Bilkent University

Department of Computer Engineering

CS 491 - Senior Design Project I

DeePaint

Project Specifications Document

Project Group Members:

Yavuz Faruk Bakman - 21703309

Hande Sena Yılmaz - 21703465

Alperen Öziş - 21703804

Zübeyir Bodur - 21702382

Duygu Nur Yaldız - 21702333

Supervisor: Uğur Doğrusöz

Jury Members: Dr. Shervin Arashloo and Dr. Hamdi Dibeklioglu

Table of Contents

1. Introduction

During the last decade, social media applications have become widely used and part of our daily lives due to the easy accessibility of mobile devices and the increasing quality of their cameras. All around the world, people take photos constantly and share them on social media. For instance, 1074 photos are uploaded on Instagram every second [1]. While sharing photos, people have an urge to get rid of the imperfections in the photo such as a stranger in the background of a selfie or a car in a beautiful scene. Moreover, many want their photos to be more artistic than the real version; therefore, they adjust lighting, change style, etc.

However, applying these changes requires knowledge and experience in photoshop. A regular person can not achieve to remove an object from a photo in a realistic way. Nevertheless, this became a need for regular people on a daily basis. Therefore, applications that make these adjustments automatically have gained more and more demand. Those applications with a user-friendly interface, fast modification speed, and low error rate in modification (i.e. more realistic results) are preferred by the users.

Considering the demand for easy and accurate photoshop applications, we came up with the DeePaint project. In the rest of the report, we provide a description of the project, constraints, professional and ethical issues, and finally, functional and non-functional requirements of the project.

1.1. Description of the Project

DeePaint is a photoshop mobile application and allows users to edit their photographs with AI power. It brings together different machine learning models and tasks in one application. The goal of the project is to minimize human interaction in the process of photo editing/shop. Meaning that users will only prompt high level of instructions to the application to edit/shop their photographs.

DeePaint provides various functions to its users. Users will be able to remove an undesired object from the image. Moreover, they will be able to change the styling of an object in the image to a similar object of the same kind. They will also be allowed to change the styling of the photo similar to a famous artist's painting style like Picasso, Munch, and Salvador Dali. Increasing the resolution of a low-quality photo is also a function of the application. Besides mentioned features, more is offered by DeePaint. In order for the users to utilize the features of the application, they should be connected to the Internet.

1.1.1. Similar Products and Innovativeness

DeePaint is not a greenfield project. Even though there are similar applications that enable image manipulation, there are no commercial products that make it as easy as DeePaint, since DeePaint will

- use segmentation to classify and draw precise boundaries around specific objects. Should the users want to remove an object, instead of drawing the boundaries themselves or a small box around them, they will need to just press a remove button close to the already segmented object, which will then be filled according to the context (See 2.1.1).
- enable their users to edit a target object with a reference object (See 2.1.2).
- enable their users to generate and add new objects into the image (See 2.1.3).

1.2. Constraints
1.2.1. Implementation Constraints

- DeePaint mobile application will be written in Java programming language, and it will be on the Android platform.
- Since the mobile application is in the Android platform, Android Studio IDE will be used for development.
- For implementing deep learning features of the application, a server-side is required. Since those libraries, such as PyTorch, TensorFlow, Keras, Theano, are libraries that all can be used in Python, the server-side will be implemented in Python.
- Object-oriented programming will be used throughout the project.
- GitHub will be used for version control and collaboration, as well as opening issues if necessary.

1.2.2. Economic Constraints

- The mobile application will be free to use, and there will be no in-app purchases. Hence, all of the users will be able to use the features without making any payment.
- Most of the APIs that will be used during development are free and open source.

- For implementing and deploying deep learning features, AWS will be used [2], which gives students free credit of $100.

- The website for the project also does not require any cost, since GitHub Pages [3] is being used.

1.2.3. Social Constraints

- A "Terms and Conditions" is required due to the following, but not limited to:

  - Manipulation of truth at the social level. Even though DeePaint is deterministic in the sense that it shows the available images to be removed, this does not change that it will generate fake images.

  - Image editing without the approval of the owner. Should the owner of the images sue one of our users, the application has no responsibility.

1.2.4. Sustainability Constraints

- The mobile application will be scalable enough that users will be able to use the application without big maintenance breaks.

- To maintain the mobile application, users will be able to give feedback for requesting a feature or reporting bugs.

1.3. Professional and Ethical Issues
1.3.1. Professional Issues

- The Agile-Scrum methodology will be used throughout the project. There will be goals, priorities, roles, and tasks; making it a lot easier to develop complex projects such as DeePaint.

- Throughout the academic year 2021-2022, the GitHub repository of the application will be private. After the final demo is presented, it will be public.

- The work distributed to the group members will be equal, in terms of work packages. Work packages, though mostly focused on implementation, will include tasks other than implementation, such as documentation, preparing a trailer, presentation.

- Decision-making will be done according to democracy in the meetings.

1.3.2. Ethical Issues

- We will be bound with the Code of Ethics in Engineering, as well as General Data Protection Regulation Law (GDPR) [4], [5]. In fact, when the user opens the application for the first time, their permission for accessing their gallery will be asked.

- As mentioned in 1.2.4 (Social Constraints), the application may lead to false information. Moreover, there is a possibility that the images are edited without the consent of the owner. To handle this, all users will be agreeing to the "Terms and Conditions" by downloading the application.

2. Requirements

2.1. Functional Requirements

2.1.1. Automatic Segmentation for Content-Aware Fill

Our program automatically makes object segmentation. The user can select the segmented objects with a simple touch and the program removes it from the image. After the removal, the area is filled with background naturally by deep learning models. In that way, the user is not required to draw the boundaries of an object which might be a hard task on a mobile phone. However, the program also allows the user to make manual drawings. The user can also draw the boundaries of the area to remove and the program can remove the area and fill it with the background.

2.1.2. Editing a Target Object with Reference Object

The user can edit a target object or part of an object with a reference object. For instance, there is a bed in a bedroom and the user does not like its appearance. The user can edit the appearance of the bed with another bed the user likes (Fig. 1).
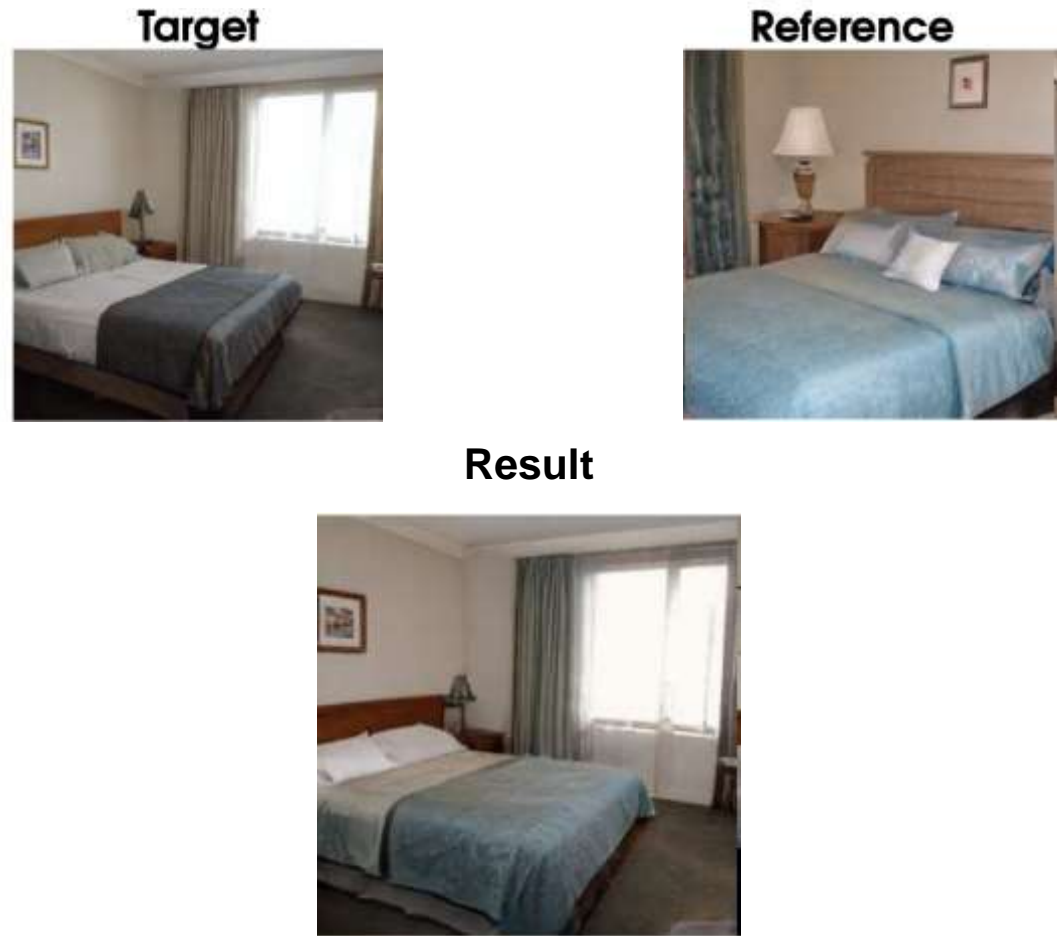
Figure 1: Style Change

2.1.3.   Generating and Adding New Objects into the Image

The user can draw the boundaries of an object and choose a class of an object. The program fills the bounded area with the selected object instance. For instance, the user draws the boundaries of a car on the road and the program puts a car regarding the shape of the drawings (Fig. 2).
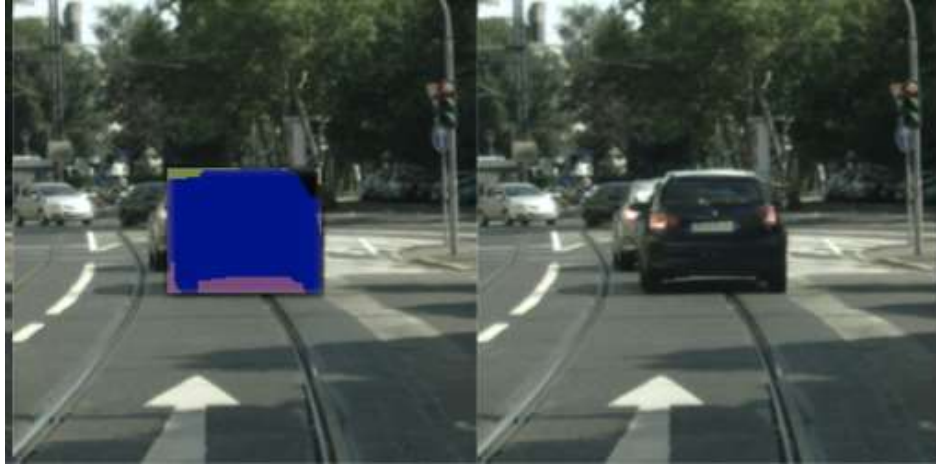
Figure 2: New object addition

2.1.4. Increasing the Resolution of Images

The user can increase the resolution of an image (Fig. 3).

Figure 3: Resolution increase

2.1.5. Making Artistic Style of Images

The user can add an artistic style to an image. For instance, a normal landscape photograph can be converted to an artistic image which is very similar to Van Gogh's artwork. The user chooses a style image to manipulate the original image's style (Fig. 4).

Figure 4: Artistic Styling

2.1.6. Cropping and Pasting Objects between Images

The user can crop an object from an image by only drawing a square around it and paste it to another image. DeePaint will determine the edges of the selected object and when the object is moved to the other photo, it will be realistic with the help of deep learning algorithms (Fig. 5).
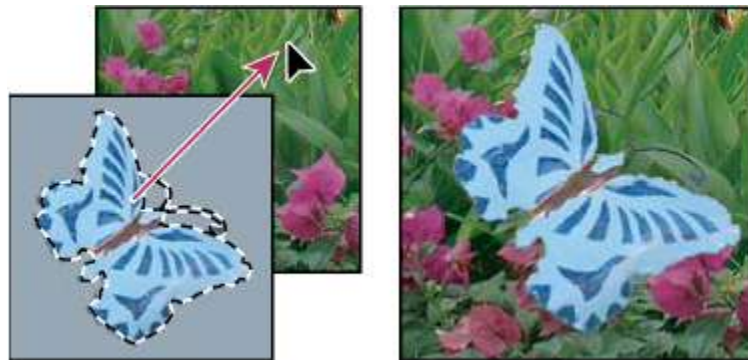

Figure 5: Cropping and Pasting Objects between Images

2.2. Non-Functional Requirements

2.2.1. Performance

While the user is choosing what to do with his photograph, (What to remove from the scene, choosing the artistic style, etc.) our program should be smooth with its reactions, namely in 100ms. Once the user has inputted his desire to our app, we should return the edited photo back to the user in at most 15 seconds.

2.2.2. Privacy

Since users will upload their personal photographs to our app, we must ensure every user can only access photographs uploaded by themselves. We will render this possible via basic authorization.

2.2.3. Usability

Since one of the rationales of our project is to rescue our users from dealing with the difficulty of the traditional photoshop apps, our app should be easy to use. A new user should be able to adapt to it in 2 minutes.

2.2.4. Extendibility

Given the highly increasing number of innovations in AI in recent years, our design patterns should allow us to easily add a new feature to our application or improve the existing ones in terms of accuracy, usability, etc.

2.2.5. Accuracy

Although there is not a universal metric to measure accuracy of our tasks numerically, our app should be able to output realistic images, to the extent of tricking a FakeorNot robot.

2.2.6. Security

Since the actual editing of the photo will be done in the cloud servers and not in the local workspace of the user, we cannot allow bots to use our application and keep our servers busy. Therefore, we will use basic authentication before allowing users to utilize our app.

3. References

[1] "Instagram by the Numbers: Stats, Demographics & Fun Facts," *Omnicore Agency*, Jan 3, 2021 [Online]. Available: https://www.omnicoreagency.com/instagram-statistics/ [Accessed Oct 10, 2021].

[2] J. Barr, "AWS Educate – Credits, Training, Content, and Collaboration for Students & Educators," *Amazon Web Services Inc* [Online]. Available: https://aws.amazon.com/blogs/aws/aws-educate-credits-training-content-and-collaboration-for-students-educators/. [Accessed Oct. 10, 2021].

[3] "GitHub Pages", *GitHub Pages.* [Online]. Available: https://pages.github.com/ [Accessed: Oct 10, 2021].

[4] "Code of Ethics," *Code of Ethics | National Society of Professional Engineers.* [Online]. Available: https://www.nspe.org/resources/ethics/code-ethics. [Accessed Oct. 10, 2021].

[5] "General Data Protection Regulation," *Intersoft Consulting.* [Online] Available: https://gdpr-info.eu/. [Accessed Oct. 10, 2021].