System Design Document

For

TurtleTech

Team members: Christopher Etinas, Laurel Dodson, Parker Brown, RoseEllen Hoke, William Wiemann

Version/Author	Date
Version 1/ TurtleTech Team	7 October 2022

TABLE OF CONTENT

1 INTRODUCTION ... 3
1.1 Purpose and Scope ... 3
1.2 Project Executive Summary ... 3
1.2.1 System Overview ... 4
1.2.2 Design Constraints ... 4
1.2.3 Future Contingencies ... 4
1.3 Document Organization ... 4
1.4 Project References ... 5
1.5 Glossary ... 5
2 SYSTEM ARCHITECTURE ... 5
2.1 System Hardware Architecture ... 5
2.2 System Software Architecture ... 5
2.3 Internal Communications Architecture ... 6
3 HUMAN-MACHINE INTERFACE ... 6
3.1 Inputs ... 6
3.2 Outputs ... 7
4 DETAILED DESIGN ... 7
4.1 Hardware Detailed Design ... 7
4.2 Software Detailed Design ... 7
4.3 Internal Communications Detailed Design ... 7
5 EXTERNAL INTERFACES ... 8
5.1 Interface Architecture ... 8
5.2 Interface Detailed Design ... 8
6 SYSTEM INTEGRITY CONTROLS ... 8

SYSTEM DESIGN DOCUMENT

1 INTRODUCTION

1.1 Purpose and Scope

The purpose of this document is to describe the system requirements, operating environment, system and subsystem architecture, files and database design, input formats, output layouts, human-machine interfaces, detailed design, processing logic, and external interfaces.

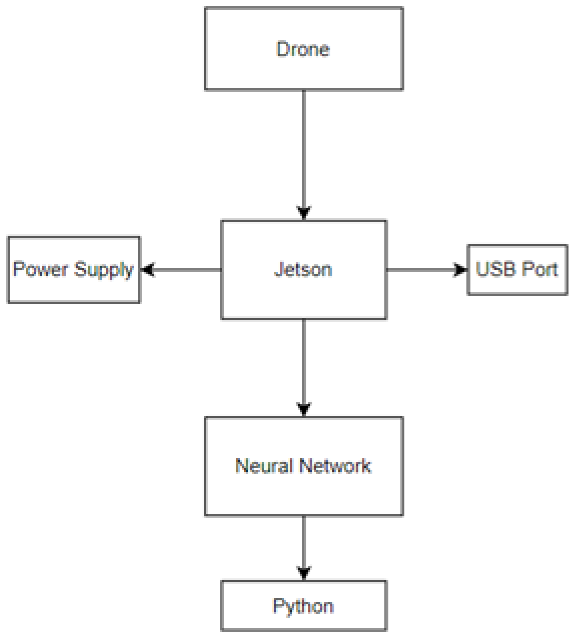
The purpose of the TurtleTech project is to produce a functioning neural network system on a Nvidia Jetson device that identifies aerial turtle images against non-turtle images in real-time. The neural networking system will only identify one specific animal at a time, it will not be able to differentiate multiple things at once. The ERAU team will produce a functioning neural networking system by the end of the Fall semester.

1.2 Project Executive Summary

TurtleTech is currently a project sponsored by Northrop Grumman and the Brevard County Zoo. Embry Riddle has been given the opportunity to partner and advance the TurtleTech project. The ERAU TurtleTech team's purpose is to create a neural network system that identifies turtles in real time while also helping troubleshoot the Jetson overheating problem. Currently the team is producing a lab procedure walkthrough on how to power on a jetson. The procedure manual will help educate and teach anyone that joins the TurtleTech project while also streamlining the amount of time it takes to learn how to get a jetson working.

The ERAU TurtleTech contributions are valuable to the project as the current state of the neural network is analyzing turtles after the fight is over. With a real time, neural network system, the scope of turtle hunting can be broadened to eventually even tracking turtle trails or even specific species of turtles. By the end of the Fall 2022 semester the ERAU team will produce a functioning neural network system,

1.2.1 System Overview



1.2.2 Design Constraints

The design constraints to the system are limited by the Nvidia Jetson Xavier. Those constraints are still being discovered as we get up to speed for the project

1.2.3 Future Contingencies

TurtleTech is a project created by Northrop Grumman and partnered with the Brevard County Zoo. This project is contingent on Northrop Grumman and the partnership with the zoo. The project is also contingent on weather, turtles in the area and the functionality of the hardware and software.

1.3 Document Organization

The organization of this document is broken up into six sections. The first section is the introduction to this project, which talks about the purpose, overview, and any constraints that the project has. The second section talks about the system hardware and software architecture. The third section talks about all the inputs and outputs within the human machine interface. The fourth section is an elaborated version of the design of the software and hardware. The fifth section is about the external interfaces within TurtleTech and the sixth section is about the system integrity of TurtleTech.

1.4 Project References

TurtleTech-erau. GitHub. (n.d.). Retrieved October 7, 2022, from https://github.com/TurtleTech-ERAU

Turtle Tech. Northrop Grumman. (2022, June 29). Retrieved October 7, 2022, from https://www.northropgrumman.com/sustainability/turtle-tech/

1.5 Glossary

- TurtleTech: sensor data collection technology for sea turtle conservation along the Florida coast.
- Nvidia Jetson Xavier: AI computer for autonomous machines, delivering the performance of a GPU workstation in an embedded module under 30W. Jetson AGX Xavier is designed for robots, drones and other autonomous machines.
- Python: high-level, general-purpose programming language.

2 SYSTEM ARCHITECTURE

2.1 System Hardware Architecture

The systems listed here are the ones that directly affect the performance and operation of the product:

- Jetson Nano:
  - This graphics processor is the main computer on which the product will run during flight. This also will be the computer that stores any image(s) saved by the product.
  - The onboard memory is 4 Gb, and it supports 4k footage at 30 fps.
- Aircraft power supply
  - For the current aircraft that the product is being deployed on a Censys Sentaero, the power provided for the product comes from the aircraft's main power supply. Along with the power supply is the voltage regulator, used to step down the voltage for the Jetson Nano to use. The power supply for the aircraft is 24 volts and is stepped down to approximately 12 volts for the Jetson processor.
- Camera (make and model)
  - We currently do not know what make or model the camera is, however, it is a crucial aspect of the product since it is the primary input.

In this section, describe the overall system hardware and organization. Include a list of hardware components (with a brief description of each item) and diagrams showing the connectivity between the components. If appropriate, use subsections to address each subsystem.

2.2 System Software Architecture

- Python 3.8
- Tensorflow 2.4
- labelimg
- python modules:
- virtualenv

- opencv-python

For the neural network to recognize the turles, we will use a python 3.8 environment. In this environment we will have pip installed the following packages: virtualenv, opencv-python, more to come. The “virtualenv” library is used to enter in and out of the virtual environment. We created a standalone python environment which the software will depend on, to avoid any additional complexity or issues related to running different python versions. OpenCV will be used in conjunction with Tensorflow. This allows us to add components of the OpenCV computer vision library. The “labelimg” software is a python program which is used to label images for the purpose of training and testing a neural network.

2.3 Internal Communications Architecture

The architecture of the system involves connecting to the Jetson Xavier via ethernet cable from a host laptop. This laptop will run a script, which will interact with the camera and signal to take pictures. The neural network will also be deployed to the target system in the same manner. Beyond that, the architecture is still being understood. The camera connects to the Jetson using some sort of cable. More to come

3 HUMAN-MACHINE INTERFACE

The system is intended to operate autonomously while the drone is in flight, but still receives inputs and generates outputs that will be of use to the operator when the flight has concluded and the captured data is reviewed. The inputs provided to the drone directly via the operator are the reference images, contained within a dataset, which train the neural network to find the expected criteria (in this case, identifying turtles). However, the datasets provided by the operator in testing will imitate, but are not exactly, the actual inputs that will be received by the system.

Instead, the main point of input capture is the onboard camera mounted on the chassis of the drone (model characteristics of the camera currently unspecified, to be clarified in the future) and the processing of said input occurs within the neural network algorithm on the Jetson chipboard, which is carried in the cargo compartment of the drone. The images captured by the camera are transferred to the Jetson via wire connection, and post-processing, the data on the Jetson is a very reduced set of photos containing potential turtles, with measures of certainty of analysis contained along with the corresponding photos. Thus, the images (dataset and captured) sent to the Jetson board are the inputs for this system, and the categorized and analyzed images showing turtles are the outputs of the system. This is described in greater detail below.

3.1 Inputs

The input media provided by the operator is training data for identifying specific attributes of a target (currently a turtle) within a larger picture. This is done by providing a dataset of images known to contain a turtle, with the aspects of the visible turtle emphasized for the network. The neural network breaks the entire image down into smaller sections, those containing relevant-to-the-target portions and those portions of the image that do not contain the target information, and records data about what aspects of a photo containing a turtle can be used for identifying the turtle. As the algorithm is fed more images in its dataset, it 'learns' criteria that are shared across varied turtle images, so that when it is given a non-dataset image, it can break it down and look for those same criteria in order to identify if a turtle is contained within the image or not. The essential inputs for the system function, thus, are an extensive database of images with a selected target present in each, and input images that may contain the target turtles from the in-flight camera captures.

3.2 Outputs

The output of the system we are creating will be identification of turtles from given images. The drone camera will take images as it travels in flight above the ocean and beach, and the images of the beach and sea will be transmitted over wire to the Jetson board for processing. After processing has occurred (intended to happen in-flight, which will require optimizing the neural network to run on less memory and processor capabilities) the outputs will be a much smaller set of sorted images, each with a selected (highlighted or boxed) section showing a high certainty of a turtle identified within the image. Depending upon how the team designs the system, some non-turtle images will also be saved for comparison purposes, but many of the images without the targets will be discarded for the purpose of saving space. In addition, the outputs may be marked with timestamps and location data if that information is available to do so. The most critical aspect of the outputs generated from the system, however, is the correct identification of turtles from the drone-captured in-flight images showcased in the output images.

4 DETAILED DESIGN

4.1 Hardware Detailed Design

The hardware design of the system is already near fully implemented, with only a few issues to sort out. The components as stated before are the power supply, voltage regulator, Jetson Nano, and camera. We currently do not have a detailed hardware design, but electrical members of the team will be going through and documenting the current system, and recording any improvements made. The one small detail that we currently know is that the Jetson requires a minimum voltage of 12V to operate.

4.2 Software Detailed Design

A software module is the lowest level of design granularity in the system. Depending on the software development approach, there may be one or more modules per system. This section should provide enough detailed information about logic and data necessary to completely write source code for all modules in the system (and/or integrate COTS software programs).

4.3 Internal Communications Detailed Design

The package, which includes the Nvidia Jetson Xavier and the camera used, is a system that has already been built. The communication systems for those devices have been created and we will not be designing anything additional for communications at this time. However, the software components will need to be able to speak with each other. The current state of the software runs a complete script, which will activate the components needed. But ideally, we wish to create a Python program, which will contain some system() commands, that will be able to communicate to each component effectively. This is the ideal form, as the former feels very segmented

5 EXTERNAL INTERFACES

5.1 Interface Architecture

The jetson is connected to the hardware of the drone through a USB port.The jetson will be powered by a battery installed into the drone. Images with embedded text are saved to the SD card in the Jetson. An ethernet cable is used to interact with the jetson through a PC. An HDMI cable is used to display the images captured.

5.2 Interface Detailed Design

Images which are captured through the drone will be processed through the neural network which uses Python to implement the image recognition program.

6 SYSTEM INTEGRITY CONTROLS

- The internal security of the device is restricted for outside users while the team members have full access to the device.
- The audit procedure will be conducted to ensure there are no risks that can be exploited and if there is a risk then an assessment of the risk will be conducted.
- The assessment will list the risk, the level of severity, and how the risk will or will not be fixed.
- The assessment will follow NIST 171-800 standards
- The Jettson is secured by a username and password
- The Microsoft teams can only be viewed by people with the link