

CSCE-585 Project Proposal

Visual M.L. System for Identifying and Minimizing Civilian Fatality in Urban War Zones

Logan Nall | Walter Pach | Vera Svensson

<https://github.com/csce585-mlsystems/MilCiv>

Problem & Interest

As A.I. systems are increasingly deployed in military contexts, there is a need to enable targeting capabilities to identify and distinguish between civilian and military vehicles and actors. This is especially true in cases of urban conflict where the availability of intelligence on location may be unavailable, and evacuation may not be efficient. Due to the efficiency of A.I. systems in enhancing human targeting officers, it is crucial that the models be trained and implemented using thorough data sets. With the increase in urban conflict in previous years, first in the middle east and now with the emerging war in Europe, it is essential that this technology be implemented to save civilian lives and prevent risk.

Background/Related Work

Convolutional Neural Networks (CNNs) can be deployed to recognize patterns in image data, including implementations for object detection. In combination with Extreme Machine Learning (EML), the accuracy of these models can be improved to upwards of 95%. This can be combined with a layered approach to distinguish between different categories of relevant objects, using a multi-level architecture. This is especially pertinent when training models to distinguish between different classifications and refining this to subclassifications later in the process.

Data

Our project relies on the use of a labelled data set of images depicting both military and civilian vehicles. This data set will contain helicopters, drones, airplanes, and land vehicles. The data must be prepared to a uniform input size, maximizing the efficiency of the model and the storage required to train it. This data set is called "Military and Civilian Vehicles Classification" and is available as FOSS, containing 6,772 images. As development continues, we may be required to supplement this data set with additional content or labels.

Methodology & Integrations

Our methodology is to place simplicity at the forefront of our development. Beginning from a single-function, multi-layered approach, we will train small models to perform object detection on specific components. This will be done using the Python TensorFlow library. We will work iteratively in a spiral development structure, employing the philosophy of working samples of our code, iteratively meeting project requirements.

Regarding the design of the system, we will first create a boilerplate for sanitizing and preparing the data and training a model, create a testing environment, determine efficiency metrics and required improvements, and then training the production model to identify the civilian/military vehicles. This will be implemented and deployed in a Jupyter Notebook, allowing for the display of graphs and statistical information alongside the models' object detection results.

Evaluation Methods

When testing our model, we plan to identify two metrics for performance: 1) accuracy, and 2) precision. Accuracy will be measured by the performance of the model at identifying vehicles as civilian or military and the rate at which failure occurs. Following the improvement of this, precision will be measured and improved by training the model's confidence about these classifications. When testing our model, we will use automated testing and a portion of our data set to test the model. We also plan to use novel images and assess its performance. These two tests will provide both statistical probabilistic measurements for the expected performance and discrete data points from which to generate visuals depicting actual performance.

Limitations

- Model predictions are limited to specific models and styles of vehicles, posing a problem for reliability
- Does not replace the human calculation and evaluation to make a final decision