Draw Bot Virtual Application

A Project Report
Presented to
The Faculty of the Computer Engineering Department

San Jose State University
In Partial Fulfillment
Of the Requirements for the Degree
Bachelor of Science in Computer Engineering

By

Jeffrey Huynh
Daniel Man
Ian Ramos
Jonathan Rojas

01/2018

APPROVED FOR THE COLLEGE OF ENGINEERING

Dr. Michael Robinson, Project Advisor

Professor Rod Fatoohi, Instructor

Dr. Xiao Su, Computer Engineering Department Chair

ABSTRACT

Draw Bot Virtual Application

By

Jeffrey Huynh, Daniel Man, Ian Ramos, Jonathan Rojas

Machine learning is an increasingly popular trend in the technology industry attracting a multitude of companies and businesses. With the rise of technological advancements made in the 1990's, the new hardware allowed for computers to perform more computationally dense tasks; hardware was the main limiting factor. The basic concept of machine learning is to grant the computer the ability to learn without explicit hard-programming. This project will utilize machine-learning concepts in order for a computer to learn how to identify objects and generate a model of the drawn object.

While there is existing software that can analyze specific objects in images, the software can only identify the object without any other application. While this may be useful for filtering objects from certain images, it poses a challenge on what the optimal use for this software can be. There is a need to improve this software to complete more tasks so this field of technology can continue to thrive. Another problem that this project will encounter is deciding which machine-learning algorithm to use for this project.

By utilizing and enhancing the several machine learning object identification software, the goal is to create an application that identifies objects with drawn images created by the user. This application will have features that define objects and be able to create a virtual model of the object, enhancing the use of machine learning object detection software. Additionally, a new algorithm and research will be a required; this is to integrate machine-learning object detection

software with the ability create virtual sketches. The goal is to simulate a human's ability to learn what objects are and the ability to draw them from memory into a software application.

Table of Contents

Chapter 1 Project Overview

1.1 Project Goals and Objectives
1.2 Problem and Motivation
1.3 Project Application and Impact
1.4 Project Results and Deliverables
1.5 Project Report Structure

Chapter 2 Background and Related Work

2.1 Background and Technologies
2.2 Literature Survey
2.3 State-of-the-art Summary

Chapter 3 Project Requirements

3.1 Domain and Business Requirements
3.2 System Functional Requirements
3.3 Non-functional Requirements
3.4 Context and Interface Requirements
3.5 Technology and Resource Requirements

Chapter 4 System Design

4.1 Architecture Design
4.2 Interface and Component Design
4.3 Structure and Logic Design
4.4 Design Constraints, Problems, Trade-offs, and Solutions

Chapter 5 System Implementation

5.1 Implementation Overview
5.2 Implementation of Developed Solutions
5.3 Implementation Problems, Challenges, and Lessons Learned

Chapter 6 Tools and Resource

6.1 Tools Used
6.2 Standards

Chapter 7 Application Testing
7.1 Testing and Experiment Scope
7.2 Testing and Experiment Approach
7.3 Testing and Experiment Results and Analysis

References

List of Figures

Figure 1. State Diagram of Program..... Page No 11
Figure 2. A general diagram of the design of the program and interface..... Page No 14.
Figure 3. Logic Diagram of program function path ..... Page No 15.

List of Tables

Table 1. Functional Requirements..... Page No 12.
Table 2. Notable Hardware and Software Used..... Page No 13.

Chapter 1. Introduction

1.1 Project Goals and Objectives

One of the main goals is to teach a computer to identify and classify objects from sketches and recreate them through a virtual model of that object. One of the main objectives in order to complete these goals are to create a scripting program to take in sketches of a desired object from the internet. Then developed object heuristics are combined to enhance the functionality to identify multiple different objects in the same image. The program identifies an object in both the quickest manner and through the least number of possible sketches. The program will be able be to classify and create a unique virtual model of the target object every time; it will have undergone a new set of image analyses to verify that it is continuously learning by using a python EDI to code the application. Through this, the program will be able to properly identify objects from sketches and display a virtual model of the users drawing.

A crucial requirement is that the user would be able to create a digital sketch in a white canvas with black ink to create a sketch as it would be created with pen a paper. Another requirement includes that the machine learning algorithm will be able to identify and classify the sketch. The final functional requirement includes the ability to display the sketch in a virtual reality environment. Some nonfunctional requirements include the complexity and the detail of the sketch drawing application. Integrating these requirements together will create the final product so that the user can have their sketch be imported into a virtual environment.

1.2 Problem and Motivation

While there is existing software that can analyze certain objects in images, the software can only identify the object without any additional functionality. This is useful for filtering objects from certain images, but poses a challenge on what is the most optimal use for the application. There is also a need to improve this software to complete more tasks so this technology can continue to thrive. This project is important because it will allow advancements to artificial intelligence by giving more human like features by using this application. Utilizing machine and deep learning algorithms, it will develop its heuristics to the point where it can use the information that it has learned; this is then extrapolated and a virtual model of the target object is created to allow a computer to create its own virtual art.

1.3 Project Application and Impact

The main application for this project is to extend beyond conventional machine learning classification. Draw bot virtual application, while it has basic machine learning classification algorithms, the project will advance beyond these basic techniques and will have an impact in variety of ways. The machine learning application will access the internet and search through thousands of sketch images of desired objects by the user, one example being a dog. Thousands of images of dogs are fed into the program and the program will be told to look for and identify dogs in the images it is being given. Learning algorithms will then aid the program to identify what a dog is and develop image data on what a dog looks like. The desired result is that the application will require as few images and as little time as possible to create an acceptable result. From the data collected, the program will be able to identify dogs accurately from later images. It will now also use the data to create a unique image of the dog and display it to the user. If the program, with the data it has already accumulated, was to search once for images to learn from, it would be desired to be able to identify dogs more easily and accurately; it will also produce a more developed image of a dog from the updated data. This simple result of this project can be used to teach computers to understand more objects that are complex and advance the study of artificial intelligence within the academic field and eventually the industry.

1.4 Project Results and Deliverables

The result of this application should be able be to create a unique image of the target object every time it has undergone a new set of images analysis to verify it is continuously learning. Processing the numerous sets of images from a database that contains a designated target object used for the learning process. Utilizing image processing and pixel recognition to discover and create an association of a desired object. Utilize machine and deep learning algorithms to develop heuristics to learn what a desired object is and be able to use it to identify the object in different sets of images will be the core of the project. Utilize machine and deep learning algorithms, it would also have developed its heuristics to the point where it can use the information it has learned and extrapolate from it to create its own image of the target object. Some of the deliverables include a project report, which describes the project in detail. In addition, a prototype of the object detection and sketching should be completed which includes starting the coding process. Also a schedule will be created in order to keep track of the progress being made and analyze if adjustments need to be made.

1.5 Project Report Structure

In the following sections, there are descriptions of the implementation of the creation of the Draw Bot Virtual Application. The introduction will begin with the current use of machine learning and image recognition. Then there is a discussion of all the requirements which helped create this project. This will contain the domain and business requirements, functional and nonfunctional requirements, context and interface requirements, and technology and resource requirements. After the requirements are established the complete design will be described in detailed leading to the actual implementation of the project. After that, implementation testing is recorded which concludes this project.

Chapter 2. Background and Related Work

2.1 Background and Technologies

Image recognition and object recognition is a well-studied area in the machine learning world. Vatsavai et. al [7] designed a wonderful tool to analyze and classify images. With the use of machine learning and neural networks, projects like Vatasavai's are able to decently classify singular objects. However, these tools and techniques are not viable for practical use since classification of singular object are not as useful as multi-object classification.
Newer hardware technologies allows GPU's to use convolutional neural networks allowing for better machine learning techniques. With the introduction of the GPU, the processor can handle more load and process the data faster than using a traditional CPU. This allows further and deeper neural networks for the computer to analyze the characteristics of objects. Caffe is developed at UCB to help achieve that goal with the integration of GPU usage [4].
The goal is to further expand on the developments of multi-object classification. First, a simple single object recognition program is assembled. By having the program identify an object from an image with multiple objects, the hope to train the program so that it can classify each object from the image. From there, different classes can be incorporated and allow the program to detect multiple objects by referring to the single object detection. This would allow multiple classification of objects from a single image.

Finally with the ability to classify and understand multiple objects, the program will understand the correlation between object to create a database or "memories" of objects. This will allow the program to understand how objects are related to another and be able to recreate an image given a word or phrase to the program. This would be a step in machine learning to allow machines to become more human-like with the way human thinks and draws based on references.

2.2 Literature Survey

Machine learning and object classification is a well studied area, but with few commercial applications. Being able to detect objects in images can be very helpful to teach computers how to identify objects. The goal is to increase the usefulness and helpfulness of machine learning and object detection so that it can learn from sketches and create virtual models based of them. Allowing the machine learning to create its own art. This art will provided a basic foundation to teach computers to create on its own knowledge, in a virtual environment.

2.3 State-of-the-art Summary

The interest and the research on Machine Learning have drastically increased since the introduction of open source tools such as Caffe and Tensorflow [4]. These tools allow a deeper understanding of how machine learning works and enables developers to design future projects. With the help of Caffe, facial and voice recognition has become more advanced and accurate in receiving and analyzing the data. The usage of machine learning has rapidly increase this last decade and the trend will continue on for another decade or so.

Machine learning was accelerated due the introduction of newer hardware technologies. With the introduction of the GPU, normal computation done by the GPU is 3x faster than the computation done on the CPU. This allows more time to create and test data without the long wait time for the data to be processed. Machine learning will continue to grow since the newer technologies such as cloud computing can allow dedicated GPU servers to do the computations when doing things on the cloud. These are leading edge technologies in development by companies like Google and Microsoft at the present time.

While there is already image recognition using these tools, the scope of the projects are too small. While analyzing the data, the convolution neural network is only useful for single target image recognition [3]. The hope to achieve multi-targeted image recognition such that the machine will behave more human-like. With the uses of multi-targeting objects in images, the main hope to achieve a greater classification so that the machine can process complex classifications. Human often process multiple things at once and with these tools the hope is to bring artificial intelligence one step closer to a more human like machine. To achieve such a feat, this project will require extensive studying and understanding how the human mind works; this will be applied by translating that into the computer.

Chapter 3. Project Requirements

3.1 Domain and Business Requirements

The domain requirements for this project are as follows. The first thing required would be

knowledge in the subject of artificial intelligence. This is mostly focused on the topic of imagery recognition and classification. The next requirement in the domain is software and coding setup, which is needed to create and build the core of the program. An additional domain is also the web interface and setup. This is required to allow for a user interface that is separate from the main hardware location for users to access the program to use.

Business requirements for this project would be focused on its resulting performance. It has a website for users to access and run the program. It will be able to take in user drawn sketches on a web interface or scrape the Internet and will be able to send it for processing. Finally, it would have the requirement to be able identify and classify what object was sketched by the user and return data to the user about it. This is shown in figure 1.

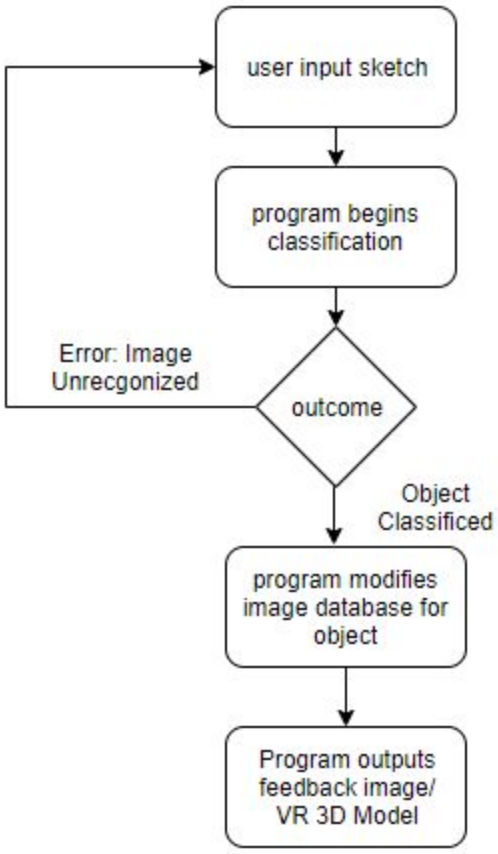

Figure 1: State Diagram of Program

3.2 System (or Component) Functional Requirements

Functional requirements for the system function would focus on what it should be capable of doing to perform its main tasks. The program shall use advance graphics hardware to allow the quick and proper processing of data. This would involve the processing of images in order to recognize main objects in it and classify that object. It shall utilize modern software

programs to allow the program to use machine learning such as classification to run the project. To allow user interface through the internet, the project would need the functional requirement of being continuous connected to internet access for use and be able to communicate with the user. It would also be able to send Virtual Reality (VR) data to the user if they so wish to view image feedback through VR.

Table 1: Functional Requirements

Take in sketches through web interface	Process sketches
Recognize and classify objects	Continuous internet connection
Send feedback to user	Send VR data to user

3.3 Non-functional Requirements

Non-functional requirements for the project would be the how it performs its operation. The first requirement it would have is to recognize and classify sketches inputted to within a certain degree of tolerance, such that images that clearly differ in style but are of the same object can be still classified. Another requirement would be for it to run and perform its task within a reasonable period of time as to allow feedback and timely interaction with the users. This can range to no more than two or three minutes to keep user engagement. Additional requirements would also include that the application is reasonably accurate in its classification, which can be judged upon how well a human can recognize it; it would be be able able to generate virtual reality imagery in a timely manner in a similar time frame listed above, and that it would work in function properly when interfaced with users for numerous transactions.

3.4 Context and Interface Requirements

In building this project, the context environment for this is the access and usage of computer components of the Engineering Virtual Reality Lab. In interfacing the hardware and software components of this lab, a special computer build was used which would allow advanced software design for machine learning use to work on powerful hardware to bring about fast results, accurate executions, and ease in creating and implementing. User interface would make use of a webpage that connects to the program on the computer online, allowing users to access the program from anywhere. There, sketches can be made and sent online to the program for processing and feedback can be return to them. The VR interface, given the user has the necessary hardware, would give access to program feedback in a VR setting.

3.5 Technology and Resource Requirements

Numerous technologies were brought and used in order to allow for the full setup and implementation of the program. On the hardware end, this involves the usage of a computer that is use to house and run the program's main code. Along with standard computer components, this includes the use of the Nvidia GeForce GTX 1080 graphics card which gives the program the processing power to examine sketches, classifies objects and simulate VR data. Along with that is a upper model Intel i7 CPU to allow processing of the main program, and a Samsung solid state drive to allow quick access of data for use. On the software end is the use of Linux Mint for a better range of capabilities and use. In it, the program Caffe2 is used. This is a powerful machine learning framework, based of the original Caffe, that is used to support the computation of image recognition and classification. Users with can use VR equipment on their end to see feedback in VR.

Table 2: Notable Hardware and Software Used

Nvidia GeForce GTX 1080 GPU	Caffe2
Intel i7 CPU	Samsung 1 TB SSD
Linux Mint	VR Equipment

Chapter 4. System Design

4.1 Architecture Design

The architecture design of the program are as follows. The system itself would consist of a program made for the project that would run on the foundation set forth by Caffe, which allows it to fully utilize machine learning. This program would run on a linux system which supports the running of caffe. Stored on a computer with the hardware listed in Table 2, it would have internet connectivity and be accessible for users with a website interface, data sent from that to it would make use of the main CPU and GPU to process the program and the image data for classification. Having an SDD, besides storing the program, also makes data interaction faster for the computer.

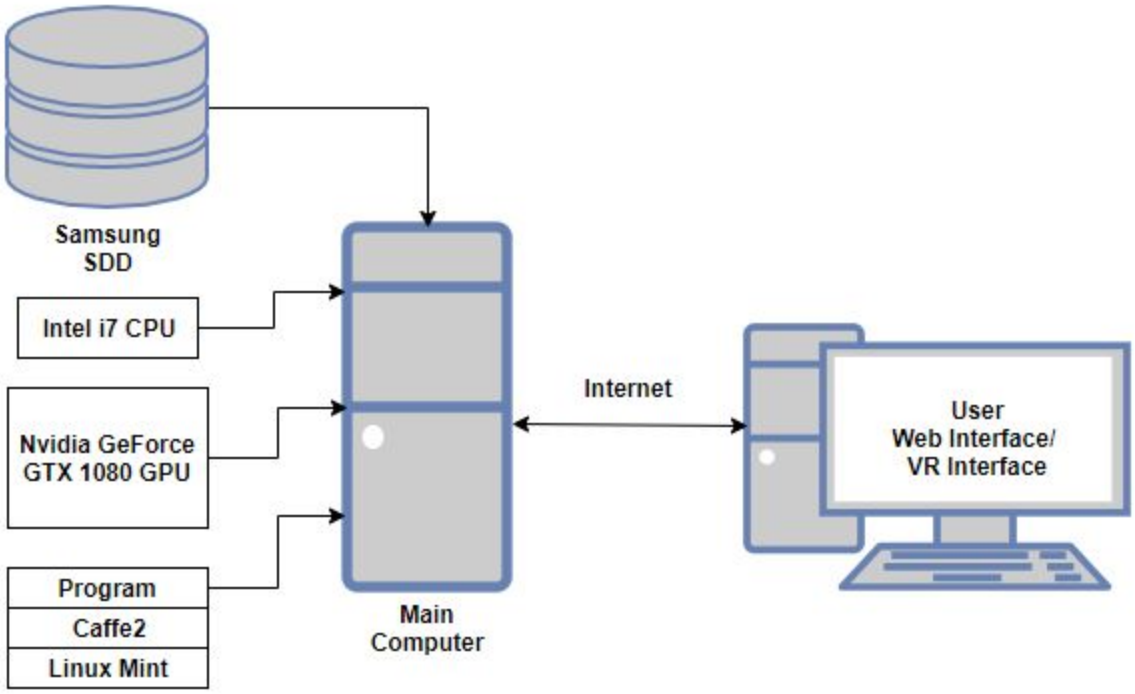

Figure 2: A general diagram of the design of the program and interface

4.2 Interface and Component Design

As shown in Figure 2 above, the main component design can be seen and its connection to the user interface. In startup, a user will go to the web interface and sketch a image of an object, using either digital pen or mouse. This data is sent to the main computer running the program. Here, the program is running with Caffe on the OS mint. The CPU and GPU run to process the program and data to identify, recognize, and classify the object, then send back feedback data of a modified image of the object based on user input. This feedback image can be seen on the web interface for the user, or can be displayed with VR equipment if they so choose.

4.3 Structure and Logic Design

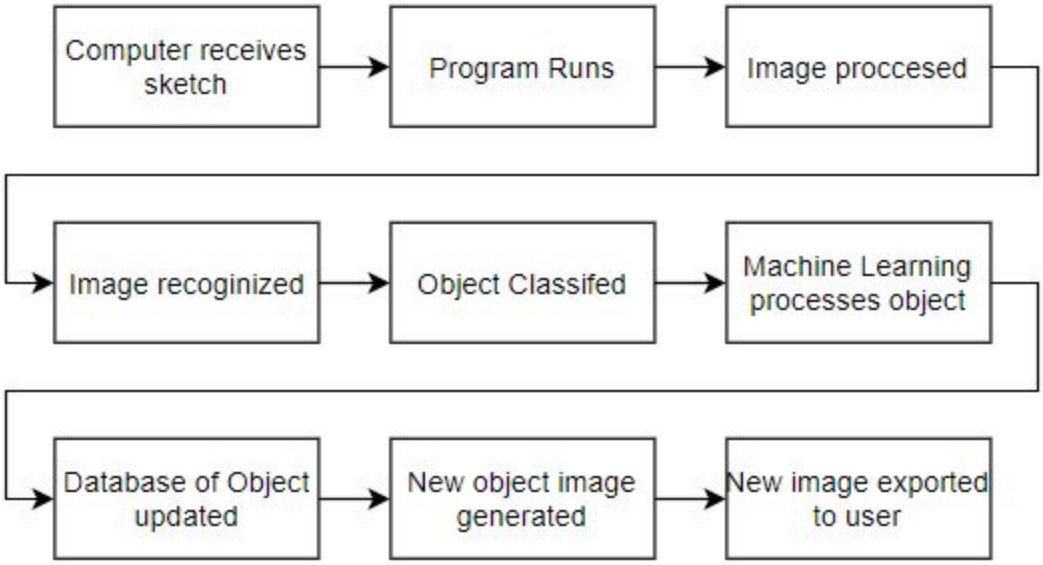

Figure 3: Logic Diagram of program function path

4.4 Design Constraints, Problems, Trade-offs, and Solutions

4.4.1 Design Constraints and Challenges

Design constraints and challenges in the scope of the project focus mainly on the availability of parts and information to construct the program. The main function of the project is to allow users to interact with a machine that learns from use drawn input and view feedback from it. This project is primarily for people to understand machine learning and has minor economic and business benefits as it is mainly for educational and research purposes. In addition, this makes finishing the physical components harder to get due to lack of funding. On a separate issue, this field of study in machine learning is an emerging, but still young and new topic. Research on it remains broad as it still begins to take shape and form.

4.4.2 Design Solutions and Trade-offs

To ensure this project can be fully implemented, key solutions were found to allow it to proceed. To build the hardware that would be needed to properly run the program, equipment from the Virtual Reality Lab was used with permission. Their supply allowed the building of the computer used to run the program and use the machine learning to full use. With the use of Caffe, the program can be written with the architecture support to run the application. To support the endeavour of constructing this and implement machine learning and VR, assistance was received from local experts and what research papers was available to allow better understanding and implementation of the program.

Chapter 5. System Implementation

5.1 Implementation Overview

The implementation scope for the project is the ability to use a user drawn sketch to have the application classify the sketch. With the classification, the application can transform the sketch into a model to be used in virtual reality. The platform used for the project is iPython and the language used for the project is python. There are many machine learning libraries being utilized from Berkeley's Caffe application. Some dependant hardware or software would be virtual reality headset and a dependant software to allow the user to create the drawn sketch. Most of the implementation dependencies are open source machine learning dependencies.

5.2 Implementation of Developed Solutions

The implementation of the project is separated into three part. The first part of the implementation is the software used to create the drawn sketch for the other parts of the project. This software is a simple implementation in javascript and can be used in HTML5 to create the application.

The second part of the implementation is the analysis of the sketch and the use of machine learning to classify the object. This step is where the bulk of the work is done by the application. The sketch is first loading into the machine learning portion via the iPython script. The sketch is decomposed into individual pixel in which the image is clean of unwanted artifacts which can affect the classification process. The image is then cross reference with an initial sketch that was used to train the machine learning algorithm. The original image is classified and improved by the machine learning process and a new image is created by the algorithm.

The third part of the implementation is to take the newly created image from the machine learning algorithm to generate a model to be used in virtual reality. The model generated would be placed into a virtual environment in which the user can visualize the change that occured between the original image and the generated image by the machine learning algorithm.

5.3 Implementation Problems, Challenges, and Lessons Learned

One of the main implementation problems is how the image was going to implementation into virtual reality. The issue pushed back progress of the design by a couple weeks because the implementation of virtual reality was an added requirement at the beginning of the 2nd semester. Another challenge face was the finalizing the product with the advisor to create a meaningful project. Discussion was made bi-weekly to create and finalize the final product of the project. This posed a huge setback to the project because nothing was agreed on prior to the finalization and many weeks were wasted.

Chapter 6. Tools and Standards

6.1 Tools Used

The main piece of hardware used to implement this project is a high end graphics card in a computer provided for use by the Engineering VR Lab. This computer contains the advanced and necessary hardware that would be capable of creating data; this data allows for the classification of objects by the Draw Bot. This hardware includes a Samsung 1TB Solid State Hard Drive for fast memory extraction, an Intel i7 Quad Processor for quick data calculation, and a Nvidia 1080 graphics card. This particular graphics card is of a very high end and can perform the image data calculation needed to create the data use by the program. Its power also makes its job accomplishable in a shorter period for better turn over.

The program software usage is decided by the application's use to create critical machine learning data, and the code used to build up the program. To create training data that would be used for classification, Caffe2 was used to run training images to create the data. This is done on the hardware stated above. For the main application that would use the data later, the main user interface to into sketch images is constructed on Python code. Its web interface used a mix of this and HTML. The process of using the trained data for classification would use use python as well. All hardware use is done on any sufficient computer.

6.2 Standards

In creating this project, boundaries were set, but also certain freedoms were accepted as needed. This application was to be known as 'DrawBot' for all future references. Its main input would be user made sketches on an application we provide. The user interface would be done on whatever manner was capable enough to

accommodate the needs and were left free to decide later until a proper decision was made with python. Documentations and notes were gathered on each members' task and filed in a shared drive for later use and use by other teammates as needed. This application would need special hardware and software to gather vital training data, but would run with this data on regular computers. All other task were left to the freedom of the members to pursue as they see fit with input from others and, as near finishing, final vote of the team if deemed sufficient to be used.

Chapter 7 Application Testing

7.1 Testing and Experiment Scope

The testing and experiment scope of the application depends on the accuracy of the trained model and the expected image relating to the original sketch. When testing the individual parts of the application, the goal is to have a working part before testing the full application. The objective of the testing is to be able to receive an image from the user, process the sketch and compute accuracy and precision of the sketch to the targeted image. The result should be correctly classified. Testing for other components of the program will be used as necessary.

[No suitable figures can be provided at this stage]

7.2 Testing and Experiment Approach

Since the application was built in three major parts, the testing approach to the application is to guarantee that each part works so that the whole application can work as a whole. The sketch part of the application is tested to be able to draw and output the image as a file format. The machine learning portion of the application is tested to be able to classify the sketch as a particular object and display the accuracy as well as the precision of that sketch to the particular object. Both parts of the application is then applied to an online web page which will symbolize the handshaking between all the parts of the application.

[No suitable figures can be provided at this stage]

7.3 Testing and Experiment Results and Analysis

The results of application depends of the user sketch of the object. If the user mimics an object with great details then the application score for accuracy and precision will be much higher with less chances of errors. The score is dependant on

the object used for training so the results of low scores would be an object not supported by the application or just a bad sketch.

[No suitable figures can be provided at this stage]

References

[1] Barbu, A., She, Y., Ding, L., & Gramajo, G. (2017). Feature selection with annealing for computer vision and big data learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 39*(2), 272-286. doi:10.1109/TPAMI.2016.2544315

[2] Dong, C., Loy, C. C., He, K., & Tang, X. (2015). Image super-resolution using deep convolutional networks. *Corr, abs/1501.00092* Retrieved from http://arxiv.org/abs/1501.00092

[3] Goodfellow, I. J., Bulatov, Y., Ibarz, J., Arnoud, S., & Shet, V. D. (2013). Multi-digit number recognition from street view imagery using deep convolutional neural networks. *Corr, abs/1312.6082* Retrieved from http://arxiv.org/abs/1312.6082

[4] Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R., . . . Darrell, T. (2014). Caffe: Convolutional architecture for fast feature embedding. Paper presented at the *Proceedings of the 22Nd ACM International Conference on Multimedia,* Orlando, Florida, USA. 675-678. doi:10.1145/2647868.2654889

[5] Sarvadevabhatla, R. K., Surya, S., Kruthiventi, S. S. S., & R., V. B. (2016). SwiDeN: Convolutional neural networks for depiction invariant object recognition. Paper presented at the *Proceedings of the 2016 ACM on Multimedia Conference,* Amsterdam, The Netherlands. 187-191. doi:10.1145/2964284.2967208

[6] Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *Corr, abs/1409.1556* Retrieved from http://arxiv.org/abs/1409.1556

[7] Vatsavai, R. R. (2013). Object based image classification: State of the art and computational challenges. Paper presented at the *Proceedings of the 2Nd ACM SIGSPATIAL International Workshop on Analytics for Big Geospatial Data,* Orlando, Florida. 73-80. doi:10.1145/2534921.2534927

[8]K. Quillin and S. Thomas, "Drawing-to-Learn: A Framework for Using Drawings to Promote Model-Based Reasoning in Biology", 2017. .

[9]L. Barraza, "Children's Drawings About the Environment", 1999. [Online]. Available: https://www.researchgate.net/publication/248965326_Children's_Drawings_About_the_Environment. [Accessed: 18- Sep- 2017].