Friend or Foe: Multi-Modal Military Target Identification

Andrew Jeon • Bassam Halabiya • Naif Ganadily • Zachary Saunders

EEP567 Final Project Checkpoint Report

Milestones/Preliminary Conclusions

As documented in our proposal, the following key milestones were identified: The creation and annotation of a specialized dataset, selection of machine learning model, initial AI model training/testing results, and the development of our Challenge Response System. Not only have these milestones been met, but we have also invested time in further testing and experimentation.

We successfully gathered 1,100 images equally distributed between the Russian, USA militaries and categorized them into four main branches: Army, Navy, Air Force, and Marine Corps. By utilizing data augmentation techniques with Roboflow we were able to expand our dataset to 2,652 images. This step was essential in enhancing the robustness of our model by introducing a range of variations in the training data.

After preparing our dataset we moved on to deploying our machine learning model through transfer learning. This stage involved coding, initial training, and deploying several versions of the model on Roboflow. We experimented with different models including YOLOv8n, YOLOv8s, YOLOv8m, YOLOv8l, YOLOv8x and YOLOv8seg. We also experimented with hyperparameter tuning such as epoch count, augmentations, optimizers, thresholds, etc. All of this said and done, we achieved a very decent mAP-50 and mAP50-95 which can be roughly used to judge the model performance. We will hold off on providing exact statistics until the final presentation where we can present a comprehensive overview. However, Figure 1 in the appendix, shows an initial detection result presented in a batch image.

In conjunction, with developing our machine learning model we designed an additional authentication method based on a challenge/response system using software defined radio (SDR) technology and directional antennas. The objective of this system is to add a layer of verification, which in turn improves the overall reliability of target identification. This system has been developed and has achieved basic functionality which is progress towards creating a multi modal identification system that combines visual and radio frequency (RF) signals for validating targets. The physical architecture of this system involves an Enhanced Weapon System (EWS) and a modified soldier uniform with an embedded receiver/transponder system. In practice, the ultimate setup will utilize a Software-Defined Radio (SDR) mounted on the EWS, however, our current demonstration employs a socket-based server-client connection for proof of concept. It’s worth noting that while the current CRS demonstration is standalone, the intended use of this system will be integrated with the machine learning model.

Bottlenecks

Our initial results for the machine learning model suffered from a few setbacks primarily due to issues with the dataset quality. The team was able to sufficiently acquire images for the US military branches due to the abundance of those images, however, collecting Russian military images proved to be far more difficult. This difficulty arose from a combination of duplicate and obscure Russian military pictures available online, compounded by misalignment between Russian military branches and their US counterparts. Additionally, in the case of Russian Marines and Navy particularly, Russia does not have a formal marine corps, instead, they have what is referred to as “Naval Infantry”. Therefore, a significant portion of our dataset in RU_navy as well as RU_marines include images from “Naval Infantry”.

Further exacerbating the dataset quality issue is the fact that we had a few blurry images which were later processed using data augmentation techniques that added noise and blurriness. It was also noted that using 8 classes to represent the custom dataset may have been overly ambitious given the fidelity of the available Russian military images. Another potential issue is that in our data labeling, we optimized for completeness, ensuring each soldier of each division was properly identified, no matter how little of their body was visible. On the surface this seems well intentioned, as if only a soldier’s head is in view, that head would be enough for a soldier to fire upon. However, let us consider the visual differences between a US soldier wearing no cap and a Russian soldier wearing no cap; how can you identify one from the other only looking at the features of their face? With each military faction full of individuals of different ethnicity, age, facial features, hair, and other variables, a face alone is not enough information to base a classification on. This is true of many partial body parts. A partially visible leg with no camouflage, a hand holding a firearm, a thigh peeking between obstructions, are all insufficient as a base for classification. Therefore another potential area of improvement could be going through and removing labeling that fails the question: when looking at this element in isolation, does this alone prove that this soldier is from X faction?

All things considered; the first few versions of the model appeared to have unreliable classification results. Fortunately, in using Roboflow to deploy our custom datasets and trained computer vision model, we were able to access various prognostics which include precision/recall/mAP results, confusion matrix, and training graphs. Said prognostics allowed the team to investigate the issues described above and address them accordingly. For instance, the custom dataset classes were consolidated into 2 classes to simplify classification and focus on optimizing the machine learning model. Additionally, subsequent iterations of the model leveraged the confusion matrix and dataset distribution graphs to address the dataset quality issues by refining the Russian military images and changing the data augmentation techniques which improved the overall efficacy and reliability of the model.

Additional Experiments

As work continues in the project we will further experiment with additional data augmentation, and hyperparameter tuning to increase robustness and generalizability of our model.

Fine-tuning with Transfer Learning - To attempt to address some of the challenges associated with dataset quality and volume, we could explore the effectiveness of transfer learning by leveraging a pre-trained model trained on another large-scale dataset, similar to COCO or other military camouflage datasets to increase dataset size and generalizability. We can then further fine-tune on our specific soldier/friend or foe classification task. This could potentially increase model performance.

Adversarial Training - To enhance the model's robustness against adversarial attacks and improve its ability to generalize to unforeseen variations in data, we could employ adversarial training techniques. We could generate adversarial examples using methods like Fast Gradient Sign Method (FGSM) and add them to our training dataset. Adversarial training encourages the model to learn more robust features and improve its resistance to perturbations. Through these additional experiments, we hope to address some of the challenges identified in the previous sections and increase real-world efficacy and resilience.

Complete overhaul of the dataset - As previously mentioned, we may remove all instances of partial body part annotation, blurry images, and outlier/rare camo uniforms for either faction. This may potentially improve model performance.

Friend or Foe: Multi-Modal Military Target Identification

Andrew Jeon • Bassam Halabiya • Naif Ganadily • Zachary Saunders

EEP567 Final Project Checkpoint Report

Appendix

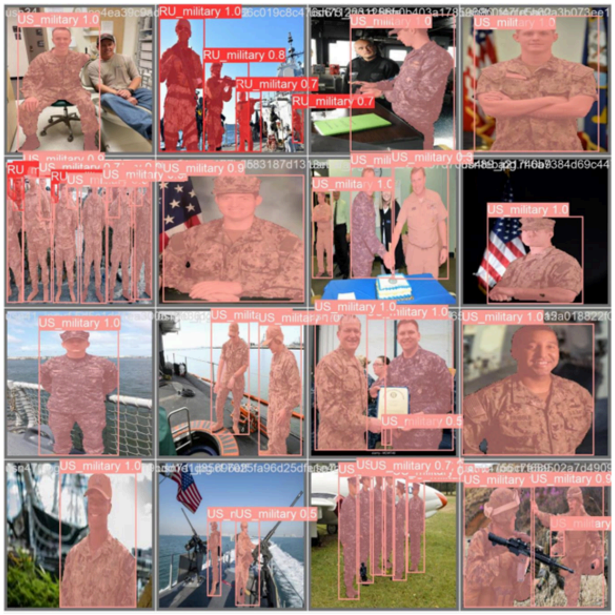
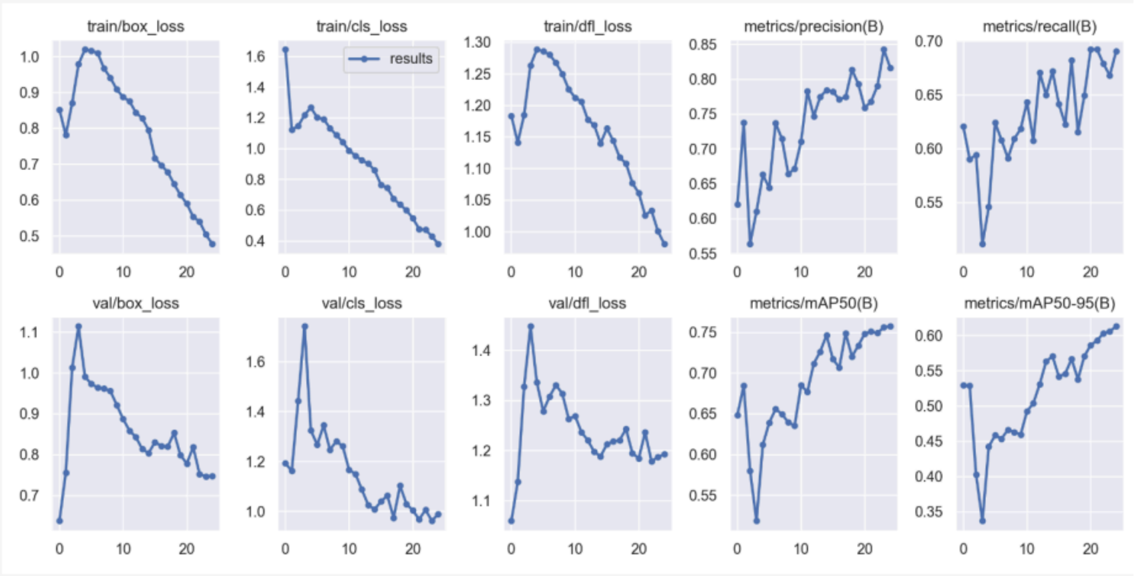


Figure 1: Model predictions on validation data with confidence scores.



Friend or Foe: Multi-Modal Military Target Identification

Andrew Jeon • Bassam Halabiya • Naif Ganadily • Zachary Saunders

EEP567 Final Project Checkpoint Report

Figure 2: Training/Validation Loss charts for YOLOv8x (obj)

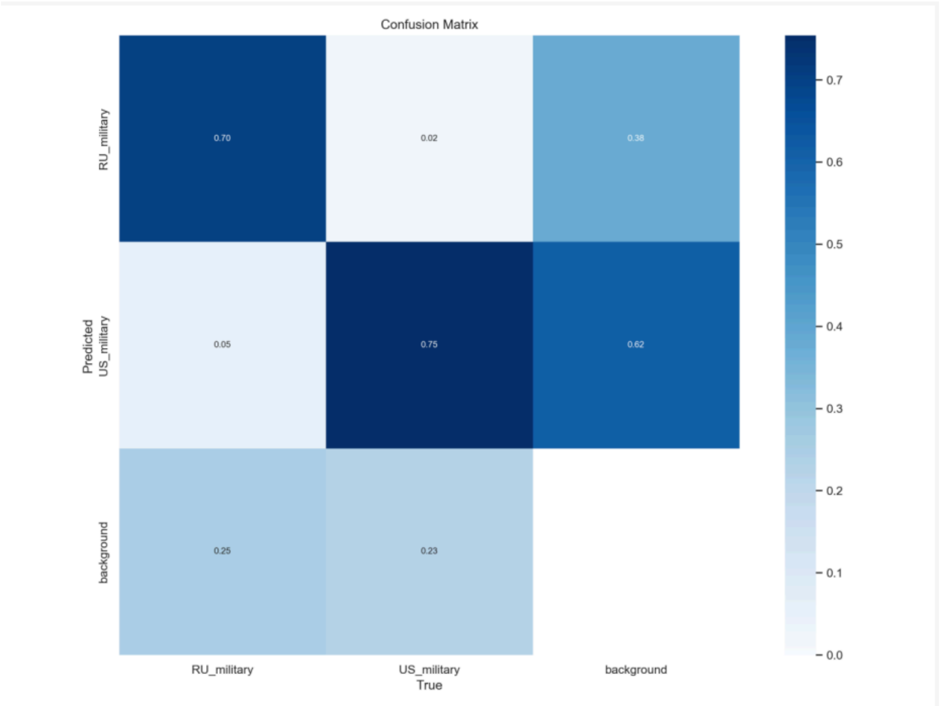


Figure 3: Confusion matrix for YOLOv8x (obj)