Test Result Report

This Test Result Report will be completed referencing the following sections of the Master Test Plan MTP:

- **MTP SECTION 10**   Planning
- **MTP SECTION 5**   Test Levels
- **MTP SECTION 4.b** Test Strategy
- **MTP SECTION 2.f.ii** Scope and Criteria

This Test Result Report is designed to be reusable for each testing phase specified in **Section 10. Planning** of the MTP. By identifying each phase and requirements the testing plan can be compiled, completed and documented.

Report Details

Test Report Date	2021-06-17
Test Report Iteration (See section 10 of the MTP)	E-4 (Elaboration 4)
Test Report Description	Elaboration 4 marks the end of the elaboration phase of the project. This test report sets out to document and validate work completed to accomplish the critical, core, risky, difficult (CCRD) use case. To target the CCRD this Test Report will focus on our most complete works contained in the Jupyter Notebook held with the SageMaker environment on Amazon Web Services (AWS)

Test Environment

Because of restrictions, resource availability and convenience, tests may be completed in environments other than AWS. If this is the case considerations need to be made to account for differences between operating systems, python versions, processing and memory limitations and data availability (eg. S3 vs local drives).

Host	AWS SageMaker
Python / Kernel	conda_python3
Notebook Name	2.4_CV_Adam_GMM_Cont
Resources (compute, Ram or instance type)	ml.t3.xlarge

Targeted Use Case

Which use case as specified within the **Initial Requirements Model** document.

Name	**Flood Area Classification**
Description	When a DCS/SS employee wants to generate a map of flooding extent, they access the AWS Sagemaker notebook to initiate a batch upload of the aerial images to the SageMaker endpoint to receive a classification of flooded areas. **This use case is also deemed the CCRD for the project.**

Tests

Testing Criteria Description

This round of testing will focus on category A (High Risk) as described in the MTP. The MTP has identified the following criteria as high risk and hence will be tested first:

- Unit Testing - The unit tests set out to validate each component works in isolation, ensuring that no component is dependent on another component (sanitized environment). This isolation also provides evidence of component operability during integration testing.
- Integration Testing - All components are executed in their intended order to achieve "Flood Area Classification" use case.
- Functional Acceptance Testing - Ensure that components and outputs align with functional requirements.
- User Acceptance Testing - Does the system pass the CCRD use case

Unit Tests
1. Ability to upload data to the service
2. Able to process aerial imagery
3. Able to locate flood affected areas
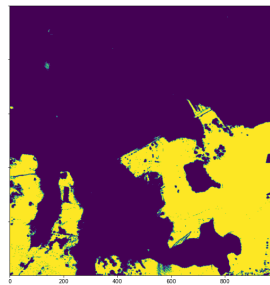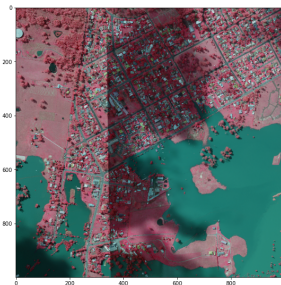4. Ability to output data in specified format

1. Ability to download data to the service

Step	Description	Expected Result	Pass /Fail
1	Open Notebook,	Notebook opens and use is able to configure script	Pass
2	Adjust settings to create a local directory to store the images (image destination)	Directory is created locally on the instance	Pass
3	Adjust settings for image retrieval location (image source). In this case images will be held in s3 Bucket	Images source is configured to download images	Pass
.4	Execute retrieval of images	Images are download in to the local directory	Pass

2. Able to process aerial imagery

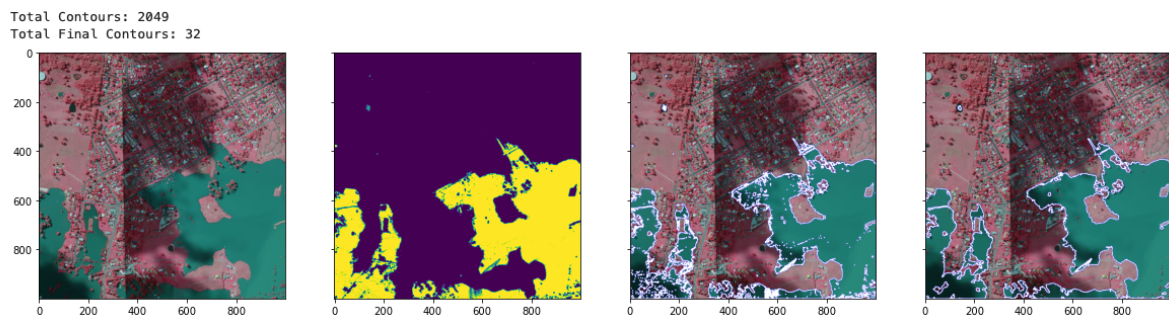| Step	Description	Expected Result	Pass /Fail
1	Execution of Notebook cells in notebook section 2.0 Extract Clusters	Images are preprocessed and clustering is performed. Clustering results can be viewed Test #1 - Test #3 of the notebook	Pass

2.1 Evidence - Original image and captured flood area

3. Able to locate flood affected areas

Step	Description	Expected Result	Pass /Fail
1	Execution of Notebook cells in notebook section 3.0 Contours	Using extracted flood clusters from 2.0. Execution of cells in section 3.0 will find contours from the binary cluster image. The number of will then be reduced by eliminating contours with an area smaller than 100. Remaining contours will be inaccurate training of flood extent	Pass

3.1 Evidence - Original image, captured flood area, contours, and contours after filtering by area

Total Contours: 2049
Total Final Contours: 32

4. Ability to output data in specified format

Step	Description	Expected Result	Pass /Fail
1	Execution of Notebook cells in notebook section 4.0 Convert	Contours are output to local file	Pass

Integration Tests

Test	Result
Able to process image from upload through to data output	Pass

Functional Tests

Test	Result
System Stability - System was able to remain stable through testing (no crashes, restarts, errors .etc)	Pass - System is able to execute from start to finish with no issues.
Time - System was able to process images and not have a major impact	Pass - Processing time is in < 15mins. This is still well within acceptable range of 1-2hrs

on the 48hr turnaround time from image capture to release	

User Acceptance Tests

Test	Result
Does the project meet requirements & scenarios provided by DCS	Pass - Currently the CSU team has access to a limited data set. However current testing proves we are able to extract flood extents from current datasets. Additional development testing will be required to

Tests Results

Completed by	Date
Adam Blewitt	2021-06-17

Unit Tests

Ability to upload data to the service	PASS
Able to process aerial imagery	PASS
Able to locate flood affected areas	PASS
Ability to output data in specified format	PASS

Integration Tests

Able to process image from upload through to data output	PASS

Functional Tests

System Stability - System was able to remain stable through testing (no crashes, restarts, errors .etc)	PASS
Time - System was able to process images and not have a major impact on the 48hr turnaround time from image capture to release	PASS

User Acceptance Tests

Does the project meet requirements & scenarios provided by DCS	PASS