Software and System Test Documentation

VehID

Version 1.0

Remington Greko, Spencer Hirsch, Thomas Johnson, Alexis Nagle

Project Advisor: Dr. Marius Silaghi
Project Client: Clayton Levins, Executive Director of Smart North Florida

October 1, 2023

Document Change Procedures and History

This section outlines the procedures for identifying, approving, implementing, and recording changes to this document, "Software and System Test Documentation" for VehID version 1.0.

Change Procedures

All changes to this document will be idfentified using the following information:

- **Document ID:** Each documented related to VehID testing shall have the unique identifier connected to the system project. The document ID for this test plan is *VehID version 1.0*
- ***Version Number:*** *Versions of this document will be numbered sequentially, starting with the first approved version, this version 1.0*

Change Approval

Changes to this document must be approved by the project advisor Dr. Marius Silhagi, and the project client, Clayton Levels, executive direcotr of Smart North Florida.

Change implemention

Once changes are approved, they will be implemented by whomever is responsible for the specific documentation.

Change Recording

All changes made to this document will be recorded, and logged with the following information:

- Document Id: Test plan for VehID version X.X.
- Version Number: Sequential version number.
- Description of Document Changes: A brief description of changes made.
- Reason for Changes: Reasoning for changes, team review, system changes, etc.
- Name of Person Making Changes: Name of the person responsible for making changes, and their role.

Change History

A change history log will be maintained at the end of this document to track all changes that have occured since its initial inception.

Contents

1 Introduction

The purpose of this test plan is to identify the highest priority features of our project VehID. This test plan will outline our testing approach, and procedures which includes our testing environment, input, and results. These procedures will be used to help us achieve our goals of vehicle identification from a live camera feed. Key interests in testing will be testing the accuracy of our CNN to identify vehicles, proper video data processing, and design of a logical database to store information.

1.1 Document Identifier

Test plan for VehID version 1.0

1.2 Scope

The testing scope for this level of testing encompasses the software product known as VehID. The software aims to identify vehicles based on specific criteria, including color, make, model, and license plate. This level of testing will verify the following aspects:

- The ability of the software to process live video feeds from roadway cameras.
- The accuracy and efficienct of the Convolutional Neural Networks in identifying vehicles, and their attributes.
- Real time image processing to ensure prompt identification.
- The reliability of web scraping mechanisms to gather vehicle information from external sources.

1.3 Context

No specific additional context is required beyond what is covered in the document. The testing is conducted as part of the VehID project, and there are no third-party Internet-based testing components.

1.4 Notation for Description

The descriptions and numbering schemes for test cases, and scenarios will follow a standardized format. Test cases to a degree will have codependency on each other, and as such will be placed into larger groups to reflect the overall functioanlity they are testing. For example “Vehicle Recognition Test Cases”. Specific test cases per category will be subsectioned, and given a unique numerical identifier local to their category.

2 Convolutional Neural Network Testing

2.1 Object Recognition

2.1.1 Test Case Identifiers

cnn-tests-obj-rec

2.1.2 Objective

Ensure model can correctly distinguish a vehicle in a frame against other objects in the frame. There will be a variety of these tests, including ones that have no vehicle, some that only have one vehicle in a variety of sizes, and finally some that contain numerous vehicles. It is important to ensure that our model can account for a numerous vehicles in a single frame.

2.1.3 Inputs

The input for these tests would be a single frame from a video feed.

2.1.4 Outcomes

Accurately distinguish objects in a frame from one another. Draw a border box around vehicles in the frame.

2.2 Color Recognition

2.2.1 Test Case Identifiers

cnn-tests-color-rec

2.2.2 Objective

Given a frame that is deemed to contain a vehicle, our system must accurately identify the color of the vehicle in the frame.

2.2.3 Inputs

The input for these tests would be a single frame from a video feed.

2.2.4 Outcomes

This test is expected to return the color of the vehicle that can then be compared with our database to narrow the search vehicles contained within the database.

2.2.5 Intercase Dependencies

cnn-tests-obj-rec

2.3 Make/Model Recognition

2.3.1 Test Case Identifiers

cnn-tests-mm-rec

2.3.2 Objective

Given a frame that is deemed to contain a vehicle of a suspected color, our system must accurately identify the make a model of a vehicle in the frame.

2.3.3 Inputs

The input for these tests would be a single frame from a video feed.

2.3.4 Outcomes

This test is expected to return the make and model of a vehicle, this data will then be used against our database containing vehicles of a specific color, this will then be used to narrow our search down even further.

2.3.5 Intercase Dependencies

cnn-tests-obj-rec, cnn-tests-color-rec

2.4 License Plate Recognition

2.4.1 Test Case Identifiers

cnn-tests-license-rec

2.4.2 Objective

Given a frame which contains a vehicle, our system must accurately identify whether or not a license plate is present in the frame. If there is a plate, it needs to then identify the plate number in the frame and compare this data to our database.

2.4.3 Inputs

The input for these tests would be a single frame from a video feed.

2.4.4 Outcomes

This test is expected to return the license plate information of the vehicle. This data will then be used against our database to determine whether or not a match as been found.

2.4.5 Intercase Dependencies

cnn-tests-obj-rec, cnn-tests-color-rec, cnn-tests-mm-rec

3 Web Scraping

3.1 Scraping from Site

3.1.1 Test Case Identifiers

scraping-tests-web-scraping

3.1.2 Objective

Ensure the web scraping tool can adequately extract necessary data from the needed websites in order to create entries that can be added to the database

3.1.3 Inputs

The input for these test cases will be a specified website that contains the necessary information to create database entries.

3.1.4 Outcomes

The expected output for these test cases is a new entry being created into the database for each instance of information scraped from the website.

3.2 Add Entry in Database

3.2.1 Test Case Identifiers

scraping-tests-add-entry

3.2.2 Objective

Ensure data extracted from the websites is in a valid format and properly create a database that includes this new data.

3.2.3 Inputs

The input for these test cases will be the data extracted from the websites.

3.2.4 Outcomes

The expected output for these test cases is a properly generated database that contains all the new information along with any existing information.

3.2.5 Intercase Dependencies

scraping-tests-web-scraping

3.3 Avoid Duplicate Entries

3.3.1 Test Case Identifiers

scraping-tests-avoid-duplicate

3.3.2 Objective

Ensure that duplicate information is not added to the database. If the same site is scraped on a regular interval or if multiple sites are scraped then there may be overlap in the information scrapped from the sites. Due to this we want to ensure that information is only being added to the database once.

3.3.3 Inputs

The inputs for these test cases will be in the form of database entries that are pending to be added to the database.

3.3.4 Outcomes

The expected output will either be an accept or reject based upon if the information is already present. If it is accepted, the entry will be added to the database. If it is rejected, the new entry will be deleted.

3.3.5 Intercase Dependencies

scraping-tests-web-scraping, scraping-tests-add-entry

4 Web Application

4.1 User Input

4.1.1 Test Case Identifiers

webapp-input-test

4.1.2 Objective

To test the function which checks for validity of a user input in the web application.

4.1.3 Inputs

The user enters a query into the database search bar.

4.1.4 Outcomes

The web application should attempt to sanitize the input and accept it, or reject the input entirely.

4.1.5 Intercase Dependencies

db-tests-webapp-input

4.2 Login

4.2.1 Test Case Identifiers

webapp-login-test

4.2.2 Objective

To test the function which reads and validates a login attempt from the user logging into the system via the user interface.

4.2.3 Inputs

The user will enter a username and password to the UI.

4.2.4 Outcomes

The user will be logged into the system.

4.2.5 Intercase Dependencies

db-tests-webapp-input, webapp-input-test

4.3 User authentication

4.3.1 Test Case Identifiers

webapp-auth-test

4.3.2 Objective

To test the funciton which authenticates the user and grants permissions based on the roles assigned to them in the system.

4.3.3 Inputs

The user submits a login attempt to the sytem.

4.3.4 Outcomes

The user is granted proper permissions based on their role (i.e. Admin, User).

4.3.5 Intercase Dependencies

db-tests-webapp-input, webapp-input-test, webapp-login-test

4.4 Notification System

4.4.1 Test Case Identifiers

webapp-notif-test

4.4.2 Objectives

To test the function which displays an alert to users logged into the system when a match is validated.

4.4.3 Inputs

Given a new match entered into the database.

4.4.4 Outcomes

Users which are logged into the system are notified of the update to the database.

4.4.5 Intercase Dependencies

db-tests-webapp-input, webapp-input-test

5 Database

5.1 Entry from web app user

5.1.1 Test Case Identifiers

db-tests-webapp-input

5.1.2 Objective

To test the functionality of adding an entry to the database through the web application interface.

5.1.3 Inputs

- User provides vehicle information including color, make, model, license plate, and other relevant details.
- User submits the information through the web application.

5.1.4 Outcomes

- The database should accept the new entry and store it correctly.
- Verify that the data entered matches what was submitted.
- Check if the database assigns a unique identifier to the new entry.
- Ensure the database maintains data integrity after the addition.

5.1.5 Intercase Dependencies

db-tests-webapp-input, webapp-login-test, webapp-input-test

5.2 Entry from web scrape

5.2.1 Test Case Identifiers

db-tests-webscrape-input

5.2.2 Objective

To test the functionality of adding an entry to the database through web scraping mechanisms.

5.2.3 Inputs

- The web scraping module fetches vehicle information from external sources.
- The retrieved data is prepared for database entry.

5.2.4 Outcomes

- The database should accept the new entry created through web scraping.
- Verify that the data retrieved from external sources is accurately stored.
- Check if the database assigns a unique identifier to the new entry.
- Ensure the database maintains data integrity after the addition.

5.2.5 Intercase Dependencies

scraping-tests-web-scraping, scraping-tests-add-entry, scraping-tests-avoid-duplicate

5.3 CNN reading from Database

5.3.1 Test Case Identifiers

db-tests-cnn-reading

5.3.2 Objective

To test the functionality of the Convolutional Neural Networks (CNNs) reading vehicle data from the database.

5.3.3 Inputs

The CNNs request vehicle data from the database for identification purposes.

5.3.4 Outcomes

- The database should provide accurate and up-to-date vehicle data to the CNNs.
- Verify that the CNNs can successfully read the required attributes from the database.
- Check if the database maintains data integrity while serving data to the CNNs.

5.3.5 Intercase Dependencies

cnn-tests-obj-rec, cnn-tests-color-rec, cnn-tests-mm-rec, cnn-tests-license-rec

6 Global

Document Change History

ID	Version Number	Desc. Doc Changes	Reason	Person Changing	Role
Test plan for VehID	Version 1.0	Initial Version	N/A	N/A	N/A