CPRE 575 Project Proposal – Virtual Backdrop

By Edward Jezisek, Nichole Dugan and Brian Schulte

# Sales Pitch:

Virtual Backdrop eliminates the need to spend hours at the Driver License Station waiting for other people to take photos. It does this by creating professional grade driver's license photos using a picture from a cell phone. Costs will be reduced, the speed will be increased and your photo will look better. Virtual Backdrop will let you take advantage of this by taking your photo, removing the background and allowing you to choose your favorite photo before you even step into the Drivers License Station. We also are aiming to ensure that hair is included inside the picture as many currently used approaches have had issues with people's hair.

Virtual Backdrop will specifically be used to separate the background from pictures/videos. This will allow it to be used in other applications besides giving a backdrop for Driver's license photos. However, the primary purpose of this application is to allow users to use their own photos as their driver's license photo.

# Target Audience:

Self-service kiosks are becoming more and more popular in states across the US. This project will focus only on kiosks being used by the state of Iowa. Currently, the following services are offered at kiosk locations: 

* Renew/Replace Iowa Driver’s License or identification card
* Update Iowa Driver’s License (removing “under 18” or “under 21”, updating address)

As a result of these select options, kiosk users must meet the following requirements:

* Possess a valid Iowa Driver’s License or ID card (not a Commercial Driver’s License)
* Be between the ages of 18 and 70
* Reside in Iowa with US citizenship
* Not be required to provide a medical or vision report during the renewal processece

Although anyone between 18 and 70 can use the kiosks, the system is designed to target drivers between the ages of 18 and 30 who routinely download applications and tend to choose self-service over in person interactions. The typical kiosk user also likes to avoid time wasted at the Driver’s License Station. We are specifically targeting users with Android based smart phones and anyone who currently uses a kiosk to replace their driver's license photo. At the current time, kiosks need to have a standing our mounted backdrop behind the photos and Virtual Backdrop could help eliminate this.

# Need for Application:

Virtual Backdrop is needed to enhance the experience of getting a Drivers license for residents of the State of Iowa. This application can further be used to create a database for facial recognition. The pictures we create will meet the standards for a driver's license and in doing so, will meet the standards for facial recognition.

Specifically, with the introduction of new technologies that require a photo with a high enough quality to be included in biometric databases used for facial recognition, photos will need to remove artifacts from their backgrounds to create consistent images of faces. Previously, colored backdrops were used to ensure noise was reduced from the photos. But this places significant limitations on the photos that can be used in the databases and the process to take said photos (i.e. cellphone cameras cannot be used). Ideally, software could be created to identify faces, remove backgrounds and produce a uniform virtual backdrop for the photo regardless of the composition of the original photo. One specific hurdle to overcome is that current algorithms can sometimes get confused with a person's hair and fill in the hair with the backdrop color instead of leaving the alone.

# Previous Approaches

# Previous Experience

Our team is comprised of three people: Edward Jezisek, Nichole Dugan and Brian Schulte. Edward Jezisek has experience with Mobile experience and a strong interest in developing consumer based applications. He is currently a programmer for Verizon Wireless, is incredibly interested in Algorithms, Data Structures, overall systems architecture and learning more. Some of his Android applications have been featured on Fox news and XDA-Developers.

Nichole Dugan is a C# developer working for the state of Iowa in the Department of Transportation. She has been working with driver’s license issuance since 2007 and has helped with the facial recognition program in the state of Iowa. She has also worked with vendors for the state of Iowa implementing self-service kiosks for the state, and recently has worked with a vendor to interface with the state of Iowa’s system of record for the mobile driver’s license project.

# Approach

Our proposed approach to this project is two-pronged. First, we want to research current algorithms being used as well as if said current algorithms could be modified to accomplish the goal. Secondly, we want to design a high-level process that is streamlined, scale-able and secure for users to import pictures taken with many types of cameras in many environments, manipulate them easily (like clicking a button rather than manually altering) and save a picture that meets the necessary specifications into the database.

To accomplish this, we have divided our project into 4 phases.

**Phase I**

This phase is how users will upload images to be used for their license. Users will be able to use a camera to capture video or upload a previously saved photo. Once the picture is loaded into the application we will be able to manipulate it to conform to the Iowa DOT standards.

**Phase II**

Phase II will involve the algorithms mentioned above. The background of the photo from Phase I will be removed and replaced with a blue background.

**Phase** **III**

Integrate virtual backdrop into an Android application

**Phase IV**

Add ICAO requirement checks

# Overall Requirements

The AAMVA (American Association of Motor Vehicle Administrators) is the governing body regulating driver’s license documents. According to AAMVA standards, the background for driver’s license photos is stated as:

Background. A uniform light blue color or white background shall be used to provide a contrast to the face and hair. Note: Preference is for uniform light blue color, such as Pantone 277 (though the specific Pantone color is not a requirement – a uniform light blue color or white background is a requirement).[[1]](#footnote-2)

An example photo from a driver’s license is shown in the below image.



A typical driver’s license station with the photo backdrop is shown in the below image.



When kiosks were introduced as a solution for citizens of the state of Iowa to be able to renew or replace their driver’s licenses without visiting a driver’s license station, certain limitations were required because of the backdrop functionality. The kiosk needs to have a standing or mounted backdrop behind the photo area to allow for the photo to have the required blue backdrop. A video showing the kiosks is located at this URL: <http://www.kcci.com/news/new-kiosks-let-you-renew-your-iowa-drivers-license/33374300>

In addition to the kiosks, the state of Iowa has recently introduced a pilot program to allow for mobile driver’s licenses. One of the suggested features has been allowing the user to take a photo from their device and use that as their driver’s license photo.

# Hardware Requirements

For this project a camera will be needed. This camera will need to take pictures of approximately 72dpi. These pictures will be approximately 4MB in size. For our project to be successful; we need to be able to process all of the drivers in Iowa in a one month time period. As there are 2.3 million drivers we need to be able to process 8 Terabytes worth of pictures in a one month time period. In addition, if this is to scale to multiple states; we need to make sure that our solution is distributive and able to scale/be run on multiple machines. Our initial product will run on the Ubuntu O/S, but future releases may need to use Redhat to ensure scalability.

Our application will be created using C++ and the gcc (Ubuntu 4.8.4-2ubuntu1~14.04) 4.8.4 compiler , the OpenCV 3.1.0 library and Boost. This will allow us to use the library based functions inside of OpenCV and develop intermediary tests for our application using the Boost::Test framework. Finally, we will provide several videos/pictures that can be used in our application to ensure that the provided photos meet necessary standards. Some of the processing will need to be done on the device as to prevent the server from receiving and processing too much of the required load.

This will require us to include parts of our code in Android. The specific version of Android that we will use is: 5.0.1. This will specifically run on an HTC One. It may run on other phones as well, but due to the scope of the project we aim to ensure that it at least works on an HTC One. Other applications may be developed to ensure access by as many users as possible. An Android phone was chosen for the convenience of having a device we already own that takes pictures/video and has a connection to the Internet. We felt that many of our customers would have a similar device and that they would be able to use the product.

# Evaluation

1. To successfully complete our project, testing will need to be performed. This testing will initially consist of a smoke test which will test the primary features of the application. This will be followed by a regression test and finally a load test.

# How do you plan to Test it?

1. The smoke test will consist of an image being processed and displaying just the face. To do this automatically we will obtain an image that has been processed and its input. We will compare the output to the result and have a threshold for differences. If there are too many differences, the smoke test will have failed. This will run for several small subsets of images. Furthermore, we will have tests that should fail. These tests will include pictures without anyone included. We will perform these tests with random objects entering the screen, and more than one individual in the picture. These tests will give a fast turnaround regarding whether or not a feature was successfully included. If the pictures are different enough a comparison image will be displayed to the developer. This will allow the developer to either accept or reject the new image.
2. Following the smoke test a regression test will be performed. This regression test will consist of the user manually checking many of the photos included to prevent inaccuracies. This test will highlight any key differences in pixels between the previous and current picture. This will allow the developer to easily notice any changes in the application; whether or not they are positive. The regression test will also take in at least one new video that the user performs and add this video to the list of tests to be performed in the full regression. This video must be taken with webcam to be determined as a success. The reason for this is to ensure that our application is able to extract an image from a new video as well as the old tests.
3. Finally a load test will be performed. This load test is needed to ensure that our server can handle the load of the inserted pictures. For this project we aim to be able to upload 100 pictures per hour for the server. If this is inserted in a kiosk, the kiosk may need a small computer or system to perform the processing. Hopefully our system runs faster; but approximately one photo processed per minute should be acceptable for our application.

# What is success?

1. If all of the previously defined tests run successfully our application will be a success. The previous test cases will prove that our application is successful with video/pictures from a webcam/mobile phone and that our application is able to withstand a specific amount of load. The tests will prove accuracy as well as speed and will automatically deduce how accurate the application is. Finally, complete success would include being included in the DMVs kiosk based product. Our attempt should be able to fix many of the issues in current approaches.
2. Finally, if our solution's pictures meet the requirement for a driver's license in 90% of the pictures taken, our application will be a success. This will require continuous analysis after the completion of the project to ensure quality; but it will hopefully remove many of the issues apparent in the current approach. These pictures must be accessible to facial recognition and ensure that a user is the same as their previous photo.

# How many test conditions?

1. Our application will consist of a minimum of three videos/photos in the regression test. These videos/photos will be of each team member. The reason for this is to ensure the quality of our solution. We will likely add more videos/photos as test evidence, but at this point three photos/videos will be required for the successful completion of our application.

# How many test subjects?

1. As our application will consist of a minimum of three videos/photos in the regression test; we will require a separate subject for each of these pictures/videos. This will ensure there are at least three subjects used as a test subject. This will ensure our application works for more than one person and will be convenient as we are a group of three people. More people will be added and more videos will be taken with respect to the necessary regression test.

# How will your results be evaluated?

1. Our results will be evaluated by comparing what we have created with the definition of success. If our application falls short of creating acceptable driver's license photos, the problem that caused us to fall short will need to be investigated. Furthermore, if it's necessary we will also perform facial recognition on past photos to ensure that users are unable to replace their photo with a picture of someone else. Essentially our results will be evaluated based on how close we get to creating an acceptable driver's license photo with a background. This background will be of varying colors. Some backgrounds will be unacceptable due to them being close in color to hair or skin. If the display does not change, it may be impossible to recognize a face correctly.
2. Furthermore, the initial results of our application will be compared manually. However, in the future we will automatically compare previous successful results with the most recent result. This will allow us to potentially speed up development and create an acceptable solution to this problem.

# How will the application be improved based on results?

1. The success rate of our application will be improved. Furthermore, we will ensure that our results are comparable with the previous results or manually input them as the new base image. This will allow for fast development and it will ensure the quality of our application. After our application is successful with a large majority of faces, the application will no longer need to be “improved” as it will be good enough.
2. Furthermore based on the initial results; how quickly we can attain our goal we plan to add specific features. Specifically running on an Android device and supporting multiple device types and Android OS. Also, as we want to improve the perception a user has over their picture; we will potentially allow them to do some image editing on the final picture. This editing naturally must be compliant with the aforementioned requirements.
3. We will also ensure compliance to the ICAO standards. This will ensure that our product is usable and acceptable to the state of Iowa. Furthermore, if performance becomes an issue we will measure various aspects of our application and ensure the quality of our project. This will provide for a solution that meets federal requirements and will ensure that our project is a success. These standards include the size of a picture needed for a driver's license and the background image that needs to be included. We hope to print at least one imaginary driver's license to prove the concept we have created. We plan to display video of us taking the picture inputting it into a driver's license format and displaying the resulting image.

1. AAMVA 2013 CDS (Card Design Standards) http://www.aamva.org/WorkArea/DownloadAsset.aspx?id=4435 [↑](#footnote-ref-2)