Threats Detection for Airport Body Scan Images

Krongrath Suwannasri (krath) and Chayawan Jaikla (cjaikla)
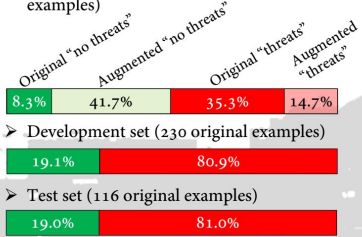
Motivations

- Strengthen algorithm accuracy to reduce false alarms and improve traveler safety
- Reduce processing time compared to the 3D CNN
- Speed-up the passenger screenings at the airport
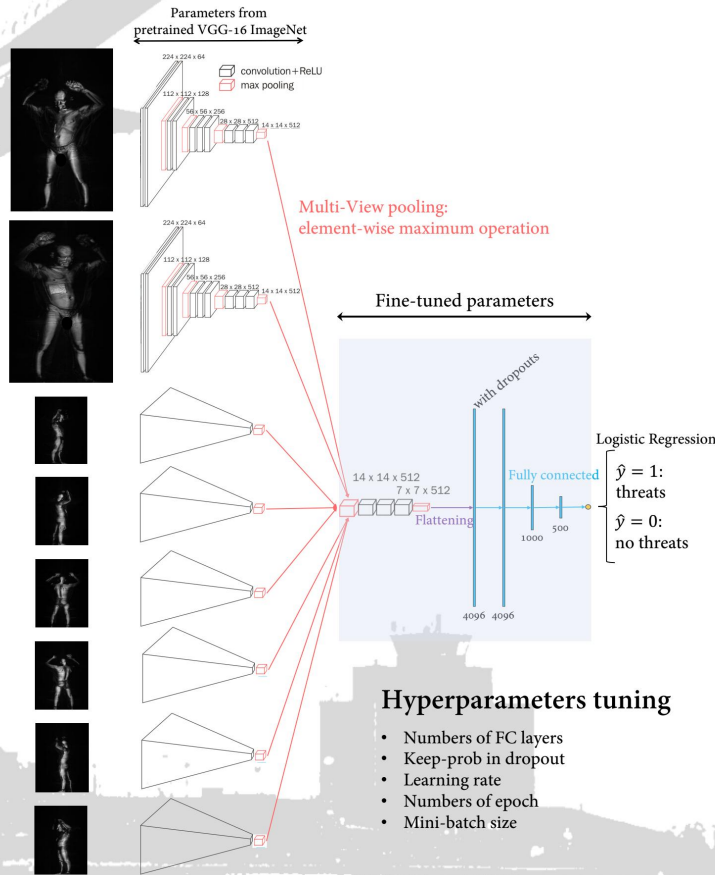
Related studies

- Multi-View Convolutional Neural Networks (MVCNN)
  - 3D shape recognition (Su et al., 2015)
  - 3D shape retrieval from ShapeNet Core55 (Savva et al., 2016)
  - Cancer screening (Geras et al., 2017)

- Pretrained CNN model
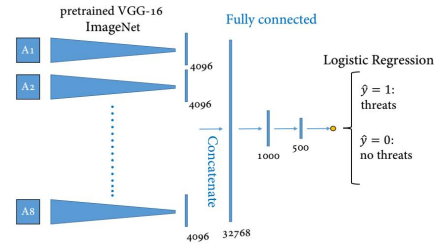 - VGG-16 on ImageNet (Simonyan and Zisserman, 2015)

Data set

- Originally 1147 examples of full body scan images
  - 219 (19%) of "no threat" examples
  - 928 (81%) of "threat" examples

- Data augmentation using combinations of:
  - Shifting 20-100 pixels
  - Sharpening and brightening

- Final data sets:
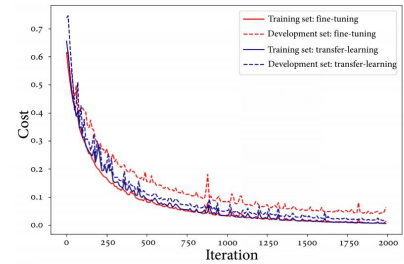 - Training set (801 original and 1035 augmented examples)

Original "no threats"	Augmented "no threats"	Original "threats"	Augmented "threats"
8.3%	41.7%	35.3%	14.7%

  - Development set (230 original examples)

19.1%	80.9%

  - Test set (116 original examples)

19.0%	81.0%

Fine-tuning Multi-View CNN (MVCNN)

Parameters from pretrained VGG-16 ImageNet

224 x 224 x 64
112 x 112 x 128
56 x 56 x 256
28 x 28 x 512
14 x 14 x 512

convolution+ReLU
max pooling

Multi-View pooling: element-wise maximum operation

Fine-tuned parameters

with dropouts

14 x 14 x 512
7 x 7 x 512

Flattening

Fully connected

4096   4096   1000   500

Logistic Regression

$\hat{y} = 1$: threats

$\hat{y} = 0$: no threats

Hyperparameters tuning

- Numbers of FC layers
- Keep-prob in dropout
- Learning rate
- Numbers of epoch
- Mini-batch size

Transfer learning MVCNN

pretrained VGG-16 ImageNet

Fully connected

A1   A2   ...   A8

4096   4096   4096

Concatenate

32768

1000   500

Logistic Regression

$\hat{y} = 1$: threats

$\hat{y} = 0$: no threats

Learning curves

- Training set: fine-tuning
- Development set: fine-tuning
- Training set: transfer-learning
- Development set: transfer-learning

Predictions

Evaluation	Fine-tuning MVCNN	Transfer learning MVCNN
Training accuracy	99.7%	100%
Dev accuracy	97.8%	100%
Test accuracy	95.7%	99.14%
Test recall	98.9%	100%
Test precision	95.9%	98.95%
Test f1 score	97.4%	99.5%

The transfer learning is slightly better than fine-tuned MVCNN on this dataset