Mini Project: Weapon Detection

Pritam Dutta (A0248400N), Mario Michelessa (A0231837A), Lalitha Ravi (A0268254X),
Anocha Sutaveephamochanon (A0268230J) and Roxana Zayeni Langroudi (A0256808U)

CS5242 2022/23-2, School of Computing, National University of Singapore

Abstract

This work has explored the object detection task using Deep Residual Learning for Image Recognition (ResNet), ResNext and Contrastive Language-Image Pre-training (CLIP). The comparison between the results is interesting and demonstrate that the highest performance levels with respect to various metrics are achieved using different models.

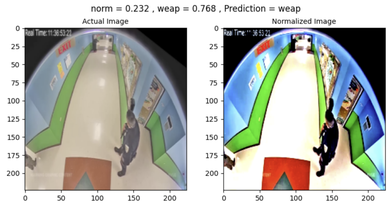


Figure 1. An example result from a fine-tuned ResNet-50 with 2 additional layers model.

Data Preparation

The data is mainly extracted from online videos, and scrapped online. The training dataset consists of 21,414 images and 4,053 images for testing. The training data is split into 80 to 20 percent for training and validation, respectively. The training dataset is slightly unbalanced (36% norm, 74% weap), and the test dataset is balanced. The models are trained and tested on different computing units due to the limitation of computing resources.

Pre-processing steps:

- The image pixels in the dataset are converted to the range of [0, 1] and then normalized for higher contrasting input by computing the Mean and Standard Deviation of the training images.
- We avoided cropping the images to not remove any weapon from the images.
- We convert all images to the size 224 x 224.

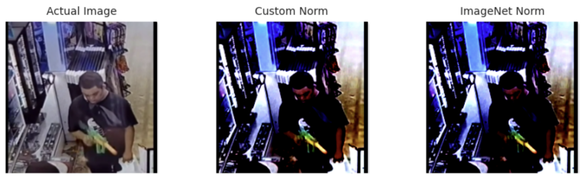


Figure 2. Example images from the training dataset where left column shows the actual image, middle column shows normalized image using derived mean and std and right column shows normalized image using ImageNet mean and std

ResNet

ResNet (Residual Network) is a deep learning model based on CNN architecture, used for computer vision applications. The key thing to note about ResNets is that they take care of the "vanishing gradient" problem, which is extremely common in very deep CNN models through the use of "skip connections".

We used the ResNet-50 model, which is pre-trained on more than a million images from the ImageNet dataset, as one of the baseline models for this project. The fully-connected layers of the pre-trained model were replaced with two linear layers - a hidden layer of dimension 2048 x 128 and an output layer of dimension 128 x 2. We also use a ReLU activation function and a dropout of 0.50 between these two layers.

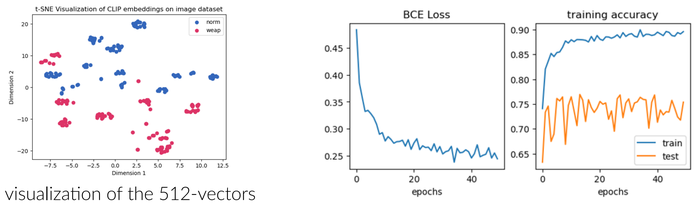
ResNeXt

ResNeXt is an Aggregated Residual Transformations (called in original paper) which exposes a new dimension called "cardinality" in the size of the set of transformations to be an essential factor in addition to the dimensions of depth and width. The original ResNet-50 and ResNeXt-50 are in similar capacity, but ResNeXt yeilds better performance.

CLIP

CLIP is a transformer-based model trained on billions of pairs of (image, caption), that is optimized to minimize the distance between the image and its caption and maximize the distance between other captions, in an unsupervised manner. The image and the caption are embedded into a latent space, the embeddings of the image can be retrieved and used to classify whether there is a weapon on the image or no. Two different methods are explored :

1. **CLIP-MLP/CLIP-SVM :** Estimate the representation of all images in the latent space, and then train an MLP or a SVM on images embeddings (openai/clip-vit-base-patch32).
2. **CLIP Vanilla :** Estimate the similarity of images between two prompts : *"a photo of a person with a weapon"* and *"a photo of a person"*. Higher similarity with the first one means that the image is classified as weapon.



(a) t-SNE visualization of the 512-vectors clusters, colored by class. The weapon and normal images seem separable.

(b) Training curves of CLIP-MLP, a MLP trained on the images embeddings.

Figure 3. caption

Models Explanations (XAI)

To verify if models use the presence of weapon in the image, saliency methods are applied to the models' predictions. Saliency methods determine which pixels are important for the models prediction. The two methods used here are GradCAM and SmoothGrad, which are complementary.

Figure 4. Saliency maps for 4 images from the test dataset. TP : The model mainly uses the barrel of the weapon. FP : The model erroneously predicts a weapon when persons stands far from the camera, or have their hands close to their belts. FN : the model focused on the unarmed subjects, which leads to a wrong prediction. TN : The true negatives with the highest confidence often contains people having their hands far from the belt.

Results

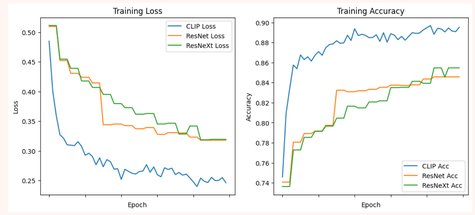


Figure 5. Training accuracy and loss during training steps. Left is depicted when training loss, and right is from accuracy.

Model	Accuracy	Precision	Recall	F1-score
RestNet-50	0.740	0.725	0.836	0.777
ResNeXt	0.728	0.693	**0.892**	**0.780**
Ensemble	0.7538	0.7615	0.7928	0.7768
CLIP-MLP	**0.754**	**0.936**	0.665	0.777
CLIP-SVM	0.732	0.923	0.646	0.760
CLIP Vanilla	0.715	0.886	0.637	0.741

Table 1. Final results table.

- **ResNeXt** performs best both in terms of recall and f1-score. Choosing a model with a high recall is important in this situation, as any false negative could have dramatic consequences. Moreover, when we created an ensemble of ResNet and ResNeXt called Ensemble, the accuracy and precision are higher than the individual models themselves, but lower recall.
- Interestingly, **CLIP-MLP** performs the best in terms of accuracy and precision, while having a much shorter training time than ResNet-50 and ResNeXt model.
- **CLIP Vanilla** performs relatively well for a model that has not been trained on the dataset. CLIP is a good alternative to ResNeXt if fine-tuning the model is impossible.

Exploring other models

YOLOV5 is a compound-scaled object detection model. It uses a single neural network to process an entire image. The image is divided into regions and the algorithm predicts probabilities and bounding boxes for each region. We trained it on a dataset of Pistols to see the results we could get on our dataset.

(a) Detection on a rifle

(b) Detection of a pistol

Figure 6. Detection examples

Conclusion

This work compares the performance of 5 models, reaching a maximum f1-score of 0.784 for ResNeXt. In forthcoming studies, we could explore data augmentation methods in greater detail, and test our methods on servers with higher resources.