Real-time low-pixel infrared human detection from unmanned aerial vehicles
Martinez-Alpiste, Ignacio; Golcarenarenji, Gelayol; Wang, Qi; Alcaraz-Calero, Jose Maria

Published in:
DIVANet 2020 - Proceedings of the 10th ACM Symposium on Design and Analysis of Intelligent Vehicular Networks and Applications

DOI:
10.1145/3416014.3424600

Published: 16/11/2020

Document Version
Peer reviewed version

Link to publication on the UWS Academic Portal

Citation for published version (APA):

General rights
Copyright and moral rights for the publications made accessible in the UWS Academic Portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

Take down policy
If you believe that this document breaches copyright please contact pure@uws.ac.uk providing details, and we will remove access to the work immediately and investigate your claim.
Real-Time Low-Pixel Infrared Human Detection From Unmanned Aerial Vehicles

Ignacio Martinez-Alpiste
University of the West of Scotland
Paisley, United Kingdom
ignacio.alpiste@uws.ac.uk

Qi Wang
University of the West of Scotland
Paisley, United Kingdom
qi.wang@uws.ac.uk

Gelayol Golcarenarenji
Paisley University of the West of Scotland
United Kingdom
g.golcarenarenji@uws.ac.uk

Jose Maria Alcaraz Calero
University of the West of Scotland
Paisley, United Kingdom
jose.alcaraz-calero@uws.ac.uk

ABSTRACT
To improve the speed and accuracy in human detection in Search and Rescue (SAR) operations, this paper presents a novel and highly efficient machine learning empowered system by extending the You Only Look Once (YOLO) algorithm, which is designed and deployed on an embedded system. The proposed approach has been evaluated under real-world conditions on a Jetson AGX Xavier platform and the results have shown a well-balanced system in terms of accuracy, speed and portability. Moreover, the system demonstrates its resilience to perform low-pixel human detection on infrared images received from an Unmanned Aerial Vehicle (UAV) at low-light conditions, different altitudes and postures such as sitting, walking and running. The proposed approach has achieved in a constrained environment a total of 89.26% of accuracy and 24.6 FPS, surpassing the barrier of real-time object recognition.

CCS CONCEPTS
• Computing methodologies → Neural networks.

KEYWORDS
UAV, Thermal imagery, Jetson AGX xavier, YOLO, machine learning

1 INTRODUCTION
Unmanned Aerial Vehicles (UAVs) have been employed in numerous applications and among others they can assist Search and Rescue (SAR) operations to find missing people in rural areas. Meanwhile, real-time and reliable human detection from UAV-based imagery is highly complex and demanding due to various factors such as different altitudes of the UAV, different view angles of the on-board camera, changes in illumination, distances from human object and so forth [20].

A myriad of AI-empowered UAV human detection system has been designed and developed using optical imagery for SAR missions (e.g.,[4], [13] and [10]). Nevertheless, due to varying illumination, performing efficient object detection is very challenging in poor contrast and illumination situations using optical cameras, making the people detection error prone. In addition, the optical solutions are not suitable for people detection for lands covered intensely with undergrowth or at nighttime. This is when thermal cameras come to play as a vital tool in SAR missions when optical cameras are not able to provide accurate and reliable outcomes.

Some methods have been proposed for people detection in thermal videos. Image segmentation, thresholding and edge detection are traditional computer vision techniques used for human detection with both optical and thermal images. Due to the simplicity, these techniques are not accurate enough for complex real-world environments. Other traditional machine learning methods such as Histogram of Oriented Gradients (HOG) combined with classification techniques such as Support Vector Machines (SVM) and Adaptive Boosting (AdaBoost) are also suitable for human detection scenarios using thermal images, although the accuracy is still low in real-life environments.

Convolutional Neural Network (CNN) based techniques such as You Only Look Once (YOLO) series [15], [16] and [17] and Faster-CNN [18] are accurate techniques that have been used in literature for human detection in thermal images techniques. Nevertheless, they are computationally very expensive for real-time scenarios when running on embedded devices such as the Nvidia Jetson Xavier platform.

To address and overcome the drawbacks of the mentioned algorithms for thermal images in literature, there is a need for a real-time human detection model with high accuracy to address the challenges of human detection from UAVs using thermal images such as the pixel size of the human from medium-high altitudes.

Hence, the aim of this paper is to design and develop a light CNN-based human detection system to detect people using thermal images from UAVs with high accuracy in real time. The proposed approach is suitable for UAV applications as it considers the challenges...
As mentioned before, YOLOv3 is an effective technique often employed due to being very accurate. It is a one stage detector which concurrently selects and classifies the Region of Interests (ROIs). Darknet-53 [1] is used as the backbone for feature extraction in this detector. It contains three different output layers with different scales to detect big, medium and small objects. YOLOv3 runs in an inference time of 29 ms at 55.3% mean Average Precision (mAP) with an input size of 416×416 when using the Common Objects in Context (COCO) dataset for training [8]. This leads to a frame rate of 35 Frames Per Second (FPS) on a powerful GPU Nvidia TITAN X. This detector is suitable for small object detection due to fusion of features at earlier and deeper layers. Nevertheless, it is computationally intensive due to many convolutional layers being involved. This makes this technique unsuitable for real-time object detection in power-limited devices.

To increase the speed of YOLOv3, Tiny-YOLOv3 has been defined as a simplified version. The backbone of Tiny-YOLOv3 contains few convolutional and pooling layers. This causes this model to detect objects at more than 200 FPS due to less computation resources and memory requirement.

### 2.2 Previous work

There are few existing works regarding human detection using thermal images in literature. As a case in point, in one study, Ivasic-Kos et al. [7] used YOLOv3 dataset of thermal images collected at night and in various ranges and movements. Although they achieved an excellent result in terms of mAP, they have not reported the speed. Apparently, as YOLOv3 is computationally expensive, it cannot achieve real-time performance on power-limited systems such as Nvidia Jetson Xavier. In addition, the study is not on UAVs which is more challenging in terms of object detection. In [21], a new approach was developed by extending YOLO using Long Short-Term Memory (LSTM). The proposed method achieved a real-time speed of 45 fps and accuracy of 0.765, 0.772 and 0.812 (IOU) for occlusion, varying size, and image contrast respectively. Nevertheless, the real-time performance is achieved on a PC and the accuracy of the model should also be improved for critical scenarios such as search and rescue missions. Furthermore, the system is not based on UAVs.

In another study, Doulmais et al. [6] used thresholding techniques to detect victims under an Urban Search and Rescue environment. They used the OpenCV library on a Raspberry Pi 2 to perform the detection. Although, they have achieved very high accuracy, the speed has not been reported. Moreover, UAV was not used in this case study, which makes the detection more challenging. In [14], YOLOV3 was utilised to train a new data set containing thermal images for mob detection. The trained model is capable of achieving accuracy of 90.52% and real-time performance of 55 fps when deployed on a modern GPU. The speed, however, was reported on a modern GPU and using YOLOv3 is computationally very expensive on embedded systems such as Nvidia Jetson Xavier.

In [19], a two-step approach was used for detecting humans in real-time using thermal long-wave infrared (LWIR) imagery captured from a camera embedded on a UAV. The application of Maximally Stable Extremal Regions (MSER) were used instead of background detector or sliding window to perform the hot spot detection. Then, Integral Channel Features (ICF) based descriptors

<table>
<thead>
<tr>
<th>Ref</th>
<th>Objective</th>
<th>Algorithm</th>
<th>Exec. Environment</th>
<th>Platform</th>
<th>Accuracy</th>
<th>Speed (FPS)</th>
<th>Model Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>[7]</td>
<td>Human</td>
<td>YOLOv3</td>
<td>PC</td>
<td>NG</td>
<td>97.93%</td>
<td>NG</td>
<td>NG</td>
</tr>
<tr>
<td>[21]</td>
<td>Human</td>
<td>Yolo+LSTM</td>
<td>PC</td>
<td>Tensorflow</td>
<td>0.812 (IOU)</td>
<td>45</td>
<td>NG</td>
</tr>
<tr>
<td>[6]</td>
<td>Human</td>
<td>Thresholding</td>
<td>Raspberry Pi 2</td>
<td>Opencv</td>
<td>96%</td>
<td>NG</td>
<td>NG</td>
</tr>
<tr>
<td>[14]</td>
<td>Crowd</td>
<td>YOLOv3</td>
<td>PC</td>
<td>NG</td>
<td>90.52%</td>
<td>55</td>
<td>NG</td>
</tr>
<tr>
<td>[19]</td>
<td>Human</td>
<td>MSER+Naive Bayes</td>
<td>PC</td>
<td>NG</td>
<td>99%</td>
<td>NG</td>
<td>NG</td>
</tr>
<tr>
<td>[5]</td>
<td>Human/animal</td>
<td>Otsu Thresh/Custom CNN</td>
<td>PC</td>
<td>NG</td>
<td>100%</td>
<td>NG</td>
<td>NG</td>
</tr>
<tr>
<td>[3]</td>
<td>Human</td>
<td>Faster RCNN</td>
<td>PC/Azure</td>
<td>Tensorflow</td>
<td>46% (F1)</td>
<td>5/100</td>
<td>NG</td>
</tr>
<tr>
<td>TP</td>
<td>Human</td>
<td>Own Approach</td>
<td>Jetson AGX Xavier</td>
<td>Darknet</td>
<td>89.26%</td>
<td>24.6</td>
<td>33 MB</td>
</tr>
</tbody>
</table>

TP = This Paper; NG = Not Given; Green = information provided; Red = information missed.
and a Naive Bayes classifier were used to verify the detected hot spots. The approach achieved very high accuracy with a low computational runtime. However, the speed was not reported in this study. In [5], a classifier and detector were developed to detect poachers and animals using thermal images on a drone. The classifier part achieved a high accuracy of 100% and the tracker part achieved a performance of 90.93% due to occlusion. Similarly, the speed was not demonstrated in this paper.

In addition, Bondi et al.[3] developed a systematic poacher detector (SPOT). To detect poachers, Faster-RCNN was used in this study, to achieve 46% (F1 score) and in near real-time performance. The performance of this detector should be improved for scenarios where human detection is highly imperative such as in search and rescue missions.

None of the existing literature covers and reports all the essential metrics for evaluation such as speed and model size[11][12]. In addition, these studies are not robust enough in terms of accuracy and/or speed for human detection in UAV use cases using thermal images in constrained environments such as Nvidia Jetson Xavier. Consequently, this paper provides an improved architecture for real-time human detection using thermal images considering all the challenges of human detection from UAVs with high accuracy suitable for UAV applications particularly for scenarios where there are resource constraints to run the CNN-based models.

3 DESIGN OF THE PROPOSED ALGORITHM

The most important part of the proposed system is the architecture of the human detector. This algorithm should be the trade-off among three main factors: accuracy, speed and portability. When flying at high altitudes, a UAV captures humans on the ground in a small range of pixels, making accuracy a key factor. Moreover, a decrease in the inference time is mandatory to make the system capable of detecting people when flying at a high speed. Finally, lighter algorithms are in demand due to deployment and execution of the algorithm on a constrained environment with limited power resources, GPU and memory. Thus, to maintain this harmony, we propose an algorithm that achieves a significant improvement in the accuracy for the detection of small people on aerial imagery taken from a UAV while fulfilling real-time requirement on embedded systems using thermal images.

3.1 Tiny-YOLOv3-based Human Detection

As stated in section 1, the standard YOLOv3 is an efficient one-stage detector yet computationally expensive. As a result, Tiny-YOLOv3 was introduced as a simplified version. It is a much faster model; however, this high speed sacrifices the accuracy. Tiny-YOLOv3 contains two output layers with various scales for object detection, which causes lower accuracy in small object detection. The accuracy of this algorithm is 33.1% mAP when using the widely adopted COCO dataset for training.

As the standard YOLOv3 lacks the speed and portability, and Tiny-YOLOv3 only lacks the required accuracy, our proposed algorithm is focused on increasing the accuracy of small object detection of Tiny-YOLOv3 without sacrificing the speed or portability.

3.1.1 Output Layers. To enhance the accuracy of detecting small humans, one extra output layer is added to the standard Tiny-YOLOv3 in our approach. This extra output layer not only leads to an increment of the accuracy but also a decrease in the speed. Nevertheless, this slight decrease in the speed is not decisive as only three new convolutional layers are added to the architecture from 13 to 16 (see table 2), which is much lower than that of the standard YOLOv3. One upsampling operation has also been added to extract finer details and improve small object detection. As apparent in Table 2, our approach outperforms the other two algorithms in terms of the significantly higher number of predictions calculated. Basically, it increments the probability of small target detection due to the improvements applied to the proposed algorithm.

### Table 2: Architecture comparison between different YOLO-based algorithms

<table>
<thead>
<tr>
<th>Architecture</th>
<th>Outputs</th>
<th>Conv.</th>
<th>Upsample</th>
<th>Grid</th>
<th>Predictions</th>
</tr>
</thead>
<tbody>
<tr>
<td>YOLOv3</td>
<td>3</td>
<td>75</td>
<td></td>
<td>2</td>
<td>13,26,52</td>
</tr>
<tr>
<td>Tiny v3</td>
<td>2</td>
<td>13</td>
<td>1</td>
<td>1</td>
<td>13,26</td>
</tr>
<tr>
<td>Approach</td>
<td>3</td>
<td>16</td>
<td>2</td>
<td>2</td>
<td>26,52,104</td>
</tr>
</tbody>
</table>

Predictions = \( \sum_{\text{Grid}[i]}^{|n|} (\text{Grid} \times \text{Grid} \times 3) \)

3.1.2 Input size. To maintain a trade-off between accuracy and speed, the width and height are two main elements. These factors depend on the image resolution. Higher accuracy is provided by higher resolution; This, however, increases the inference time. To execute YOLOv3 accurately and quickly, the input size of 416 x 416 pixels is recommended. Hence, the algorithms in the results section (section V) were all trained using input size of 416 x 416 pixels to find the best trade-off between accuracy and speed.

3.1.3 Anchor Boxes. To provide information regarding the size of objects, anchor boxes are used. The anchor boxes are linked to the output detection layers and each has a shape of "width, height". Anchor boxes are connected to the stride of the architecture. Smaller objects can be detected with smaller strides. The anchor boxes should be calculated based on the dataset to increase the accuracy of small objects. If anchor boxes are not customised properly, the detector will miss the detection of several objects.

YOLOv3 usually uses three anchor boxes per output layer to improve the detection. As this research is focused on detecting humans in thermal images at very low pixel size, we need to regenerate the anchors values. The technique "K-Means" clustering was applied on the collected dataset (4.2). Nine pairs of values were obtained, the biggest anchors should be linked to the output layer in charge of big-scale detection and the smallest anchors to the small-scale detection output layer.

3.1.4 Masks. To determine which of the anchor boxes are in charge of prediction, masks are used as indices of anchor boxes in YOLO output layers. The first YOLOv3 layer predicts the largest boxes, whilst the last YOLOv3 layer predicts the smallest ones. The mask should be changed based on anchor boxes and the sizes of the objects to be detected.
In order to increase the probability of small target detection, one mask was removed from the first output layer and added to the third output layer. While only two masks are linked to the first output layer, four masks are used in the third output detection layer. The number of masks in the second output layer remains constant.

3.1.5 Grid Cells. YOLO-based algorithms divide the input image into a grid. The size of the grid cell depends on the image input size and the stride applied. The stride downsamples the dimensions of the image. The smaller the stride, the smaller the grid size, and thus the smaller objects can be detected.

As shown in Fig. 1, the controller sends the video from the drone to a controller and a Jetson Xavier. The UAV flies and transmits the video recorded by the camera to the controller. The camera is a “DJI Matrice 210” which is able to operate at lower power consumption. It includes a 512-core Volta GPU with Tensor Cores as GPU and 8-core ARM v8.2 64-bit CPU as CPU.s Darknet as the machine learning platform was deployed on Ubuntu 16.04 to run the human detection algorithm. Darknet [1] is a widely used platform written in C++ with CUDA support totally compatible with YOLO-based convolutional neural networks.

Deep and dataset creation. The collection was performed by authors at different altitudes and distances from the human object up to 75 meters. The footage was recorded in 640 × 512 pixels and the images were extracted for further processing. The dataset consists of 4,141 labelled images extracted from 47 video footage with 2,878 of positive and 1,263 of negative images. YOLO-Mark2 [2] was the tool employed to manually annotated each image.

The dataset was shuffled and split into two subsets: training and testing, in the ratios of 80:20. To increase the entropy and diversity of the dataset, the videos were recorded at different times of the day, in different weather conditions (sunny, cloudy and foggy) and different locations.

4.3 Training

The training was performed on a workstation with a Nvidia Titan X graphics card. The machine learning platform to train the model is the same as the one deployed on the Jetson Xavier for testing. The proposed algorithm was trained on Darknet to create the model. The hyperparameters used to train our proposed algorithm are defined in table 3. The total number of training iterations and the batch size were set to 15,000 and 64 respectively. The Stochastic Gradient Descent with Warm Restarts (SGDR) [9] was chosen as the solver. In order to reduce the overfitting probability and better feature learning, the image input size was randomised including other sizes such as 608 × 608 and 320 × 320, although 416 × 416 is the priority size.

<table>
<thead>
<tr>
<th>Hyperparameters</th>
<th>Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>Image size in pixel</td>
<td>416 × 416</td>
</tr>
<tr>
<td>Number of iteration</td>
<td>15,000</td>
</tr>
<tr>
<td>Batch Size</td>
<td>64</td>
</tr>
<tr>
<td>Initial Learning Rate</td>
<td>0.001</td>
</tr>
<tr>
<td>Solver</td>
<td>SGDR</td>
</tr>
<tr>
<td>Momentum Coefficient</td>
<td>0.9</td>
</tr>
<tr>
<td>Weight Decay</td>
<td>0.0005</td>
</tr>
</tbody>
</table>

Figure 1: Scenario of the proposed architecture.
algorithms: YOLOv3 and Tiny-YOLOv3. Every result was obtained from the same system executed in a Nvidia Jetson Xavier.

These outcomes consider three main factors when exploring the human detection using UAVs: accuracy, speed and portability. In this section, a thorough evaluation is performed to show a stable balance among the aforementioned factors.

5.1.1 Accuracy. A set of videos sequences were taken from the UAV for the purpose of testing the validity of our proposed human detection algorithm. Due to the fact that it is highly imperative not to miss any people in a SAR mission, we have assessed the confusion matrix of the dataset by comparing the different algorithms. The bar graph 2 presents a confusion matrix of True Positives (TP), False Positives (FP) and False Negatives (FN). More TPs and fewer FPs and FNs mean better performance of the algorithms in terms of accuracy.

Fig. 2 shows the reliability of the standard YOLOv3 algorithms and our approach when detecting humans based on the collected dataset. Although the standard YOLOv3 obtained more TP detection and fewer FNs than our approach, our approach is more trustworthy by taking into account the number of False Positive detection. Tiny-YOLOv3 performed worst for all the metrics in the confusion matrix.

The overall accuracy was calculated with the mean Average Precision metric (mAP) at 0.5 Intersection over Union (IoU) threshold of 50%. Due to its less complex architecture, Tiny-YOLOv3 just obtained 70.23% of accuracy. By contrast, the standard YOLOv3 and our approach achieved greater performance with less than a 1% of difference at 90.19% and 89.26% respectively.

5.1.2 Speed. Furthermore, to calculate the speed on Nvidia Jetson Xavier, the power supply was adjusted to 10W, 15W and MAXN mode. Table 4 depicts the performance in terms of speed at various power modes.

In terms of speed, Tiny-YOLOv3 is faster than our approach and YOLOv3 but with far less accuracy (32.9 FPS versus 24.6 and 10.1 FPS @MAXN mode, 16.5 versus 12.1 and 5 @15W and 10.3,6 and 2.6 @10W) on Jetson AGX Xavier.

5.1.3 Portability. The portability factor is mainly measured by the size of the model and BFLOPS (Billions of Floating-Point Operations per Second). BFLOPS show the overload at the processor in order to execute the human detector algorithm. While our approach executed 19.971 BFLOPS, Tiny-YOLOv3 merely needed 5.448 BFLOPS, which is 3.7 times less than ours. By contrast, the standard YOLOv3 required 65.304 BFLOPS, 3.2 times more than ours.

The model size results in memory usage. While the model size of the standard YOLOv3 is 235 MB, the model size of Tiny-YOLOv3 and our approach are smaller with 34 MB and 33 MB respectively and thus are suitable for devices with constraints on power and computation.

5.2 Qualitative results
The images in Fig. 3 show the detection results on four images at different altitudes on our test set. The results reveal the successful detection of our proposed human detection system with no false negative detection. Humans were found at different locations and in different positions. Fig. 3(a) shows a detection of a human sitting on the bushes and Fig. 3(b) illustrates three humans siting and lying down at different scales. Finally, challenging situations such as varying distances (Fig. 3(c)) and flying altitudes (Fig. 3(d)) have validated the resilience and robustness of our proposed system.

6 CONCLUDING REMARKS AND FUTURE WORK
A new efficient machine learning-based human detection system has been designed, implemented and deployed to address UAV-based SAR challenges in a realistic scenario. The architecture counted on a balanced algorithm of the standard YOLOv3 and Tiny-YOLOv3 algorithms and it features several modifications such as “finer strides” and an extra “third output layer” to improve low-pixel human detection on thermal images. The proposed algorithm has demonstrated reliable performance in challenging situations. The detection of humans in a small range of pixels in different positions and weather conditions has shown the high accuracy of the algorithm. In addition, it has achieved sufficient speed surpassing the barrier of real-time detection in an embedded system. Finally, the solution is considered as a light model due to its smoothness when porting the system in a constrained system. The proposed system improves real-time object recognition in low illumination condition on constrained environments and it is expected to contribute significantly to the wider use of UAV applications.
ACKNOWLEDGMENTS
This work was in part funded by the Innovation Centre for Sensor and Imaging Systems (CENSIS) and Thales UK, and in collaboration with Police Scotland, under the grant “Smart Unmanned Aerial System for Real-Time Object Detection” (Ref. CAF440). The authors would like to thank all the partners in this project for their support.

REFERENCES


