AI-based Pilgrim Detection using Convolutional Neural Networks

MARWA BEN JABRA, ADEL AMMAR, ANIS KOUBAA, OMAR CHEIKHROUHOU, HABIB HAMAM
Plan

- INTRODUCTION
- ALGORITHMS BACKGROUND
- THE PILGRIMS DATASET
- EXPERIMENTAL EVALUATION
- CONCLUSION
- PERSPECTIVES
INTRODUCTION

- The safety and security of pilgrims is the highest priority for the authorities.
- CNN to detect and identify Pilgrims and their features.
- Dataset for the detection of pilgrims and their genders.
- Two CNN models based on YOLOv3 and Faster-RCNN.
ALGORITHMS BACKGROUND

**YOLOv3** is a one-stage detector that is known to be the fastest detection algorithm.

**Faster R-CNN** is an improvement of R-CNN that represents the most efficient region-based CNN algorithm for image detection.
Faster R-CNN

- The RPN module generate the region proposals.
- The Fast R-CNN detector:
  - The extraction of feature vectors from the region of interest.
  - The feature vector obtained is the input of the classifier

The classification output are:
- A sequence of probabilities estimated of the different objects considered.
- The coordinates of the regions proposals.
YOLOv3

- Multi-label classification based on logistic regression.
- Cross-entropy loss function.
- The prediction of bounding boxes.
- The concept of Feature Pyramid Network for the prediction.
- Darknet-53 CNN features extractor.
THE PILGRIMS DATASET

Pilgrim, Not Pilgrim classes designate a male

Woman classes with no additional feature
THE PILGRIMS DATASET

- 622 images of people in the holy places of Makkah and Madinah.

- Using the LabelImg software, we labeled the collected dataset into three chosen labels.

<table>
<thead>
<tr>
<th></th>
<th>Training</th>
<th>Testing</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of images</td>
<td>560</td>
<td>62</td>
<td>622</td>
</tr>
<tr>
<td>Number of instances</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Pilgrim men</td>
<td>1228</td>
<td>111</td>
<td>1339</td>
</tr>
<tr>
<td>Non-pilgrim men</td>
<td>859</td>
<td>111</td>
<td>970</td>
</tr>
<tr>
<td>Women</td>
<td>1016</td>
<td>162</td>
<td>1178</td>
</tr>
</tbody>
</table>
EXPERIMENTAL SETUP

- For Faster R-CNN: Inception-v2 and ResNet50
- For YOLOv3: (320x320), (416x416), and (608x608) pixels.
- For the learning rate:
  - YOLOv3: an initial rate of 0.001,
  - Faster R-CNN: an initial rate of 0.0002 with Inception-v2 and 0.0003 with ResNet50.
- The weight decay value of 0.0005.
- Stochastic Gradient Descent (SGD) of momentum (0.9).

<table>
<thead>
<tr>
<th></th>
<th>Machine 1</th>
<th>Machine 2</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>CPU</strong></td>
<td>Intel Core i7-8700K (3.7 GHz)</td>
<td>Intel Core i9-9900K (Octa-core)</td>
</tr>
<tr>
<td><strong>Graphics card</strong></td>
<td>NVIDIA GeForce 1080 (8 GB) GPU</td>
<td>NVIDIA GeForce RTX 2080T (11 GB) GPU</td>
</tr>
<tr>
<td><strong>RAM</strong></td>
<td>32GB</td>
<td>64GB</td>
</tr>
<tr>
<td><strong>Operating system</strong></td>
<td>Linux (Ubuntu 16.04 TLS)</td>
<td>Linux (Ubuntu 16.04 TLS)</td>
</tr>
</tbody>
</table>
EXPERIMENTAL METRICS

For the evaluation used six metrics:

- **True Positive (TP)**
- **False Positive (FP)**
- **False Negative (FN)**
- **Precision** = $\frac{TP}{TP + FP}$
- **Recall** = $\frac{TP}{TP + FN}$
- **F1score** = $\frac{(2 \times Precision \times Recall)}{(Precision + Recall)}$
- **Quality** = $\frac{TP}{TP + FP + FN}$
- **mIoU**: mean of the Intersection over Union.
- **mAP**: mean Average Precision
- **FPS**: frame per second.
## Comparison between Faster R-CNN and YOLO v3

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>YOLOv3 (320x320)px</th>
<th>YOLOv3 (416x416)px</th>
<th>YOLOv3 (608x608)px</th>
<th>Faster R-CNN (Inception v2)</th>
<th>Faster R-CNN (ResNet 50)</th>
</tr>
</thead>
<tbody>
<tr>
<td>FP</td>
<td>40</td>
<td>66</td>
<td>51</td>
<td>164</td>
<td>137</td>
</tr>
<tr>
<td>TP</td>
<td>159</td>
<td>171</td>
<td>165</td>
<td>228</td>
<td>195</td>
</tr>
<tr>
<td>FN</td>
<td>225</td>
<td>213</td>
<td>219</td>
<td>156</td>
<td>189</td>
</tr>
<tr>
<td>Precision</td>
<td>0.8058</td>
<td>0.7288</td>
<td>0.7735</td>
<td>0.6091</td>
<td>0.6022</td>
</tr>
<tr>
<td>Recall</td>
<td>0.4349</td>
<td>0.4577</td>
<td>0.4557</td>
<td>0.5929</td>
<td>0.5042</td>
</tr>
<tr>
<td>Quality</td>
<td>0.3905</td>
<td>0.3849</td>
<td>0.3923</td>
<td>0.4172</td>
<td>0.3744</td>
</tr>
<tr>
<td>F1score</td>
<td>0.5541</td>
<td>0.5546</td>
<td>0.5546</td>
<td>0.5887</td>
<td>0.5446</td>
</tr>
<tr>
<td>mAP</td>
<td>0.3999</td>
<td>0.4152</td>
<td>0.4214</td>
<td>0.5162</td>
<td>0.4317</td>
</tr>
<tr>
<td>mIoU</td>
<td>0.6352</td>
<td>0.5988</td>
<td>0.6192</td>
<td>0.5710</td>
<td>0.5850</td>
</tr>
<tr>
<td>FPS</td>
<td>91.28</td>
<td>65.31</td>
<td>43.84</td>
<td>3.35</td>
<td>3.8</td>
</tr>
</tbody>
</table>
TP, FP and FN

![Algorithm / dataset bar chart]

**Algorithm / dataset**
- Faster RCNN
  - Pilgrim
  - Non-Pilgrim
  - Woman
  - Total
- YOLO v3
  - Pilgrim
  - Non-Pilgrim
  - Woman
  - Total

**Value**
- FP
- TP
- FN

<table>
<thead>
<tr>
<th>Algorithm / dataset</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Faster RCNN Pilgrim</td>
<td>22.0</td>
</tr>
<tr>
<td>Faster RCNN Non-Pilgrim</td>
<td>52.0</td>
</tr>
<tr>
<td>Faster RCNN Woman</td>
<td>56.0</td>
</tr>
<tr>
<td>Faster RCNN Total</td>
<td>70.0</td>
</tr>
<tr>
<td>YOLO v3 Pilgrim</td>
<td>27.0</td>
</tr>
<tr>
<td>YOLO v3 Non-Pilgrim</td>
<td>40.0</td>
</tr>
<tr>
<td>YOLO v3 Woman</td>
<td>47.0</td>
</tr>
<tr>
<td>YOLO v3 Total</td>
<td>64.0</td>
</tr>
</tbody>
</table>

**FP**
- 22.0
- 52.0
- 40.0
- 27.0
- 49.0

**TP**
- 60.0
- 68.0
- 72.0
- 64.0
- 35.0
- 52.0

**FN**
- 52.0
- 40.0
- 47.0
- 49.0
- 35.0
- 14.0

**Total**
- 212.0
- 158.0
- 219.0
- 165.0
- 113.0
- 52.0
**mAP**: mean Average Precision

![Diagram showing mAP values for different algorithms and datasets.](image-url)
**FPS:** frame per second.

![Graph showing mAP vs Inference Time](image)

- Darknet53 - 608x608
- Darknet53 - 416x416
- Darknet53 - 320x320
- Inception_v2 - 0
- ResNet-50 - 0

Legend:
- Faster RCNN
- YOLOv3
We developed convolutional neural network models for pilgrim detection for Al-Hajj based on YOLOv3 and Faster RCNN.

We have built a dataset containing three classes of pilgrims, non-pilgrims and women.

Experimental results show that Faster RCNN with Inception v2 feature extractor provides the best mean average precision over all classes with 51%, comparable to state-of-the-art object detection algorithms.
Perspective

- We will extend the dataset to have several tens of thousands of instances to improve the overall accuracy and precision,
- We will consider more classes.
- We aim at developing a search application for lost people during Hajj and Umrah.
Demo

Link of demo: https://www.youtube.com/watch?v=L-nmYBY2pvE
Thanks for your attention