BULLETIN OF THE POLISH ACADEMY OF SCIENCES
TECHNICAL SCIENCES, Vol. 71(6), 2023, Article number: e147340
DOI: 10.24425/bpasts.2023.147340

A deep learning method for hard-hat-wearing detection based on head center localization

Bartosz WÓJCIK[1], Mateusz ŻARSKI[1]*, Kamil KSIĄŻEK[1], Jarosław A. MISZCZAK[1], and Mirosław J. SKIBNIEWSKI[1,2]

[1] Institute of Theoretical and Applied Informatics, Polish Academy of Sciences, 44-100 Gliwice, Poland
[2] A. James Clark School of Engineering, University of Maryland, College Park, MD 20742-3021, USA

Abstract. In recent years, a lot of attention has been paid to deep learning methods in the context of vision-based construction site safety systems. However, there is still more to be done to establish the relationship between supervised construction workers and their essential personal protective equipment, like hard hats. A deep learning method combining object detection, head center localization, and simple rule-based reasoning is proposed in this article. In tests, this solution surpassed the previous methods based on the relative bounding box position of different instances and direct detection of hard hat wearers and non-wearers. Achieving MS COCO style overall AP of 67.5% compared to 66.4% and 66.3% achieved by the approaches mentioned above, with class-specific AP for hard hat non-wearers of 64.1% compared to 63.0% and 60.3%. The results show that using deep learning methods with a humanly interpretable rule-based algorithm is better suited for detecting hard hat non-wearers.

Key words: hard hat; personal protective equipment; construction safety; deep learning; keypoint R-CNN.

1. INTRODUCTION

Construction is one of the most dangerous industries. Together with manufacturing, it leads in the number of non-fatal and fatal accidents [1, 2]. Among accidents occurring on a construction site, particularly dangerous are those in which the head is injured. While in non-fatal incidents, the share of head injuries is only about 7%, in fatal ones, they account for over 30% of all occurrences [1]. This makes them a significant problem that has a crucial impact on the safety of construction workers. The most common head injury occurring is TBI – traumatic brain injury [3]. The injury itself can be fatal [4, 5] and occurs when the rapid acceleration or deceleration of the head causes the brain to move and collide with the skull. It has been identified that the most common causes of TBIs on construction sites are falls and being struck by or against an object [3, 6, 7].

Recognition of head injuries as a significant factor influencing the safety of the construction site has led to the legal regulation of the approach to Personal Protective Equipment (PPE) around the world [8, 9]. These regulations oblige the employer to provide personal protection measures for employees. To help ensure the appropriate usage of PPE, various methods are based on wearable sensors and vision monitoring. Vision-based methods use the on-site CCTV systems [10] or UAVs [11] for obtaining the image data from the construction site and pair it with shallow [12] or deep learning algorithms [13] for real-time hard hats detection.

The vision-based methods appear to be straightforward. However, this problem is more complex than it might seem. Simply finding hard hats and workers in the image is not enough. The relationship between these instances has to be established. The real problem is not to find people who correctly wear PPE but to find people who do not comply with the safety rules by not using it.

The currently used solutions are based on direct detection of hard hat non-wearers or separate detection of workers and hard hats. According to a study performed in [14], the second approach achieves worse results in direct comparison. This is likely caused by the fact that reasoning based on the relative position of instances on the image is too simple to capture the relationship between people and their head protection. On the other hand, the first approach still suffers from high inter-class similarity, as instances of hard hat non-wearers differ from hard hat wearers only by fine detail, presence of the hard hat. Some custom approaches are based on face detection [15] or human pose estimation [16–18]. However, these work only in some situations, as additional features have to be visible.

In this article, a novel approach to hard-hat-wearing detection is proposed. It couples object detection with human joint localization and rule-based reasoning. However, rather than using known models for human pose estimation, the model was trained to locate the person's head while finding instances of people and hard hats simultaneously. This unique problem formulation provides a way to determine the correct relationship between workers and their head protection. At the same time, it achieves this with simple human–interpretable rules. Additionally, it overcomes the drawbacks of currently used approaches as direct hard-hat-wearing detection suffers from high inter-

*e-mail: mzarski@iitis.pl

Manuscript submitted 2023-05-23, revised 2023-08-22, initially accepted for publication 2023-09-04, published in December 2023.

class similarity, whereas solutions based on bounding box relative position lack information to reliably establish worker–hard hat relationship. This results in better performance, especially regarding the detection of hard hat non-wearers. The latter is critical from a construction safety point of view, as detecting people who do not wear hard hats is the actual task. We believe that this kind of work is crucial to developing reliable construction sites safety systems based on deep learning.

The rest of this paper is organized as follows. In Section 2 literature review regarding deep learning in the context of construction safety and hard-hat-wearing detection is presented. Section 3 describes the research methodology. Dataset, training and model tuning is described in Section 4, and solution evaluation is presented in Section 5, and in Section 6, comparisons with other methods have been made. Finally, all the results are discussed in Section 7, and conclusions drawn with suggestions for future work are presented in Section 8.

2. RELATED WORK

Machine learning has found applications in many fields related to Civil Engineering. The methods used vary from simple artificial neural networks [19] to complex, image-based deep neural networks [20, 21]. Additionally, such an approach as transfer learning makes the network even easier to train and deploy [22, 23].

It is no different in construction safety, as deep learning finds application in a variety of its aspects. In [24] authors presented a framework enabling safety monitoring with computer vision, review of computer vision applications for behaviour–based safety was performed in [25]. A comprehensive review of computer vision in construction safety and management was carried out in [26].

2.1. Vision-based detection in construction safety

In [27] a deep learning-based framework to detect work performed by unauthorized workers was proposed. The framework, composed of three modules: key video clips extraction, trade recognition and worker competency judgment can extract and identify activities performed on the construction site, identify workers, and check in the predefined database whether they are authorized to carry out this work. In [28] authors developed a method for safety harness wearing detection. They paired Faster R-CNN [29] with custom-developed CNN to detect workers and verify if the safety harness is worn. In [30] Mask R-CNN [31] combined with developed overlapping detection module (ODM) to recognize workers traversing structural supports to prevent falls was used. The presented ODM can determine the relationship between workers and structural supports based on mask relative positions. Authors of [32] proposed an approach for safety officer trajectory tracking on the construction site. In this work, the authors use YOLOv3 [33] for safety officer detection and Kalman filter with Hungarian matching algorithm for tracking. Another research [34] presented an approach utilizing a spatial and temporal attention pooling network that enables worker identification. On the other hand, [35] used human-object interaction recognition to claim whether workers wear the correct PPE during tool usage. In [36] authors developed a real-time system capable of detecting if workers enter hazardous areas, and in [37] Mask R-CNN based object correlation detection for mobile scaffolding safety checks was developed.

2.2. Hard-hat-wearing detection

Detection of hard hat wearers and non-wearers also has been addressed recently. In [13] Faster R- CNN framework for this task was used and the impact of different visual conditions on the detection performance specific to construction sites was analyzed. A multi-staged method composed of a histogram of the orientated gradients and colors was presented in [38]. In the first stage, workers are detected in a video feed. Head protection presence in the upper body part is established by an object detector coupled with color-based classifier. Authors in [39] described a model based on the single shot detector framework [40] and provided a benchmark dataset containing 3174 images. In [14] three different approaches to detecting PPE based on YOLOv3: detecting PPE and people to then establish workers – PPE relationship based on bounding box relative position, detecting PPE wearers and non-wearers directly and finally detecting only people to determine if they are wearing PPE with a different model was compared. In [41] researchers focused on real-time processing with MobileNet [42] architecture and in [43] tested YOLOv5 for this application.

2.3. Shortcomings of existing hard-hat-wearing solutions

Most of the solutions used to detect the wearing of the hard hats presented in this section fall into one of the two general categories:

- detection of people or people and hard hats, wearing head protection is determined in different steps according to rules or another model,
- detecting hard-hat-wearers and non-wearers as separate classes.

Both categories suffer their problems. The main problem of the first one is to establish the correct relationship between the person and the hard hat. Reasoning based on the bounding box relative position seems too simple to capture it and the solutions human pose estimation and geometrical dependencies [16–18] fail to set it properly for all cases.

The second category suffers from a significant inter-class similarity problem. The person wearing a hard hat and the person not wearing it are a subclass of a person's class. This problem is well known in subcategory classification [44–46] as it is harder to develop a model that can correctly distinguish fine details between subcategories. For this reason, the best-performing models in this group look for the human head instead of the whole person [27, 39, 41, 43], making them less suitable for direct transfer learning from well-trained person detection models.

Additionally, in both cases, researchers tend to disregard situations where a person is partially detected, or the head is obstructed, and it is not possible to tell if a person is wearing head protection. In the majority, that person is incorrectly classified as a worker without a hard hat. A good example could be the

Pictor-v3 dataset provided by [14], in which even a person visible from waist down is annotated as hard hat non-wearer when in reality it cannot be determined. This is a severe oversight and makes it impossible to apply such a solution in practice. The real problem is finding people who are not following the safety rules reliably.

3. METHODOLOGY

3.1. Head center based hard-hat-wearing detection

To address the challenges of solutions based on a hard hat and person detection, we are introducing a new approach to detecting hard hat wearers based on workflow focusing on skeleton head joint localization (Fig. 1).

In our proposed solution, models perform two tasks in parallel – the first one is responsible for hardhat detection, while the second one detects the human silhouette and marks the head center. Then, the proposed algorithm checks whether the head center and the hardhat bounding box coincide. Accordingly, the detected person is classified as hardhat wearer or non wearer. The algorithm simplicity allows checking if the helmet is in most likely the proper position with respect to the head, while keeping computational complexity to a minimum.

In the context of deep learning, keypoints are understood as points of interest in the image. Their strongest advantage is that they are invariant for transformations, so scaling will not affect them.

The most common keypoint application is human pose estimation, where they represent human joints. However, instead of using an existing human pose estimation model like [16, 17] or [18], as stated before, we define only one joint representing the localization of the human head. This model formulation enables us to correctly establish the relationship between a hard hat and a hard hat wearer, with a simple rule-based algorithm presented in Algorithm 1.

Algorithm 1 Head center-based hard-hat-wearing detection algorithm

input: *inst* – list containing person (*p*) and hard hat (*hh*) instances
output: *newInst* – list containing person (*p*), hard hat wearer (*hhw*) and hard hat non-wearer (*hhnw*) instances

```
for each p ∈ inst do
    p.copyTo(newInst)
    if p.hasHeadKp then
        p ← hhnw
        for each hh ∈ inst do
            if p.headKp ∈ hh.bBox then
                p ← hhw
                break loop
            end if
        end for
    end if
end for
return newInst
```

3.2. Architecture

The proposed solution was implemented based on the Generalized Region-based Convolutional Neural Network (Fig. 2) framework described for the first time in the Mask R-CNN [31]. This natural and flexible extension to Faster R-CNN [29] enables the creation of models capable of performing a variety of tasks simultaneously. In this case, we dropped the part of the network responsible for mask prediction leaving only object detection and joint localization (making it Keypoint R-CNN).

Three models were implemented, each with a different backbone network featuring Feature Pyramid Network [47], although they were all ResNet [48] variants (The ResNet part of Fig. 2). The ResNet model was used because of its deep architecture, made trainable through the use of residual blocks, which allows for high obtainable metrics. Two of the backbones

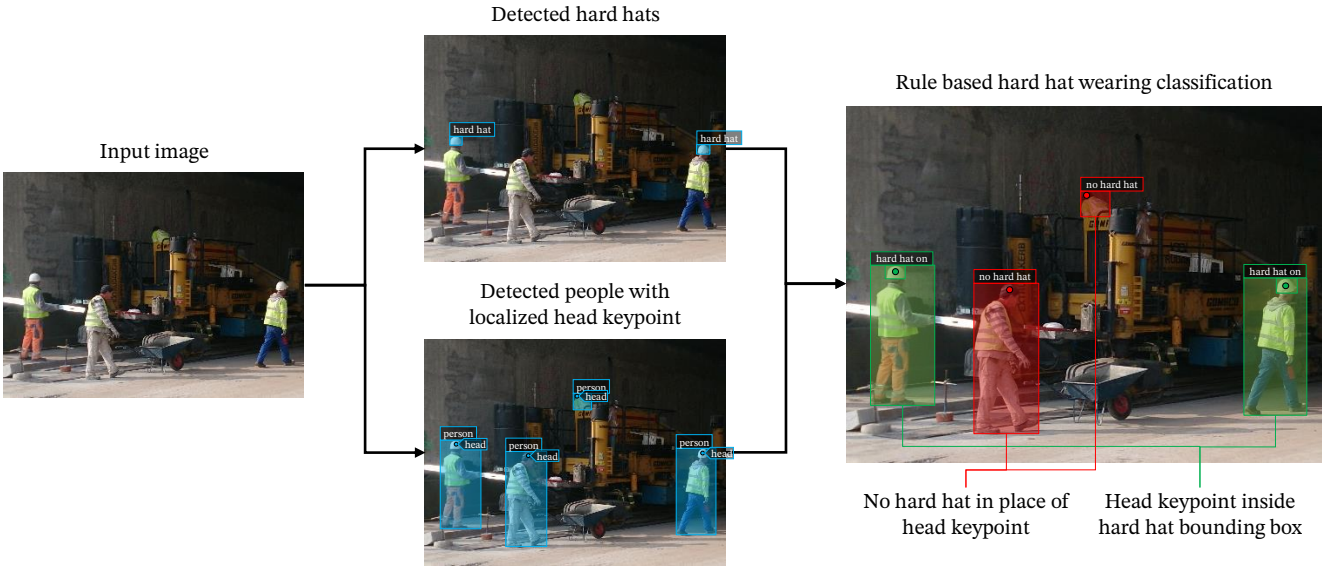


Fig. 1. Workflow of the proposed solution, showing the parallel operation of the two algorithm models and the final hardhat wearer classification

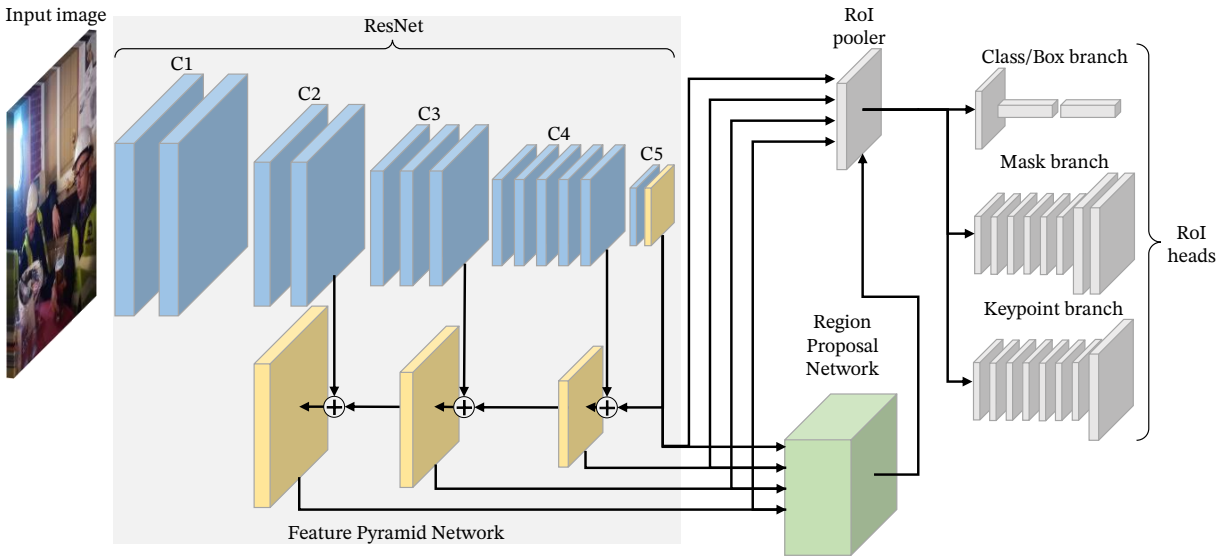


Fig. 2. Generalized Region-based Convolutional Neural Network with ResNet based FPN backbone

were built using ResNet architecture with layers depth of 50 and 101 (denoted as R50 and R101). Except for the use of residual blocks, these architectures are similar to sequential CNNs. The last one was built with ResNeXt [49] architecture with layers depth of 101, block cardinality of 32 and depth of 8 (denoted as X101). This model is distinguished by its use of a deep microarchitecture, so that operations in the network are performed in fewer, parallel blocks, albeit containing similar number of layers. At the same time, this design does not significantly increase the complexity of the computations performed by the network.

A network head (far right part of Fig. 2) was built combining standard Faster R-CNN with FPN classification and box regression branch, as proposed in [47]. Additional keypoint conv-deconv upscaling branch described in [31] was also added. Detailed architecture of Keypoint R-CNN head used (as opposed to general Mask R-CNN in Fig. 2), containing both FPN and keypoint branches is shown in Fig. 3.

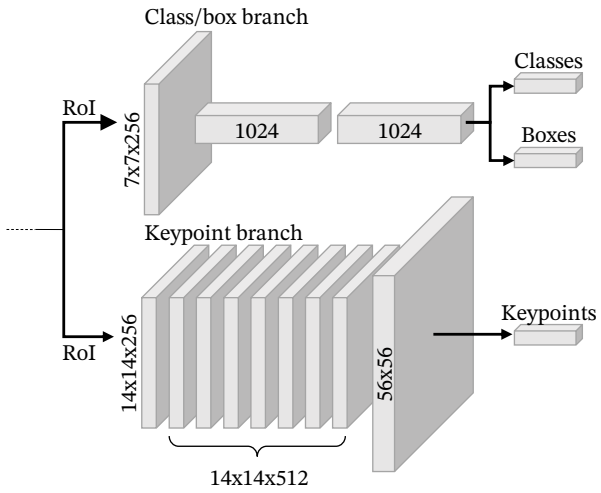


Fig. 3. Keypoint R-CNN head derived from standard Mask R-CNN composed of standard Faster R-CNN with FPN [47] classification/box regression branch and keypoint conv-deconv upsample branch [31]

3.3. Loss function

The Generalized R-CNN defines multi-task loss as the sum of losses of each task. For our network – Keypoint R-CNN (with two-part head Fig. 3) it can be expressed by the following formula

$$L = L_{\text{cls,RPN}} + L_{\text{bbox,RPN}} + L_{\text{cls,head}} + L_{\text{bbox,head}} + L_{\text{kp,head}} \,. \tag{1}$$

The loss function components were adopted from [29] and [31] and then slightly modified. $L_{\text{cls,RPN}}$ and $L_{\text{bbox,RPN}}$ are standard classification and bounding box losses of region proposal network for detecting objects in the foreground and background. Classification loss provides information on whether the proposed region contains an object or not, and bounding box loss checks the bounding box actual alignment with the ground truth object. $L_{\text{cls,head}}$, $L_{\text{bbox,head}}$, and $L_{\text{kp,head}}$ are losses computed for each sampled proposed region. The first two are computed similarly to RPN losses, but for true regression targets and their actual class and position prediction. Lastly, keypoint branch loss was set as adjusted for head center detection mask loss as in [31].

3.4. Evaluation metrics

Average precision (AP) and mean average precision (mAP) are the most commonly used metrics for evaluating object detectors. Both metrics were developed to address a need to quantify both classification and localization performance simultaneously. AP averages precision (p) values across recall (r) range for a specific class, whereas mAP provides an overall metric by averaging APs for the collection of classes. Where precision

$$p = \frac{\text{true positives}}{\text{true positives} + \text{false positives}} \tag{2}$$

measures percentage of correct predictions out of all predictions and recall

$$r = \frac{\text{true positives}}{\text{true positives} + \text{false negatives}} \tag{3}$$

percentage of instances found.

AP has a value from 0 to 1 as both precision and recall fall in the same range and can be interpreted as an area under a precision-recall curve. Thus, AP can be defined by the following formula

$$\text{AP} = \int_0^1 p(r)\,\mathrm{d}r. \tag{4}$$

Therefore, mAP also ranges from 0 to 1 and can be calculated accordingly

$$\text{mAP} = \frac{1}{n}\sum_{i=1}^{n} \text{AP}_i\,, \tag{5}$$

where AP_i is AP calculated for is i–th element of n element collection of classes.

The theoretical formula for AP calculation presented in Equation 4 is impractical as it requires the precision-recall relationship to be a known continuous function. Instead, the precision-recall curve is approximated by sampling precision values for defined recall thresholds, and AP is calculated by numerical integration. Additionally, precision values are interpolated to reduce the influence of minor variations in instance rankings. This interpolated AP (APint) can be expressed by the following formula

$$\text{AP}_\text{int} = \frac{1}{m}\sum_{i=1}^{m} p_{int}(r_{thr,i}), \tag{6}$$

where, $\text{r}_\text{thr,i}$ is i–th recall threshold out of m element recall threshold collection and interpolated precision

$$p_{int}(r_{thr}) = \max_{\widetilde{r}_{thr} \geq r_{thr}} p(\widetilde{r}_{thr}) \tag{7}$$

is the maximal precision value out of precision values achieved at recall thresholds equal or greater than r_thr.

However, for the detection performance measurements in this article, MS COCO style AP metrics were used. These are stricter and thus provide more insight into the detector performance. Traditionally, AP and mAP metrics are computed at the intersection over union (IoU) of 50%. That means that detection is treated as positive if the ratio between a common part of its bounding box and ground truth (intersection) and the area encompassed by both (union) is greater than or equal to 0.5. Whereas MS COCO style AP (AP_COCO) averages 101- point AP_int over ten IoU thresholds, from 50% to 95% with the step of 5%. This cloud is expressed by the following formula

$$\text{AP}_{COCO} = \frac{1}{j}\sum_{i=1}^{j} \text{AP}_{int,i}\,, \tag{8}$$

where, $\text{AP}_\text{int,i}$ is interpolated AP defined in Equation 6 computed for i–th IoU threshold out of j element collection of IoU thresholds. Additionally, MS COCO enables the usage of similar metrics for head center evaluation. However, one fundamental distinction as opposed to bounding boxes – IoU cannot be computed for point representations. Therefore, positive detection is determined with the use of the object keypoint similarity (OKS) metric.

OKS computes the Euclidean distance between the detected keypoint and its ground truth, normalized by the scale of the bounding box. The exact formula can be expressed in the following manner

$$OKS = \frac{\sum_i \exp\{-d_i^2/2s^2k_i^2\}\delta(\vartheta_i > 0)}{\sum_i \delta(\vartheta_i > 0)}\,, \tag{9}$$

where s is an object scale computed from the bounding box, d_i is keypoint-ground truth distance for the i–th keypoint, ϑ_i is visibility flat that takes positive values if i–th keypoint is indicated. A value k_i is a constant specific to the i–th keypoint, according to the following formula

$$k_i = 2\sigma_i\,. \tag{10}$$

The value σ_i is the i-th keypoint standard deviation computed relative to the object scale over a set of redundantly annotated images. In our case, the head center σ value was set to 0.026 according to the value provided for AI Challenge Keypoint Dataset [50]. There were not enough redundantly annotated images in our dataset to compute k constant. This value aligns well with MS COCO values form on head features.

MS COCO style metrics also include AP computed at IoU/OKS of 50% and 75%, denoted as AP^{50} and AP^{75}, and AP computed for objects at different scales, denoted as AP_S, AP_M and AP_L. Moreover, creators of these metrics abandoned the distinction between AP and mAP as both are, in fact, a mean value but computed over different collections. Instead, the difference between these should be well stated in the context.

4. IMPLEMENTATION

4.1. Dataset

A publicly available dataset [51] was used for the training and testing of our solution. This dataset contains 7035 images of different sizes, split into a train (5269 images) and test (1766 images) part. The average image size is 358×476 px for the train and 360×480 px for the test part, with images of size 332×499 px being the largest group in both parts. A full breakdown of image size distribution in both parts of the dataset is presented in Fig. 4.

As available annotations were incompatible with our solution, it has been labelled as a hard hat and person with head center detection, which resulted in over 55 thousand object instances in MS COCO format [52]. A detailed breakdown for the training and testing dataset, broken down by category, subcategory, and according to the bounding box area, is presented in Table 1.

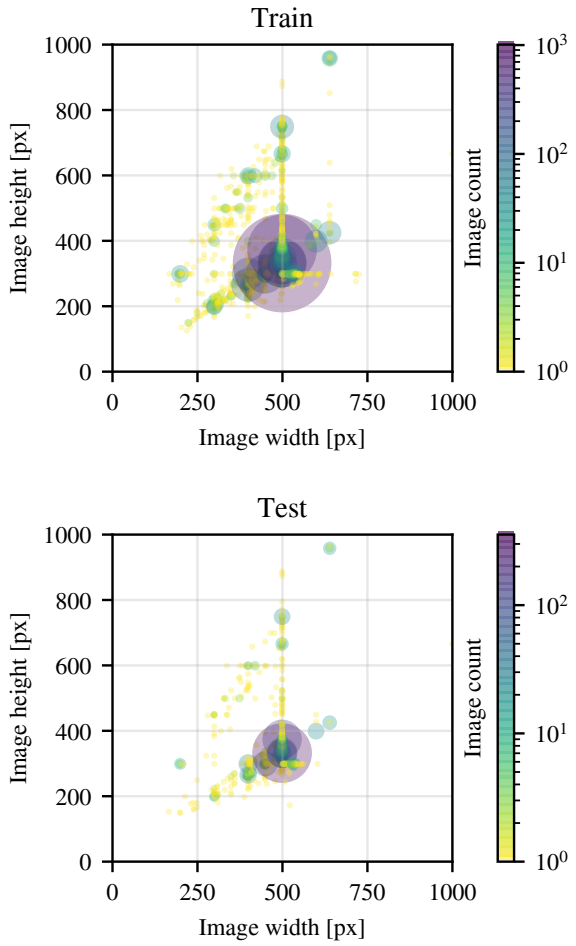


Fig. 4. Image size distribution in training and testing part of dataset [51]

Table 1
Breakdown of object instances in training and testing part of dataset

Instances	all	small	medium	large
Train:				
hard hat	17,741	11,340	5,922	479
person	23,882	2,805	9,729	11,348
- w/head center	22,983	2,602	9,232	11,149
- w/head center and hard hat	16,700	1,715	6,459	8,526
Test:				
hard hat	5746	3727	1841	178
person	7992	1077	3200	3715
- w/head center	7775	1036	3065	3674
- w/head center and hard hat	5353	509	2071	2773

Where:
all – sum of all available instances in dataset
small – instances with bounding box area smaller than 1024 px
medium – instances with bounding box area between 1024 and 9216 px
large – instances with bounding box area greater than 9216 px

Compared to original annotations, ours contained almost 4 thousand more person instances in training and over 1.3 thousand more person instances in the testing part. This difference probably comes from the number of small instances, as the dataset originally contained annotations of people heads that are smaller than the silhouette of a whole person.

4.2. Training

Transfer learning, which is a popular technique in deep learning, was used to accelerate training. All backbones were initiated from model weights trained in human pose estimation for around 37 epochs on MS COCO 2017 dataset [53], on which they achieved scores close to state-of-the-art models. Additionally, the first two layers of the backbone were frozen, as they extract general features that do not have to be retrained.

Each model was then trained on the annotated train part of [49] dataset for 50 thousand steps with a batch size of 4, resulting in almost 38 training epochs. Following data augmentation was used: images could be randomly flipped horizontally, vertically, or in both axes simultaneously. Furthermore, the shorter edge was randomly resized to 640, 672, 704, 736, 768 or 800 pixels. At the same time, the dimension of the longer edge could not exceed 1333 pixels. The value of loss function and classification accuracy measured for each model throughout training steps is shown in Fig. 5.

The hyperparameters were set according to [53]. Thus the original data set divided into training and testing was kept. Instead of monitoring loss function value on the validation dataset, models were evaluated each 5 thousand training iterations on both training and testing datasets to ensure lack of overfitting in the final model. MS COCO style AP for all models computed on train and test datasets are shown in Fig. 6.

A series of experiments were performed using the test part of the dataset to evaluate trained models. Additionally, due to significant size variations of the dataset and already used augmentation in training, the shorter edge of the test images was resized to 800 pixels. At the same time, the dimension of the longer edge could not exceed 1333 pixels during inference on the test dataset.

4.3. Detection threshold moving

The algorithm presented in Algorithm 1 does not take the detection probability score into account. It makes it vulnerable to the detection confidence threshold, as low-scoring hard hat instances would be treated in the same manner as instances detected with nearly 100% confidence. This means that the detector used for hard-hat-wearing evaluation has to be properly tuned.

The confidence threshold below which objects are not treated as positive detection is called the decision threshold. The process of finding the optimal threshold is referred to as detection threshold moving. There are a few strategies for this task depending on the preferences. In this case, the detection threshold for each model was selected by maximization of the F1 score. This was done to balance precision and recall as F1 is a harmonic mean of these metrics and can be expressed by the fol-

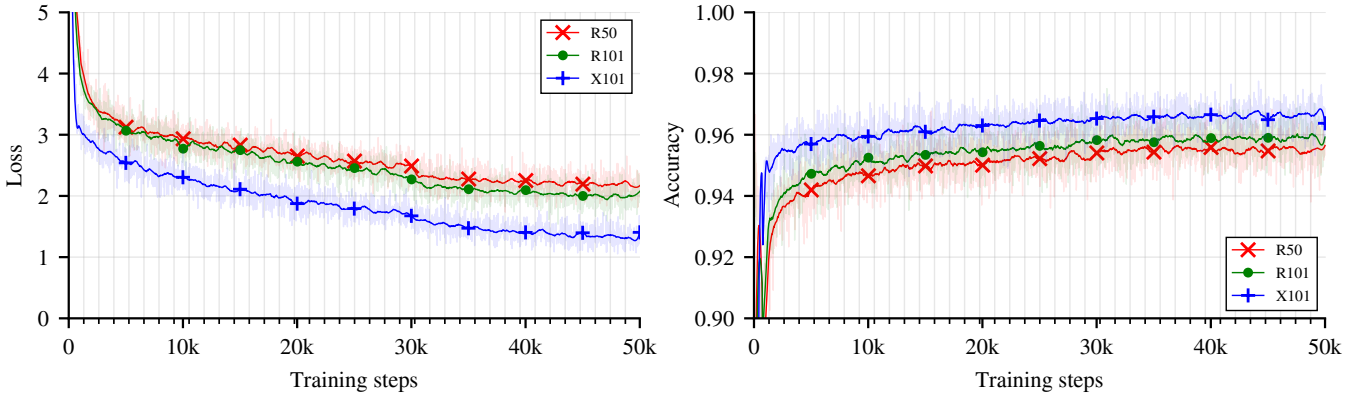


Fig. 5. Value of loss function (left) and (right) accuracy on the training set throughout training for all models. Solid color lines represent smoothed values whereas transparent ones raw data. Additionally, vertical grid lines mark the end of the training epoch

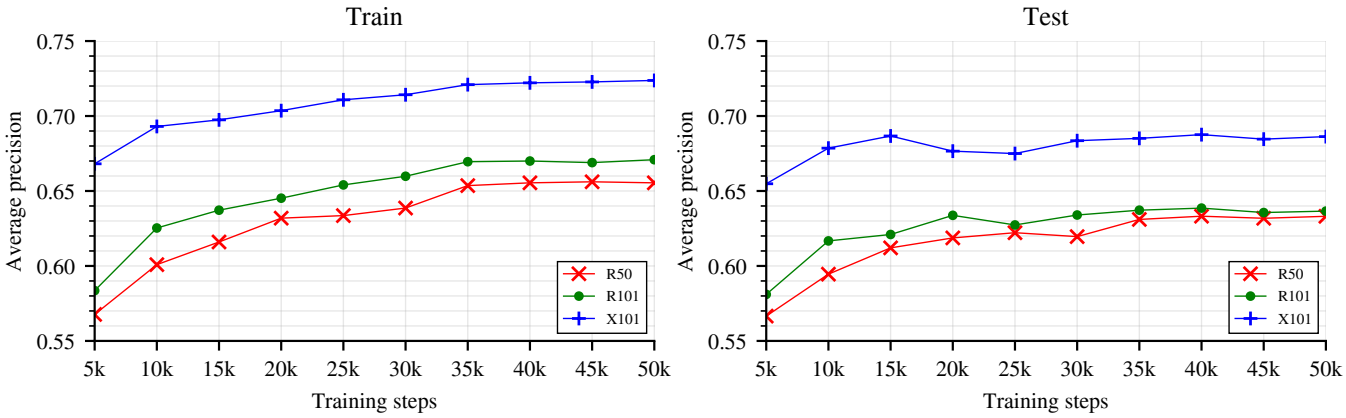


Fig. 6. MS COCO style AP of each model computed every 5 thousand training steps on the train (left) and test (right) dataset

lowing formula

$$F1 = \frac{2(p \cdot r)}{p + r}. \tag{11}$$

The F1 scores were calculated for each class over the set of decision thresholds starting from 5% to 99% with the step of 1%. The scores obtained this way were then averaged to get the overall F1 metric, and a threshold value with the highest F1 score value was selected. The resulting decision thresholds and corresponding overall F1 scores achieved by each model are summarized in Table 2.

Table 2
Detection threshold with corresponding overall F1 score for each model

Model	Detection threshold [%]	Overall F1 score [%]
R50	79	88.7
R101	82	89.2
X101	81	91.8

5. RESULTS

5.1. Bounding box detection

Detection results show that models were trained correctly, as all three perform well, achieving AP^{50} over 90% and AP over 60%. The model denoted as *R50* performed worst, with *R101* slightly ahead and the *X101* being the best of the considered ones. This came with no surprise, as in the benchmarks performed [54], ResNeXt outperformed even much deeper, but sequential architectures. At the same time, it is clear that ResNet-50 being the shallowest network performs worst. The trend observed here is present in all the experiments described in the paper – thus, it can be assumed that with further development of CNN architectures, our solution can perform even better, if the backbone is swapped again. The full breakdown of the results containing overall and class-specific MS COCO style metrics for each model is summarized in Table 3.

Examining class-specific results, it can be noticed that models achieve worse AP for hard hat class, as seen in Table 3. Better performance in person detection is no surprise, as the training part of the dataset contains more person instances. Additionally, these models were derived from models trained only for person detection. Considering the above, this bias would be

Table 3
Bounding box detection results

Model	AP	AP^{50}	AP^{75}	AP_S	AP_M	AP_L
Overall:						
R50	0.633	0.909	0.727	0.402	0.699	0.774
R101	0.637	0.912	0.721	0.401	0.706	0.778
X101	**0.686**	**0.940**	**0.782**	**0.466**	**0.756**	**0.820**
Hard hat:						
R50	0.599	0.900	0.700	0.511	0.756	0.771
R101	0.591	0.901	0.679	0.499	0.752	0.765
X101	**0.626**	**0.927**	**0.731**	**0.536**	**0.776**	**0.801**
Person:						
R50	0.667	0.918	0.753	0.294	0.643	0.777
R101	0.682	0.923	0.763	0.303	0.661	0.792
X101	**0.746**	**0.953**	**0.833**	**0.395**	**0.736**	**0.839**

Where:
AP – MS COCO style AP computed at different IoUs (from 50% to 95% with a step of 5%)
AP^{50}, AP^{75} – AP computed at IoU of 50% and 75%
AP_S, AP_M, AP_L – AP computed for small, medium and large objects (see Table 1)

expected. However, it should also be pointed out that the difference in AP^{50} is not that significant and hard hats, as smaller objects compared to people, achieved better scores in AP_S and AP_M metrics.

5.2. Head center localization

Head center localization results

All models performed very well in the person head center localization, achieving AP over 70% and AP^{50} over 80%. The results are even more impressive, considering that head center localization is perceived as harder than bounding box detection. The full breakdown of the head center localization for the person class is summarized in Table 4.

Table 4
Person head center localization results

Model	AP	AP^{50}	AP^{75}	AP_M	AP_L
R50	0.704	0.814	0.736	0.697	0.854
R101	0.707	0.819	0.740	0.705	0.856
X101	**0.747**	**0.838**	**0.767**	**0.748**	**0.884**

Where:
AP – MS COCO style mean AP computed at different OKSs (from 50% to 95% with a step of 5%)
AP^{50}, AP^{75} – AP computed at OKS of 50% and 75%
AP_M, AP_L – AP computed for medium and large objects (see Table 1)

Head and head with hard hat

Apart from the above, head center localization evaluation was performed on person sub-classes that represented hard hat wearers and non-wearers. This was done to check if our solution can generalize head center between both groups, thus localizing head center whether the hard hat is worn. The full results of this evaluation are summarized in Table 5.

Table 5
Comparison of person head center localization results for person with and without hard hat

Model	AP	AP^{50}	AP^{75}	AP_M	AP_L
Person w/hard hat:					
R50	0.727	0.809	0.756	0.697	0.847
R101	0.732	0.816	0.763	0.700	0.852
X101	**0.774**	**0.841**	**0.799**	**0.751**	**0.883**
Person w/o hard hat:					
R50	0.536	0.646	0.567	0.573	0.742
R101	0.555	0.671	0.582	0.590	0.766
X101	**0.609**	**0.713**	**0.628**	**0.650**	**0.817**

For definition of AP, AP^{50}, AP^{75}, AP_M and AP_L see Table 4

As seen in the results, all models display a bias towards a more prominent person sub-class representing hard hat non-wearers which is in line with instance imbalance written in Table 1.

However, head center heatmaps were inspected to assure that the solution performs well. Some of these joints heatmaps overlaid on instances of both hard hat wearers and non-wearers are presented in Fig. 7. From these heatmaps, joints are selected as points with the highest score. It can be seen that the head center is correctly localized for both groups considering different poses and scales and partial visibility.

(a) hard hat wearers

(b) hard hat non-wearers

Fig. 7. Comparison of head center heatmaps between hard hat wearers (a) and non-wearers (b), overlaid on object instances from test dataset [51] detected by best performing model (*X101*)

5.3. Hard-hat-wearing

Once again the model denoted as *X101* performed best, which was expected as classification is based on previously evaluated detection and head center localization. The detailed breakdown of results of hard-hat-wearing detection is presented in Table 6.

Table 6
Results of hard-hat-wearing detection

Model	AP	AP^{50}	AP^{75}	AP_S	AP_M	AP_L
Overall:						
R50	0.575	0.752	0.663	0.124	0.572	0.728
R101	0.595	0.765	0.681	0.140	0.587	0.757
X101	**0.675**	**0.826**	**0.759**	**0.211**	**0.682**	**0.817**
Hard hat wearer:						
R50	0.620	0.823	0.719	0.167	0.586	0.723
R101	0.637	0.827	0.736	0.186	0.600	0.746
X101	**0.710**	**0.871**	**0.805**	**0.247**	**0.693**	**0.805**
Hard hat non-wearer:						
R50	0.531	0.682	0.606	0.082	0.559	0.733
R101	0.553	0.704	0.626	0.094	0.574	0.768
X101	**0.641**	**0.780**	**0.714**	**0.175**	**0.671**	**0.828**

For definition of AP, AP^{50}, AP^{75}, AP_S, AP_M and AP_L see Table 3

As seen in the results, only the above model achieved AP^{50} over 80% and AP over 60%, with other models closer to AP^{50} value of 75% and AP below 60%. However, to fully assess the performance of the proposed approach, it is necessary to put it into perspective by comparing it with different already presented ones.

Moreover, by examining class-specific scores, it can be seen that again all models show bias towards hard hat wearer class. As stated before, it is not surprising because hard-hat-non-wearers account for about 30% of people instances in both the train and test part of the dataset. However, a troubling fact is small-scale performance as AP_S for all models did not exceed 25% and mostly scored well below 20%.

6. COMPARATIVE STUDIES

Authors in [14] compared three different approaches to PPE detection: direct detection of PPE wearers, separate detection of workers and PEE equipment coupled with a decision tree (DT), and a two-staged approach where the first stage localizes people and the second one classifies if PPE is worn. Out of these, the first one archived the best results for hard hats, with an AP^{50} value of 73.97%, and class-specific values of 79.81% for hard hat wearers and 63.12% for non-wearers. Whereas, solution based on bounding box relative position performed the worst with AP^{50} value of 69.09% and class-specific values of 74.29% and 63.84%. However, direct comparison with this work is not feasible as our solution is based on different detectors that, opposed to YOLOv3, are focused on accuracy instead of real-time performance. Moreover, the dataset used in [14] in their study was smaller. Therefore, to provide a fair comparison, all the models were compared using the same testing dataset and a new model for direct detection and a new DT were developed with our training dataset. Additionally, for the comparison to be entirely unbiased, the DT developed in [14] was also tested, as it can be used with our detector.

For the direct detection (naming consistent with Table 7 and 8), another model based on the ResNeXt backbone has been trained. Excluding the keypoint branch, it was identical to the model previously denoted as *X101* and was trained to start from the same weights, with the same parameters and for the same number of training steps. However, instead of detecting hard hat and person instances, it was trained to detect hard-hat-wearers and non-wearers directly. Additionally, detection threshold moving was also performed for this model. The overall F1 score achieved the highest value of 87.6% at a decision threshold of 83%. This newly trained network was the Faster R-CNN previously used for this task [13].

The architecture of our DT was selected using the grid search technique with 5-fold cross-validation on the training set. The optimized parameters were as follows:

- a split criterion: Gini impurity or entropy information gain,
- the maximum depth of the tree m_d to prevent overfitting: $\{2, 3, ..., 15\}$ or no limit,
- the minimum samples m_s necessary to split.

Results of experiments indicate that the best set of hyperparameters is: Gini impurity criterion, $m_d = 10$ and $m_s = 14$.

The developed DT and original [14] were paired with *X101* model as a sole detector for this comparison, as it performed best among all trained ones.

The full breakdown of the comparison is summarized in Table 7. It could be seen that our solution achieved the highest

Table 7
Comparison of our head center based approach with proposed in [14] decision tree based on the bounding box relative position and direct detection of hard hat wearers/non-wearers

Classifier	AP	AP^{50}	AP^{75}	AP_S	AP_M	AP_L
Overall:						
Our solution	**0.675**	**0.826**	**0.759**	0.211	**0.682**	**0.817**
Our DT	0.664	0.815	0.746	0.222	0.668	0.799
[14] DT	0.654	0.806	0.736	0.222	0.662	0.775
Direct detection	0.663	**0.826**	0.757	**0.248**	0.670	0.809
Hard hat wearer:						
Our solution	0.710	0.871	0.805	0.247	0.693	0.805
Our DT	0.698	0.860	0.794	0.248	0.681	0.789
[14] DT	0.696	0.860	0.795	0.250	0.682	0.788
Direct detection	**0.723**	**0.905**	**0.824**	**0.331**	**0.700**	**0.809**
Hard hat non-wearer:						
Our solution	**0.641**	**0.780**	**0.714**	0.175	**0.671**	**0.828**
Our DT	0.630	0.769	0.698	**0.197**	0.655	0.808
[14] DT	0.611	0.751	0.677	0.194	0.642	0.762
Direct detection	0.603	0.747	0.690	0.165	0.639	0.810

For definition of AP, AP^{50}, AP^{75}, AP_S, AP_M and AP_L see Table 3

overall AP out of all tested models. It was followed by DT fitted to our dataset, direct detection and finally original [14] DT.

Regarding class-specific performance, our solution performs slightly worse than direct detection in the detection of hard hat wearers, with an AP difference of 71.0% to 72.3%. However, this situation is reversed in the detection of hard hat non-wearers, where our solution achieves an AP of 64.1% compared to 60.3% for direct detection. Additionally, in the latter, direct detection was also outperformed by both DTs. The above is true for almost all metrics, excluding AP_S. While the gain may seem to be insignificant, our solution shows its advantage in more complicated cases, as seen in Figs. 8 and 9. Furthermore, even small percentage of metric gain is significant in the task of object detection and image recognition.

(a) Most common direct approach failure – duplicated labels

(b) Direct approach missclassification in crowded environment

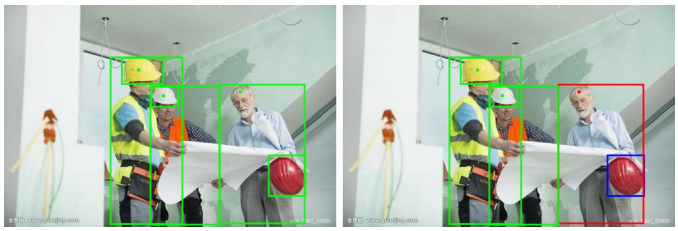
(c) Misclassification of shadow as a separate person

(d) Misclassification due to inter-class similarity

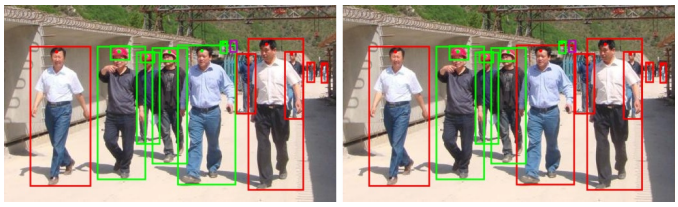
Fig. 8. Comparison of our solution (each image right) with direct approach (each image left), hard hat wearers with their head protection marked in green, non-wearers in red, people without head joint in magenta and not worn hard hats in blue

(a) Misclassification in crowded environment

(b) Misclassification due to proximity of hardhat bounding



(c) Misclassification due to hardhat bounding box within person bounding box



(d) Misclassification due to inter-class similarity

Fig. 9. Comparison of our solution (each image right) with [14] DT (each image left), for legend of instanceses see Fig. 8

The class-specific breakdown shows that the difference in performance comes solely from hard hat non-wearer detection. It seems that our solution delivers a more balanced performance at the cost of hard-hat-wearing detection.

Also, it has to be mentioned that [14] DT achieved higher overall and class-specific AP^{50} values on our dataset with our detector. That only underlines the inability to direct the comparison of methods developed in different conditions. The relative performance of both decision trees is also worth addressing. As seen in Table 7, overall AP is only slightly better for DT developed from scratch, and regarding hard hat wearers, [14] DT is on par. The main difference comes from hard hat non-wearers. However, our DT is significantly more complex. It has a depth of 10 layers and is composed of 485 nodes, compared to 3 layers and 10 nodes of [14] DT. This means that it lost one of the main advantages of the DTs – human interpretability. Whereas simpler trees can be acquired, a decrease in depth and node number leads to performance degradation. Given the interpretability aspect, it can be concluded that [14] has already reached the limits of the decision trees.

Person detection comparison

Additionally, Table 8 shows a comparison of person detection evaluation between the model trained for direct detection of hard-hat-wearing (denoted as *Direct detection*, described above, and one trained in detecting hard hat and person instances (denoted as *Our solution*). As seen in the table, our solution slightly outperforms direct detection in person detection when all instances detected by the latter are treated as one class. The difference in performance is not very significant. However, it should be noted that, in general, adding a keypoint branch, as opposed to a mask branch, hinders object detection [31]. Therefore, direct detection should perform better than our solution.

Table 8
Comparison of person detection results as separate category (Our solution) and super-category (Direct detection)

Model	AP	AP^{50}	AP^{75}	AP_S	AP_M	AP_L
Our solution	**0.746**	**0.953**	**0.833**	**0.395**	**0.736**	**0.839**
Direct detection	0.729	0.940	0.827	0.369	0.714	0.830

For definition of AP, AP^{50}, AP^{75}, AP_S, AP_M and AP_L see Table 3.

This difference in performance could be linked to the problem mentioned earlier – high inter-class similarity. In this case, the detector focuses on learning specific image features linked to the hardhat (its shape, color and position in the bounding box) and loses the ability to recognize more general features of a person. This is even more evident at higher IoU thresholds (Fig. 10) as precision values drop faster as recall rises.

7. DISCUSSION

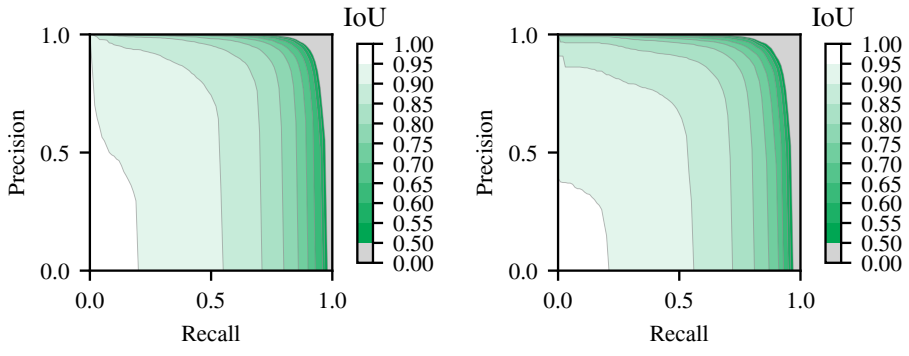
There are many advantages of using computer vision in safety monitoring and PPE detection, and some of them are highlighted by our solution. Vision-based methods are not worker-focused and can supervise multiple workers simultaneously, while other systems (*e.g.* RFID technology) rely heavily on workers' cooperation in the process. They also do not require a separate system specific to safety monitoring, as they can use existing CCTV cameras already present on-site.

The solution proposed in this paper addresses the main issues observed in the vision-based hard hat detection approaches currently presented in the literature. It is based on the separate detection of people and hard hats, which means it does not suffer from the intraclass similarity problem found in hard hat wearer/non–wearer detection. Moreover, this allows direct transfer learning from well-trained person detection models, making it easier to train and deploy. The addition of head center enables the hard-hat-wearing to be determined with a simple, human-interpretable rule-based algorithm. Moreover, no distance threshold [16, 17, 55] or additional features like neck, hips [16, 17], ears, nose [18] or face [15] are needed to establish a worker – hard hat relationship. This alone makes our solution more flexible, as it will work in more situations where additional information will not be available for others.

In tests, it surpassed the previous solution based on the relative bounding box position of people and hard hats and the direct detection of hard hat wearers and non-wearers. The MS COCO style overall AP of 67.5% compared to 66.4% and 66.3% achieved by the approaches mentioned above, with class-specific AP for hard hat non-wearers of 64.1% compared to 63.0% and 60.3%. The performance gain in the latter task should be highlighted, as detecting workers that do not comply with the rules is the real problem.

Additionally, to fully understand the performance and limitations of our solution, raw detection results were also examined. Some examples of images with marked object instances are presented in Fig. 11.

Most observed errors came from hard hat or person detection failures (Fig. 11 b, c and e). Another problem found in the results was related to the head center detection. It turned out that the solution detects them even for instances where the head is not visible (Fig. 11 f), which is caused by a minimal number of such cases in the dataset. However, the main surprise is that head joints can be localized correctly even at a small scale (Fig. 11 c and e). Causing other errors for small-scale worker instances, matching hard hats are even smaller, making them impossible to detect. This hinders hard-hat-wearing detection at a small scale, and whilst it would explain the worse AP_S achieved by our approach in Section 6. It cannot fully explain the difference in performance to DTs, as these also suffer for



(a) Our solution **(b)** Direct detection

Fig. 10. Person class Precision-Recall curves for *Our solution* (a) and *Direct detection* (b)

(a) perfect detection

(b) failure to detect one of the hard hats

(c) failure to detect both person and hard hat due to occlusion

(d) failure in head center localization

(e) detection failure at small scale

(f) failure in head center detection for partially visible person

Fig. 11. Results of inference on test part of the dataset [51] achieved by best performing model (*X101*), for legend of instanceses see Fig. 8

the same reason. It seems that the difference in performance is also influenced by joint localization errors at a small scale.

Moreover, it has to be mentioned that due to Algorithm 1 simplicity, it cannot handle significant scale difference between people and hard hats, leading to a situation where a small-scale worker instance can be classified as a hard-hat-wearer with a hard hat at a much larger scale, worn by someone else. However, this was not observed in the results.

Apart from the above, some problems with currently available datasets also have to be acknowledged. In general, these

datasets are significantly imbalanced. In the dataset used in this study, the ratio between people not wearing and wearing hard hats is close to 7:3. This is clearly visible in the results presented in Sections 5 and 6. However, more worrying is the fact that most of the hard hat non-wearers are not workers. In the majority of cases, people instances in this group are wearing civilian clothes and the cases in which they are workers performing some tasks are rare. This should be avoided, particularly in the case of direct detection of hard hat wearers and non-wearers. It may lead to a situation where the model will learn to distinguish workers from civilians instead.

Finally, there is no benchmark dataset with multiple different labels available. Researchers tend to develop and test their solutions on custom datasets, making fair comparison impossible.

8. CONCLUSIONS AND FUTURE WORK

This article proposed a novel approach to hard-hat-wearing detection based on the detection of people and hard hats combined with person head center localization. This unique combination allows one to determine the correct relationship between these instances and differentiate hard-hat-wearers and non-wearers. Results show that it surpassed both the solution based on the relative bounding box position of people and hard hats and direct detection of hard hat wearers and non-wearers. Achieving MS COCO style overall AP of 67.5% compared to 66.4% and 66.3% achieved by the approaches mentioned above. Even more important, the main gains come from detecting hard-hat-non-wearers, with class-specific AP of 64.1% compared to 63.0% and 60.3%. This aspect matters the most, as this kind of solution should focus on detecting safety breaches. Additionally, in-depth comparisons proved that our approach does not suffer from the problem of intraclass similarity. Moreover, the addition of the person head center enables the solution to be reduced to simple human-interpretable rules, rather than an overly complex decision tree that cannot provide such results.

However, reliable detection of hard hat non-wearers is only the first step in developing a deep learning supported safety system for construction site monitoring purposes. For such a system to be effective, workers who break the rules have to be identified so that they can be reprimanded, fined or sent to additional OHS training. Hard-hat-wearing detection based on face detection, like the one described in [15], is not an answer to that problem as it will not work in situations when worker's face is not visible. Instead, a solution similar to one described in [32] should be considered. Face detection and identification should be made simultaneously with safety rule checking as each worker is tracked, ideally in multiple views at once. Moreover, all these tasks should be done in real-time as the construction site is a dynamic environment.

This brings us to a hardware problem, as little attention is paid to the infrastructure needed on the construction site to deploy these models in real-time. This is an essential aspect, especially considering the recent surge in GPUs. Simply moving processing to cloud services will not be enough. Real-time streaming of multiple high-quality video feeds still needs a lightning-fast internet connection while providing an awful amount of data to analyze.

An answer to this problem could be the usage of solutions aiming at efficient computation on edge devices. Recently, some methods delivering lightened deep learning architectures were proposed [56], along with ones tailored explicitly for embedded applications [42, 57]. These are slowly used in recent studies. An excellent example of the application of the latter in the construction safety context is [41]. The use of such algorithms and appropriate devices would allow the creation of a distributed computing system in which each node, starting from the input, would gradually analyze the data. Therefore, decreasing data throughput needed and lowering hardware demand.

REFERENCES

[1] Eurostat, "Accidents at work – statistics by economic activity," 2020. [Online]. Available: https://ec.europa.eu/ eurostat/ statistics- explained/ index.php?title=Accidents_at_work_-_statistics_by_economic_activity.

[2] USDL, "National Census of Fatal Occupational Injuries in 2019," pp. 1–9, 2020. [Online]. Available: https://www.bls.gov/news.release/pdf/cfoi.pdf.

[3] A. Colantonio, D. McVittie, J. Lewko, and J. Yin, "Traumatic brain injuries in the construction industry," *Brain Inj.*, vol. 23, no. 11, pp. 873–878, 2009, doi: 10.1080/02699050903036033.

[4] S. Konda, H.M. Tiesman, and A.A. Reichard, "Fatal traumatic brain injuries in the construction industry, 2003-2010," *Am. J. Ind. Med.*, vol. 59, no. 3, pp. 212–220, 2016, doi: 10.1002/ajim.22557.

[5] C.A. Taylor, J.M. Bell, M.J. Breiding, and L. Xu, "Traumatic brain injury-related emergency department visits, hospitalizations, and deaths - United States, 2007 and 2013," *MMWR Surv. Summ.*, vol. 66, no. 9, pp. 1–16, 2017, doi: 10.15585/mmwr.ss6609a1.

[6] A. Colantonio, D. Mroczek, J. Patel, J. Lewko, J. Fergenbaum, and R. Brison, "Examining occupational traumatic brain injury in Ontario." *Can. J. Public Health-Rev. Can. Sante Publ.*, vol. 101 Suppl 1, no. S1, pp. S58–S62, mar 2010, doi: 10.1007/bf03403848.

[7] A.M. Salem, B.A. Jaumally, K. Bayanzay, K. Khoury, and A. Torkaman, "Traumatic brain injuries from work accidents: A retrospective study," *Occup. Med.*, vol. 63, no. 5, pp. 358–360, 2013, doi: 10.1093/occmed/kqt037.

[8] EU-OSHA, "Directive 89/656/EEC – use of personal protective equipment," 1989.

[9] G.P. Jirka and W. Thompson, "Personal protective equipment," pp. 493–508, 2009, doi: 10.1201/9781420071825-29.

[10] M.-W. Park, N. Elsafty, and Z. Zhu, "Hardhat-Wearing Detection for Enhancing On-Site Safety of Construction Workers," *J. Constr. Eng. Manage.*, vol. 141, no. 9, p. 04015024, 2015, doi: 10.1061/(asce)co.1943-7862.0000974.

[11] W. Tun, J.-H. Kim, Y. Jeon, S. Kim, and J.-W. Lee, "Safety Helmet and Vest Wearing Detection Approach by Integrating YOLO and SVM for UAV," in *Korean Society for Aeronautical and Space Sciences 2020 Spring Conference*, 2020. [Online]. Available: https://www.dbpia.co.kr/Journal/articleDetail?nodeId=NODE10442178.

[12] M. Memarzadeh, A. Heydarian, M. Golparvar-Fard, and J.C. Niebles, "Real-time and automated recognition and 2D tracking

of construction workers and equipment from site video streams," *Congress on Computing in Civil Engineering, Proceedings*, pp. 429–436, 2012, doi: 10.1061/9780784412343.0054.

[13] Q. Fang, H. Li, X. Luo, L. Ding, H. Luo, T. M. Rose, and W. An, "Detecting non-hardhat-use by a deep learning method from far-field surveillance videos," *Autom. Constr.*, vol. 85, pp. 1–9, 2018, doi: 10.1016/j.autcon.2017.09.018.

[14] N.D. Nath, A.H. Behzadan, and S.G. Paal, "Deep learning for site safety: Real-time detection of personal protective equipment," *Autom. Constr.*, vol. 112, p. 103085, 2020, doi: 10.1016/j.autcon.2020.103085.

[15] J. Shen, X. Xiong, Y. Li, W. He, P. Li, and X. Zheng, "Detecting safety helmet wearing on construction sites with bounding-box regression and deep transfer learning," *Comput.-Aided Civil Infrastruct. Eng.*, vol. 36, no. 2, pp. 180–196, 2021, doi: 10.1111/mice.12579.

[16] S. Chen and K. Demachi, "A vision-based approach for ensuring proper use of personal protective equipment (PPE) in decommissioning of fukushima daiichi nuclear power station," *Appl. Sci.*, vol. 10, no. 15, p. 5129, 2020, doi: 10.3390/app10155129.

[17] S. Chen and K. Demachi, "Towards on-site hazards identification of improper use of personal protective equipment using deep learning-based geometric relationships and hierarchical scene graph," *Autom. Constr.*, vol. 125, p. 103619, 2021, doi: 10.1016/j.autcon.2021.103619.

[18] R. Xiong and P. Tang, "Pose guided anchoring for detecting proper use of personal protective equipment," *Autom. Constr.*, vol. 130, p. 103828, 2021, doi: 10.1016/j.autcon.2021.103828.

[19] M. Ochmański, G. Modoni, and J. Bzówka, "Prediction of the diameter of jet grouting columns with artificial neural networks," *Soils Found.*, vol. 55, no. 2, pp. 425–436, apr 2015, doi: 10.1016/j.sandf.2015.02.016.

[20] J. Shen, X. Xiong, Z. Xue, and Y. Bian, "A convolutional neural-network-based pedestrian counting model for various crowded scenes," *Comput.-Aided Civil Infrastruct. Eng.*, vol. 34, no. 10, pp. 897–914, 2019, doi: 10.1111/mice.12454.

[21] X. Luo, H. Li, Y. Yu, C. Zhou, and D. Cao, "Combining deep features and activity context to improve recognition of activities of workers in groups," *Comput.-Aided Civil Infrastruct. Eng.*, vol. 35, no. 9, pp. 965–978, 2020, doi: 10.1111/mice.12538.

[22] Y. Gao and K.M. Mosalam, "Deep Transfer Learning for Image-Based Structural Damage Recognition," *Comput.-Aided Civil Infrastruct. Eng.*, vol. 33, no. 9, pp. 748–768, sep 2018, doi: 10.1111/mice.12363.

[23] M. Żarski, B. Wójcik, and J.A. Miszczak, "KrakN: Transfer Learning framework for thin crack detection in infrastructure maintenance," *arXiv*, apr 2020. [Online]. Available: http://arxiv.org/abs/2004.12337.

[24] W. Fang, L. Ding, P. E. Love, H. Luo, H. Li, F. Peña-Mora, B. Zhong, and C. Zhou, "Computer vision applications in construction safety assurance," *Autom. Constr.*, vol. 110, p. 103013, 2020, doi: 10.1016/j.autcon.2019.103013.

[25] W. Fang, P.E. Love, H. Luo, and L. Ding, "Computer vision for behaviour-based safety in construction: A review and future directions," *Adv. Eng. Inform.*, vol. 43, p. 100980, 2020, doi: 10.1016/j.aei.2019.100980.

[26] B.H. Guo, Y. Zou, Y. Fang, Y.M. Goh, and P.X. Zou, "Computer vision technologies for Saf. Sci. and management in construction: A critical review and future research directions," *Saf. Sci.*, vol. 135, p. 105130, 2021, doi: 10.1016/j.ssci.2020.105130.

[27] Q. Fang, H. Li, X. Luo, L. Ding, T.M. Rose, W. An, and Y. Yu, "A deep learning-based method for detecting non-certified work on construction sites," *Adv. Eng. Inform.*, vol. 35, pp. 56–68, 2018, doi: 10.1016/j.aei.2018.01.001.

[28] W. Fang, L. Ding, H. Luo, and P. E. Love, "Falls from heights: A computer vision-based approach for safety harness detection," *Autom. Constr.*, vol. 91, pp. 53–61, 2018, doi: 10.1016/j.autcon.2018.02.018.

[29] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1137–1149, jun 2017, doi: 10.1109/TPAMI.2016.2577031. [Online]. Available: http://ieeexplore.ieee.org/document/7485869/.

[30] W. Fang, B. Zhong, N. Zhao, P. E. Love, H. Luo, J. Xue, and S. Xu, "A deep learning-based approach for mitigating falls from height with computer vision: Convolutional neural network," *Adv. Eng. Inform.*, vol. 39, pp. 170–177, 2019, doi: 10.1016/j.aei.2018.12.005.

[31] K. He, G. Gkioxari, P. Dollar, and R. Girshick, "Mask R-CNN," in *Proceedings of the IEEE International Conference on Computer Vision*, vol. 2017-October, 2017, pp. 2980–2988, doi: 10.1109/ICCV.2017.322.

[32] Y. Zhao, Q. Chen, W. Cao, J. Yang, J. Xiong, and G. Gui, "Deep Learning for Risk Detection and Trajectory Tracking at Construction Sites," *IEEE Access*, vol. 7, pp. 30 905–30 912, 2019, doi: 10.1109/ACCESS.2019.2902658.

[33] J. Redmon and A. Farhadi, "YOLOv3: An Incremental Improvement," 2018. [Online]. Available: http://arxiv.org/abs/1804.02767

[34] R. Wei, P.E. Love, W. Fang, H. Luo, and S. Xu, "Recognizing people's identity in construction sites with computer vision: A spatial and temporal attention pooling network," *Adv. Eng. Inform.*, vol. 42, p. 100981, 2019, doi: 10.1016/j.aei.2019.100981.

[35] S. Tang, D. Roberts, and M. Golparvar-Fard, "Human-object interaction recognition for automatic construction site safety inspection," *Autom. Constr.*, vol. 120, p. 103356, 2020, doi: 10.1016/j.autcon.2020.103356.

[36] H. Luo, J. Liu, W. Fang, P.E. Love, Q. Yu, and Z. Lu, "Real-time smart video surveillance to manage safety: A case study of a transport mega-project," *Adv. Eng. Inform.*, vol. 45, p. 101100, aug 2020, doi: 10.1016/j.aei.2020.101100.

[37] N. Khan, M. R. Saleem, D. Lee, M.W. Park, and C. Park, "Utilizing safety rule correlation for mobile scaffolds monitoring leveraging deep convolution neural networks," *Comput. Ind.*, vol. 129, p. 103448, 2021, doi: 10.1016/j.compind.2021.103448.

[38] B.E. Mneymneh, M. Abbas, and H. Khoury, "Vision-Based Framework for Intelligent Monitoring of Hardhat Wearing on Construction Sites," *J. Comput. Civil. Eng.*, vol. 33, no. 2, p. 04018066, 2019, doi: 10.1061/(asce)cp.1943-5487.0000813.

[39] J. Wu, N. Cai, W. Chen, H. Wang, and G. Wang, "Automatic detection of hardhats worn by construction personnel: A deep learning approach and benchmark dataset," *Autom. Constr.*, vol. 106, p. 102894, 2019, doi: 10.1016/j.autcon.2019.102894.

[40] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.Y. Fu, and A.C. Berg, "SSD: Single shot multibox detector," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 9905 LNCS, 2016, pp. 21–37, doi: 10.1007/978-3-319-46448-0_2.

[41] L. Wang, L. Xie, P. Yang, Q. Deng, S. Du, and L. Xu, "Hardhat-wearing detection based on a lightweight convolutional neural network with multi-scale features and a top-down module," *Sensors*, vol. 20, no. 7, 2020, doi: 10.3390/s20071868.

[42] A.G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications," 2017. [Online]. Available: http://arxiv.org/abs/1704.04861.

[43] F. Zhou, H. Zhao, and Z. Nie, "Safety Helmet Detection Based on YOLOv5," in *Proceedings of 2021 IEEE International Conference on Power Electronics, Computer Applications, ICPECA 2021*, 2021, pp. 6–11, doi: 10.1109/ICPECA51329.2021.9362711.

[44] S. Cai, W. Zuo, and L. Zhang, "Higher-Order Integration of Hierarchical Convolutional Activations for Fine-Grained Visual Categorization," in *Proceedings of the IEEE International Conference on Computer Vision*, vol. 2017-October, 2017, pp. 511–520, doi: 10.1109/ICCV.2017.63.

[45] W. Luo, X. Yang, X. Mo, Y. Lu, L. Davis, J. Li, J. Yang, and S.N. Lim, "Cross-X learning for fine-grained visual categorization," in *Proceedings of the IEEE International Conference on Computer Vision*, vol. 2019-October, 2019, pp. 8241–8250, doi: 10.1109/ICCV.2019.00833.

[46] J. Han, X. Yao, G. Cheng, X. Feng, and D. Xu, "P-CNN: Part-Based Convolutional Neural Networks for Fine-Grained Visual Categorization," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1–1, 2019, doi: 10.1109/tpami.2019.2933510.

[47] T.Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, vol. 2017-January. IEEE, jul 2017, pp. 936–944, doi: 10.1109/CVPR.2017.106.

[48] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2016-December. IEEE, jun 2016, pp. 770–778, doi: 10.1109/CVPR.2016.90. [Online]. Available: http://ieeexplore.ieee.org/document/7780459/.

[49] S. Xie, R. Girshick, P. Dollár, Z. Tu, and K. He, "Aggregated residual transformations for deep neural networks," in *Proceedings – 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, vol. 2017-January, 2017, pp. 5987–5995, doi: 10.1109/CVPR.2017.634.

[50] J. Wu, H. Zheng, B. Zhao, Y. Li, B. Yan, R. Liang, W. Wang, S. Zhou, G. Lin, Y. Fu, Y. Weng, and Y. Wang, "Large-scale datasets for going deeper in image understanding," pp. 1480–1485, 2019, doi: 10.1109/ICME.2019.00256.

[51] L. Xie, "Hardhat," 2019, doi: 10.7910/DVN/7CBGOS. [Online]. Available: https://dataverse.harvard.edu/dataset.xhtml?persistentId=doi:10.7910/DVN/7CBGOS.

[52] T.-Y. Lin, M. Maire, S. Belongie, L. Bourdev, R. Girshick, J. Hays, P. Perona, D. Ramanan, C.L. Zitnick, and P. Dollár, "Microsoft {COCO}: {Common Objects in Context}," 2015.

[53] Y. Wu, A. Kirillov, F. Massa, W.-Y. Lo, and R. Girshick, "Detectron2," 2019. [Online]. Available: https://github.com/facebookresearch/detectron2.

[54] S. Bianco, R. Cadene, L. Celona, and P. Napoletano, "Benchmark analysis of representative deep neural network architectures," *IEEE Access*, vol. 6, pp. 64 270–64 277, 2018, doi: 10.1109/access.2018.2877890.

[55] S. Guo, D. Li, Z. Wang, and X. Zhou, "Safety Helmet Detection Method Based on Faster R-CNN," in *Communications in Computer and Information Science*, X. Sun, J. Wang, and E. Bertino, Eds., vol. 1253 CCIS, 2020, pp. 423–434, doi: 10.1007/978-981-15-8086-4_40.

[56] D. Blalock, J. J. G. Ortiz, J. Frankle, and J. Guttag, "What is the State of Neural Network Pruning?" 2020. [Online]. Available: http://arxiv.org/abs/2003.03033.

[57] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L. C. Chen, "MobileNetV2: Inverted Residuals and Linear Bottlenecks," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2018, pp. 4510–4520, doi: 10.1109/CVPR.2018.00474.