



Bilkent University
Department of Computer Engineering

Senior Design Project

Project short-name: Autoshop

Project Specifications Report

Efe Acer, Hikmet Demir, Mehmet Mert Duman, Talha Murathan Göktaş,
Burak Yaşar

Supervisor: Fazlı Can

Jury Members: Cevdet Aykanat and Ercüment Çiçek

Progress Report
October 14, 2019

This report is submitted to the Department of Computer Engineering of Bilkent University in partial fulfillment of the requirements of the Senior Design Project course CS491/2.

Contents

Introduction	3
1.1 Description	4
1.2 Similar Products and Technologies	4
1.3 Constraints	5
1.3.1 Development Constraints	5
1.3.2 Economic Constraints	6
1.3.3 Language Constraints	6
1.3.4 Social Constraints	6
1.4 Professional and Ethical Issues	7
1.4.1 Ethical Issues	7
1.4.1 Professional Issues	7
Requirements	7
2.1 Functional Requirements	7
2.1.1 System Functionality	7
2.1.2 User Functionality	9
2.2 Non-functional Requirements	10
2.2.1 Extensibility	10
2.2.2 Reliability	10
2.2.3 Usability	10
2.2.4 Accessibility	11
The system should:	11
be downloadable for free.	11
2.2.5 Portability	11
2.2.6 Efficiency	11
2.3 Pseudo Requirements	11
2.3.1 Issue Tracking	11
2.3.2 Testing	12
2.3.3 External Tools and Technologies	12
References	13

1 Introduction

With social media becoming prominent as a byproduct of the technology era, we began to see a very intense online photo traffic. To be more specific, over 300 million photos are uploaded to Facebook daily and there are around 40 billion photos shared on Instagram since its creation [1]. The direct implication of these numbers is that people love to take and share photos. They spend hours trying to capture effectual moments or just to look good. In an effort to make it easier for people to be happy with their photos; a new industry, "Photoshop" had emerged.

Photoshop itself, is a demanding task. People train themselves to become good photoshoppers and spend heaps of time in front of the screen to make realistic photo edits. The departure point of our project, "Autoshop", is to make this arduous task accessible to everyone, even the people knowing the absolute minimum of Photoshop.

The nature of innovation behind Autoshop is to create a new and powerful photo editing tool using state of the art Computer Science techniques. Autoshop will help people add objects to and remove objects from their photos without requiring any photoshop knowledge. Autoshop makes advanced technologies accessible in multiple platforms, aiming to dilate its user profile.

Imagine yourself missing a friend gathering, you would probably say "I wish I was there.". Autoshop helps you exactly at these moments. Using it, you will be able to effortlessly add yourself into the moment. You may argue that this is possible using tools such as Adobe's Photoshop, however, you would have to work quite a bit to do that. Autoshop, as its name suggests, automizes the process and makes your computer or your mobile phone your personal photoshopper.

Tons of use cases can be found regarding photo editing. For instance, one may want to purchase some furniture for her living room, but she may be indecisive about how well the furniture will fit into the room. With the help of Autoshop; she can take the photo of the piece of furniture and the living room, combine these as if they are actually in the same place, and then decide. Also, people can add their faces on top of any photo and any person's face, for example, their favorite rock band Queen's poster or the famous movie Marvel Avengers' cover photo.

In this report, we intend to provide an overall specification of the system. A brief description of the proposed system, Autoshop, will be given by emphasizing its unique features first. Later, existing systems together with their

scopes, differences and missing features will be discussed. Then system constraints surrounding economic, social, and other types of issues will be argued. Afterward, the professional and ethical responsibilities Autoshop brings along will be stated. Finally, functional and non-functional requirements will be listed.

1.1 Description

Autoshop is a photo-editing tool that makes use of cutting edge deep learning techniques to add objects to and remove objects from images. The tool is innovative in the sense that there is no commercial product that can perform context-aware addition and removal of objects. The task of context-aware object addition and removal is challenging since the texture, opacity, brightness and other photographic details of the added object should match those of the background image. Currently, this process is handled by professional photoshoppers and it is quite time-consuming. Hence, Autoshop will allow people to edit their photos effortlessly.

Advanced models and techniques such as neural networks, smart-cropping, padding, and image inpainting will constitute the backbone of Autoshop. The software will mostly revolve around the "sub" Computer Science topics; Deep Learning and Image Processing. Autoshop will demonstrate that such advance technologies can indeed be useful as solutions to practical problems.

Autoshop aims to have an extensive user profile. Thus, the software will have two versions; desktop and mobile. The desktop will enable professionals to use Autoshop in unison with the other photoshopping software they have. Besides that, the rationale behind the idea was to make photoshopping easy and accessible to everyone, so a mobile version is a must for us.

1.2 Similar Products and Technologies

Although NVIDIA open-sourced sophisticated deep learning models to remove objects from and add objects to images, there is no comprehensive commercial product that makes use of these technologies. There are also web-applications that offer the removal of unwanted objects from photos [2], however, these applications corrupt the inherent style of the photos while removing the objects from them. Additionally, there are industry-standard tools such as Adobe's Photoshop [3], albeit, these tools require advanced photoshop

knowledge to perform addition and removal of objects. On top of that, Adobe products are relatively expensive [4] and not fully-functional on mobile platforms; which implies that these products have serious accessibility issues.

1.3 Constraints

1.3.1 Development Constraints

- The mobile version of the application will target the Android platform, hence Android Studio IDE will be used for development. Android Studio primarily accepts Java and Kotlin as the development languages; since we are all fluent in Java we will proceed with it.
- The application will necessarily require a server-side. The server-side will be written mostly in Python, since there will be strong dependencies with Python deep learning and image processing libraries.
- The Graphical User Interface (GUI) of the desktop version of the application is planned to be written in Java. This way, we can make use of Java Virtual Machine (JVM) and run the application on any OS that has a JVM and can execute Python scripts.
- The application will be powered by NVIDIA's open-source libraries FastPhotoStyle and Image Inpainting. These libraries will be very beneficial since they allow us to use powerful pre-trained neural networks, which in turn reduces our dependency on data.
- The models used in the application will draw on certain academic papers such as "Image Inpainting for Irregular Holes Using Partial Convolutions" [5]. This paper presents many useful neural network implementations such as PConv-Keras from Mathias Gruber and an official implementation from the authors of the paper "partialconv - NVIDIA" [6], [7], [8], [9].
- Git and Github will be used for version controlling and tracking. Also, Github provides a wiki page and a project management tool which offer issue tracking [10].
- Slack will be used to provide a better communication within the team, in a professional manner.
- Object Oriented Programming paradigm will be applied in the entire project.

1.3.2 Economic Constraints

- The libraries that are planned to be used are open source and have the permissive licenses for free usage
- The application will require cloud computing, which will create server costs. Google Colab provides free GPU usage for educational purposes, this will be beneficial in the development phase. However, for the final product dedicated servers will be needed, the cost of these servers are to be determined.
- Google charges a one time fee of 25\$ for the Android Platform developers. This will be one of the required initial costs. Other IDEs that will be used for the development of the desktop version are free to use.
- A suitable domain name for "Autoshop" is needed for the application website

1.3.3 Language Constraints

- English is chosen as the language of the application because of the fact that it is the most widespread language globally. Also, photoshopping jargon is mainly developed in English because of commonly used tools like Adobe Photoshop.

1.3.4 Social Constraints

- With the application, faces of people can be pasted to irrelevant photos. This fake editing may be done without the approval of the owner of the face. Hence, a well-defined terms and conditions is necessary to protect the integrity of the original photos.
- The application, in a sense, will generate fake photos and this can cause manipulation of truth at a social level. Again, the terms and conditions should point on this issue.
- The mobile version of the application requires good use of touch-screen controls. Hence, elderly people, toddlers and preschoolers will be probably excluded from the user profile.

1.4 Professional and Ethical Issues

1.4.1 Ethical Issues

- We will adhere to the Code of Ethics and General Data Protection Regulation (GDPR) [11] [12].
- The generated fake photos by the application may result in false truths and possibly open up security issues. For instance, a criminal may attempt to remove himself from a digital evidence. Such scenarios can be potentially prevented using crypto-paradigms such as digital signatures.
- The application should not store photos of the users without their consent, and store an encrypted version of the photos if the necessary permission is given. This is basically a prerequisite of GDPR.

1.4.1 Professional Issues

- The source code of the project will be kept private.
- Agile-Scrum development strategy will be adopted [13]. This will help us define roles, meetings and tools in a well-structured manner.
- Our team will meet at least once a week. In the later phases of development, the meetings will transform into Agile Scrum meetings .
- Work division will be equal among the team members, and the team will carry out any decision making process democratically.

2 Requirements

2.1 Functional Requirements

2.1.1 System Functionality

The system should:

- ask the user's permission to access the local machine's resources (photo gallery, camera, etc.).

- display a gallery of user photos from which the user can make selections.
- provide an option to receive camera input.
- receive a background image from the user as an input, the background image refers to the image to be edited.
- receive additional images, which are the image(s) to add on the background image, as a user input.
- get the position of the additional images over the background image by drag and drop.
- get the size of the additional images by resizing.
- get the borders of the additional images by cropping and scanning; if the user does not crop or scan, then use smart-cropping methods to extract the object of interest.
- get the borders of the background image by cropping.
- get the scaling factor of the background image by rescaling.
- display the images to edit, in other words, the background image and the additional images on top of it, in a frame.
- communicate with a server, which runs a pre-trained neural network. In the desktop application, the server will be the localhost itself since it can run GPU-heavy tasks. However, in the mobile application, the server will be a remote machine that is capable of running the neural network efficiently.
- send the background images, the additional images; the positions, sizes, and borders of those images to the server.
- use the pre-trained neural network to adjust the photographic properties (opacity, texture, brightness, color, etc.) of the additional images according to the background image, and generate a photoshopped output.
- receive the photoshopped output from the server.
- display the photoshopped image in another frame. This will be done in real-time, as the system receives input, in the desktop application since the server is the local machine itself; but the mobile application will require some time to establish server communication and perform data transfers.
- provide an option that allows the user to save the photoshopped output to the local machine's photo gallery.
- provide an option that allows the user to share the photoshopped output using the various social media and messaging platforms (Instagram, Whatsapp, E-mail, etc.) in the local machine.

- ask the user's permission to use the photoshopped output for further training of the neural network.
- delete the photoshopped output and the inputs from the server-side, if the user refuses the use of his/her data.
- store the photoshopped output and the inputs in the server-side, if the user grants permission for the system to use his/her data.
- periodically download "photo - photoshopped photo" data from certain sources; and use this data, together with the stored user data to retrain the neural network. Period is to be determined.

2.1.2 User Functionality

The user should/can:

- allow or deny permission for the system to use the local machine's resources.
- select or take a background photo and upload it to the system.
- select or take additional images and upload them to the system.
- drag and drop the additional images over the background image to properly position them.
- rotate the additional images and the background image.
- resize the additional images.
- rescale (zoom in or out) the background image.
- crop or scan the additional images.
- crop the background image.
- see the original image and the photoshopped output. The user can also watch the process of neural network photoshopping the images in action in the desktop application.
- save the photoshopped output to his/her photo gallery.
- share the photoshopped output directly in his/her preferred social media or messaging platform.
- allow or deny permission for the system to use his/her data (original and photoshopped image) for system improvement purposes.

2.2 Non-functional Requirements

2.2.1 Extensibility

The system should:

- be easy to maintain, in other words, open to updates.
- be available on multiple platforms (desktop, mobile, etc.).

2.2.2 Reliability

The system should:

- ensure that the user data (the additional images and the background image) is not stored in the server-side unless the user gives permission.
- roll-back in case of a failure in the server communication. To be clearer, delete any data that is not completely processed.
- be resistant to adversarial cyber-attacks on the server-side. Methods such as hashing, signing and encrypting can be used to ensure data confidentiality and integrity.
- generate a realistic, hence reliable, output. For this, NVIDIA's libraries such as FastPhotoStyle and Image Inpainting will be integrated into the system [14].

2.2.3 Usability

The system should:

- be self-explanatory and user-friendly.
- be clear in terms of display and language when prompting.
- present a neat and well-organized user interface with themes that the user can select.
- require the absolute minimum photoshop knowledge from users.

2.2.4 Accessibility

The system should:

- be downloadable for free.
- be downloadable from the Autoshop official website for the Desktop version.
- be downloadable from the GooglePlay Store for the mobile version.

2.2.5 Portability

The system should:

- run in hardware and OS independent manner. Here, OS independent refers to the flexibility to run on any OS that has a Java Virtual Machine (JVM) and can run Python scripts.
- be accessible from multiple platforms (desktop and mobile).

2.2.6 Efficiency

The system should:

- not lag when communicating with the server. In case of a lag, there should be a time-out event, possibly 5 seconds (subject to change), that leads to server roll-back.
- not delay much in the mobile version when receiving touch-input from the user. A long delay is conventionally considered as 2-3 frames.
- not delay much in the desktop version when receiving keyboard and mouse input. A long delay is usually considered as 100 milliseconds.
- not wait longer than a second (subject to change) for the neural network to generate the photoshopped output.

2.3 Pseudo Requirements

2.3.1 Issue Tracking

- Git and Github will be used for version control.
- The project board of Github will be used for issue tracking. The Agile strategy will be adopted in the development process. Short intervals of

development; in other words, Sprints will be executed to ensure progressive improvement. Developers in the team will be equally assigned tasks and issues through Github's project board.

2.3.2 Testing

- Travis CI will be used as a continuous integration service used to build and test Autosshop directly from its Github repository [15].
- CodeClimate will be used as an automated code review service to ensure that the style conventions determined by the team are followed [16].
- A certain threshold of test-coverage will be enforced to merge any code with the Master branch using Travis CI. This will make sure that the Master branch is always a functional version of Autosshop.

2.3.3 External Tools and Technologies

- Android Studio will be used for the development of the mobile application [17].
- NVIDIA's technologies for GPU accelerated Python programming will be potentially used, depending on how GPU-dependent the performance of the neural network will be.
- Grafana[18] will be used as an observability platform to monitor and analyze database analytics.

3 References

- [1] D. W. Stout, J. Lyles, James, A. Sarkar, Barbora, Kyle, Betty, A. Brown, B. W. Robison, Panos, N. Sanden, and A. Torti, "Social Media Statistics: Top Social Networks by Popularity," *Dustin Stout*, 16-Jul-2019. [Online]. Available: <https://dustinstout.com/social-media-statistics/>. [Accessed: 13-Oct-2019].
- [2] "Remove unwanted objects & Fix imperfections with Inpaint Online!," *Inpaint*. [Online]. Available: <https://online.theinpaint.com/>. [Accessed: 13-Oct-2019].
- [3] "Gerçekliği yeniden keşfedin.," *Adobe Photoshop'u Satın Alın | En iyi fotoğraf, görüntü ve tasarım düzenleme yazılımı*. [Online]. Available: https://www.adobe.com/tr/products/photoshop.html?gclid=CjwKCAjwlovvBRBrEiwAG3XJ-7aBdUeNEE6tF24ijWMUmdH52nQO3P7waIyasAwxcUa8eD7rw3u60xoC00EQAvD_BwE&sdid=8DN85NTT&mv=search&skwid=AL!3085!3!340872550298!e!!g!!adobe photoshop&ef_id=CjwKCAjwlovvBRBrEiwAG3XJ-7aBdUeNEE6tF24ijWMUmdH52nQO3P7waIyasAwxcUa8eD7rw3u60xoC00EQAvD_BwE:G:s&s_kwid=AL!3085!3!340872550298!e!!g!!adobe photoshop. [Accessed: 13-Oct-2019].
- [4] Creative Cloud deneyimini keşfedin," *Creative Cloud fiyatlandırma ve üyelik planları | Adobe Creative Cloud*. [Online]. Available: <https://www.adobe.com/tr/creativecloud/plans.html>. [Accessed: 13-Oct-2019].
- [5] G. Liu, F. A. Reda, K. J. Shih, T.-C. Wang, A. Tao, and B. Catanzaro, "Image Inpainting for Irregular Holes Using Partial Convolutions," *Computer Vision – ECCV 2018 Lecture Notes in Computer Science*, pp. 89–105, 2018.
- [6] "Partial Convolution Layer for Padding and Image Inpainting," Github. [Online]. Available: <https://github.com/NVIDIA/partialconv>. [Accessed: 13-Oct-2019].
- [7] Li Y., Liu MY., Li X., Yang MH., Kautz J. (2018) A Closed-Form Solution to Photorealistic Image Stylization. In: Ferrari V., Hebert M., Sminchisescu C., Weiss Y. (eds) *Computer Vision – ECCV 2018*. ECCV 2018. Lecture Notes in Computer Science, vol 11207. Springer, Cham
- [8] "Add Objects to Paintings and Images seamlessly with this Amazing Python Script," *Analytics Vidhya*. [Online]. Available: <https://www.analyticsvidhya.com/blog/2018/04/add-objects-paintings-images-seamlessly-amazing-python-script/>. [Accessed: 13-Oct-2019].
- [9] - "NeuroNuggets: Cut-and-Paste in Deep Learning," *Medium*. [Online]. Available: <https://medium.com/neuromation-blog/neuronuggets-cut-and-paste-in-deep-learning-a296d3e7e876>. [Accessed: 13-Oct-2019].
- [10] "Build for developers," *GitHub*. [Online]. Available: <https://github.com/>. [Accessed: 13-Oct-2019].
- [11] "Code of Ethics," *Code of Ethics | National Society of Professional Engineers*. [Online]. Available: <https://www.nspe.org/resources/ethics/code-ethics>. [Accessed: 13-Oct-2019].

- [12] "General Data Protection Regulation," intersoft consulting. [Online] Available: <https://gdpr-info.eu/>. [Accessed: 13-Oct-2019].
- [13] Atlassian, "Scrum - what it is, how it works, and why it's awesome," *Atlassian*. [Online]. Available: <https://www.atlassian.com/agile/scrum>. [Accessed: 13-Oct-2019].
- [14] "FastPhotoStyle," Github. [Online]. Available: <https://github.com/NVIDIA/FastPhotoStyle>. [Accessed: 13-Oct-2019].
- [15] "Hey there!," *Travis CI - Test and Deploy Your Code with Confidence*. [Online]. Available: <https://travis-ci.org/>. [Accessed: 13-Oct-2019].
- [16] "Engineering Metrics to Improve Continuous Delivery Practices: Velocity," *Engineering Metrics to Improve Continuous Delivery Practices | Velocity*. [Online]. Available: <https://codeclimate.com/>. [Accessed: 13-Oct-2019].
- [17] "Download Android Studio and SDK tools : Android Developers," *Android Developers*. [Online]. Available: <https://developer.android.com/studio>. [Accessed: 13-Oct-2019].
- [18] "Grafana: The open observability platform," *Grafana Labs*. [Online]. Available: <https://grafana.com/>. [Accessed: 13-Oct-2019].