This ICCV workshop paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

DriPE: A Dataset for Human Pose Estimation in Real-World Driving Settings

Romain Guesdon Carlos Crispim-Junior Laure Tougne
Univ Lyon, Lyon 2, LIRIS
Lyon, France, F-69676

{romain.guesdon, carlos.crispim-junior, laure.tougne} @liris.cnrs.fr

Abstract

The task of 2D human pose estimation has known a significant gain of performance with the advent of deep learning. This task aims to estimate the body keypoints of people in an image or a video. However, real-life applications of such methods bring new challenges that are under-represented in the general context datasets. For instance, driver status monitoring on consumer road vehicles introduces new difficulties, like self- and background body-part occlusions, varying illumination conditions, cramped view angles, etc. These monitoring conditions are currently absent in general purposes datasets. This paper proposes two main contributions. Firstly, we introduce DriPE (Driver Pose Estimation), a new dataset to foster the development and evaluation of methods for human pose estimation of drivers in consumer vehicles. This is the first publicly available dataset depicting drivers in real scenes. It contains 10k images of 19 different driver subjects, manually annotated with human body keypoints and an object bounding box. Secondly, we propose a new keypoint-based metric for human pose estimation. This metric highlights the limitations of current metrics for HPE evaluation and of current deep neural networks on pose estimation, both on general and driving-related datasets.

1. Introduction

Human Pose Estimation (HPE) is a well-known task in computer vision. This problem aims to find the position of keypoints in the 2D plane or the 3D space. Keypoints are generally placed on the body joints (shoulders, elbows, wrists, hips, knees, ankles), and the head. Additional points can be placed on hands, feet, or face.

State-of-the-art methods have reached good performances on HPE challenges on both single-person [1, 19, 30] and multiperson datasets [24], especially through deep learning. However, these general-purpose datasets do not depict challenging scenes that might occur very often in real-life applications, *e.g.,* strong body occlusion or varying illumination.

Pose estimation inside of a vehicle brings new difficulties that are under-represented in general datasets (Fig. 1). First, the camera placement causes a strong side viewing angle, producing both self- and background occlusion (*e.g.,* by the dashboard and the wheel). By consequence, the side of the subject's body opposite to the camera becomes more difficult to detect (Fig. 1C). Luminance is also an important factor in HPE. For instance, body parts can be fully visible in a regular pose but be missed by the network due to strong illumination (Fig. 1A). Also, the outside light may visually split the upper body into two halves, and hence deceive the network (Fig. 1B). Finally, the low contrast of the car interior can make the detection of body parts difficult, like the right forearm in the picture (Fig. 1D), depending on the color of the subject's clothes. To evaluate the open challenges on human pose estimation in consumer cars, we propose the first publicly-available dataset in real-world conditions called DriPE (Driver Pose Estimation) [1].

Moreover, we study the limitations of existing metrics [12, 24, 40] for the evaluation of the HPE task on keypoint detection, on both general and driving contexts. Based on our observations, we propose a new metric called mAPK to characterize the observed limitations. This metric is essential to highlight the challenges presented by DriPE, and up to now ignored in general datasets, such as background and self-occlusion.

This paper is organized as follows. Section 2 presents related work on human pose estimation. In Section 3, we present DriPE dataset. We describe in Section 4 the proposed mAPK metric. Section 5 introduces the evaluated networks and describes their architecture. We present and discuss in Section 6 the experimental results. Finally, Section 7 presents our conclusions and future work.

[1] DriPE dataset is publicly available on: https://gitlab.liris.cnrs.fr/aura_autobehave/dripe

Motive Technologies, Exhibit 2014
IPR2016-00108

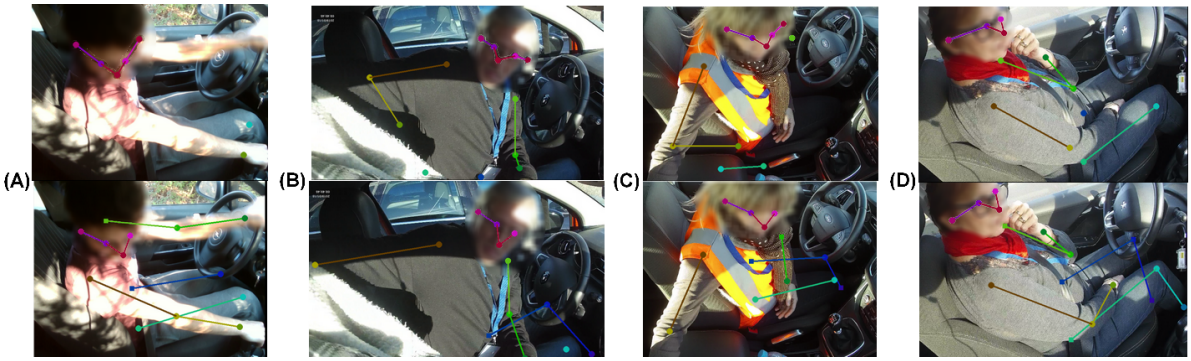

Figure 1: Samples of DriPE dataset. The top and bottom rows show, respectively, pose predictions by Simple Baseline network [39] and ground truth data. Faces have been blurred on this figure to anonymize the participants' identities.

2. Related Work

This section presents the work related to keypoint detection for human pose estimation. More precisely, we discuss the datasets used for this task, the current methods for pose estimation, and the metrics used to evaluate their accuracy.

2.1. Datasets

Datasets play an important role in the performance of deep learning methods. Improvements in the human pose estimation using deep learning networks have been partly justified by new datasets with more subjects' pictures and more variability in their poses, the angles of view, the background, etc.

Leeds Sports Pose (LSP) [19] dataset is the first HPE dataset released with more than 1k training images, which was later extended to 11k. It contains pictures of full-body subjects practicing different sports extracted from Flickr. Frames Labeled In Cinema (FLIC) dataset [30] is formed of around 5k pictures extracted from Hollywood movies. The Max Planck Institute for Informatics (MPII) dataset [1] contains around 25k images extracted from various YouTube videos. Microsoft Common Objects in Context (COCO) [24] is originally an object detection and segmentation dataset, which was then expanded to a multiperson HPE dataset. It is composed of more than 250k pictures extracted from Bing, Flickr, and Google.

Even if these general datasets can be useful for training or benchmarking, they might not present certain challenging situations that might occur in domain-specific datasets. Therefore, several datasets have been published in the last years focusing on monitoring people inside cars [3, 4, 13, 18, 25]. However, they are mostly focused on the action recognition task. Furthermore, most of the available datasets are recorded in studios and do not represent natural foreground nor illumination changes present in vehicle cockpit during a daily routine ride, which are true challenges for HPE methods. For instance, authors in [25] propose Drive&Act dataset, depicting multi-view and multi-modal (RGB, NIR, depth) actions in a static driving simulator, with labeled actions and predicted 3D human poses. DFKI [13] describes a new test platform to record in-cabin scenes. However, no public dataset for HPE in a vehicle using this setup has been recorded or published up to now.

Besides, HPE datasets do not use exactly the same keypoints to represent the body. Most of the representations, commonly called skeletons, include one joint marker per major body limb articulation (shoulder, elbow, wrist, hip, knee, ankle). However, while some datasets [1, 19] only put markers on the top of the head and the base of the neck, others adopt a finer representation (eyes, nose, ears) [24]. Some works also extend the human pose representation to hands and feet [16, 6].

In the end, the most prominent general datasets in the state of the art of HPE are MPII [1] and LSP [19] for single-person and COCO [24] for multiperson pose estimation. Regarding the pose estimation inside of a vehicle, there is no publicly available dataset for HPE which presents real driving conditions.

2.2. HPE Methods

The pose estimation methods may be divided into two types: single-person and multiperson methods.

2.2.1 Single-person Pose Estimation

Single-person methods for HPE using convolutional neural networks can be split into two categories: regression-based and detection-based methods.

Regression-based CNN methods aim to directly predict the keypoints coordinates from pictures. AlexNet [21] is the first CNN baseline used for HPE. Toshev and Szegedy [36] use AlexNet as a multi-stage coordinate estimator and refiner. Carreira *et al.* [8] propose an Iterative Error Feedback network based on the deep convolution network GoogleNet [33]. Finally, Sun *et al.* [32] propose a parametrized pose repre-

sentation using bones instead of keypoints, paired up with the ResNet-50 [14] for both 2D and 3D HPE.

However, regression-based networks usually lack robustness due to the high non-linearity of the end-to-end structure between the image and the coordinates of the keypoints. To overcome this issue, many methods have proposed a detection-based approach instead. The majority of these methods aim to predict heatmaps, *i.e.,* maps where each pixel represents the probability for the keypoint to be located here. Newell *et al.* [27] propose an architecture composed of new modules called Hourglasses, which aim to extract features from different scales using a network built based on Residual Modules [15]. This architecture has inspired several other works [11, 20, 34, 35]. In addition to Hourglass-based methods, other detection-based architectures have been developed. Chen *et al.* [9] propose an adversarial learning architecture that combines a heatmap pose generator with two discriminators. Xiao *et al.* [39] use the ResNet-50 [14] network but add deconvolution layers in the last convolution stage to predict the heatmaps. Unipose [2] combines a ResNet backbone for feature extraction with a waterfall module to perform HPE. Sun *et al.* [31] use a parallel multi-scale approach similar to the Hourglass with exchange units.

The networks mentioned previously achieve state-of-the-art performances on recent challenges. However, ResNet Simple Baseline [39] presents a competitive performance while preserving a light architecture compared to others.

2.2.2 Multiperson Pose Estimation

Multiperson HPE brings two difficulties to the problem: find the locations of keypoints on the image and associate the detected keypoints to the different subjects. Multiperson approaches can be divided into two categories: top-down and bottom-up methods.

Top-down approaches first detect the people in the image and then find the keypoints of each person. Most of the top-down methods use a single-person HPE architecture preceded by a person detection step: Xiao *et al.* [39] and Sun *et al.* [31] both use a faster R-CNN [29] while Chen *et al.* [10] use a feature pyramid network [23]. Li *et al.* [22] propose a multi-stage network with cross-stage feature aggregation. Cai *et al.* [5] use a similar structure combined with an original residual steps block.

Conversely, bottom-up methods first detect every keypoint in the image and then infer people instances from them. Newell *et al.* [26] reuse their stacked hourglass network for single-person HPE and adapt it to multiperson by predicting an additional association map for each keypoint. Cao *et al.* [7] propose an iterative architecture with part affinity fields used to associate the keypoints to people.

Among the described architectures, top-down methods currently present the highest performance on HPE. For instance, MSPN [22] and RSN [5] have won the COCO Keypoint Challenge in 2018 and 2019, respectively.

2.3. Evaluation Metrics

The performances of the general 2D HPE methods can be difficult to evaluate since it depends on many criteria (number of visible keypoints, number of visible people, size of the subjects, etc.).

One of the first commonly used metrics is Percentage of Correct Parts (PCP) [12]. Each keypoint prediction is considered correct if its distance to the ground truth is inferior to a fraction of the limb length (*e.g.,* 0.5). Thereby, this metric punishes more severely smaller limbs, which are already hard to predict due to their size. To mitigate this issue, Percentage of Correct Keypoints (PCK) [40] sets the threshold for every keypoint of a subject on a fraction of a specific limb's length. Two thresholds are commonly chosen to evaluate the performance in the literature. These metrics are mostly employed to evaluate algorithms on single-person datasets, like MPII and LSP.

Another common metric is Average Precision (AP), paired up with Average Recall (AR). For single-person networks, APK [40] is computed on keypoint detections. A detection is considered as a true positive if it falls under a set range of the ground truth, similarly to that PCP and PCK metrics, and a false positive otherwise.

In a multiperson context, most metrics compute the performance of a method at a person detection level instead of a keypoint level. For instance, the mAP metric [1] first pairs up each person detection with the ground truth using PCK metric. Then, the matched and unmatched people are used to compute the average precision and recall. COCO dataset proposes a second metric for the evaluation of the HPE task that we will refer to as AP OKS. This metric uses the Object Keypoint Similarity (OKS) score [24], which is similar to the Intersection over Union (IoU), to calculate the distance between the people detections and ground truth based on keypoints. The final scores are still computed over people.

One of the main limitations of both PCK and AP OKS evaluation metrics is that they both put aside false-positive keypoints. Moreover, because the COCO dataset is mostly used in a multiperson context, its metric measures precision and recall based on people detection, instead of keypoints. To address the limitations of previous evaluation procedures, we define a new general metric based on keypoints detection called mAPK.

3. DriPE Dataset

We propose DriPE, a dataset to evaluate HPE methods on real-world driving conditions, containing illumination changes, occluding shadows, moving foreground, etc. The dataset is composed of 10k pictures of drivers in real-world

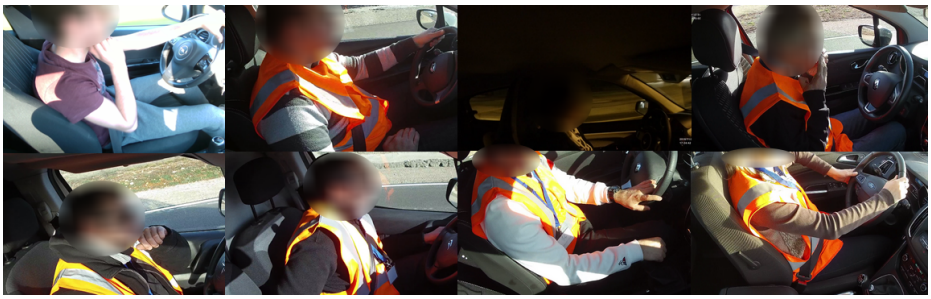

Figure 2: Image samples from DriPE dataset. Faces on the figure have only been blurred for the purpose of this paper.

conditions, split into 7.4k images for training, and 2.6k images equally divided into validation and testing sets. Table 1 presents a detailed description of the dataset and compares it to prior work.

3.1. Data Collection

To build DriPE, we extracted pictures from videos recorded during several driving experiments. In each experiment, we installed an RGB camera inside the car on top of the passenger's door, directed towards the driver. The subjects drive either in a real-size replica of a city (closed track) or on actual roads. In total, we recorded 19 drivers, allowing us to collect over 100 hours of video clips. We based the image selection process using two metrics: structural similarity index measure (SSIM) [37] and brightness differential. We chose these two metrics with the objective of extracting pictures with both distinct luminance and structure. Therefore, we computed the SSIM and the light differential between two successive frames, with a step of three frames per second. Then, we selected 10k pictures, half with the highest absolute light differential, and half with the lowest SSIM. We defined a minimum time gap between two selected frames to increase variability.

3.2. Annotations

We have chosen to follow the COCO dataset's annotation style for DriPE since face keypoints are particularly interesting to describe driver attention. For each image, we annotated the person bounding box and 17 keypoints: arms and legs with three keypoints each, and 5 additional markers for the eyes, ears, and nose. We split the annotated keypoints into two categories: visible and non-visible. The non-visible category corresponds to the occluded points, either by an object or by the subject body, but which position can still be deducted from the visible body parts. Note that in this study, both categories are treated equally by the evaluation methods. Following the COCO dataset policy, the face keypoints were annotated only if visible.

The ground truth heatmaps were generated using centered 2D Gaussian with a standard deviation of 1px, centered around the keypoint location.

	Drive&Act [25]	DriPE
N° subjects	15	**19**
Female / Male	4 / 11	**7 / 12**
Annotations	HPE network	**Manual**
RGB	✓	✓
Depth	✓	-
NIR	✓	-
N° images	**9.6M** (videos)	10k
Driving context	Simulator	**Real world**

Table 1: Comparison of driving-related datasets for HPE.

4. Evaluation Metric

Following the state of the art, we only evaluate in this study detection-based networks, which predict heatmaps. Each heatmap is a matrix where the elements represent the probability of a particular keypoint to be located at a pixel. Therefore, the output of the evaluated network models contains one heatmap per skeleton keypoint. Following the common practice in 2D single-person HPE [27, 35, 38, 39], the position of a given keypoint corresponds to the maximum value of its heatmap. To separate predictions from noise, a minimum confidence threshold is applied to this maximum. From these coordinates, several metrics can be calculated to evaluate the network performances.

4.1. Background

First, we describe and discuss in detail two evaluation metrics from the literature: AP OKS and APK.

4.1.1 AP OKS

To evaluate the performance of each network on the COCO dataset, the official multiperson metric is based on average precision (AP) and recall (AR). This evaluation is carried out following three steps: 1) compute the distance between each detected person and each ground-truth subject, 2) pair up the best person detection with its ground-truth, and 3) compute the precision and recall.

The metric used to compute the distance between a person's prediction and its ground truth is the OKS (Equation 1).

$$\text{OKS} = \frac{\sum_i \text{KS}_i * \delta(v_i > 0)}{\sum_i \delta(v_i > 0)} \quad (1)$$

where KS_i is defined as follows:

$$\text{KS}_i = \exp -\frac{d_i^2}{2.s^2.k_i^2} \quad (2)$$

where i iterates over each detected keypoint, d_i is the Euclidean distance between the predicted and the ground-truth keypoints, s is the image scale computed from the bounding box size, k_i a per-keypoint constant that tries to homogenize the standard deviations between each body part. Non-annotated keypoints have visibility v_i equal to 0, therefore their associated false positives are ignored by OKS computation.

Secondly, the OKS scores are used to select the best paired-up people, starting from the highest score. All unmatched detected people or paired-up couples with an OKS score lesser than a selected threshold (ranging from 0.5 to 0.95) are discarded. Finally, considering matched and discarded people as true and false positives, respectively, the metric computes the mean average precision and recall at a person-level detection.

Regarding our problem, this metric has two main limitations. Firstly, the OKS metric only considers the annotated body points. This decision prevents the metric to properly measure the keypoint detection's precision of the evaluated methods. This bias can be problematic in contexts where many keypoints cannot be annotated, *e.g.,* in a car context with the strong occlusion (mostly the legs and the bodyside opposite to the camera). Therefore, we want to integrate false-positive keypoints into the performance evaluation of HPE methods. Secondly, the true and false positives are computed at the level of person detections instead of keypoints. In summary, this procedure does not properly characterize the performance of the evaluated methods on the task of keypoint detection.

4.1.2 APK

Average Precision over Keypoints (APK) [40] is a metric that aims to compute precision and recall scores based on keypoints. For each keypoint, a prediction is considered as a true positive if it is located within a defined radial distance from the ground truth. The original work sets this threshold to half the size of the hand. A similar threshold is used to compute Percentage of Correct Keypoints (PCK) [40], and it is defined as a fifth of the torso size (PCK@0.2[19]) or half the head size (PCKh@0.5[19]). Then, non-detected keypoints are counted as false negatives, while points that are detected but not annotated in the ground truth count as false positives. Finally, average precision and recall are computed.

This metric is interesting since it handles the two problems of the COCO OKS metric: it is keypoint-based, and it considers false positives of non-annotated keypoints. This metric has not been used in recent HPE work [2, 20, 34, 39]. One of its main limitations is the use of a distance threshold based on body part size. In fact, the COCO annotation style does not provide hand or head size. The use of the torso is also not an appropriate option in the car cockpit context since, depending on the viewing angle, the torso's full length is not always fully visible on the image.

4.2. mAPK

To address the problems mentioned previously, we propose to compute an evaluation metric based on keypoints instead of people. The mAPK metric reuses the concept from APK of computing average precision and recall based on keypoints but changes the acceptance method. Algorithm 1 summarizes the computation process. The algorithm takes as input a list of matched person (gt, dt) from the ground truth and the detection, respectively, as well as two lists representing unmatched ground truth and detected people. A person (in gt or dt) is defined as a list of keypoint coordinates (if a keypoint is not annotated or detected, the corresponding element in the list is empty). The output of the algorithm is the average precision AP and recall AR.

For single-person settings, the list of matched people consists of the ground-truth annotations and the predicted keypoints. For multiperson settings, a person detector is generally used to compute the people candidates in the scene. In this case, we first carry out a pairing phase to match ground truth and people predictions. We use for this step the pairing algorithm from COCO based on OKS. We set the OKS threshold which controls the pair acceptance to 0 to avoid discarding any person (see [24] for more details).

The calculation of mAPK is carried out as follows. Firstly, we compute a keypoint score KS (Equation 2) for each keypoint which is both annotated and detected. A keypoint is considered as correctly detected, *i.e.,* true positive (TP), if its KS score exceeds a threshold selected between 0 and 1. Otherwise, we consider the ground truth and the prediction keypoint unmatched. Then, we count all unmatched keypoint predictions as false positives and unmatched ground-truth keypoints as false negatives. Finally, we compute precision and recall for each type of keypoint. This process is repeated with different acceptance-threshold values (e.g., from 0.5 to 0.95, with a step of 0.05) and then averaged to obtain the final performance of the evaluated method.

5. Evaluated Architectures

This section describes the HPE methods in evaluated this study. From the state of the art, we selected three recent net-

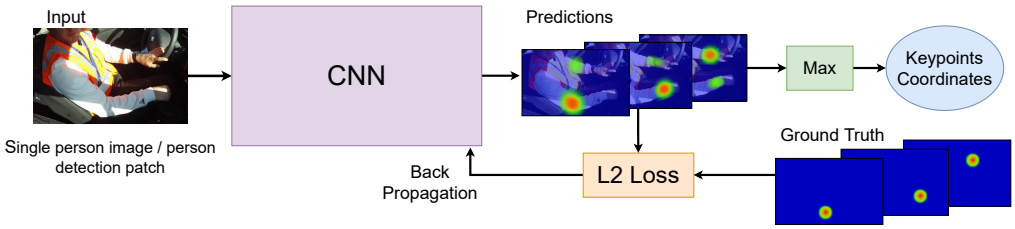

Figure 3: Generic pipeline of HPE methods based on heatmap generation.

Algorithm 1: mAPK computation

Input :
matched_person: pairs of (*gt*, *dt*) of matched ground truth and detected people
unmatched_dts: unmatched detected people
unmatched_gts: unmatched ground-truth people
acceptance_score: acceptance-score threshold
Output : AP, AR

true_positives=0, false_positives=0, false_negatives = 0
for *each (gt, dt) in matched_person* **do**
  for *keypoint kp in the skeleton_representation* **do**
    if *not empty(*$dt[kp]$*) and empty(*$gt[kp]$*)* **then**
      false_positives += 1
    else if *empty(*$dt[kp]$*) and not empty(*$gt[kp]$*)* **then**
      false_negatives += 1
    else
      if $KS(gt[kp], dt[gp]) > acceptance_score$ **then**
        true_positives += 1
      else
        false_positives += 1
        false_negatives += 1

for *each keypoint in all unmatched_gts* **do**
  false_negatives += 1
for *each keypoint in all unmatched_dts* **do**
  false_positives += 1
AP = compute_AP(true_positives, false_positives)
AR = compute_AR(true_positives, false_negatives)

works [5, 22, 39] with competitive performances on single and multiperson settings, as discussed in Section 2.2. Using these two categories of methods will allow us to evaluate the relevance of the mAPK metric for both single-person and multiperson settings. These networks are detection-based architectures (Fig. 3). At last, we describe the procedure followed for training and evaluation of the selected networks.

5.1. Simple Baseline ResNet

Simple Baseline (SBl) architecture [39] bases its feature extraction process on the ResNet architecture [14]. ResNet model has been proved well efficient for image-feature extraction [32, 2] and is often used in other image processing tasks. This backbone is based on several convolution layers gathered as blocks, with skip connections between each module adding the input of the module to the output.

Xiao *et al.* [39] propose to implement ResNet 50 with a different output module for human pose estimation. First, the ResNet 50 backbone learns to extract the features while reducing the shape of the feature maps. Then, the last stage is composed of three upsampling convolutions combined with BatchNorm [17] and ReLu layers, instead of the original ResNet *C5* stage. This deconvolution stage brings back the feature maps to their input size and generates the heatmaps for each keypoint.

5.2. MSPN and RSN

MSPN [22] is a top-down multiperson HPE network. It is built around two steps. First, MegDet [28] object detector identifies the bounding boxes of each person in the images. Then, the picture is cropped around the boxes, and each part serves as input for the multi-stage pose estimator. A stage of the MSPN has a U-shape architecture that processes features at 4 different scales. A bottleneck residual module processes the features at each scale, and skip connections are used between the downsizing stage and its symmetric counterpart in the upsizing stage. Intermediate supervision is applied to each scale of the upsizing stage. Indeed, the loss is applied on heatmaps generated at each scale and which are previously upsampled to the network's output shape. Stages are then stacked several times (four times in this implementation). To reduce information loss between stages, the architecture uses cross-stage aggregation.

RSN [5] follows the same global architecture as MSPN. However, a novel residual steps block module (RSB) replaces the regular residual block in the downsizing stages. The RSB module aims to learn delicate local representations, by splitting the features into four channels. At the end of the multi-stage network before the final loss, a pose refine machine (PRM) is used as an attention mechanism to generate the final heatmaps.

5.3. Model Training and Inference

The training of the models has been done using the code provided by the respective authors in public repositories, following their recommendations for hyperparameters. All

training stages were done on the COCO 2017 train set, with mini-batches of 32 images and data augmentation operations (horizontal flipping, rotation, etc.). The training set is composed of 118k pictures, while the validation set contains 5k images. We used ResNet-50 based Simple Baseline architecture, trained for 140 epochs on the COCO dataset with a learning rate of 1e-3. RSN and MSPN are trained for 384k iterations, with a 5e-4 base learning rate divided by 10 at epochs 90 and 120. The networks were trained on two 24GB Nvidia Titan RTX with 64GB of RAM and an Intel i9900k processor.

Also, since DriPE is a single-person dataset, all network models took as input the full image. However, for COCO which is a multiperson dataset, the models took as input a patch cropped around the output of a person detection algorithm.

6. Results and Discussion

We first present the performance of the three described networks trained on COCO 2017 and tested on both the COCO validation set and the DriPE test set. Then, we present the results of these models after finetuning them on the training set of DriPE dataset. We first use AP metric based on OKS, then compare the results with mAPK metric results.

6.1. Performance of Networks trained on COCO Dataset

This evaluation studies the performance of the trained networks on the COCO validation set (Table 2) using the official dataset evaluation procedure. We validate that the trained models achieves a performance close to the original work (around 2% less on average).

AP OKS (%)	AP	AP^{50}	AP^{75}	AP^{L}	AR	AR^{50}	AR^{75}	AR^{L}
SBl [39]	72	92	80	77	76	93	82	80
MSPN [22]	**77**	94	85	82	**80**	95	87	85
RSN [5]	76	94	84	81	79	94	85	84

Table 2: HPE on the COCO 2017 validation set.

Then, we evaluate the performance of these methods on DriPE test set (Table 3) using the models trained on COCO 2017. Due to the camera placement in the car, DriPE contains only "Large" subjects (subjects with a bounding box containing more than 96^2 pixels [24]). Therefore, it is more suitable to compare COCO and DriPE datasets using AP^L and AR^L column values.

AP OKS (%)	AP	AP^{50}	AP^{75}	AP^{L}	AR	AR^{50}	AR^{75}	AR^{L}
SBl [39]	75	99	91	75	81	99	94	81
MSPN [22]	81	99	97	**81**	85	99	97	**85**
RSN [5]	75	99	93	75	79	99	95	79

Table 3: HPE on the DriPE test set.

The state-of-the-art networks show slightly lower performances on DriPE dataset than on the COCO dataset (Tables 2 and 3). On one hand, we note that on average, AP^L and AR^L are lower on DriPE than on COCO. On another hand, we observe higher precision and recall scores on the three networks when using an OKS threshold of 50% (AP^{50}) or 75% threshold (AP^{75}). The results suggest that most of the improvements to be made in the car context concern the precision of the localization of keypoint predictions (AR / AP threshold superior to 75 %).

6.2. Finetuning on DriPE Dataset

We finetune the three networks on DriPE training set. Finetuning has been done for 10 epochs with a learning rate 10 times lower than the original learning rate used for the COCO base training (Table 4).

AP OKS (%)	AP	AP^{50}	AP^{75}	AP^{L}	AR	AR^{50}	AR^{75}	AR^{L}
SBl [39]	97	100	80	**97** ↑	97	100	99	99
MSPN [22]	97	100	99	**97** ↑	98	100	99	**98**
RSN [5]	91	99	98	91 ↑	94	100	99	94

Table 4: HPE of finetuned networks on the DriPE test set.

Results indicate a gain from 20 to 25% in AP and 10 to 15% in AR after finetuning the networks. This increase can be partially explained by the relatively small variance of the dataset. Therefore, the networks could have overfitted the training set without experiencing an important performance loss on the test set. Despite that, the improvement of performance suggests that the networks learned specific features on DriPE that they did not learn on a general dataset, which highlights the relevance of DriPE dataset to the field. Eventually, AP OKS results may suggest that HPE inside of a car cockpit would be a nearly solved problem, at least when evaluating the performance of keypoint detections methods at a people level.

6.3. Comparison with mAPK Metric

This evaluation assesses the performance of the same models but at the level of keypoint predictions. We recomputed the performance of the evaluated models (Tables 2 and 3) using mAPK metric (Table 5 and Table 6).

mAPK (%)		Head	Sho.	Elb.	Wri.	Hip	Knee	Ank.	Mean
AP	SBl [39]	44	69	59	55	65	62	60	**59**
	MSPN [22]	49	76	60	53	62	47	40	55
	RSN [5]	49	76	59	52	61	46	39	55
AR	SBl [39]	82	86	83	79	80	81	80	82
	MSPN [22]	87	88	87	84	82	85	85	**86**
	RSN [5]	86	88	86	83	82	84	84	85

Table 5: HPE on the COCO 2017 validation set.

We observe that even if AP OKS and mAPK metrics values are not directly comparable, the recall scores are close

mAPK (%)		Head	Sho.	Elb.	Wri.	Hip	Knee	Ank.	Mean
AP	SBl [39]	29	86	78	92	91	75	14	**66**
	MSPN [22]	25	80	77	90	91	77	13	65
	RSN [5]	25	78	76	89	88	68	11	62
AR	SBl [39]	89	92	93	96	88	61	09	75
	MSPN [22]	96	87	96	97	92	77	45	**85**
	RSN [5]	94	85	95	96	89	68	33	81

Table 6: HPE on the DriPE test set.

mAPK (%)		Head	Sho.	Elb.	Wri.	Hip	Knee	Ank.	Mean
AP	SBl [39]	24	90	79	94	98	98	40	**75** ↑
	MSPN [22]	25	89	79	91	97	94	38	73 ↓
	RSN [5]	25	88	78	91	95	86	30	70 ↓
AR	SBl [39]	93	97	98	98	98	98	94	**97** ↑
	MSPN [22]	97	97	98	99	98	94	87	96 ↑
	RSN [5]	91	95	98	98	95	86	73	91 ↑

Table 8: HPE on the DriPE test set of finetuned networks.

between the two metrics (around 75%) (Tables 2, 3, 5, and 6). However, we note that the average precision scores are lower with mAPK. This decay in precision is explained by the high number of false positives that are considered by mAPK but ignored by OKS (Table 7). After analysis, we determined that most of the false positives come from the non-annotated points, particularly for the MSPN and RSN architectures. These results show that the networks are overconfident in their prediction and cannot properly detect the absence of a keypoint on the image. Note that this information cannot be found with AP OKS since the score is not computed at a keypoint level.

	Head	Should.	Elbow	Wrist	Hip	Knee	Ankle	Total
GT	17k	25k	21k	26k	26k	26k	11k	152k
TP	**16k**	21k	20k	23k	23k	18k	**2.8k**	124k
FP	**50k**	5.7k	6.4k	3.1k	3.1k	8.4k	**24k**	100k
FN	0.7k	3.8k	1.1k	2.9k	3.0k	8.3k	8.2k	28k

Table 7: Performance of RSN model on DriPE test set with mAPK metric.

It is worth noticing that even if the head keypoints are considered as some of the easiest keypoints to detect in HPE, trained models have attained a very low average precision on their detection. The overall number of false positives is almost twice higher than the number of true positives (Table 7). In fact, the COCO annotation policy does not annotate occluded keypoints on the head. Therefore, these results highlight that the current models have difficulties not detecting keypoints, *i.e.,* to identify when a keypoint is not visible. Also, the models on DriPE have very low performance on ankles detection, both in precision and recall. The ankles are usually difficult to predict, particularly inside of a car, where the lower limbs are almost totally occluded by the dashboard. This occlusion difficulty paired up with the low contrast and luminosity makes the detection of ankles very challenging.

Finally, we compare the evaluation of the finetuned network using mAPK (Table 8). First, we may observe that this metric confirms the increase of prediction performances indicated by AP OKS (Table 4). Then, we notice that the precision did not increase as much as the recall. These results highlight the importance of DriPE to improve the performance of current models on monitoring people in the consumer car context. But they also bring attention to open challenges on keypoint prediction that cannot be solved by simply finetuning the current models on a dataset-specific task. Astonishingly, Simple Baseline ranks higher than more recent methods according to mAPK. This can be observed on both datasets and it is especially true for precision values. It reveals that Simple Baseline has a lower number of false positives, which shows a better ability to not predict non-annotated keypoints.

7. Conclusion and Perspectives

This paper has presented two contributions: firstly, a new keypoint-based metric, named mAPK, to measure the performance of HPE methods. Secondly, a novel dataset, named DriPE, to benchmark methods for monitoring the pose of drivers in consumer vehicles. The mAPK metric is an extension of APK and OKS evaluation metrics. Results indicate it characterizes more precisely the performance of HPE methods in terms of keypoint detection, both on general and driving datasets.

The DriPE dataset is the first publicly available dataset depicting images of drivers in real-world conditions. We have shown that it may contribute to further improve the performance of deep neural networks on the driver monitoring task. Moreover, the mAPK metric indicates that simply finetuning current methods on the DriPE dataset is insufficient to fully address the driver monitoring task. These results imply that more precise methods must be developed to tackle the existing challenges.

Future work will investigate how to include other evaluation aspects in the proposed metric. For instance, the impact of the confidence threshold on the performance measured. Also, the proposed metric ignores the varying difficulty of predicting keypoints of different limbs and treats equally keypoints of different levels of visibility. Predicting the visibility of keypoints could provide interesting information for a spatial understanding of the interactions of the person with the scene.

Acknowledgements

This work was supported by the Pack Ambition Recherche 2019 funding of the French AURA Region in the context of the AutoBehave project.

References

[1] Mykhaylo Andriluka, Leonid Pishchulin, Peter Gehler, and Bernt Schiele. 2d human pose estimation: New benchmark and state of the art analysis. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2014. 1, 2, 3

[2] Bruno Artacho and Andreas Savakis. Unipose: Unified human pose estimation in single images and videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7035–7044, 2020. 3, 5, 6

[3] Guido Borghi, Stefano Pini, Roberto Vezzani, and Rita Cucchiara. Mercury: a vision-based framework for driver monitoring. In *International Conference on Intelligent Human Systems Integration*, pages 104–110. Springer, 2020. 2

[4] Guido Borghi, Marco Venturelli, Roberto Vezzani, and Rita Cucchiara. Poseidon: Face-from-depth for driver pose estimation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4661–4670, 2017. 2

[5] Yuanhao Cai, Zhicheng Wang, Zhengxiong Luo, Binyi Yin, Angang Du, Haoqian Wang, Xinyu Zhou, Erjin Zhou, Xiangyu Zhang, and Jian Sun. Learning delicate local representations for multi-person pose estimation. In *ECCV*, 2020. 3, 6, 7, 8

[6] Zhe Cao, Gines Hidalgo, Tomas Simon, Shih-En Wei, and Yaser Sheikh. Openpose: realtime multi-person 2d pose estimation using part affinity fields. *IEEE transactions on pattern analysis and machine intelligence*, 43(1):172–186, 2019. 2

[7] Zhe Cao, Tomas Simon, Shih-En Wei, and Yaser Sheikh. Realtime multi-person 2d pose estimation using part affinity fields. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017. 3

[8] Joao Carreira, Pulkit Agrawal, Katerina Fragkiadaki, and Jitendra Malik. Human pose estimation with iterative error feedback. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016. 2

[9] Yu Chen, Chunhua Shen, Xiu-Shen Wei, Lingqiao Liu, and Jian Yang. Adversarial posenet: A structure-aware convolutional network for human pose estimation. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Oct 2017. 3

[10] Yilun Chen, Zhicheng Wang, Yuxiang Peng, Zhiqiang Zhang, Gang Yu, and Jian Sun. Cascaded pyramid network for multi-person pose estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018. 3

[11] Xiao Chu, Wei Yang, Wanli Ouyang, Cheng Ma, Alan L. Yuille, and Xiaogang Wang. Multi-context attention for human pose estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017. 3

[12] Marcin Eichner, Manuel Marin-Jimenez, Andrew Zisserman, and Vittorio Ferrari. 2d articulated human pose estimation and retrieval in (almost) unconstrained still images. *International Journal of Computer Vision*, 99(2):190–214, 2012. 1, 3

[13] Hartmut Feld, Bruno Mirbach, Jigyasa Singh Katrolia, Mohamed Selim, Oliver Wasenmüller, and Didier Stricker. Dfki cabin simulator: A test platform for visual in-cabin monitoring functions. In *Proceedings of the 6th Commercial Vehicle Technology Symposium (CVT), 6th International*, University of Kaiserslautern, 2020. University of Kaiserslautern, Springer. 2

[14] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016. 3, 6

[15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016. 3

[16] Gines Hidalgo, Yaadhav Raaj, Haroon Idrees, Donglai Xiang, Hanbyul Joo, Tomas Simon, and Yaser Sheikh. Single-network whole-body pose estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6982–6991, 2019. 2

[17] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *CoRR*, abs/1502.03167, 2015. 6

[18] Imen Jegham, Anouar Ben Khalifa, Ihsen Alouani, and Mohamed Ali Mahjoub. Mdad: A multimodal and multiview in-vehicle driver action dataset. In *International Conference on Computer Analysis of Images and Patterns*, pages 518–529. Springer, 2019. 2

[19] Sam Johnson and Mark Everingham. Clustered pose and nonlinear appearance models for human pose estimation. In *Proceedings of the British Machine Vision Conference*, 2010. doi:10.5244/C.24.12. 1, 2, 5

[20] Lipeng Ke, Ming-Ching Chang, Honggang Qi, and Siwei Lyu. Multi-scale structure-aware network for human pose estimation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, September 2018. 3, 5

[21] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 25*, pages 1097–1105. Curran Associates, Inc., 2012. 2

[22] Wenbo Li, Zhicheng Wang, Binyi Yin, Qixiang Peng, Yuming Du, Tianzi Xiao, Gang Yu, Hongtao Lu, Yichen Wei, and Jian Sun. Rethinking on multi-stage networks for human pose estimation. https://github.com/megvii-detection/MSPN.git, 2019. 3, 6, 7, 8

[23] Tsung-Yi Lin, Piotr Dollar, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017. 3

[24] Tsung-Yi Lin, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, C. Lawrence Zitnick, and Piotr Dollár. Microsoft coco: Common objects in context, 2015. 1, 2, 3, 5, 7

[25] Manuel Martin, Alina Roitberg, Monica Haurilet, Matthias Horne, Simon Reiß, Michael Voit, and Rainer Stiefelhagen. Drive&act: A multi-modal dataset for fine-grained driver behavior recognition in autonomous vehicles. In *The IEEE*

International Conference on Computer Vision (ICCV), Oct 2019. 2, 4

[26] Alejandro Newell, Zhiao Huang, and Jia Deng. Associative embedding: End-to-end learning for joint detection and grouping. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 2277–2287. Curran Associates, Inc., 2017. 3

[27] Alejandro Newell, Kaiyu Yang, and Jia Deng. Stacked Hourglass Networks for Human Pose Estimation. In Bastian Leibe, Jiri Matas, Nicu Sebe, and Max Welling, editors, *Computer Vision – ECCV 2016*, pages 483–499, Cham, 2016. Springer International Publishing. 3, 4

[28] Chao Peng, Tete Xiao, Zeming Li, Yuning Jiang, Xiangyu Zhang, Kai Jia, Gang Yu, and Jian Sun. Megdet: A large mini-batch object detector. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6181–6189, 2018. 6

[29] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems 28*, pages 91–99. Curran Associates, Inc., 2015. 3

[30] Benjamin Sapp and Ben Taskar. Modec: Multimodal decomposable models for human pose estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013. 1, 2

[31] Ke Sun, Bin Xiao, Dong Liu, and Jingdong Wang. Deep high-resolution representation learning for human pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 3

[32] Xiao Sun, Jiaxiang Shang, Shuang Liang, and Yichen Wei. Compositional human pose regression. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Oct 2017. 2, 6

[33] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015. 2

[34] Wei Tang and Ying Wu. Does learning specific features for related parts help human pose estimation? In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 3, 5

[35] Wei Tang, Pei Yu, and Ying Wu. Deeply learned compositional models for human pose estimation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, September 2018. 3, 4

[36] A. Toshev and C. Szegedy. Deeppose: Human pose estimation via deep neural networks. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1653–1660, 2014. 2

[37] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004. 4

[38] Shih-En Wei, Varun Ramakrishna, Takeo Kanade, and Yaser Sheikh. Convolutional pose machines. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016. 4

[39] Bin Xiao, Haiping Wu, and Yichen Wei. Simple baselines for human pose estimation and tracking. In *Proceedings of the European Conference on Computer Vision (ECCV)*, September 2018. 2, 3, 4, 5, 6, 7, 8

[40] Yi Yang and Deva Ramanan. Articulated human detection with flexible mixtures of parts. *IEEE transactions on pattern analysis and machine intelligence*, 35(12):2878–2890, 2012. 1, 3, 5