FAST – Flood Assessment Structure Tool

FAST is a python-based structure level assessment tool for floods. It has a pre-processing tool that helps the user prep the data for the analysis tool.

Prerequisites to setup the tool:

Python Packages: GDAL, numpy, tkinter

Folders: ../Rasters, ../UDF, ../Log

Rasters
The program will look for rasters in the .tif, .tiff format in the 'rasters' in folder in the main directory. You may add or remove raster options as you see fit.

The default rasters included with this demo are>
Rasters:

- Honolulu_GAT.tif: Depth grid developed for Honolulu City/County based on potential inundation from the Great Aleutian tsunami scenario.
- NYC_rpd100.tif: Depth grid created for the 5 Boroughs of New York City based on the FEMA 100 year mapping.
- ND_Minot_PNNL100.tif, ND_Minot_PNNL500.tif, ND_Minot_rpd100.tif, ND_Minot_rpd500.tif

UDF:
This folder contains some sample UDF input files:

Each row must have columns corresponding to the fields below:

Input	Required?
A UDF in a .csv file-format	Yes
User Defined Flty Id:	Yes
Occupancy Class: One of 33 Hazus-defined types, e.g., {RES1, RES2, COM3, IND4, AGR1, GOV2, REL1}. Script will skip row if not specified, or if an unrecognized value is provided.	Yes
Cost: Replacement Cost of Structure, in US dollars. Records with '0' cost: the script will accept a zero value, but the record is essentially useless, as any estimated dollar damage to the structure will be 0. Consider correcting the UDF record or deleting it.	Yes
Number of Stories: Number of stories of building. Must be an integer.	Yes
Foundation Type: Foundation Type of the building. Text type, per Hazus-MH Flood Model convention. Must be an integer from 1 to 7, inclusively.	Yes

First Floor Height: Must be a float greater than 0.	Yes
Area: Total Area for the structure, in square feet. Used for Inventory Loss calculation when Inventory Cost is not supplied. Must be greater than 0.	Yes
Coastal Flooding attribute (flC): Identifies particular UDFs in a coastal flooding zone. If BldgDamageFnID, ContDamageFnId are not provided or populated, the script will use coastal flooding DDFs instead of Riverine DDFs. Only CoastalV, CoastalA, and Riverine are recognized by the script. For all other values, Riverine DDFs will be used.	Yes
Content Cost: If attribute is supplied, DOGAMI script will use the attribute value; otherwise, the script will assume Content Cost is 50% or 100% or 150% of Building Cost, depending on Occupancy Class. Must be greater than or equal to 0.	No
Inventory Cost: If user has better information than what Hazus estimates based on Occupancy Class and Square Foot. Must be greater than or equal to 0.	No
Building DDF: User-supplied Building Depth Damage Function ID. Used by Hazus-MH flood model and DOGAMI script to override the standard Damage Depth Functions for buildings. Text type, per Hazus convention. This is an optional attribute: if attribute exists, and the record is populated with a legitimate value, the script will use it, else the script will use the standard (default) DDF for the given Occupancy Class/Number of Stories/FoundationType. The 'ID' capitalization is per the Hazus naming convention. If the attribute is supplied, not all records need to be populated; supply with NoData or "" (blank) where there is no need to override the standard (default) DDF assignment.	No
Content DDF: User-supplied Content Depth Damage Function ID. Used by Hazus-MH flood model and DOGAMI script to override the standard Damage Depth Functions for building content. Text type, per Hazus convention. This is an optional attribute: if attribute exists, and the record is populated with a legitimate value, the script will use it, else the script will use the standard (default) DDF for the given Occupancy Class/Number of Stories/FoundationType. If the attribute is supplied, not all records need to be populated; supply with NoData or "" (blank) where there is no need to override the standard (default) DDF assignment.	No
Inventory DDF: User-supplied Inventory Depth Damage Function ID. Used by Hazus-MH flood model and DOGAMI script to override the standard Damage Depth Functions for building inventory. Text type, per Hazus convention. This is an optional attribute: if attribute exists, and the record is populated with a legitimate value, the script will use it, else the script will use the standard (default) DDF for the given Occupancy Class/Number	No

of Stories/FoundationType. If the attribute is supplied, not all records need to be populated; supply with NoData or "" (blank) where there is no need to override the standard (default) DDF assignment.	

HI_Honolulu_UDF.csv: A csv file containing (>80K) buildings with required attributes
NY_NYC_UDF.csv: A csv file containing (>800K) buildings with required attributes
ND_Minot_UDF.csv: A csv file containing (>12K) building with different occupancies and the required attributes

Lookup Tables
The program will look for the lookup tables in the .csv format in the 'lookuptables' folder in the main directory.

1. How to start:

Double click on OpenHazus_POC.bat in the main directory.

A windowed GUI should launch with field inputs.
A console log should also launch; check here for errors.

2. The GUI:

The GUI of the program allows for custom field mapping and checking. If valid input UDF (in a .csv format) is not selected, the fields will be color coded as RED.

If an input UDF is selected the, the program will search through the input UDF's field names and cross-check them against what is currently in the corresponding text entry box. It also checks against the default name of the field, according to its field name on the left of the entry box.
If the field is colored YELLOW, that field is has not been successfully mapped, but is NOT critical.
NOTE: We encourage the user to customize field-names to be recognized by default by changing the code in gui_program.py.

If the field is colored RED and a UDF is already selected, the field has not been usefully mapped and IS critical.

If the field is colored GREEN, the field has been mapped successfully.

3. Input:

CSV file with fields corresponding to program requirements.
You will be asked to browse for this file. The included UDF folder has some ready-to-go samples with pre-mapped fields. You must be sure to select a depth grid that spans the coordinates in the UDF, otherwise depth calculations will not results in any loss calculations. NOTE: multiple grids can be selected if you want to run them sequentially; they may be selected by shift or control clicking multiple selections.

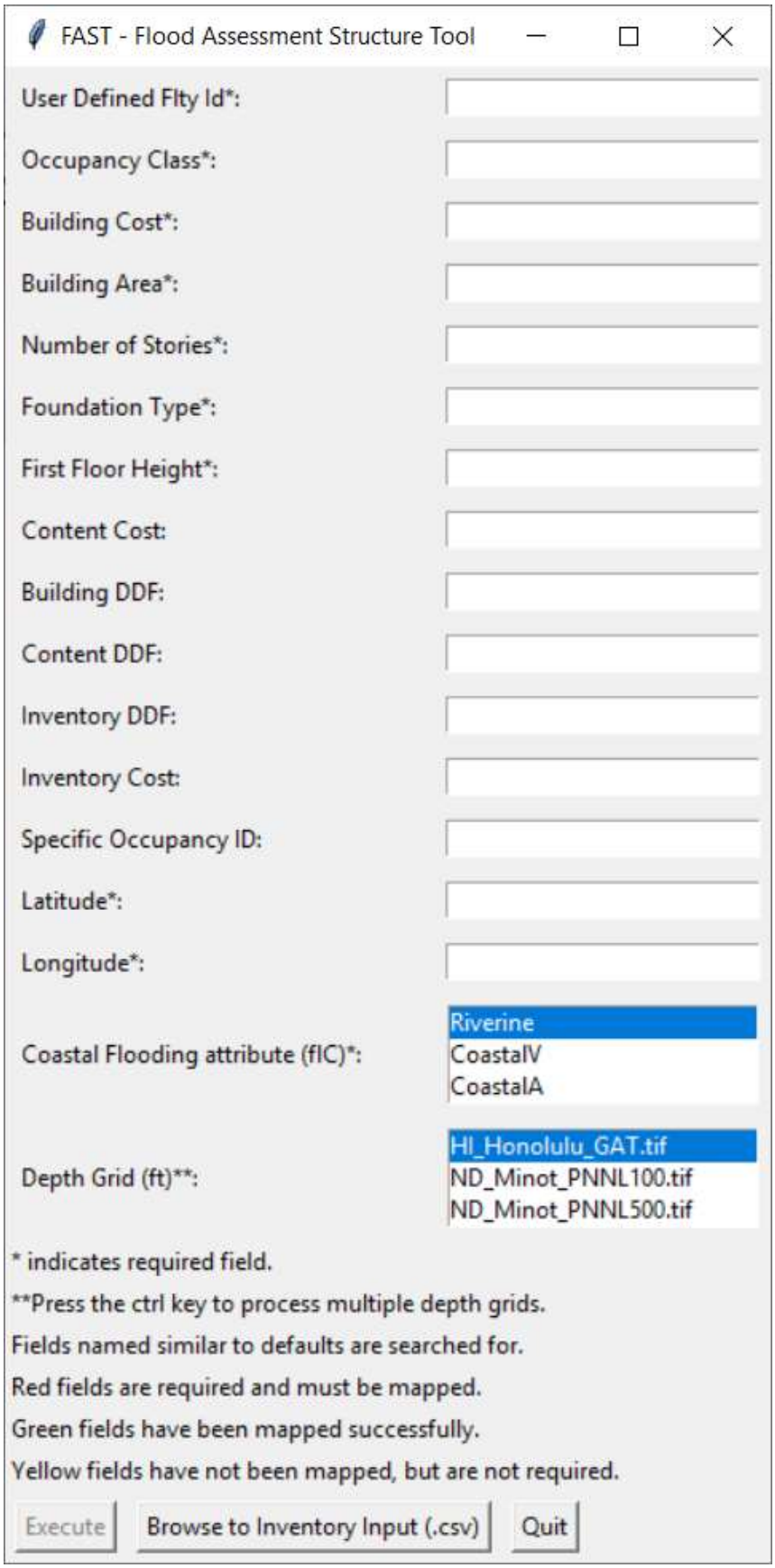
4. Output:

If the program runs successfully, you will find the final product in .csv file-format in the same location of the original input .csv file. The name should be the original name of the .csv file with an added _RASTERNAME to the end.

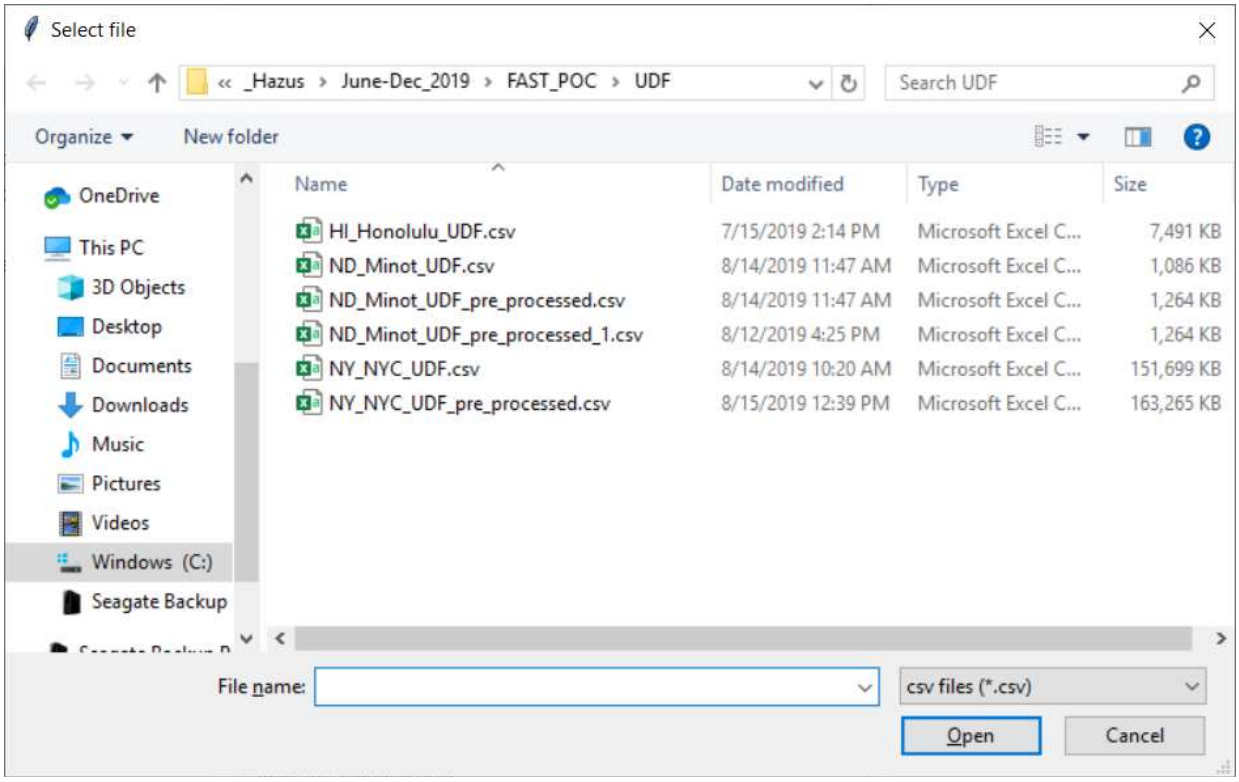
DEMO INSTRUCTIONS:

Open the program using FAST_POC.bat,

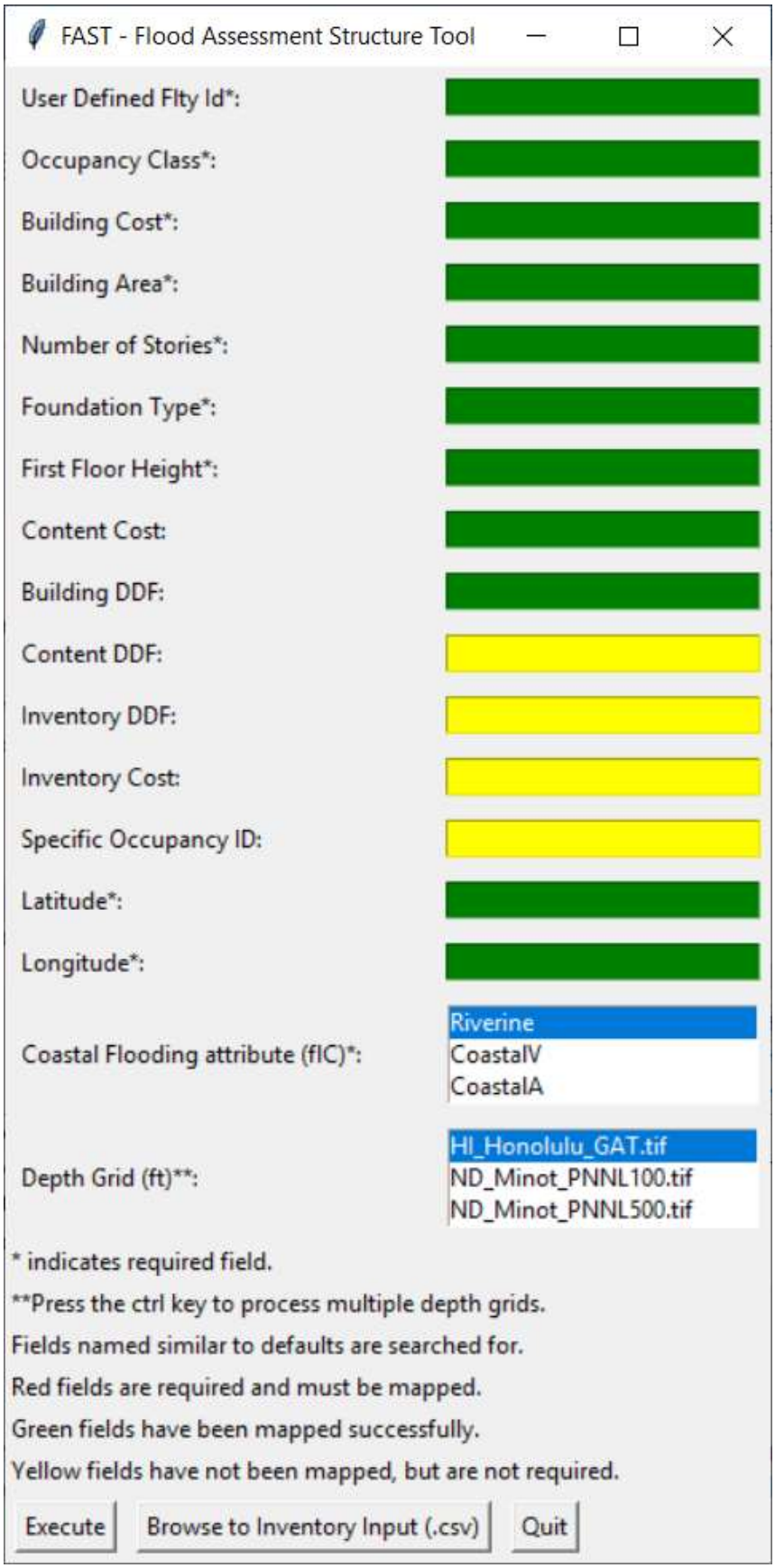
1. Leave all the entry text fields blank select the Honolulu_GAT.tif raster. The window should look like this:


FAST - Flood Assessment Structure Tool
User Defined Flty Id*:
Occupancy Class*:
Building Cost*:
Building Area*:
Number of Stories*:
Foundation Type*:
First Floor Height*:
Content Cost:
Building DDF:
Content DDF:
Inventory DDF:
Inventory Cost:
Specific Occupancy ID:
Latitude*:
Longitude*:
Coastal Flooding attribute (fIC)*:
Riverine
CoastalV
CoastalA
Depth Grid (ft)**:
HI_Honolulu_GAT.tif
ND_Minot_PNNL100.tif
ND_Minot_PNNL500.tif
* indicates required field.
**Press the ctrl key to process multiple depth grids.
Fields named similar to defaults are searched for.
Red fields are required and must be mapped.
Green fields have been mapped successfully.
Yellow fields have not been mapped, but are not required.
Execute
Browse to Inventory Input (.csv)
Quit

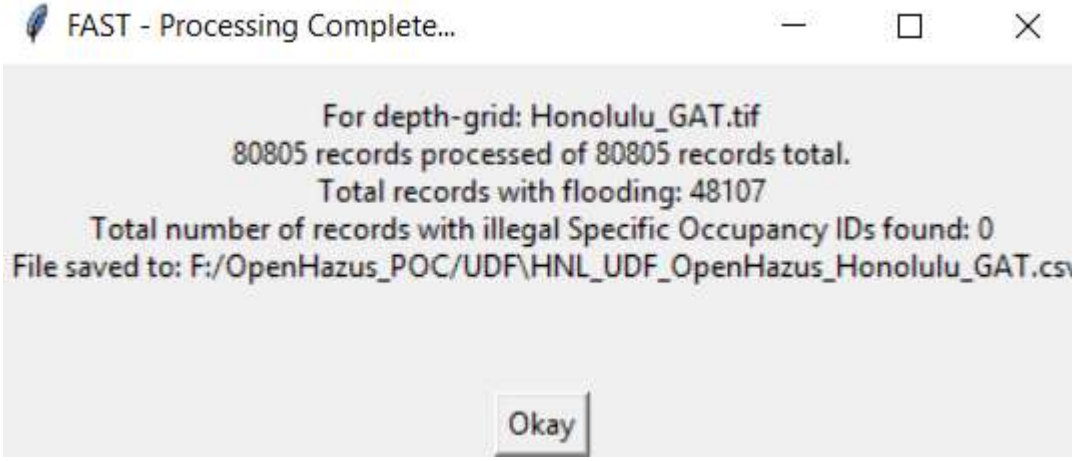
2. Press the "Browse to Inventory Input (.csv)" button and you will be brought to the UDF folder in the main directory; pick the HI_Honolulu_UDF.csv. It should look something like this:



3. Open it, the window should then look like this:


FAST - Flood Assessment Structure Tool
User Defined Flty Id*:
Occupancy Class*:
Building Cost*:
Building Area*:
Number of Stories*:
Foundation Type*:
First Floor Height*:
Content Cost:
Building DDF:
Content DDF:
Inventory DDF:
Inventory Cost:
Specific Occupancy ID:
Latitude*:
Longitude*:
Coastal Flooding attribute (fIC)*:
Riverine
CoastalV
CoastalA
Depth Grid (ft)**:
HI_Honolulu_GAT.tif
ND_Minot_PNNL100.tif
ND_Minot_PNNL500.tif
* indicates required field.
**Press the ctrl key to process multiple depth grids.
Fields named similar to defaults are searched for.
Red fields are required and must be mapped.
Green fields have been mapped successfully.
Yellow fields have not been mapped, but are not required.
Execute
Browse to Inventory Input (.csv)
Quit

4. Double-click on the HI_Honolulu_UDF.csv file, then press execute. After the program finished you should have a summary window like this:



Wait for the program to run (check the log window for any changes or errors), then go to the UDF folder for the new processed .csv file.

Troubleshooting:

If the required fields aren't found using either the given or default field names, the program cannot run. Send log info to the administration.

If a .csv file is not selected, the program cannot run.

NOTES:

Rasters used for processing must have coordinate systems of either Latitude, Longitude, or UTM. The input UDF must be in Latitude Longitude format.

Raster files and input .csv files must share the same coordinate system or the program will fail to run.

For example, if the raster file in question is using a UTM coordinate system, and the input .csv has fields mapped to latitude and longitude, the program will fail to run.

NOTE: Coordinate transforms may result in insignificant change to depth based calculations. For e.g. for the HI, Honolulu dataset here are the percent difference:

Small changes that occur as a result of the transformations. The most significant but still less than 0.3% is with UTM.

Honolulu		Diff			
$	14,611,808,813	0.0000%	Bldg losses, WGS84 geographic		
$	14,612,663,276	0.0058%	Bldg losses, NAD83 geographic		
$	14,568,790,444	-0.2944%	Bldg losses, UTMZ4N projected		

Enhancements made to the tool as of 8/15/19

1. Error logging and Messaging – The application stores basic log in the ../Log/app.log
2. Field names and aliases can be modified by the user in the file ../Python_env/guiprogram.py to match with the input data so that the user does not have to type them at each run.
3. RES2 pre-processing fixed (was not assigning DDF_ID)
4. The Browse button now opens to the UDF folder
5. The records with negative depths (from the raster) are not processed during loss calcs - Modified the DOGAMI code.
6. Title modified to FAST - Flood Assessment Structure Tool
7. The final message now displays - Total records processed, Total records that actually had flooding, (Total records that had unmatched occupancy Ids - CBH)
8. The name of the file that saves the results is now displayed on the final message
9. In addition to the normal results file there is another results file created with **_sorted** in the name which has results sorted on the Depth in structure field values
10. Special case handled for COM6 & RES2 building losses
11. Minimized string hard coding as much as possible
12. The tool appears on the center of the screen
13. The Quit button now closes the complete app
14. The .bat background window is now minimized
15. The messages on the main screen are left justified at the bottom of the page
16. Adjusted for max float processing over integer
17. Account for missing values in input UDF