Artificial Intelligence Driving
Assistant

MAK Ting Hin, Alex 3035123253
Group FYP17021

supervised by
Dr. Chim T W

April 15, 2018
Abstract

According to the statistic provided by the government, there are more than 40 road accidents in every single day in 2016[1]. Among all reasons that caused the accident, inattentively driving is one of the major causes. For maintaining the future driving safety, our team believe that artificial intelligence could be the next solution on helping people to get away from potential dangers.

Despite of there are already a lot of different systems being invented to prevent accidents from happening, there is a lack of solutions to reconstruct the scene after it has happened. In some accidents, the involved vehicles might either not have a camera installed or the video is not retrievable for several reasons. The police would ask for people to provide car camera records if they were there or passing by. Our team believe that in the future with this project released, a database of camera and trip record will be available. Eliminating the need of spending a huge amount of human resources on reconstructing the accident.

The following will illustrate this system in details by stating its functions and followed with the information on its principle behind. The status of the project will also be discussed, starting from Phase 1 (1st Oct 2017) to the end of Phase 3 (15th Apr 2018). A future development plan and remaining works will be set out afterwards.
Acknowledgement

I would like to express my great gratitude to our supervisor Dr. T. W. Chim for his guidance and support. Dr. Chim assists our group in enriching the project’s content as well as technical advice when we encountered problems in the implementation stage. His insights and advice had cleared the path for us. I would also like to give a special thanks to Dr. C. K. Chui that he raised a lot of critical questions that have prevented a lot of problems and issues might occur in the later stage of development.
Contents

Abstract 1

Acknowledgement 2

Contents 3

List of Figures 5

1 Introduction 6
   1.1 Background 6
   1.2 Problems 6
      1.2.1 Careless drivers in car accidents 6
      1.2.2 Reconstructing the scene of car accidents 6
   1.3 Solution 7
   1.4 Scope and deliverable 7
   1.5 Outline 8

2 Previous Works 9
   2.1 Nexar 9
      2.1.1 Introduction of Nexar 9
      2.1.2 Comparison between Nexar and Aucar 10
   2.2 DailyRoads Voyager 14
      2.2.1 Introduction of DailyRoads Voyager 14
      2.2.2 Comparison between DailyRoads Voyager and Aucar 14
   2.3 Differentiation and contribution of Aucar 16

3 Methodology 19
   3.1 System architecture 19
   3.2 Mobile application 20
      3.2.1 Supported platform 20
      3.2.2 Reason of creating mobile application 20
List of Figures

1. Nexar ADAS warning ............................................... 9
2. Nexar video recording ................................................ 10
3. Nexar object recognition .............................................. 11
4. Aucar object recognition .............................................. 12
5. Nexar collision reports ................................................. 13
6. Dailyroads Voyager screenshot ......................................... 14
7. Aucar recording configurations ........................................ 15
8. Dailyroads Voyager loop recording ..................................... 15
9. Dailyroads Voyager video upload ....................................... 16
10. Aucar system architecture ............................................. 19
11. Aucar video player interface .......................................... 22
12. Aucar number plate recognition process ............................ 23
13. Aucar number plate example image in 480p .......................... 24
14. Aucar number plate example image in 480p and grayscale .......... 25
15. Aucar number plate example image in 480p and grayscale with detected corners .................................................. 26
16. Aucar number plate example image in 480p and grayscale with recognized plate .................................................. 27
17. Aucar number plate example cropped number plate .................. 27
18. Aucar number plate example cropped number plate with histogram equalization .................................................. 27
19. Aucar.io sign in form ................................................ 29
20. Aucar.io OAuth 2.0 token request sequence diagram .................. 31
1 Introduction

1.1 Background

Today’s world is technologically advanced, increasingly advanced safety systems are being invented and installed into vehicles. However, there are still car accidents happening every single day in the world, and some of them are unsolvable because the lack of evidence. Technologies like machine learning, neural network and computer visions are maturer than they used to be, such that it makes artificial intelligence a more realistic but sophisticated answer to reduce the number of car accidents and reconstruction of the scene.

1.2 Problems

1.2.1 Careless drivers in car accidents

According to a summary of key statistics released by the transport department of Hong Kong [2], an average of 44 car accidents happened in every single day of 2016. The first two main causes of these accidents, according to a figure released from the same department [3], were inattentive driving and driving too close with other cars. It is observed that people are unaware of the current dangerous situation or were not being able to respond to events in time.

1.2.2 Reconstructing the scene of car accidents

Reconstructing the scene after an accidents has happened is important to identify the responsibility of involved parties. Video and trip records are believed to be the strongest and useful evidence in reconstruction. In some cases that do not have video recordings or a driver has evaded the scene of accident, sometimes it is hard to reconstruct what has happened. It then relies on the statement of drivers or witnesses, if any, and the environment evidences like skid marks left by vehicles. The police might need to ask of drivers and witness to provide evidence or statement voluntarily, which is a timely passive method that requires people’s voluntarily cooperation.
1.3 Solution

A system called AUCAR is the proposed solution to problems mentioned in 1.2.1 and 1.2.2. Aucar consists of two major components: An Android application and a website.

Regarding problem 1.2.1, it is believed that with a proper alert or notification of the potential dangers ahead to the driver, it can actively draws drivers’ caution and to react. In such way that it would prevent an accident from happening. In order to achieve this, AUCAR needs to be able to identify and predicts the potential dangers. For instance, it is going to let the driver knows that if the driver keeps the current speed and direction, the vehicle will collide with an object ahead. In case the driver was not aware of that, it would alert the driver such that the driver can take actions earlier.

Regrading problem 1.2.2, it involves multiple functions of Aucar. First, the mobile application of Aucar streams the camera video to the server of Aucar, along with the current location from the device’s global positioning system (GPS). After that, the server will add time and location as the meta of video stream, then save as file clunks. An index will be created afterwards, such that it would be able to search for vehicles and drivers by location and time. Therefore, in case of an accident has happened, Aucar users involved in the accident would not need to make a copy of the video as if they are using a traditional car camera. A copy of the video is already available in their personal account on server. Moreover, as solely for technical and academic purpose, the consideration of legal and privacy concern is out of the scope of this report: it would be possible for Aucar to reconstruct the accident by searching for other Aucar users or non-Aucar vehicles identified from videos. The movement time-line of each objects and a list of witnesses can be provided.

1.4 Scope and deliverable

There are two objectives and deliverables in Aucar.

The first objective and deliverable is to create a mobile application which can replace all other accessories including navigation, car camera and head-up display (HUD) that drivers install to their dashboard. The application is able to record video and stream to Aucar’s server, provide navigation, recognize number plates and calculate the distance
while recording. Hence when the application has detected a potential collision to vehicles ahead, it immediately alerts the driver via notification and alarm. In the current implementation, the application supports Android only.

The second objective and deliverable is to create a website which user can logs into and replay their video records. The website also provides a "god view" of the overall traffics on a map. Moreover, it also serves as a marketing material to introduce Aucar and information for the project required by the University of Hong Kong, such as progress and schedule. The website is available on https://www.aucar.io/.

1.5 Outline

This report is going to first discuss the difference between AUCAR and some previous works in section 2 previous works. Then explain how the functions and features are going to be implemented in section 3 methodology. Including the hardware and software libraries that have been used, and the design of AUCAR. After that, the result and limitations encountered will be discussed in section 4 results and section ?? limitations. This report will be finished with a conclusion of the project in section 6.
2 Previous Works

There are some solutions available in the market which provide video recording using smart phone, including Nexar and DailyRoads. In this section, both solutions will be reviewed and compared with Aucar.

2.1 Nexar

2.1.1 Introduction of Nexar

Nexar is a mobile application that supports both iOS and Android. It allows users to record video like a dash camera by mounting their smart phone on the dashboard, and it will upload the recordings to Nexar’s server. In addition to video recording, Nexar also created a vehicle-to-vehicle network with Nexar users, in which the applications communicate with each other directly for events like braking or slowing down. See figure 1 for an example screenshot, the bottom left of the figure has shown that the application notified the nearby Nexar users when the driver applied brake.

Figure 1: Nexar ADAS warning
2.1.2 Comparison between Nexar and Aucar

2.1.2.1 Video recording and streaming  Both Nexar and Aucar provides video recording function. Nexar records both outside and insider of the vehicle (see figure 2), which Aucar records outside only. However, Nexar does not allows user to live stream video from their website or other applications. Aucar allows user to live stream their video stream from active Aucar applications, this is useful for fleet management. For example, a bus company would be able to see the live stream from a bus when they had received an accident report.

![Figure 2: Nexar video recording](image)

2.1.2.2 Vehicle to vehicle (V2V) network  Nexar creates a V2V network and joins vehicle when the driver is using Nexar. Vehicles inside the network communicate proximity warnings, claims that dangers beyond drivers’ field of view are alerted. Aucar is not planned to add a similar network due to the consideration of complexity and security. It would increase the smart phone’s workload and exposed the system to a potential security weakness, as the application would form a peer-to-peer network themselves. It is possible to send fake signals within the network and it requires a complicated encryption and protocol to protect users.
2.1.2.3 Object recognition  Nexar claims that machine learning and computer vision are used to constantly scan the road and detect when objects ahead become dangerous, for example, when the car ahead suddenly stops (see figure 3). The figure shows that Nexar recognizes objects in the scene and tags the objects with label. However, no other information or screens can be found except the shown figure which is downloaded from Nexar’s website (https://www.getnexar.com/). The principle and functions of the system remain unclear in the time of writing this report (April, 2018).

Initially, a similar approach had been tried in the early development stage of Aucar (see figure 4). However, the implementation of such system is not ideal and realistic. It also fails to serve the purpose of recognizing potential dangers, the reason and detail of this conclusion is covered in the result section. As a result, number plate recognition using computer vision has replaced the general object recognition. Aucar is focused on recognizing number plates and estimate the distance using camera calibration.
2.1.2.4 Collision report  Nexar generates a collision report when requested by users (see figure 5). The report includes the following items [4]: images, video, driving path map, collision location, force analysis, speed analysis, collision diagram, weather conditions, timestamps, and complete driver information.

Aucar provides a function for the same purpose, although the implementation is different. Aucar has the driving data for all Aucar users, including the movement timeline, which is similar to the concept of driving path map. Aucar also has the other assets including images and video recordings. However, Aucar is not generating a report of any side’s perspective. Aucar is reconstructing the accident in a general view, by utilizing the data from all involved parties. In addition, Aucar can generate a report not only for collision, but a report for the general environment of a specific time and location.

For example, when an animal has been hit-and-run by a vehicle, in most cases there will not be sufficient evidence for the police to identify which vehicle, even when there were other drivers driving by. Since the police would need to ask for witnesses and passing by drivers to provide evidence voluntarily. This is a very inefficient method since either the witnesses or drivers are not aware of this request, or they just do not have time to do so. Instead, the police can provide the time and location of accident, then Aucar will
actively notify the drivers that were at the scene during that time period, and ask for permission or consent to provide the data that Aucar has on the server.

Figure 5: Nexar collision reports
2.2 DailyRoads Voyager

2.2.1 Introduction of DailyRoads Voyager

DailyRoads Voyager is an Android application that allows continuous video recording from vehicles (see figure 6). The application records video from the camera and saves to the local storage. It can recycle old recordings when the storage space is full, while keeping the important events which can be tagged both automatically or manually. It can also capture photos at specific intervals. Captured videos and photos can be automatically uploaded to DailyRoads, or manually upload to third-party storage providers like YouTube, Facebook, Dropbox, etc.

Figure 6: Dailyroads Voyager screenshot

2.2.2 Comparison between DailyRoads Voyager and Aucar

2.2.2.1 Video recording configurations

DailyRoads Voyager allows users to configure the variables of video recording (see figure 6). Users can specify the resolution and camera mode used in recordings. Aucar provides the same function which allows users to configure the resolution of recordings (see figure 7).
2.2.2.2 Loop recording  DailyRoads Voyager offers loop recording, which automatically delete the oldest video file when the storage space is full. In addition, it also skips important recordings when deleting old files. The application uses accelerometer to detect sudden changes in speed while recording, and mark the recording as important automatically when such events is detected. Users can also mark recording as important manually in the application. Aucar also provides the same function which automatically deletes old files when the storage space is full.

2.2.2.3 Meta data recording  DailyRoads Voyager also records speed, elevation, timestamp and GPS coordinates alongside of the video recordings on each second. Depends on the permission granted and available sensors, Aucar also record the same set
of data while recording. Meta data is stored in a separate file and sorted by timestamp, and is uploaded to the server afterwards. Aucar server will match the video and meta data according to the timestamp.

2.2.2.4 Video storage and upload DailyRoads Voyager stores video recordings in local storage, and provides function to upload user’s videos to server of other third-parties storage provides (see figure). However, video recording in Aucar are stored in server instead of local storage. Local storage is only used as cache in Aucar, such that the length of recorded videos is not limited by the size of local storage.

2.3 Differentiation and contribution of Aucar

2.3.0.1 Dash camera replacement Replacing dash cameras with smart phones is not a new idea and is already exists since the invention of smart phones. Until recent, some new applications are also trying to utilize machine learning and modern cloud server infrastructure in car safety. Aucar is built with a similar idea in mind, enhancing driving safety and experience using the latest technologies.

However, it is believed that the live camera preview is only needed when the user is mounting their smart phone. When the position is set and started recording, the
screen is spared for information other than the live camera preview. Therefore, Aucar integrates OpenStreetMap (OSM) to provide map and navigation. Unlike other dash camera replacement platform, Nexar and DailyRoads Voyager for example, Aucar means to be an all-in-one replacement. A basic usage scenario is the user mount their smartphone on their vehicle when getting in, then Aucar is automatically launched because of connected to the vehicle’s Bluetooth automatically when the user starts the engine. After that, Aucar will start recording since the mount is already in position after the first setup, and shows a map which centered to the vehicle’s current location. Nearby traffic and point of interest are shown on the map, and user is good to go.

2.3.0.2 Rethinking video recordings
In general, there are two types of recordings, driving and parking recording. For driving recordings, drivers would like to have a video recording to show what had happened in case of accident, and as backup evidence to support their statement. For parking recordings, vehicle owners needs to know what happened to their vehicles when they are not there. For example, they wanted to know who hit their vehicle and run away, they need a video for proof and report to the police. It is observed that both recordings are in fact to serve as an evidence in case of accidents or damages.

2.3.0.3 Drawbacks of dash cameras
After an accident had happened, the police will take away the memory card in dash cameras as evidence. If the driver wanted to keep a copy of the video himself, the backup process needs to be done on scene and before the police take it. Sometimes it is impossible to do so since the dash camera might not provide such function or connectivity itself, a card reader and a computer is needed to backup the videos. Despite having such equipment or the dash camera allows the user to backup the video using a smart phone, it might be dangerous to do so in some situation like on a highway.

2.3.0.4 Benefits of Aucar recordings
However, drivers do not need to do anything if they are using Aucar. Videos are already stored in Aucar’s server and there are multiple ways to submit the video as evidence to the police. For example, they will be able to download the video from Aucar’s website, or just give a consent to allow Aucar provides
their video recording to the police for them. As a neutral third-party, Aucar can also prove that the video is unedited.

2.3.0.5 Smart city and data collection Aucar and other platforms have one common characteristic: The more users it has, the more detail and higher accuracy it is. Data is being collected and analyzed from the application. For example, a map for the density of traffic can be generated by collecting the location of vehicles. In addition, when smart city has been implemented and governments are provide application programming interfaces (api) to traffic and transport information, for example traffic accidents happened in tunnels and disabling lanes. Aucar can provide accurate information by connecting to these interfaces.

2.3.0.6 Functions of number plate recognition However, smart city is still in concept stage. Therefore, Aucar still depends on collecting data from users. Aucar tries to collect more data from a single user, so number plate recognition is used. The location of other vehicles that is in the field of view can also be collected by recognizing number plate and calculate the distance between it and host vehicle. Hence even if a vehicle is not using Aucar, its location can still be collected. In theory, fewer installations are needed to acquire the same coverage in comparison to not using this method. In addition, the distance to number plate also serves as a danger indicator. Aucar uses it to predict potential collisions and warns drivers beforehand.
3 Methodology

There are three components in Aucar: A mobile application, a website and an application programming interface (API) server are going to be developed. The mobile application and website will be connected to the API. Users will be able to sign in on both application and website using the same account. In the following sections, the architecture of whole system will first be introduced, then the design and principle of each component will be discussed.

3.1 System architecture

Aucar’s system is hosted in two servers, one for both API and website, another for database (see figure 10). Both servers are hosted by Microsoft Azure in Hong Kong, and there is no plan of moving or changing. The mobile application and website serves as an user interactive interface while the API handles all logic. For example, the website provides a form for user to input the credentials when a user is signing in, and then send a request to API and authenticate the credentials. In such way, the sign in function in mobile application can also utilize the same API to authenticate credentials. Hence, redundant code has been reduced to minimal level, and it is easy to add any other interactive clients to the system.

Figure 10: Aucar system architecture
3.2 Mobile application

A mobile application is developed to serve as an edge client of Aucar, this has replaced the custom hardware which was adopted in the early development stage of Aucar.

3.2.1 Supported platform

In 2017, there are two operating systems dominating the whole mobile market, iOS and Android. According to Statista [5], Android has 87.7% share of the market by Q2 2017. Therefore, Android has been chosen as the first supporting platform, as in comparison to iOS that less effort is required to cover more users. Despite only Android is going to be supported because of the resource limitation, iOS support is possible in the future development.

3.2.2 Reason of creating mobile application

In the early development stage of Aucar, video recording and object recognition were planned to be accomplished by using a custom hardware. The hardware was gathered around by installing high quality sensors to single board computers. However, it was found harder to develop a whole software suite on it because of the computational power limitation. Smart phones in 2018 are a lot more powerful than the single board computers that was used, and users do already own a smart phone. In order to overcome the limitation, and it is believed that convincing users to install a free application is easier than purchasing a new hardware, the hardware was replaced by a mobile application. In addition, it is much easier to publish a mobile application internationally than a hardware.

3.2.3 Development environment

Java has been used as the programming language in the development of Aucar, it is the major supported language by Android officially. The minimum supported sdk version of Aucar is 21, which is also known as Android 5.0 Lollipop. This version is chosen because there are 82.3% of Android devices running Android 5.0 or above [6].
3.2.4 Video recording and upload

The mobile application records video and upload to Aucar’s server. It shows a camera preview when the user has signed in, and starts recording when user touched the record button. It records video from the back camera on a smart phone, and save the file to local storage. Video recordings are truncated into small video fragments of 10 seconds, then being uploaded to Aucar’s server one by one. The uploaded fragments will be deleted only after it has been uploaded, hence it is always available until the fragment is stored in Aucar’s server. In case of no network connectivity or upload failures, the user would not need to worry about losing the video recordings.

3.2.5 Meta data recording and upload

Meta data including position, speed, elevation and timestamps are also recorded while the application is recording videos. Meta data are also being cached to the local storage and uploaded to Aucar’s server in another background thread. Data are being uploaded in a frequency of 1 second, and will only be deleted after it has been successfully uploaded. Therefore, users would not need to worry about losing the cached data.

3.2.6 Carrier data package consideration

Since it is not a standard for every user to have unlimited carrier data package, or the speed of transmission, for example, 2G and 3G mobile network, is not suitable for the usage of Aucar. Therefore, a controllable time of upload is a must have function. As mentioned in the previous sections, video and meta data are cached until they have been successfully uploaded. Aucar also provides an option for user to only upload these data when they are connected to Wifi. However, since videos need to be cached until they connect to Wifi. Users are reminded that in this delayed mode, the length of recordings are limited by the available storage on their smart phone.

3.2.7 Video playback

Users can replay their recorded videos in the application, both uploaded and local cached videos. The application can stream their recorded videos from api, it provides a native
video player interface for the playback. Therefore, a regular user would be able to use the player to replay the playback, including functions like jump to a specific time and pause are also provided. Figure 11 shows the interface of video player, a pitch black video is recorded by covering the camera in order to demonstrate the interface clearly. Users can jump to a specific time by touching the timeline on the bottom. The current battery level is shown on the top right corner, and users can toggle fullscreen mode using the bottom right button. When the playback is done, users can back to the list of videos using the top left button.

Figure 11: Aucar video player interface
3.2.8 number plate recognition

The mobile application also recognizes number plates and calculate the distance to the recognized plates while recording. This process is handled by a separate background thread (recognizer) with OpenGL off-screen rendering while the application is recording videos. See figure 12 for the general process, each step will be explained in the following paragraphs.

Figure 12: Aucar number plate recognition process
3.2.8.1 Image input  Video frames are captured from camera and sent to the recognizer. Usually, users would like to have a high resolution video for video recordings. However, it is not suitable for a smart phone to analyze a high resolution video frame as the total pixel count is too high. Therefore, when a frame arrives the recognizer, it is first being scaled to a new image with a height of 480 pixels. As a result, for a regular 16:9 video frame, the total pixel count is only $853 \times 480 = 409440$ in comparison to a 1080p, which has $1920 \times 1080 = 2073600$. The pixel count is reduced by over 80%, while remains sufficient for identifying the content of number plates. See figure 13 for sample output.

![Figure 13: Aucar number plate example image in 480p](image-url)
3.2.8.2 Grayscale transform  Since the colour of each pixel is not useful in the process, only the light intensity is needed. Therefore, the image is transformed to grayscale. The intensity of each pixel is calculated by \( I(x, y) = 0.299 \times R(x, y) + 0.587 \times G(x, y) + 0.114 \times B(x, y) \), where \( I(x, y) \) means the intensity of pixel on \((x, y)\), \( R(x, y) \) means the red value of pixel \((x, y)\), and \( G(x, y) \) and \( B(x, y) \) represent green and blue respectively. OpenGL is used since the graphics processing unit (gpu) can processes the pixels in parallel, which is much faster than doing the same on a central processing unit (cpu). So a custom shader is used to transform an input of 24-bit RGB texture to an output of 8-bit floating point texture. See figure 14 for sample output.

Figure 14: Aucar number plate example image in 480p and grayscale
3.2.8.3 **Corner detection** Before recognizing number plates, corner positions are needed from the image. Harris corner detection is used in Aucar to detect corners. The algorithm takes a smoothed grayscale image and output the position of corners. Since the image is already transformed to grayscale, hence it is only smoothed by convolving a Gaussian Kernel with a size of 7. The smoothed image is then sent to the Harris corner detector, which is implemented in the application. See figure 15 for sample output.

![Aucar number plate example image in 480p and grayscale with detected corners](image.png)

Figure 15: Aucar number plate example image in 480p and grayscale with detected corners
3.2.8.4 Number plate recognition  After the corners are located, the 8-adjacency connectivity with a threshold of 32 is calculated on each corners. This gives a set of connected components in the image which has a similar colours, then each components is being cropped. After that, the components will be filtered by the ratio of width and length of their bounding box. Only boxes that has a ratio within 10% range of 1:4.5217 is considered as number plates. See figure 16 for sample output. After the number plate is being recognized, it will be cropped and applied histogram equalization (see figure 17). The cropped image will be applied optical character recognition (ocr) and camera calibration to estimate the distance (see figure 18).

Figure 16: Aucar number plate example image in 480p and grayscale with recognized plate

Figure 17: Aucar number plate example cropped number plate

Figure 18: Aucar number plate example cropped number plate with histogram equalization
3.2.8.5 Distance calculation In order to calculate the distance between number plate and the smart phone, camera calibration is used. Camera calibration is a process of estimating the intrinsic and extrinsic parameters. Since the intrinsic parameters can be retrieved from the operating system, the extrinsic parameters can be estimated from the image. The top left corner of recognized number plate is assumed to be (0,0) in world coordinate and the bottom right corner is (115, 520), as the number plate has a size of $11.5cm \times 52cm$. Then the regular camera calibration process is done and the z-axis value is used as the estimated distance to the number plate.

3.3 Website

A website (https://www.aucar.io) is created to serve as the homepage of Aucar and also a web application for Aucar’s users. Users can sign up on the website, and sign in to the homepage of their Aucar account. The website provides an interface for users to replay their video recording stored in Aucar’s server. In the following sections, the design and functions of the website will be explained.

3.3.1 Server environment

The hosted server is a virtual machine running Ubuntu Server 17.04, and an Apache web server of the version 2.4.33 has been installed. It has been configured to enable gzip compression to compress the transmission, and http/2 for faster respond. PHP 7.2 is also installed and configured.

3.3.2 Development environment

The website is developed in PHP, and Laravel framework is used. The source code of whole project is managed by git, and a continuous integration setup has been done. Therefore, every commit will trigger an integration which will automatically build and test the code. If it passes the tests, the website will be automatically deployed to Aucar’s server. Laravel is used because of it eliminates the need of programming common code, for example, cookies and sessions handling. It also provides an object-relational mapping (ORM) called Eloquent to speed up the development process, it helped developers operate
the database without writing queries for every single operation.

### 3.3.3 Sign up and in

The website is connected to the api and allows users to sign up and in. It provides a form for user to input their credentials (see figure 19). The website itself does not process user’s credentials, but request api to authenticate the credentials. It redirect users to their homepage if they are authenticated, otherwise a corresponding error message will be shown.

![Figure 19: Aucar.io sign in form](image)

### 3.3.4 Video playback

The website provides a list of the recorded videos for authenticated users, which is sorted by the begin time of recording. A native interface to replay their recorded videos is also provided by the browser since it is using HTML5 video element, users will be have the common video controls like jump to specific time and pause. The video source is retrieved from the api.
3.3.5 God view

The website has integrated Open Street Map (osm) to provide a god view of traffic, users will be able to see the location of vehicles on map. The data is retrieved from the api and the map is real-time, which means that the map will be updated automatically without refreshing the web page.

3.4 Application Programming Interface

An application programming interface (api) has been developed to support the system of Aucar. It shares the same server as the website and the interface is served as a representational state transfer (REST) api through HTTP. Details of the api will be explained in the following sections.

3.4.1 Server and development environment

The api shares the same server as the website, it is also served by the same Apache web server. It is also developed in PHP and Laravel framework. The source code of whole project is also managed by git, and is automatically integrated on every commit, like the website. A database has been installed on another server, it is a MySQL server of version 5.7.21. In Aucar, only the api can connect to the database and execute queries. The api identifies the incoming HTTP request by their requesting endpoints, for example, "/oauth/token" is an endpoint for requesting OAuth 2.0 tokens. Then the verb of request is considered, for example, GET and POST to the same endpoint will triggers a different action and result. Sending GET request to the videos endpoint means getting all the video recordings that a user has. However, sending a POST request to the same endpoint means the user is uploading a video.

3.4.2 User authentication

The api has implemented the complete OAuth 2.0, and authenticate each request to identify the authorized user. The website and mobile application are considered as the first-class clients, hence it is not required to redirect users to Aucar’s website to sign in. However, the complete OAuth 2.0 token exchange procedure is still completed in the
background. When a user is signing through the first-class clients, a new access token is generated. These first-class tokens are long lived by default, hence users do not need to sign in again unless they have explicitly signed out. The client only stores the access token for security concern, users will be able to sign out clients remotely just by revoking the tokens. However, in the current stage of Aucar this function is not released to normal users, but only administrators. See figure 20 for the sequence diagram of OAuth 2.0 token request procedure and request authentication [7].

![Authorization Code Grant Flow](image)

Figure 20: Aucar.io OAuth 2.0 token request sequence diagram

### 3.4.3 Video recording and playback

The mobile application records video and meta data, then send to api. When the api received the video fragments and meta data, it combines the videos into sessions. A sessions starts from the recording button being touched and ends when it has been touched
again. Therefore, the whole trip will be combined as one video in api, although the video has been truncated into fragments when caching and uploading. When a video is being replayed in clients, the meta data and video are streamed together to the client.

3.4.4 God view

When number plate is recognized by the mobile application, the recognized number and distance is also sent to the api. The global coordinate will be calculated using the uploader’s current coordinate and the distance. Then the license number, global coordinate and timestamp will be stored into database, the location of uploader is also stored. When the god view is requested, the latest position of each license plate will be returned. For those license plate that isn’t updated after 15 seconds, it will be filtered out from the god view and considered it is already not on the same location.
4 Results

At this stage, the mobile application, website and api are developed. The crucial functions of Aucar are implemented, including video recording and playback, navigation and member system. The detail of result will be discussed in the following sections.

4.1 Mobile application

The Android version of Aucar mobile application had been developed, but it is not published in Google Play at the time of this report. Users can sign up and in in the application, then mount their smart phone to their vehicle and use it as a dash camera. While it is recording, the user can use the application as navigation. After that, recordings can be replayed in the application. Number plate recognition is also implemented in the application, a real number plate with custom license number "AUCAR" is purchased to test this function.

4.2 Website

A server had been rented and configured on Microsoft Azure, a website had been developed and available on https://www.aucar.io/. It serves as an introduction to Aucar and also the web application for Aucar’s users. Users can sign up and in on the website, and replay their video recordings.

4.3 Application programming interface

Another server had been rented and configured on Microsoft Azure, a database had been installed to the server. The interface is developed and can connect to the database. It is supporting the operation of mobile application and website.
5 Future Plan

There are two fundamental goals planned. The first goal is publishing the Android version of mobile application, and the second goal is to develop the iOS version of the mobile application.

5.1 Publishing the Android version of mobile application

At this stage, the application can not be published yet. The user interface of Aucar’s mobile application is not production ready, although it is functional. In order to attract and keep users, the user interface and experience is very important. Therefore, the user interface has to be finished and polished before publishing the application.

5.2 Developing the iOS version of mobile application

Although Android has over 80% of share in the market of smart phone, iOS users cannot be neglected. The development of the iOS version is planned after the publication of Android version.
6 Conclusion

A new system called Aucar had been developed to enhance driving safety and experience. It replaces the traditional dash cameras and navigation devices, by providing cloud connected video recording and integrated navigation.

An Android application, website and api were developed, although it is not published yet. During July 2017 to April in 2018, Aucar has gone through planning, research and then the development stage. The first objective was to develop the application and number plate recognition, these were developed in parallel to each other. The next objective will be finishing and polishing the user interface and publish the application.
References


