Machine Threat Detection: Learning to Spot AK-47s in Images

Milestone Report
Jacob Jurmain

Abstract & Rationale

This project's goal is to use a latent SVM (LSVM) on a Histogram of Oriented Gradients (HOG) representation of diverse images of a target object, in this case an AK-47. The specific HOG formulation is described in [1]; the LSVM is described in [2].

The application concept is that security cameras and military and law enforcement drones should be able to autonomously detect threats such as carried weapons and alert humans. This would reduce the amount of attention humans would need to pay to watching a screen. In security cameras, this would save guards the often uncomfortably dull task of maintaining constant vigilance despite a very low rate of threat occurrence. In drones, this would allow the operator, who is often in a combat situation, to keep his or her attention on threats to their person.

System Architecture & Operation

The LSVM is trained on HOG representations of AK-47s cut out from their surroundings, and encoded with appropriate resolution to ensure they all measure m cells high by n cells wide. The training set will include each image's z-values, i.e. the coordinates of the upper-left corners of their part models. The part filters are also trained on selections of uniform cellular size, but at double the resolution of the top-level filter. Negatives and "hard negatives", i.e. potential false positives, will be included in the training set. Hard negatives are necessary to ensure a convex optimization problem [2].

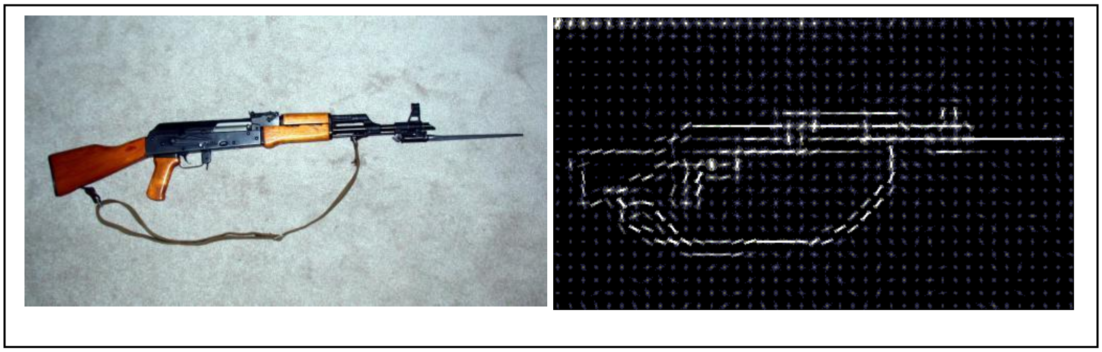
The LSVM is tested on an image which may or may not contain an AK-47 somewhere within at some unknown scale. To overcome these uncertainties, the image is encoded as a pyramid of HOGs, i.e. a series of HOGs of the image at increasing/decreasing resolutions; these form the "levels" of the pyramid. The trained filter scrolls through each possible position at each level of the pyramid. When it scores a location to a threshold, it engages a local scan with the part filters and scores them at each of their possible locations. The best-scoring location of the image is returned as a positive detection if it passes a threshold.

Both the training and testing algorithms are computationally expensive. Too expensive in fact to be implemented in real time by anything less than a supercomputer or server farm. Use in drones will likely remain a dream until quantum computers except perhaps as a minor component of a less expensive system.

Results

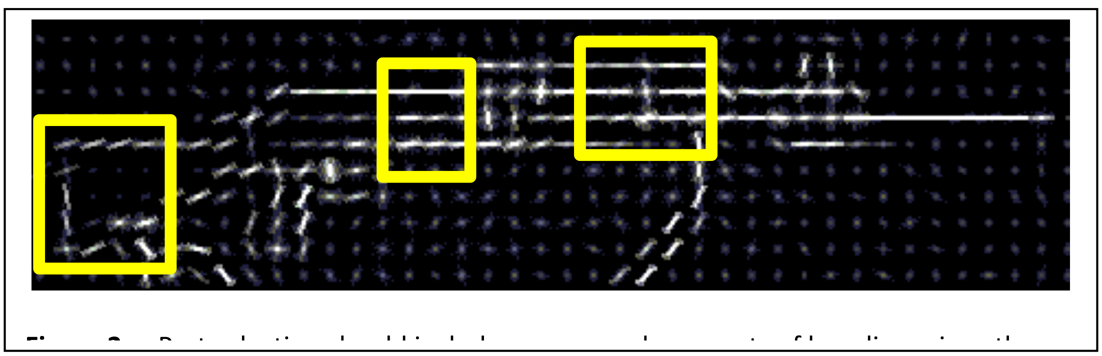
At this point only the HOG image encoding has been coded. Several useful insights have been derived from it however. Figure 1 presents a visualization of its output. The HOG itself and the visualization were generated entirely with original code.

The first major insight is that small detection windows will be critical to minimizing computational expense, but the fact that AK-47s may rotate in the plane of the image makes the detection window necessarily a square with side length the largest dimension of the object. It may be more efficient to use a classifier for vertical orientation and one for horizontal orientation and let them each only rotate by 45° instead of 90°. That leads into the second insight, which is that unlike many object detection



problems LSVMs have been applied to, AK-47s are not left-right symmetrical. It may be necessary to in fact use four classifiers, one in each direction of possible orientation. To avoid actually training four classifiers for the same object, this project could, with instructor approval, constrain this project to pick just one direction to work with. Images in the training and test sets could be manually rotated to place them within the classifier's angular detection range.

The second major insight is that AK-47s may rotate perpendicular to the plane of the image. In this case, they will appear to squeeze and stretch as they are rotated. To minimize the issues this will cause the filter, model parts should be chosen which will remain relatively unchanged by this effect, yet will still be distinct and common to the class. Specifically, parts should be corners and segments of long lines (figure 2a) and complex structures (figure 2b).



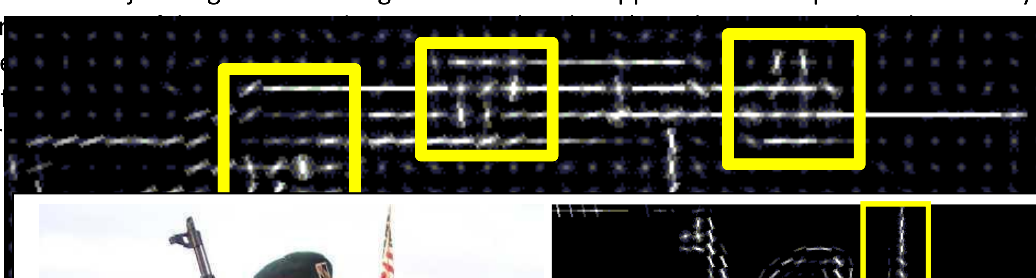
The third major insight is that images realistic to our application concepts will have many features

which r ... se

positive ... will be

very di ... an

interpr ...



A third, lower-level insight is that the minimum detection window appears (according to calibrated eyeball) to be roughly 18 cells along the long axis of the object and 5 cells along the short axis (figure 4a). Testing of this assertion will of course have to wait for the completed system; it is only based on the subjective judgment that lower resolution causes the object to lose easy recognizability to human perception (figure 4b). Note that the detection window will need to be larger than 5 cells high to allow for in-plane rotation.

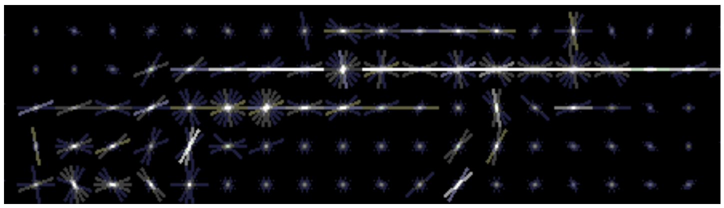
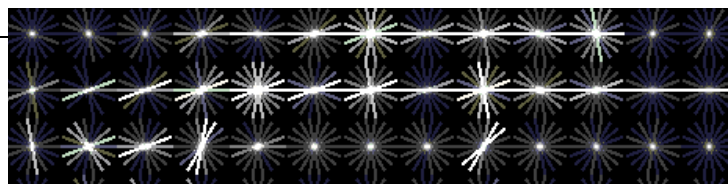


Figure 4a: AK47 at 18 x 5 cells (24^2 pixels/cell).



Conclusion

Progress on this project has been slow. Two contingency plans are in place. The primary is to use varying quantities of off the shelf code and interface between it and MATLAB via text files due to difficulties encountered in calling C functions from MATLAB. The secondary (less likely to be necessary) is to assume the object will only be observed in an ideal orientation and forgo the use of part models. Since this would not meet the project requirement to use an algorithm not covered in class it will be avoided unless the alternative is incompleteness.