Neural Computing and Applications (2021) 33:17273–17286
https://doi.org/10.1007/s00521-021-06317-8

ORIGINAL ARTICLE

Using human pose information for handgun detection

Alberto Velasco-Mata[1] · Jesus Ruiz-Santaquiteria[1] · Noelia Vallez[1] · Oscar Deniz[1]

Received: 18 August 2020 / Accepted: 6 July 2021 / Published online: 21 July 2021

Abstract
Fast automatic handgun detection can be very useful to avoid or mitigate risks in public spaces. Detectors based on deep learning methods have been proposed in the literature to trigger an alarm if a handgun is detected in the image. However, those detectors are solely based on the weapon appearance on the image. In this work, we propose to combine the detector with the individual's pose information in order to improve overall performance. To this end, a model that integrates grayscale images from the output of the handgun detector and heatmap-like images that represent pose is proposed. The results show an improvement over the original handgun detector. The proposed network provides a maximum improvement of a 17.5% in AP of the proposed combinational model over the baseline handgun detector.

Keywords Handgun detection · Human pose information · Deep learning

1 Introduction

The rapid detection of handguns in public spaces is essential to avoid or mitigate risks [1]. Surveillance by means of closed-circuit television (CCTV) has been widely used to detect those situations, although it requires continuous supervision of the images. This task is usually handled by a human operator, which is likely to miss them due to fatigue or visual distraction.

Deep learning techniques have proven to be specially powerful at automating this kind of visual tasks, where novel methods such as convolutional neural networks (CNNs) achieve good results in object detection. However, these methods are based on the generalization from a set of training samples, which may differ from the actual scenarios where they are used. When the difference is significant, these methods are known to either fail detecting the object they were trained to detect, or increase the false alarm rate [2]. This problem is exacerbated by the characteristics of the specific problem of threat detection in CCTV systems, where the cameras usually have poor resolution, the images often include artifacts and the threatening objects are small and in a far plane, represented by a small set of pixels on the image. The unreliable output in these conditions leads to a common case where the operator shuts down the automatic system because of the high rate of false alarms.

The addition of extra information to the image might be a solution for this kind of problems. In this work, the human body pose is considered as an improvement factor that is agnostic to the specific scenario, which could improve the handgun detector performance when it is used in the real scenario.

The pose of a person has proven to be effective to recognize human actions in images [3]. This work proposes the use of neural networks to integrate that information into an already existing handgun detector, analyzing the different possibilities that would improve its results on a new scenario that differs from the one used to train it. As far as the authors know, this is the first time that pose is leveraged to extend and improve an appearance-based handgun detector.

The paper is organized as follows. Section 2 explains briefly the previous work done on the handgun detection problem as well as the use of the pose as a source of information for activity recognition. Then, the datasets used are described in Sect. 3. Section 4 shows the handgun detector that has been used as a base to obtain a relative improvement. Section 5 provides a detailed explanation of the proposed combinational model, the training process and

✉ Noelia Vallez
noelia.vallez@uclm.es

[1] VISILAB, ETSI Industriales, Avda. Camilo Jose Cela SN, 13071 Ciudad Real, Spain

some additional considerations taken into account. In Sect. 6, the results of both the baseline detector and the combinational model are shown and compared. Finally, Sect. 7 summarizes the main conclusions.

2 Previous work

Traditionally, X-ray and millimetric wave imaging has been used to detect concealed handguns at the entrance of public places such as airports and train stations. Several approaches for automatic detection using those images have appeared in the literature. In Nercessian et al. [4] introduced a system that used segmentation and edge-based feature vectors to automate the detection of handguns in X-ray scan images of luggage. Other methods were proposed for passive millimetric wave images, such as the one by Xiao et al. [5] that used Haar-like features and the AdaBoost algorithm to detect metallic pistols accurately. However, although these methods achieve good results on these kind of images, there is a strong limitation in terms of the required machinery. This limitation leads to other proposals based on CCTV systems, which are cheaper and widely extended as they only require RGB images.

A first step towards automatic visual handgun detection was made in 2015 by Tiwari and Verma. They used color-based segmentation and k-means clustering to remove unrelated objects from the image and then applied the Harris interest point detector and fast retina keypoint to locate the handgun [6].

More recently, deep learning algorithms have been used for the handgun detection problem in surveillance camera images. In Olmos et al. [7] made a seminal contribution using two different approaches. The first one uses a sliding window and a CNN classifier, and the second uses a faster-RCNN [8] based model that provides the most promising results. Several other authors have used similar deep learning techniques to solve the handgun detection problem [9, 10].

To decrease the false-positive and false-negative rates, some authors have proposed methods such as using a symmetric dual camera system to improve the selection of candidate regions, thus increasing the performance of the model [11]. Another approach used is to model false positives as anomalies to filter false positives and increase the overall performance of the detector [12].

Although some of these methods address specifically the problem of erroneous detections when the systems are used in different scenarios, they are all based on the visual appearance of the handgun in the images. The novelty of our work is to use additional information (the human pose) in order to improve the output of the model.

Over the years, activity recognition has been tackled using extensive feature engineering along with classical data science techniques. In 2008, Thurau and Hlavac presented a method that recognizes human actions in still images using pose primitives [13]. This method extends a HOG descriptor to deal with articulated poses, recognizing the activity by comparing histograms. In Reiss et al. [14] introduced another approach that combined a signal-oriented classifier with model-based features that were calculated by means of joint angles and torso orientation.

In 2016, Chevalier introduced the use of recurrent neural networks in human action recognition [15]. In that work, he used long short-term memory cells to classify the type of movement. However, he used not the pose but the accelerometer and gyroscope data from a phone attached to the waist. Later, Eiffert extended the work using 2D pose keypoints as an input for a similar LSTM model, proving that two-dimensional pose estimation can be used for human activity recognition [16]. The human pose was obtained using OpenPose [17], a pose estimator that predicts the keypoints from the image.

Further research has been made in this topic, combining the pose and the appearance of the images to get accurate results recognizing the actions and activities performed on the scene [3]. Since the human pose provides enough information to determine the activity, it should be useful when it comes to the handgun detection problem. While body pose information has been extensively used in classic computer vision problems such as gesture and activity recognition (see recent survey [18]), it has not been widely used for handgun detection yet. Recently, some methods try to use the body pose information for handgun detection but using an indirect approach. Lomas proposed a preprocessing method that consists in modifying the input images by adding an overlay with the human skeleton shapes retrieved from OpenPose framework [19]. Although emphasizing the body pose in the input images in this way may help to detect handguns close to the human body, the base detection architecture is not modified, which leads to maintaining the classic limitations of these detection methods. Basit et al. [20] proposed a method that associates weapons with persons carrying them. In this approach, persons and handguns are detected separately and then a network is trained to detect which paired bounding box corresponds to the person that has the handgun. This method behaves like a false-positive filter, removing detected handguns which are not associated with a detected person, but false negatives cannot be solved.

To the best of our knowledge, our proposed work is the first one that applies pose information directly in a deep neural network architecture to increase the performance of a handgun detector, both reducing false positives and false

negatives. Thus, the objective shifts from that of 'weapon detection' to one of a 'threat detection.'

3 Dataset acquisition and processing

3.1 Dataset selection

Nowadays, in the Big Data era, many companies provide tons of data about almost every topic you can imagine. However, finding good data that is properly labeled is still not trivial at all. Getting more and more specific on the required topic exacerbates this problem. Furthermore, the common problem of the amount of false positives and false negatives that appear when the scene is different to the one used in the training process should be taken into account. Thus, it makes sense to select different datasets in order to simulate different scenarios. Taking this into account, a collection process has been made to obtain enough data. One of the sources considered was the academic community, although the existing publications usually do not publish the datasets and resources used. Nevertheless, one of the first datasets considered for our work was the Gun Movies Database [10], with frames depicting an individual holding a handgun and walking through a room in several positions. However, handgun labels were not available online, so manual labeling was performed. The camera in this dataset is fixed, with a background that does not change, and therefore data augmentation was used to introduce variability. A total of 181 augmented frames were randomly selected from over 800 initial labeled frames, in order to avoid consecutive frames that are almost identical.

In addition, another obvious source of CCTV-like images was considered: the Internet. Many video-based platforms have several recordings of people training with real guns. Therefore, a set of videos containing this kind of images was downloaded and labeled properly to be used in this work. Unfortunately, many of these frames are blurry due to bad quality cameras, so an appropriate frame selection process was required too. A set of 837 frames was obtained from eleven videos downloaded from the YouTube platform.

In any case, although there were many videos on the Internet, the camera perspective is usually positioned in first person, and the full bodies and skeleton poses where not as visible as it would be expected from a surveillance camera. Our requirement of different poses holding a handgun led us to resort to synthetic images. Although the synthetic generation of images for handgun detection purposes has already been studied [21], the creation of realistic skeleton animations is out of the scope of this work. Therefore, videogames like first-person shooters were considered as a source of images with handgun-like poses and a high level of realism.

An automatic mechanism for image retrieval was mandatory to obtain a large amount of frames with different poses. The NVIDIA Ansel technology was considered for this task, since it 'is a powerful photo mode that lets you take photographs of your games' and allows 360-degree movement, according to its official website. This technology provides the ability to pause the game (if the game is supported by the library), move the camera freely around the scene and save a picture of it. However, the supported games list is rather limited, so the selected videogame was Watch Dogs 2 which has a high level of realism and allows the selection of different guns. It also provides different shooting poses that can be recorded. The Ansel technology allowed the acquisition of eight videos of different poses and different angles, which were then labeled and integrated with the previous datasets. Examples of the final dataset are shown in Fig. 1.

3.2 Human pose acquisition

Once all images were gathered and labeled, the additional information required for this work is the human pose estimation on those images.

There are two typical approaches for the problem of human pose estimation, top-down and bottom-up, according to the way the detection is made:

- *Top-down approaches* are based on the sequential combination of a person detector and a single body part estimator, which is applied over each of the detections from the first step.
- *Bottom-up approaches* follow an inverse order, in which all body parts are detected directly on the image and then a grouping step is executed to associate limbs belonging to the same person.

Top-down approaches are easier to implement than the grouping algorithm of the bottom-up ones. However, it is hard to make a distinction on which of the two approaches provides better results.

A distinction on the output these estimators provide can also be made. The provided information is usually common between all of them, providing the keypoints of the detected limbs on the image, but the difference relies on the dimensions of those keypoints, where some of them provide a two-dimensional output and others estimate the 3D position of the keypoints.

In our case, *OpenPose* [17] was selected as the body pose estimator because of its speed and capability of detecting up to 25 pose keypoints of human bodies in the image. This pose estimator offers a bottom-up approach that detects the keypoints of the limbs on the image and

Fig. 1 Sample images of the different sources for the dataset

(a) Gun Movies Database **(b)** Videos from YouTube **(c)** Watch Dogs 2

then uses a set of convolutional layers to predict Part Affinity Fields (PAFs), which represent the association between the detected parts. Using a bottom-up approach offers the advantage of independence in execution speed with respect to the number of individuals in the image, as opposed to top-down approaches that execute code per person detected.

OpenPose offers 2D and 3D estimation of the human pose, but the later is only obtained from multiple views of the same scene. Other architectures such as VNect provide a three-dimensional output from monocular images, but 2D estimation was chosen for this project.

As a result, *OpenPose* was applied to all the images of the dataset. This step stores the detected keypoints on the images in JSON format, as well as the PAF maps in the form of grayscale images.

3.3 Dataset split

The dataset was split into training, validation and test subsets. A special consideration was taken into account: The test subset should simulate the original problem of using the detector in a different scene than the ones it was trained with. Since the dataset is composed of many videos with different scenarios, they were separated according to this principle and balancing the number of synthetic and real images on both the training and test sets.

- *Training and validation* The YouTube videos and half of the eight videos obtained from the Watch Dogs 2 videogame were selected for the training step of the models. 60% of each video was dedicated to training and 20% was used for validation.
- *Test* The remaining 20% of the previous videos was reserved for testing the models. The Gun Movies Database frames and the other half of the Watch Dogs 2 videos were reserved for the final comparison between the models.

To make the comparison as fair as possible, both the baseline detector and the detection and pose combination model will be trained with the same images, and then the performance of both will be evaluated using the reserved videos that contain different scenarios. A visual representation of the dataset split can be seen in Fig. 2.

4 Baseline handgun detector

Since the addition of the body pose information is not tied to a specific model architecture, any modern object detector that is available in the literature is suitable for the comparison intended in our work. Most of them, such as faster-RCNN, are based on a region proposal network that focuses on the interesting regions of the image, followed by a classifier that determines if the region is one of the selected classes. These networks are usually slow, both to be trained and to run over new images. However, there are other architectures with similar accuracy results that follow an approach based on the examination of the complete image at once which makes them much faster. This has been the reason why we have selected YOLOv3 as the baseline handgun detector, since it is in general faster than other methods while keeping a good detection rate.

This network was trained and tested using the dataset samples described in Sect. 3.3. The performance on the test dataset is shown and compared later in Sect. 6.

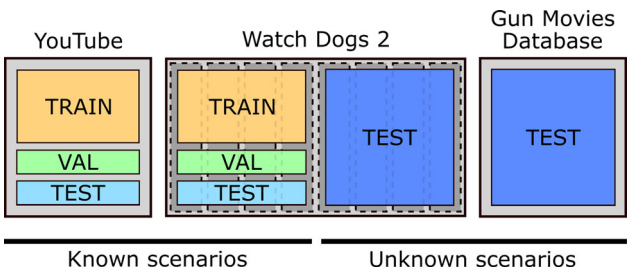


Fig. 2 Separation of the dataset videos into training, validation and test subsets. Two kinds of test subsets are shown, as for known and unknown scenarios

5 Proposed combinational model

5.1 Architecture proposal

The proposed model takes the pose as an extra step after the handgun detection takes place. This conforms a modular architecture where an existing handgun detection is leveraged and extended, without modifying its internal architecture. Thus, the combinational architecture takes two inputs: the result of the object detection network and the pose information. It can be seen as a 'horizontal Y'-shaped architecture, having two input branches and a final combination trunk (see Fig. 3).

Potentially, this architecture can both reduce the false-positive and false-negatives rates. Regions that are detected as handgun by mistake can be removed from the final result because that regions do not match with any human on the scene. Besides, a handgun that was not detected in the first place might still be considered by this network if the person holding it has a typical handgun-holding pose.

The detailed layer description of the model is shown in Table 1, and a visual representation of the layers and expected inputs and outputs is shown in Fig. 4.

5.2 Training

The characteristics of the problem considered in this work have conditioned the training of the combinational network. Again, the whole system should improve the results measured in a different environment than the one it was trained with. That is why there are different scenarios in the training and test datasets, as explained in Sect. 3. However, to be fair in the comparison, the dataset used to train the combinational network must be the same as the one used to train the baseline detector (otherwise it could be argued that any advantages of the combinational network are due to a different dataset). This might result in a problem during training, because the baseline detector performs really well on the training dataset and that is likely to make the combinator always leverage the output of the detector, discarding the pose information. Obviously, this is not the desired behavior. To solve these kinds of problems, two countermeasures have been considered: separating training in two phases and balancing the training process of the combinational segment.

5.2.1 Two-phase training

The aforementioned problem affects the combination process. However, the proposed network is divided into three parts: two input branches and a combinational trunk. The combinational segment is just part of the network, so we can train the pose input branch first and keep it away from the problem. In this way, we can assure that the pose input branch is accomplishing its purpose of detecting handgun-holding poses and emphasizing their hands. After that, the weights of those layers can be frozen and the combinational training takes place.

To do so, the implementation of the network actually provides two models: the pose detector and the full combinator. These two models share most of the pose detector layers, but the pose detector adds one extra layer to get as an output a grayscale image that represents the focused region mask. The full model concatenates the last of the shared layers to the detector input and applies several convolutions until a final monochrome image is obtained.

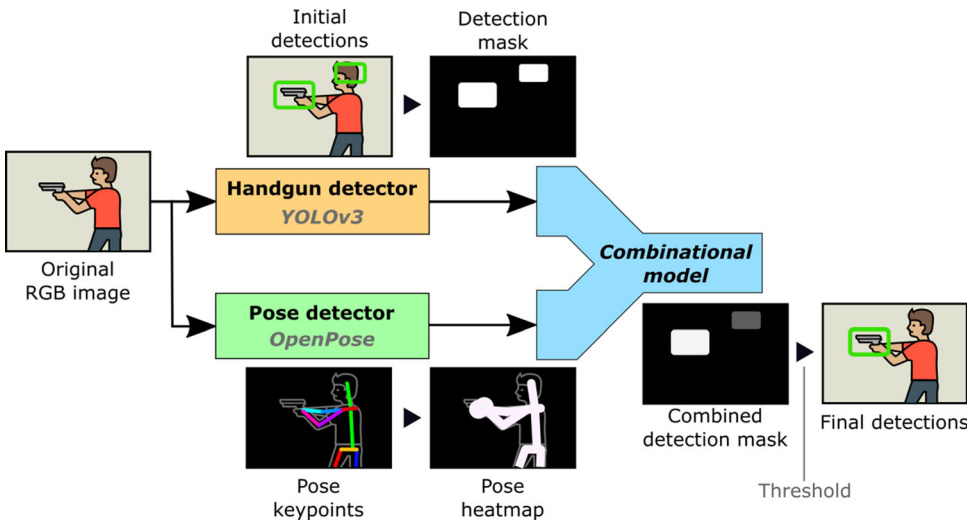


Fig. 3 Architecture proposal where the pose is considered after the handgun detector. A threshold can be applied on the final step, where the image is converted to bounding boxes

Table 1 Detailed description of the combinational network architecture used, separated by logic blocks as two input branches and a combination trunk

Layer (type)	Output shape	Param #
Pose Detector Input Branch		
input_1 (InputLayer)	(None, 416, 416, 1)	0
conv2d_1 (Conv2D)	(None, 416, 416, 64)	640
conv2d_2 (Conv2D)	(None, 416, 416, 64)	36928
conv2d_3 (Conv2D)	(None, 416, 416, 64)	36928
conv2d_4 (Conv2D)	(None, 416, 416, 64)	36928
conv2d_5 (Conv2D)	(None, 416, 416, 64)	36928
conv2d_6 (Conv2D)	(None, 416, 416, 64)	36928
conv2d_7 (Conv2D)	(None, 416, 416, 64)	36928
conv2d_8 (Conv2D)	(None, 416, 416, 64)	36928
conv2d_9 (Conv2D)	(None, 416, 416, 32)	18464
conv2d_10 (Conv2D)	(None, 416, 416, 16)	4624
conv2d_11 (Conv2D)	(None, 416, 416, 8)	1160
conv2d_12 (Conv2D)	(None, 416, 416, 4)	292
conv2d_13 (Conv2D)	(None, 416, 416, 2)	74
Handgun Detector Input Branch		
input_2 (InputLayer)	(None, 416, 416, 1)	0
Combination Trunk		
concatenate_1 (Concatenate)	(None, 416, 416, 3)	0
conv2d_15 (Conv2D)	(None, 416, 416, 64)	832
conv2d_16 (Conv2D)	(None, 416, 416, 64)	16448
conv2d_17 (Conv2D)	(None, 416, 416, 32)	8224
conv2d_18 (Conv2D)	(None, 416, 416, 16)	2064
conv2d_19 (Conv2D)	(None, 416, 416, 8)	520
conv2d_20 (Conv2D)	(None, 416, 416, 4)	132
conv2d_21 (Conv2D)	(None, 416, 416, 2)	34
conv2d_22 (Conv2D)	(None, 416, 416, 1)	9

The concatenate_1 layer merges layers conv2d_13 (pose detector branch, with processed pose as heatmap of handgun-like regions) and input_2 (handgun detector branch, with mask of the handgun detections)

5.2.2 Training balancing for the combinational segment

Once the pose detector segment is trained, the remaining step to get the combinational model is to train the combinational trunk. The detector dataset leads to this segment omitting the pose information because the detector input is almost always a safe choice that assures a low loss in training. Data augmentation is used to balance the training process and avoid this kind of problems. It is implemented by randomly introducing false positives and false negatives in the baseline detector input. About half of the input images that represent the bounding boxes over a black background are replaced by a full black image, simulating a false-negative scenario where the detector failed to detect the handgun and the final result depends on pose information. After that, a similar approach is used to introduce false positives, inserting random white rectangles on the image and thus simulating a false detection coming from the detector. This way, the training process is balanced and the network must learn to combine the images coming from the pose branch and the detector image to decrease the loss on the output.

The advantage of working with monochrome images is that the output is visually representative. When the trained model is applied to the images of the test set, the expected behavior is clearly seen in Fig. 5: The output has distinctive white spots when the detector is really sure about the handgun (i.e., both the baseline detector and the pose detector agreed on the handgun region), and grayish spots when only one of them has considered the region as a handgun. Thus, the output of the final model is still a grayscale heatmap-like image that represents the region probabilities of finding a handgun. This output must be then converted to the classical bounding boxes in order to do the comparison.

5.3 Mask-to-bounding box conversion

The development of the algorithm that converts grayscale images into bounding boxes is based on the thresholding of the white intensity, using other computer vision techniques to reduce noise and obtain a clear result. The conversion algorithm has the following steps:

1. *Apply a threshold on the white intensity level* Since the output of the network is a heatmap that represents the probabilities of each pixel being part of a handgun, this threshold binarizes that information into classes gun/not-gun.
2. *Get white spots on the image* Computer vision techniques such as topological structural analysis are applied to obtain the contours of the resulting white spots from the previous step. These spots are candidates of being part of a handgun.
3. *Group spots by nearness to hand* Due to the thresholding step, there might be multiple spots that belong to a single handgun. This step obtains their centroids and matches each of them with one of the hands (as estimated by the pose detector) based on the distance to that keypoint. The closest spots are then grouped together and considered part of the same handgun.
4. *Select handgun candidates* A ratio between the white spots and the total hand area is calculated for each detected hand. Those hands whose ratio is greater than a 15% are considered handguns.
5. *Convert handgun detections to bounding boxes* Once the regions of the hands holding handguns are detected,

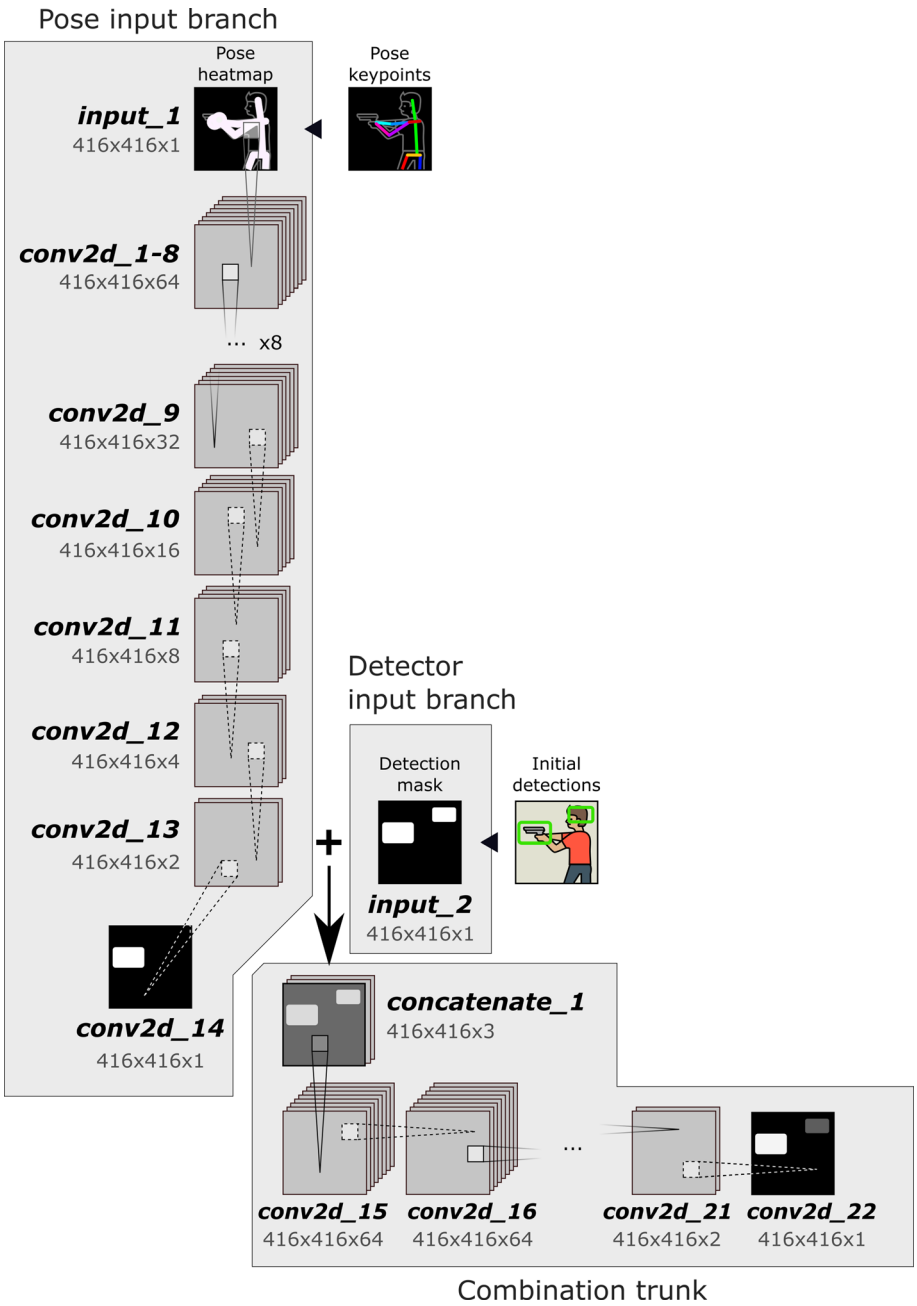


Fig. 4 Visualization of the combinational network layers and the expected inputs and outputs. Notice the extra convolutional layer *conv2d_14*, which is only used in the pose branch training but not on the final combinational network

they are converted into rectangles by stretching the hand regions horizontally.

A visualization of the steps can be seen in Fig. 6.

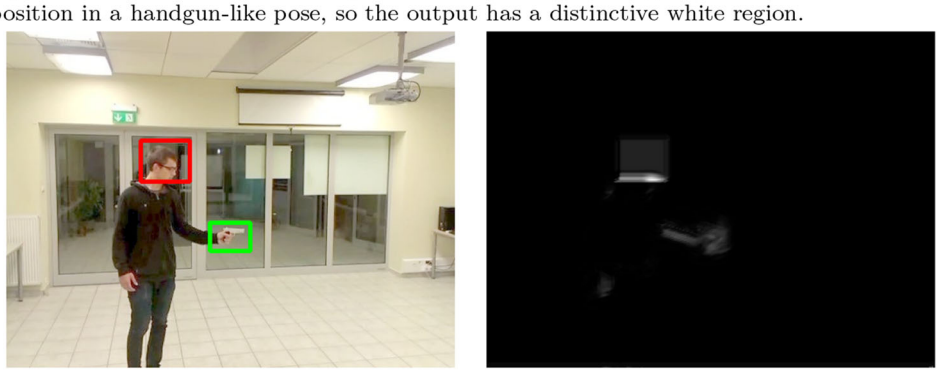
5.4 Ground-truth intersection measure

One of the most common metrics that determines if a detected box matches the ground-truth (labeled detection) is the intersection over union (IoU), which evaluates how the detected box fits the target box.

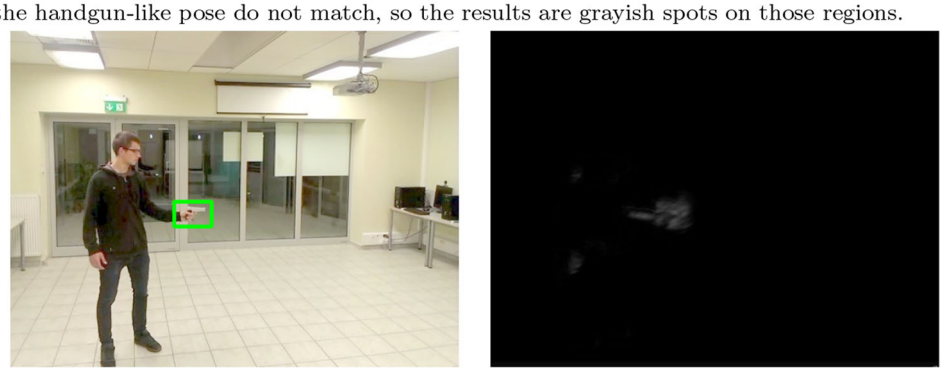
The converted output shows that the detected handguns have very tight bounding boxes, adjusting closely to the handgun region. However, the labeling of some images of the dataset was very different: It had part of the forearm and background included in the bounding box. This resulted in a huge difference in size between the predicted box and the target box, which produces very low IoU values even when almost all detections are inside the ground-truth boxes. Examples of these cases are shown in Fig. 7.

For the problem tackled in this work, these detections are considered correct despite the size difference. The main reason for this is that the purpose of the model is to detect if a handgun is present in a CCTV image (and which

(a) True Positive. The detector predicted a correct bounding box that matches the hand position in a handgun-like pose, so the output has a distinctive white region.



(b) False Positive. The detector prediction (false positive on the individual's head) and the handgun-like pose do not match, so the results are grayish spots on those regions.



(c) False Negative. The detector could not get the handgun region, but the pose indicates that there might be a handgun there, so there is a grayish spot on the output image.

Fig. 5 Combinational network results in different situations where the base detector gets correct and incorrect bounding boxes. On the left, the frame images with green rectangles on the ground truth, red rectangles on detected regions considered false positives and blue rectangles on detections that are correct. On the right, resulting mask images of the combinational model. Best viewed in color

individual is holding it), not its exact limits. Besides that, most of these cases were caused by the criteria used in the labeling process. Thus, the IoU is not considered appropriate for the measurements, and another metric is used: intersection over minimum area (Eq. 1). This metric still takes into account the overlap between the two bounding boxes, but it does not penalize the difference in size. Some examples of the values obtained with both metrics in different overlapping situations are shown in Fig. 8.

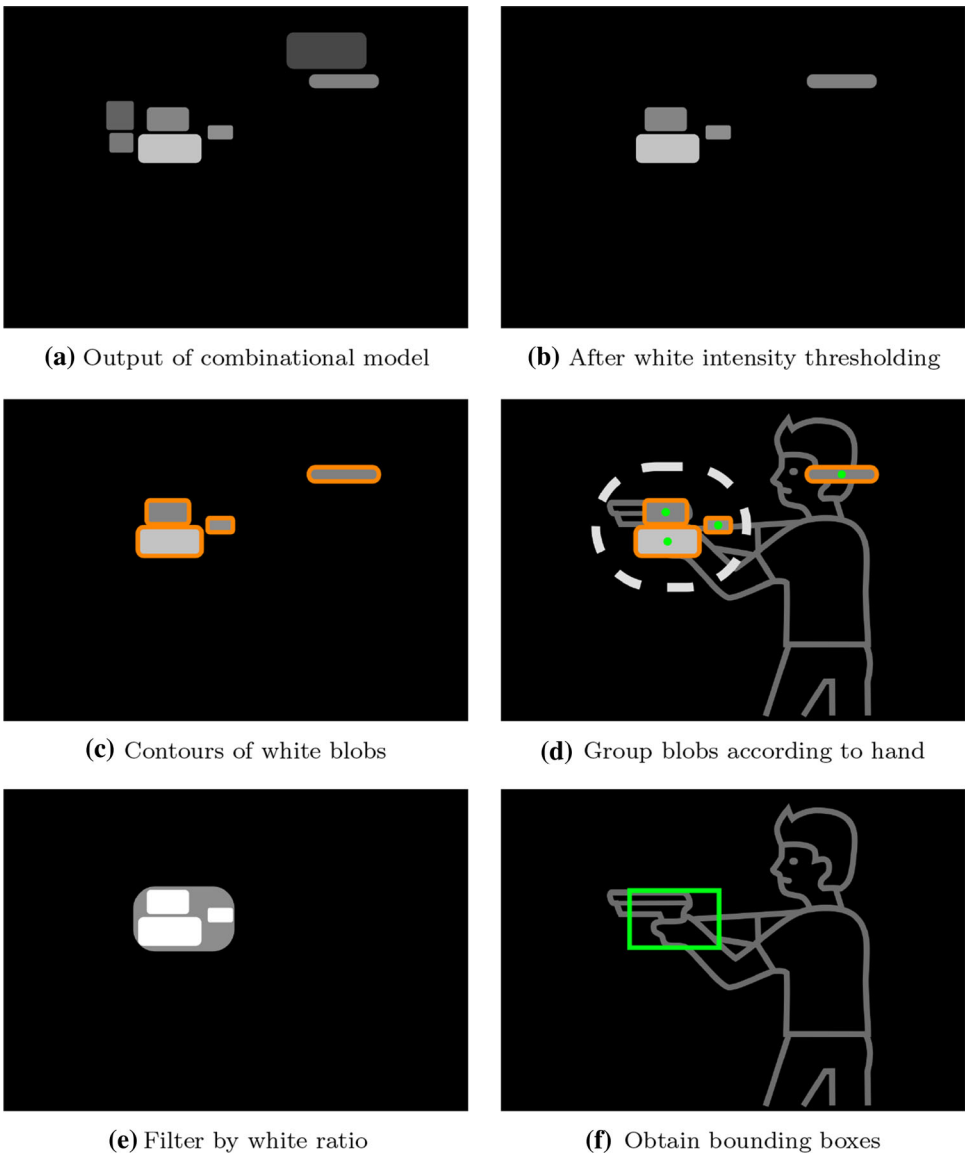
$$IoMin(A,B) = \frac{A \cap B}{\min(Area(A), Area(B))} \quad (1)$$

6 Results

A comparison between the results of the baseline detector and the proposed model has been made. Performance is measured on the test samples for both the known scenario seen in the training step and the unknown scenarios that were reserved for the comparison.

The IoMin threshold to classify a detection as true or false positive has been set to a 50% of overlap for all measurements.

The achieved results on the baseline detector are shown in Table 2. The output is the expected: The results for the test subset of the videos used in training are really good, obtaining both high precision and high recall. However,



(a) Output of combinational model

(b) After white intensity thresholding

(c) Contours of white blobs

(d) Group blobs according to hand

(e) Filter by white ratio

(f) Obtain bounding boxes

Fig. 6 Visualization of the steps in the algorithm that converts output images into bounding boxes. Best viewed in color

Fig. 7 Output examples where the detected bounding box is smaller than the labeled box. Rectangles in green represent the ground-truth bounding boxes, and red ones represent the detections considered false positives due to the small IoU value. Best viewed in color

when the detector is used on a new scenario, its performance is affected. In this case, the impact on the false-negative rate is more noticeable than the false-positive rate. This is likely to be caused by the differences in the weapons color and size, as well as the illumination on the scene and the distance respect to the camera. An example on this is shown in Fig. 9, where the known scenario used on training shows a black gun, close to the camera and

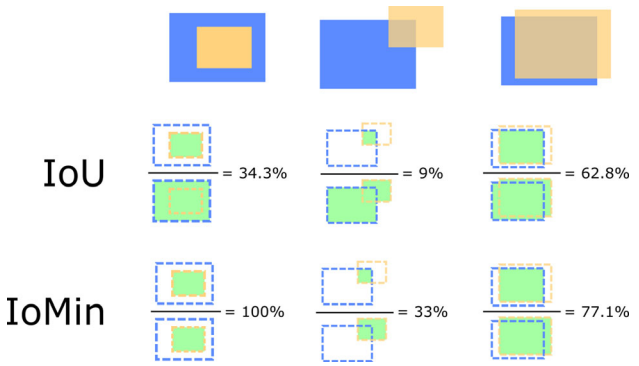


Fig. 8 Examples of IoU and IoMin values for 3 different overlaps between detection and groundtruth

recorded outdoors, while the unknown scenario used for test displays a gray gun, further away from the camera and recorded indoors. Also, the YOLOv3 network architecture is known to be worse when dealing with small objects compared to other two-step detection architectures [22].

If the confidence of the bounding boxes is thresholded, a precision–recall curve is obtained that represents how they are affected by the variations on the threshold. This curve makes it possible to obtain the average precision (AP) as the area under the curve. This metric is clearly worse on the scenarios that have significant differences to the ones used in training, and it should be improved by the proposed combinational model.

In order to obtain the same measurements for the combinational model, it is necessary to set a threshold on the white level, which is used on the conversion of the output to bounding boxes. The approach to obtain the best threshold possible is to calculate the precision–recall curves and APs on the test dataset for the known scenarios. In this way, the final test dataset that contains images of new scenarios is still not used in any part of the training process to make the final comparison fair.

For the dataset used in this paper, the AP measurement has been calculated with white thresholds between 55 and 90, as seen in Fig. 10. It shows a bi-modal distribution with peak values at 65 and 85 thresholds.

The resulting curves for those AP values are shown in Fig. 11. The variation on the white threshold makes the model either decrease the false-negative rate or the false-positive rate. The two peak threshold values (65 and 85) correspond to an improvement in precision and recall, respectively. Depending on the desired output result, one of them will be chosen.

Once the threshold has been selected, the performance measurements for comparison can be calculated. In the following, we have included results for both thresholds 65 and 85, depending on whether false positives or false

Table 2 Baseline detector performance measured over the test subsets of both the known scenarios and the unknown ones

	Known scenarios			Unknown scenarios		
	Global	YouTube	Watch Dogs 2	Global	Gun Movies	Watch Dogs 2
TP	236	120	116	331	63	268
FP	9	8	1	36	28	8
FN	46	21	25	375	80	295
Precision	96.3%	93.8%	99.1%	90.2%	69.2%	97.1%
Recall	83.7%	85.1%	82.3%	46.9%	44.1%	47.6%
AP	**83.3%**			**45.0%**		

Bold values indicate the global performance of each method in each scenario

Fig. 9 Examples of images from the train dataset with known scenarios (**a**) and the test dataset with unknown scenarios (**b**)

(a) Sample image from YouTube dataset

(b) Sample image from Gun Movies Database

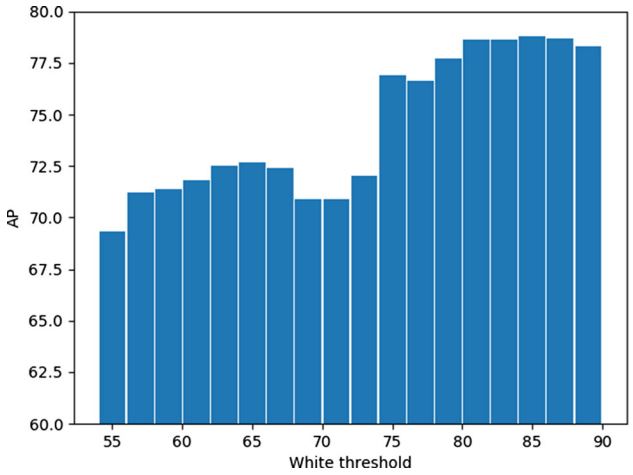


Fig. 10 AP measurements of the combinational model on known scenarios, depending on the white threshold applied. A bimodal distribution is shown with optimal peak values on 65 and 85

negatives should be avoided. Tables 3 and 4 depict the results, and Fig. 12 shows the precision–recall curves compared to the detector one.

As results show, the baseline detector obtains an AP of 45% on the scenarios that differ from the ones it was trained with. However, with the proposed combinational model that integrates the pose, an AP of 62.5% is achieved when the white level of the output image is thresholded to values higher than 65. These results also show a maximum improvement of a 17.5% on the AP metric. Furthermore, as seen in Table 4, with a threshold set to 85 on the white level, the detection capabilities of the baseline detector are kept, while reducing the false-positive and false-negative rates.

Nevertheless, when the combinational model is applied to the known scenarios, there is no improvement over the

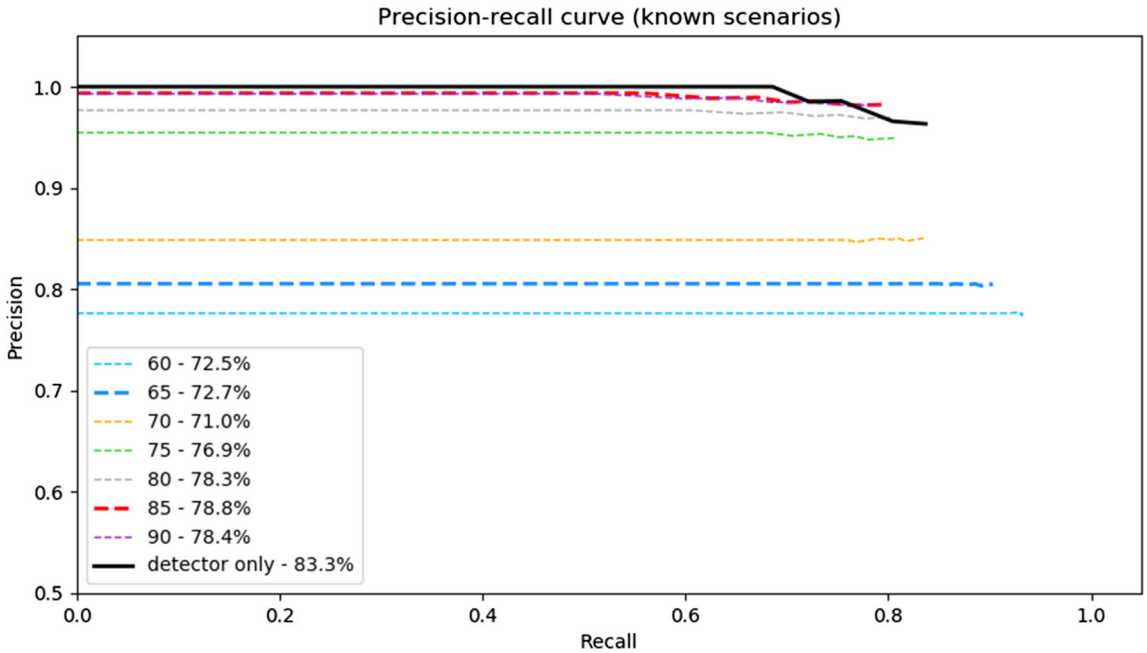


Fig. 11 Precision–recall curves of the combinational model on known scenarios. Several thresholds are used to determine which one is optimal comparing them with the baseline detector performance

Table 3 Combinational model performance measured over the test subsets of both the known scenarios and the unknown ones, thresholding the white level to values higher than 65

	Known scenarios			Unknown scenarios		
	Global	YouTube	Watch Dogs 2	Global	Gun Movies	Watch Dogs 2
TP	252	121	131	597	71	526
FP	61	49	12	197	2	195
FN	27	21	6	122	72	50
Precision	80.5%	71.2%	91.6%	75.2%	97.3%	73%
Recall	90.3%	85.2%	95.6%	83%	49.7%	91.3%
AP	**72.7%**			**62.5%**		

Bold values indicate the global performance of each method in each scenario

Table 4 Combinational model performance measured over the test subsets of both the known scenarios and the unknown ones, thresholding the white level to values higher than 85

	Known scenarios			Unknown scenarios		
	Global	YouTube	Watch Dogs 2	Global	Gun Movies	Watch Dogs 2
TP	221	108	113	332	63	269
FP	4	4	0	11	1	10
FN	57	33	24	374	80	294
Precision	98.2%	96.4%	100%	96.8%	98.4%	96.4%
Recall	79.5%	76.6%	82.5%	47%	44.1%	47.8%
AP	**78.8%**			**45.8%**		

Bold values indicate the global performance of each method in each scenario

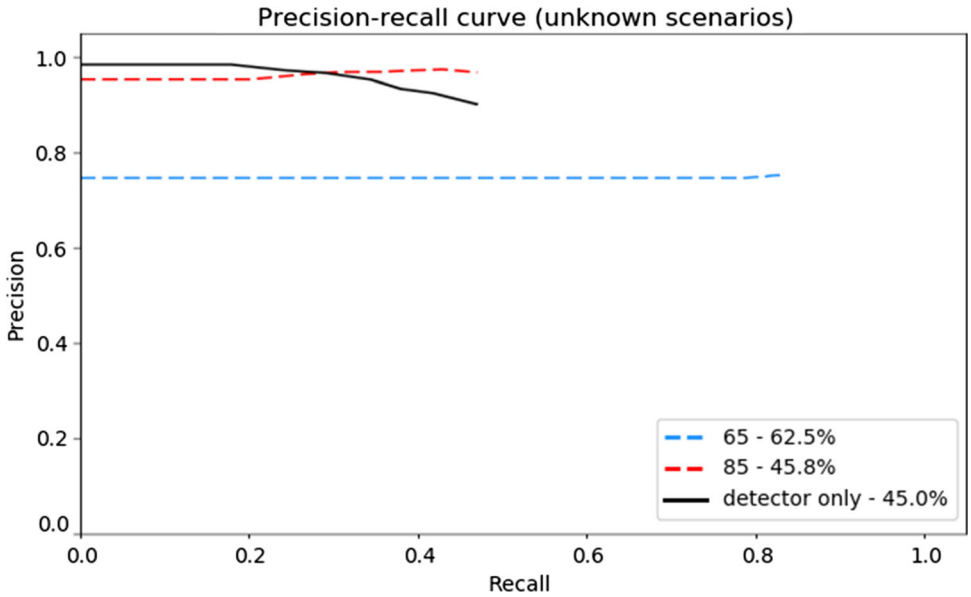


Fig. 12 Precision–recall curves of the combinational model on unknown scenarios. The selected thresholds are applied to compare the performance with the baseline detector (solid black line)

detector. The main reasons for this might be that the baseline detector is already trained on similar scenarios and also that the pose might provide misleading information when people on the image extend their arms to point at something.

Overall, the proposed combinational model provides an improvement over the baseline detector for unknown scenarios depending on the threshold selected, which heavily affects the precision and/or recall increase rates and how they are balanced.

Finally, to check the performance of the proposed method against other handgun detection approaches which also use body pose information, a comparison with Lomas [19] and Basit et al. [20] works has been made. Table 5 depicts a summary of the performance of all methods. In unknown scenarios, all methods improve the results of the baseline detector except the work proposed by Basit et al. This might be explained by the fact that this method is based on a filter after the initial detections, removing bounding boxes which are not associated with a person,

Table 5 Performance comparison between combinational model and other methods

	Known scenarios			Unknown scenarios		
	Precision (%)	Recall (%)	AP (%)	Precision (%)	Recall (%)	AP (%)
YOLOv3	96.3	83.7	83.3	90.2	46.9	45.0
Proposed method (threshold = 65)	80.5	90.3	72.7	75.2	83.0	**62.5**
Proposed method (threshold = 85)	98.2	79.5	78.81	96.8	47.0	45.8
Lomas	98.3	83.4	83.24	94.19	55.1	53.35
Basit et al.	88.62	78.7	74.52	93.33	45.61	43.99

Bold values indicate the summarize the global performance of each method in each scenario

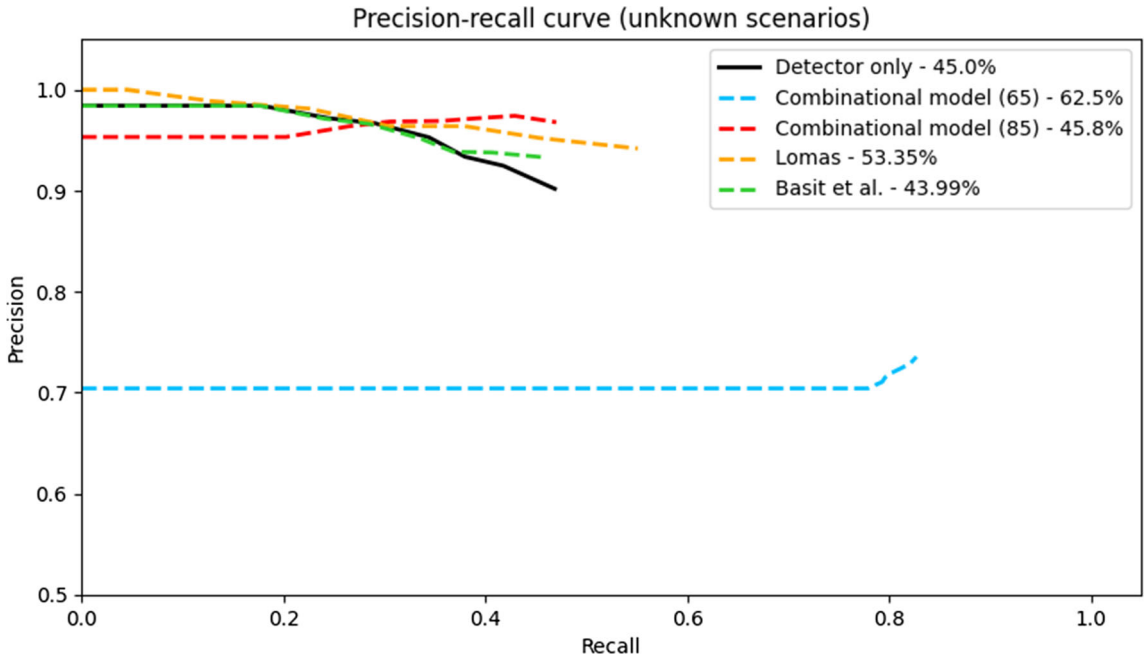


Fig. 13 Comparison between precision–recall curves of the combinational model and other methods on unknown scenarios

without adding new correct detections. Applying this method on a new scenario leads to the removal of some valid bounding boxes, decreasing the performance of the initial detector. On the other hand, an explicit trade-off between false positives and false negatives is shown comparing the combinational model with the work proposed by Lomas. Depending on the threshold applied on the proposed model, the performance is better or worse. A visual representation of this effect is shown in Fig. 13. Nevertheless, the proposed combinational model outperforms the existing methods due to the direct integration of the human pose into the architecture.

7 Conclusions

The results obtain an improvement over a handgun detector by leveraging the human pose. The proposed network provides successful results, with a maximum improvement of a 17.5% in AP of the proposed combinational model over the baseline handgun detector (*YOLOv3*). Our work shows that the individual's pose can be effectively used to improve the threat detection accuracy. Furthermore, a variable threshold is available to make this method favor new detections or reduce the false-positive ratio, so it would be useful in a real use case scenario. These kind of techniques might represent an improvement in the security against potential threats in buildings where CCTV cameras are already installed.

Acknowledgements This work was partially funded by projects TIN2017-82113-C2-2-R by the Spanish Ministry of Economy and Business and SBPLY/17/180501/000543 by the Autonomous Government of Castilla-La Mancha and the ERDF.

Funding Open Access funding provided thanks to the CRUE-CSIC agreement with Springer Nature.

Declarations

Conflict of Interest The authors declare that they have no conflict of interest.

References

1. Enríquez F, Soria LM, Álvarez-García JA, Caparrini FS, Velasco F, Deniz O, Vallez N (2019) Vision and crowdsensing technology for an optimal response in physical-security. In: International conference on computational science, pp 15–26

2. Vállez N, Bueno G, Déniz O (2013) False positive reduction in detector implantation. In: Conference on artificial intelligence in medicine in Europe (AIME), pp 181–185
3. Luvizon DC, Picard D, Tabia H (2018) 2D/3D pose estimation and action recognition using multitask deep learning. CoRR arXiv:abs/1802.09232
4. Nercessian S, Panetta K, Agaian S (2008) Automatic detection of potential threat objects in X-ray luggage scan images. In: 2008 IEEE conference on technologies for homeland security, pp 504–509
5. Xiao Z, Lu X, Yan J, Wu L, Ren L (2015) Automatic detection of concealed pistols using passive millimeter wave imaging. In: 2015 IEEE international conference on imaging systems and techniques (IST), pp 1–4
6. Tiwari RK, Verma GK (2015) A computer vision based framework for visual gun detection using Harris interest point detector. Procedia Comput Sci 54:703–712
7. Olmos R, Tabik S, Herrera F (2018) Automatic handgun detection alarm in videos using deep learning. Neurocomputing 275:66–72
8. Ren S, He K, Girshick R, Sun J (2015) Faster R-CNN: towards real-time object detection with region proposal networks. Adv Neural Inf Process Syst 91–99
9. Gelana F, Yadav A (2019) Firearm detection from surveillance cameras using image processing and machine learning techniques. In: Smart innovations in communication and computational sciences, pp 25–34
10. Grega M, Matiolański A, Guzik P, Leszczuk M (2016) Automated detection of firearms and knives in a CCTV image. Sensors 16(1):47
11. Olmos R, Tabik S, Lamas A, Pérez-Hernández F, Herrera F (2019) A binocular image fusion approach for minimizing false positives in handgun detection with deep learning. Inf Fus 49:271–280
12. Vallez N, Velasco-Mata A, Corroto JJ, Deniz O (2019) Weapon detection for particular scenarios using deep learning. In: Iberian conference on pattern recognition and image analysis, pp 371–382
13. Thurau C, Hlavac V (2008) Pose primitive based human action recognition in videos or still images. In: 2008 IEEE conference on computer vision and pattern recognition, pp 1–8
14. Reiss A, Hendeby G, Bleser G, Stricker D (2010) Activity recognition using biomechanical model based pose estimation. In: Smart sensing and context, pp 42–55
15. Chevalier G (2016) LSTMs for human activity recognition. https://github.com/guillaume-chevalier/LSTM-Human-Activity-Recognition. Accessed 01 Aug 2020
16. Eiffert S (2018) Activity Recognition from 2D pose using an LSTM RNN. https://github.com/stuarteiffert/RNN-for-Human-Activity-Recognition-using-2D-Pose-Input. Accessed 01 Aug 2020
17. Cao Z, Hidalgo Martinez G, Simon T, Wei S, Sheikh YA (2019) OpenPose: realtime multi-person 2D pose estimation using part affinity fields. In: IEEE transactions on pattern analysis and machine intelligence, pp 1–1
18. Boualia SN, Essoukri BAN (2019) Pose-based Human Activity Recognition: a review. In: 2019 15th international wireless communications mobile computing conference (IWCMC), pp 1468–1475
19. Lomas V (2020) Handgun detection in video. Master's thesis, Escuela Superior de Informática (UCLM)
20. Basit A, Munir MA, Ali M, Werghi N, Mahmood A (2020) Localizing firearm carriers by identifying human-object pairs. In: 2020 IEEE international conference on image processing (ICIP), pp 2031–2035. IEEE
21. Vallez N, Velasco-Mata A, Cotorro JJ, Deniz O (2019) Es posible entrenar modelos de aprendizaje profundo con datos sintéticos? In: XL Jornadas de Automática, pp 859–865
22. Nguyen ND, Do T, Ngo TD, Le DD (2020) An evaluation of deep learning methods for small object detection. J Electr Comput Eng. https://doi.org/10.1155/2020/3189691