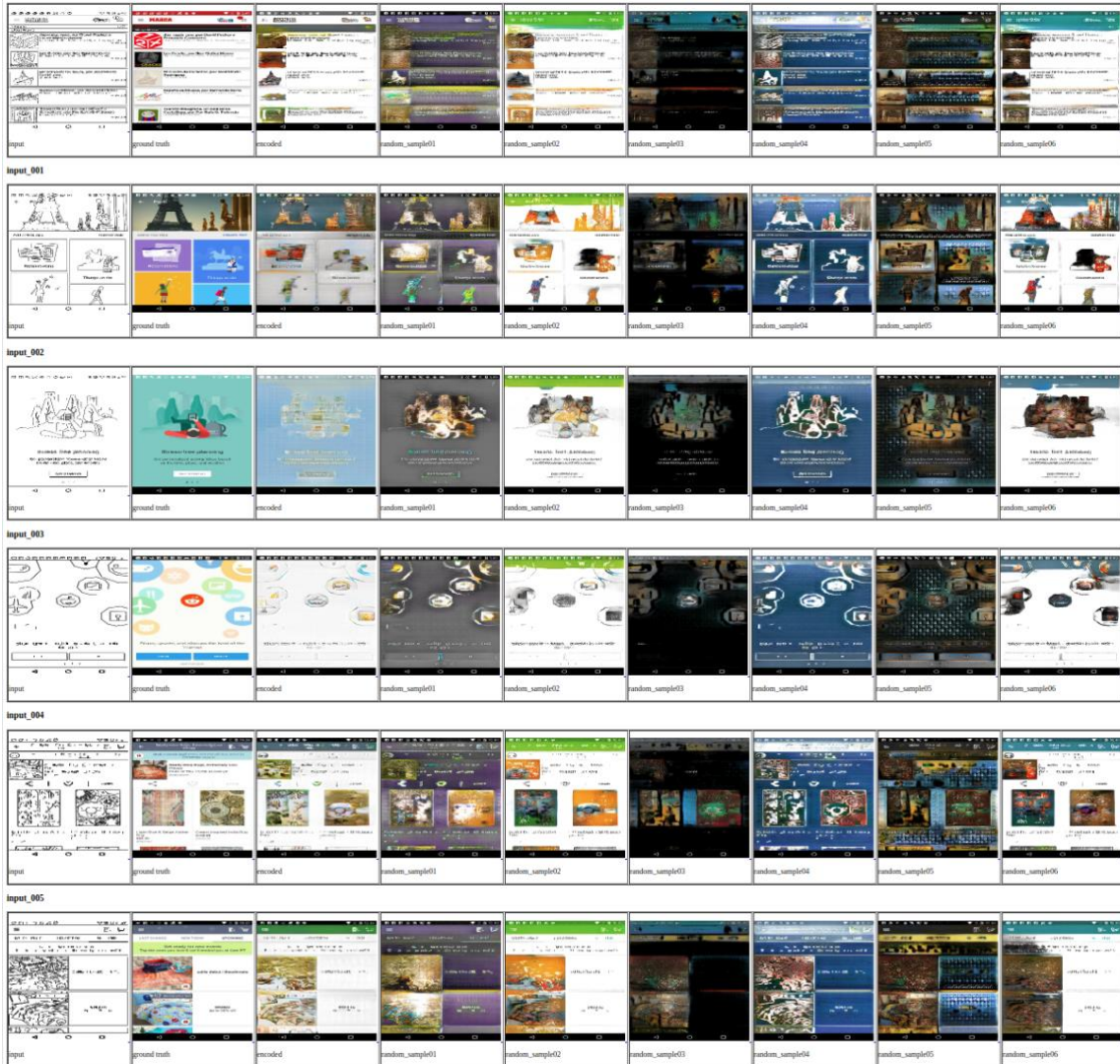
Xu, J., Li, H., & Zhou, S. (2015). An overview of deep generative models. *IETE Technical Review*, 32(2), 131-139.

Yonetsuji, T. (2017). Paintschainer. [https://paintschainer.preferred.tech/index\\_en.html](https://paintschainer.preferred.tech/index_en.html)

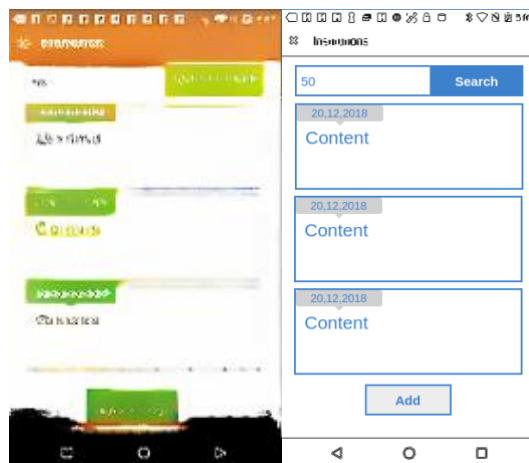
Zhang, R., Zhu, J. Y., Isola, P., Geng, X., Lin, A. S., Yu, T., & Efros, A. A. (2017). Real-time user-guided image colorization with learned deep priors. *arXiv preprint arXiv:1705.02999*.

Zhu, J. Y., Zhang, R., Pathak, D., Darrell, T., Efros, A. A., Wang, O., & Shechtman, E. (2017). Toward multimodal image-to-image translation. In *Advances in Neural Information Processing Systems* (pp. 465-476).

### The results generated by AI (BicycleGAN)



### The partial results of user-AI cooperation



## The questionnaire for participants

29/05/2019

User experience test consent form

### User experience test consent form

Please read and sign this form.

You have been invited to participate in a user experience test which is part of my master's thesis work at the University of Tampere. By participating in the test, you will help to evaluate the user experience of 360-degree video application.

In this user experience test:

- You will be assigned 3 simple UI designing tasks.
- You will be asked to make 3 mockups during the test.
- You will be asked to fill in 2 questionnaires.
- You will be asked to answer few questions regarding the demo you designed.
- The interview part of the experiment will be recorded as an audio recording.

Participation in this usability study is voluntary. All information will remain strictly confidential. The results and findings may be used to help improve the 360-degree video application. By participating the experiment, you can get a Finnkino movie ticket as a compensation.

You can withdraw your consent to the experiment and stop participation at any time. Feel free to ask any questions you may have about your participation.

If you have any questions after the experiment, please contact: [wenyan.yang@tut.fi](mailto:wenyan.yang@tut.fi)

I have read and understood the information on this form and had all of my questions answered

**1. Date and Place:**

---

**2. Signature**

---

**3. Name clarification**

---

**4. Email Address**

---

### Background questionnaires

Please take a few minutes to answer the following questions to help me better understand your background. I will use this information only to provide background and usage context in which to interpret the feedback you'll give me in the user study. I will keep your information confidential.

**5. Participate number**

---



29/05/2019

User experience test consent form

**6. Age**

*Tick all that apply.*

- Option 1
- 25-30
- 30-35
- 35+

**7. Gender**

*Mark only one oval.*

- Male
- Female
- Other: \_\_\_\_\_

**8. What is your occupation**

\_\_\_\_\_

**9. Do you have experience with UI design (or just mockups)**

*Mark only one oval.*

- yes
- no
- Other: \_\_\_\_\_

**10. Have you ever heard of AI or any AI related applications**

*Mark only one oval.*

- yes
- No
- Maybe

**11. Do you know what weak-AI is?**

*Mark only one oval.*

- yes
- No
- Maybe

**Show examples of weak-AI**

---

**12. What is your opinion about weak-AI**

*Mark only one oval.*

	1	2	3	4	5	
Negative (AI should be highly restricted)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Positive (I believe it will improve our civilization)

---

### Test scripts

In our test, we will ask you to refine a sketch to a better UI mockups. We will provide three UI sketches, and your work is to pick one of them and use the software "Pencil" to make it into a mockup.

You need to pick one of the given UI design tasks to make a UI wireframe. The goal of the task is to make a UI wireframe which you think is good enough for visual designing. You will be asked to choose 3 simple UI wireframe tasks.

steps:

### Criteria of UX evaluation (no-AI)

We selected 12 items from the criteria commonly used for user interface usability and user experience evaluations [1, 35] in consideration of the characteristics of the tasks: 1) useful, 2) easy to use, 3) easy to learn, 4) effective, 5) efficient, 6) comfortable, 7) communicative, 8) consistent, 9) fulfilling, 10) fun, and 11) satisfying

13. **Useful: It is useful/helpful for me to make UI prototype.**

Mark only one oval.

	1	2	3	4	5	6	7	
disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	agree

14. **Easy to learn: I easily remembered how to use it.**

Mark only one oval.

	1	2	3	4	5	6	7	
disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	agree

15. **Effective: I made a very useful UI prototype for visual designing.**

Mark only one oval.

	1	2	3	4	5	6	7	
disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	agree

16. **Efficient: will be measured by time recording.**

Example: 8.30 a.m. \_\_\_\_\_

17. **Comfortable: I felt comfortable/natural with all the interactions.**

Mark only one oval.

	1	2	3	4	5	6	7	
disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	agree

18. **Communicative: I was communicating my idea with the system during the design.**

Mark only one oval.

	1	2	3	4	5	6	7	
disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	agree

29/05/2019

User experience test consent form

**19. I did not notice any inconsistency as I used it.**

*Mark only one oval.*

	1	2	3	4	5	6	7	
disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	agree

**20. I felt fulfilled when I completed the prototype**

*Mark only one oval.*

	1	2	3	4	5	6	7	
disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	agree

**21. It is fun to use**

*Mark only one oval.*

	1	2	3	4	5	6	7	
disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	agree

**22. It worked the way I wanted it to work**

*Mark only one oval.*

	1	2	3	4	5	6	7	
disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	agree

### Criteria of UX evaluation (single)

We selected 12 items from the criteria commonly used for user interface usability and user experience evaluations [1, 35] in consideration of the characteristics of the tasks: 1) useful, 2) easy to use, 3) easy to learn, 4) effective, 5) efficient, 6) comfortable, 7) communicative, 8) consistent, 9) fulfilling, 10) fun, and 11) satisfying

**23. Useful: It is useful/helpful for me to make UI prototype.**

*Mark only one oval.*

	1	2	3	4	5	6	7	
disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	agree

**24. Easy to learn: I easily remembered how to use it.**

*Mark only one oval.*

	1	2	3	4	5	6	7	
disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	agree

**25. Effective: I made a very useful UI prototype for visual designing.**

*Mark only one oval.*

	1	2	3	4	5	6	7	
disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	agree

26. **Efficient: will be measured by time recording.**

*Example: 8.30 a.m.*

27. **Comfortable: I felt comfortable/natural with all the interactions.**

*Mark only one oval.*

	1	2	3	4	5	6	7	
disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	agree

28. **Communicative: I was communicating my idea with the system during the design.**

*Mark only one oval.*

	1	2	3	4	5	6	7	
disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	agree

29. **I did not notice any inconsistency as I used it.**

*Mark only one oval.*

	1	2	3	4	5	6	7	
disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	agree

30. **I felt fulfilled when I completed the prototype**

*Mark only one oval.*

	1	2	3	4	5	6	7	
disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	agree

31. **It is fun to use**

*Mark only one oval.*

	1	2	3	4	5	6	7	
disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	agree

32. **It worked the way I wanted it to work**

*Mark only one oval.*

	1	2	3	4	5	6	7	
disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	agree

**AI interface issue (single hint)**

three extra criteria that have been pointed out in the AI interface issue [18, 23, 48

predictability, comprehensibility, controllability

**33. Every feedback of the interaction matches my intention**

*Mark only one oval.*

	1	2	3	4	5	6	7	
disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	agree

**34. The results generated exactly as I expected it to be**

*Mark only one oval.*

	1	2	3	4	5	6	7	
disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	agree

**35. I think I was leading the design and all the interactions**

*Mark only one oval.*

	1	2	3	4	5	6	7	
disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	agree

**36. The system actions were totally controllable**

*Mark only one oval.*

	1	2	3	4	5	6	7	
disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	agree

**37. The generated results are unedrstandable**

*Mark only one oval.*

	1	2	3	4	5	6	7	
disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	agree

**38. I got positive feedbacks from the weak-AI hints**

*Mark only one oval.*

	1	2	3	4	5	6	7	
disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	agree

**39. My thoughts were affected and constrained by the weak-AI hints**

*Mark only one oval.*

	1	2	3	4	5	6	7	
disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	agree

**Criteria of UX evaluation (multi)**

We selected 12 items from the criteria commonly used for user interface usability and user experience evaluations [1, 35] in consideration of the characteristics of the tasks: 1) useful, 2) easy to use, 3) easy to learn, 4) effective, 5) efficient, 6) comfortable, 7) communicative, 8) consistent, 9) fulfilling, 10) fun, and 11) satisfying

29/05/2019

User experience test consent form

**48. It is fun to use**

Mark only one oval.

	1	2	3	4	5	6	7	
disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	agree

**49. It worked the way I wanted it to work**

Mark only one oval.

	1	2	3	4	5	6	7	
disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	agree

**AI interface issue (multi)**

three extra criteria that have been pointed out in the AI interface issue [18, 23, 48

predictability, comprehensibility, controllability

**50. Every feedback of the interaction matches my intention**

Mark only one oval.

	1	2	3	4	5	6	7	
disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	agree

**51. The results generated exactly as I expected it to be**

Mark only one oval.

	1	2	3	4	5	6	7	
disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	agree

**52. I think I was leading the design and all the interactions**

Mark only one oval.

	1	2	3	4	5	6	7	
disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	agree

**53. The system actions were totally controllable**

Mark only one oval.

	1	2	3	4	5	6	7	
disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	agree

**54. The generated results are unedrstandable**

Mark only one oval.

	1	2	3	4	5	6	7	
disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	agree

29/05/2019

User experience test consent form

40. **Useful: It is useful/helpful for me to make UI prototype.**

Mark only one oval.

	1	2	3	4	5	6	7	
disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	agree

41. **Easy to learn: I easily remembered how to use it.**

Mark only one oval.

	1	2	3	4	5	6	7	
disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	agree

42. **Effective: I made a very useful UI prototype for visual designing.**

Mark only one oval.

	1	2	3	4	5	6	7	
disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	agree

43. **Efficient: will be measured by time recording.**

Example: 8.30 a.m.

44. **Comfortable: I felt comfortable/natural with all the interactions.**

Mark only one oval.

	1	2	3	4	5	6	7	
disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	agree

45. **Communicative: I was communicating my idea with the system during the design.**

Mark only one oval.

	1	2	3	4	5	6	7	
disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	agree

46. **I did not notice any inconsistency as I used it.**

Mark only one oval.

	1	2	3	4	5	6	7	
disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	agree

47. **I felt fulfilled when I completed the prototype**

Mark only one oval.

	1	2	3	4	5	6	7	
disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	agree

29/05/2019

User experience test consent form

**55. I got positive feedbacks from the weak-AI hints**

*Mark only one oval.*

	1	2	3	4	5	6	7	
disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	agree

**56. My thoughts were affected and constrained by the weak-AI hints**

*Mark only one oval.*

	1	2	3	4	5	6	7	
disagree	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	agree

**Semi-structured questionnaires**

The drawn pictures will be showed to better recall the details.

**57. Can you describe how you are taking the weak-AI hint into your design strategy?**

---

---

---

---

---

**58. What is the most difficult part when you try to understand weak-AI generated results?**


---

---

---

---

---

Powered by  
 Google Forms