



**HAL**  
open science

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

Romain Guesdon, Carlos F Crispim-Junior, Laure Tougne

► **To cite this version:**

Romain Guesdon, Carlos F Crispim-Junior, Laure Tougne. DriPE: A Dataset for Human Pose Estimation in Real-World Driving Settings. 2nd Autonomous Vehicle Vision (AVVision) - International Conference on Computer Vision (ICCV) Workshop, Oct 2021, Virtual Conference, France. 10.1109/ICCVW54120.2021.00321 . hal-03380579

**HAL Id: hal-03380579**

**<https://hal.science/hal-03380579v1>**

Submitted on 15 Oct 2021

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

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

Romain Guesdon

Carlos Crispim-Junior  
Univ Lyon, Lyon 2, LIRIS  
Lyon, France, F-69676

Laure Tougne

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

## Abstract

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

## 1. Introduction

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

30 State-of-the-art methods have reached good performances  
31 on HPE challenges on both single-person [1, 19, 30] and  
32 multiperson datasets [24], especially through deep learn-  
33 ing. However, these general-purpose datasets do not depict  
34 challenging scenes that might occur very often in real-life

35 applications, e.g., strong body occlusion or varying illumina-  
36 tion.

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

55 Moreover, we study the limitations of existing metrics  
56 [12, 24, 40] for the evaluation of the HPE task on keypoint  
57 detection, on both general and driving contexts. Based on  
58 our observations, we propose a new metric called mAPK to  
59 characterize the observed limitations. This metric is essential  
60 to highlight the challenges presented by DriPE, and up to  
61 now ignored in general datasets, such as background and  
62 self-occlusion.

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

<sup>1</sup>DriPE dataset is publicly available on: [https://gitlab.liris.cnrs.fr/aura\\_autobehave/dripe](https://gitlab.liris.cnrs.fr/aura_autobehave/dripe)

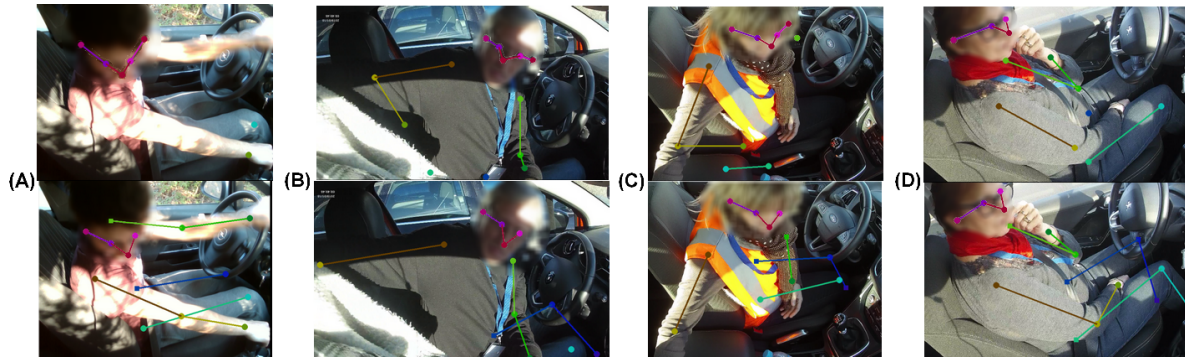


Figure 1: Samples of DriPE dataset. The top and bottom rows show, respectively, pose predictions by Simple Baseline network [39] and ground truth data. Faces have been blurred on this figure to anonymize the participants’ identities.

## 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-

141 sentation using bones instead of keypoints, paired up with 192  
142 the ResNet-50 [14] for both 2D and 3D HPE. 193

143 However, regression-based networks usually lack robust-  
144 ness due to the high non-linearity of the end-to-end structure 194  
145 between the image and the coordinates of the keypoints. 195  
146 To overcome this issue, many methods have proposed a 196  
147 detection-based approach instead. The majority of these 197  
148 methods aim to predict heatmaps, *i.e.*, maps where each pixel 198  
149 represents the probability for the keypoint to be located here. 199  
150 Newell *et al.* [27] propose an architecture composed of new 200  
151 modules called Hourglasses, which aim to extract features 201  
152 from different scales using a network built based on Residual 202  
153 Modules [15]. This architecture has inspired several other 203  
154 works [11, 20, 34, 35]. In addition to Hourglass-based meth- 204  
155 ods, other detection-based architectures have been developed. 205  
156 Chen *et al.* [9] propose an adversarial learning architecture 206  
157 that combines a heatmap pose generator with two discrimina- 207  
158 tors. Xiao *et al.* [39] use the ResNet-50 [14] network but add 208  
159 deconvolution layers in the last convolution stage to predict 209  
160 the heatmaps. Unipose [2] combines a ResNet backbone for 210  
161 feature extraction with a waterfall module to perform HPE. 211  
162 Sun *et al.* [?] use a parallel multi-scale approach similar to 212  
163 the Hourglass with exchange units. 213

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

### 2.2.2 Multiperson Pose Estimation

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

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

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

190 Among the described architectures, top-down methods  
191 currently present the highest performance on HPE. For in-

192 stance, MSPN [22] and RSN [5] have won the COCO Key-  
193 point 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 in-  
ferior 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 net-  
works, 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 per-  
formance 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.

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

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

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

### 246 3.1. Data Collection

247 To build DriPE, we extracted pictures from videos  
 248 recorded during several driving experiments. In each experiment,  
 249 we installed an RGB camera inside the car on top of the passenger's  
 250 door, directed towards the driver. The subjects drive either in a  
 251 real-size replica of a city (closed track) or on actual roads. In  
 252 total, we recorded 19 drivers, allowing us to collect over 100  
 253 hours of video clips. We based the image selection process using  
 254 two metrics: structural similarity index measure (SSIM) [37] and  
 255 brightness differential. We chose these two metrics with the  
 256 objective of extracting pictures with both distinct luminance and  
 257 structure. Therefore, we computed the SSIM and the light  
 258 differential between two successive frames, with a step of  
 259 three frames per second. Then, we selected 10k pictures,  
 260 half with the highest absolute light differential, and half with  
 261 the lowest SSIM. We defined a minimum time gap between  
 262 two selected frames to increase variability.  
 263

### 264 3.2. Annotations

265 We have chosen to follow the COCO dataset's annotation  
 266 style for DriPE since face keypoints are particularly  
 267 interesting to describe driver attention. For each image, we  
 268 annotated the person bounding box and 17 keypoints: arms  
 269 and legs with three keypoints each, and 5 additional markers

270 for the eyes, ears, and nose. We split the annotated keypoints  
 271 into two categories: visible and non-visible. The non-visible  
 272 category corresponds to the occluded points, either by an  
 273 object or by the subject body, but which position can still be  
 274 deducted from the visible body parts. Note that in this study,  
 275 both categories are treated equally by the evaluation methods.  
 276 Following the COCO dataset policy, the face keypoints were  
 277 annotated only if visible.

278 The ground truth heatmaps were generated using centered  
 279 2D Gaussian with a standard deviation of 1px, centered  
 280 around the keypoint location.

## 281 4. Evaluation Metric

282 Following the state of the art, we only evaluate in this  
 283 study detection-based networks, which predict heatmaps. Each  
 284 heatmap is a matrix where the elements represent the probability  
 285 of a particular keypoint to be located at a pixel. Therefore,  
 286 the output of the evaluated network models contains one heatmap  
 287 per skeleton keypoint. Following the common practice in 2D  
 288 single-person HPE [27, 35, 38, 39], the position of a given  
 289 keypoint corresponds to the maximum value of its heatmap. To  
 290 separate predictions from noise, a minimum confidence threshold  
 291 is applied to this maximum. From these coordinates, several  
 292 metrics can be calculated to evaluate the network performances.  
 293

### 294 4.1. Background

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

#### 297 4.1.1 AP OKS

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

305 The metric used to compute the distance between a person's prediction and its ground truth is the OKS (Equation 1). 350

$$351 \text{OKS} = \frac{\sum_i \text{KS}_i * \delta(v_i > 0)}{\sum_i \delta(v_i > 0)} \quad (1) \quad 352$$

353 where  $\text{KS}_i$  is defined as follows: 354

$$355 \text{KS}_i = \exp - \frac{d_i^2}{2 \cdot s^2 \cdot k_i^2} \quad (2) \quad 356$$

357 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. 360

361 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. 365

366 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. 375

### 376 4.1.2 APK

377 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 385

386 as false positives. Finally, average precision and recall are computed. 387

388 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. 395

## 396 4.2. mAPK

397 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. 400

401 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). 405

406 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. 410

## 411 5. Evaluated Architectures

412 This section describes the HPE methods in evaluated this study. From the state of the art, we selected three recent net- 415

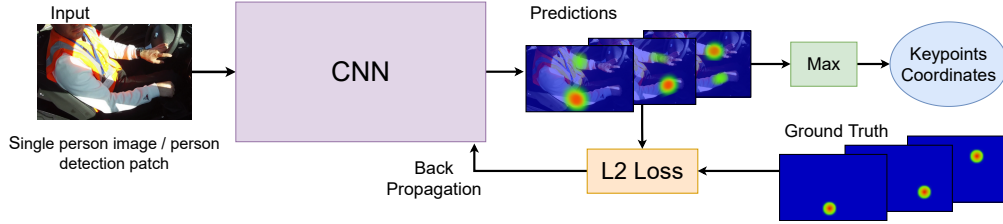


Figure 3: Generic pipeline of HPE methods based on heatmap generation.

---

### 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 (SBI) 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

453 training stages were done on the COCO 2017 train set, with 503  
454 mini-batches of 32 images and data augmentation operations 504  
455 (horizontal flipping, rotation, etc.). The training set is com- 505  
456 posed of 118k pictures, while the validation set contains 5k 506  
457 images. We used ResNet-50 based Simple Baseline archi- 507  
458 tecture, trained for 140 epochs on the COCO dataset with a 508  
459 learning rate of  $1e-3$ . RSN and MSPN are trained for 384k 509  
460 iterations, with a  $5e-4$  base learning rate divided by 10 at 510  
461 epochs 90 and 120. The networks were trained on two 24GB 511  
462 Nvidia Titan RTX with 64GB of RAM and an Intel i9900k 512  
463 processor. 513

464 Also, since DriPE is a single-person dataset, all network 514  
465 models took as input the full image. However, for COCO 515  
466 which is a multiperson dataset, the models took as input 516  
467 a patch cropped around the output of a person detection 517  
468 algorithm. 518

## 469 6. Results and Discussion 519

470 We first present the performance of the three described 520  
471 networks trained on COCO 2017 and tested on both the 521  
472 COCO validation set and the DriPE test set. Then, we present 522  
473 the results of these models after finetuning them on the 523  
474 training set of DriPE dataset. We first use AP metric based 524  
475 on OKS, then compare the results with mAPK metric results. 525

### 476 6.1. Performance of Networks trained on COCO 526 477 Dataset 527

478 This evaluation studies the performance of the trained 529  
479 networks on the COCO validation set (Table 2) using the 530  
480 official dataset evaluation procedure. We validate that the 531  
481 trained models achieves a performance close to the original 532  
482 work (around 2% less on average). 533

483 Then, we evaluate the performance of these methods 534  
484 on DriPE test set (Table 3) using the models trained on 535  
485 COCO 2017. Due to the camera placement in the car, DriPE 536  
486 contains only "Large" subjects (subjects with a bounding 537  
487 box containing more than  $96^2$  pixels [24]). Therefore, it is 538  
488 more suitable to compare COCO and DriPE datasets using 539  
489  $AP^L$  and  $AR^L$  column values. 540

490 The state-of-the-art networks show slightly lower perfor- 541  
491 mances on DriPE dataset than on the COCO dataset (Tables 2 542  
492 and 3). On one hand, we note that on average,  $AP^L$  and 543  
493  $AR^L$  are lower on DriPE than on COCO. On another hand, 544  
494 we observe higher precision and recall scores on the three 545  
495 networks when using an OKS threshold of 50% ( $AP^{50}$ ) or 546  
496 75% threshold ( $AP^{75}$ ). The results suggest that most of the 547  
497 improvements to be made in the car context concern the pre- 548  
498 cision of the localization of keypoint predictions ( $AR / AP$  549  
499 threshold superior to 75 %). 550

### 500 6.2. Finetuning on DriPE Dataset 551

501 We finetune the three networks on DriPE training set. 553  
502 Finetuning has been done for 10 epochs with a learning rate 554

10 times lower than the original learning rate used for the  
COCO base training (Table 4).

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 perfor-  
mance 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. Even-  
tually, 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.

### 518 6.3. Comparison with mAPK Metric 519

519 This evaluation assesses the performance of the same 519  
520 models but at the level of keypoint predictions. We recom- 520  
521 puted the performance of the evaluated models (Tables 2 and 521  
522 3) using mAPK metric (Table 5 and Table 6). 522

523 We observe that even if AP OKS and mAPK metrics 523  
524 values are not directly comparable, the recall scores are close 524  
525 between the two metrics (around 75%) (Tables 2, 3, 5, and 6). 525  
526 However, we note that the average precision scores are lower 526  
527 with mAPK. This decay in precision is explained by the high 527  
528 number of false positives that are considered by mAPK but 528  
529 ignored by OKS (Table 7). After analysis, we determined 529  
530 that most of the false positives come from the non-annotated 530  
531 points, particularly for the MSPN and RSN architectures. 531  
532 These results show that the networks are overconfident in 532  
533 their prediction and cannot properly detect the absence of 533  
534 a keypoint on the image. Note that this information cannot 534  
535 be found with AP OKS since the score is not computed at a 535  
536 keypoint level. 536

537 It is worth noticing that even if the head keypoints are 537  
538 considered as some of the easiest keypoints to detect in HPE, 538  
539 trained models have attained a very low average precision 539  
540 on their detection. The overall number of false positives 540  
541 is almost twice higher than the number of true positives 541  
542 (Table 7). In fact, the COCO annotation policy does not 542  
543 annotate occluded keypoints on the head. Therefore, these 543  
544 results highlight that the current models have difficulties 544  
545 not detecting keypoints, *i.e.*, to identify when a keypoint 545  
546 is not visible. Also, the models on DriPE have very low 546  
547 performance on ankles detection, both in precision and recall. 547  
548 The ankles are usually difficult to predict, particularly inside 548  
549 of a car, where the lower limbs are almost totally occluded by 549  
550 the dashboard. This occlusion difficulty paired up with the 550  
551 low contrast and luminosity makes the detection of ankles 551  
552 very challenging. 552

553 Finally, we compare the evaluation of the finetuned net- 553  
554 work using mAPK (Table 8). First, we may observe that



555 this metric confirms the increase of prediction performances  
 556 indicated by AP OKS (Table 4). Then, we notice that the  
 557 precision did not increase as much as the recall. These  
 558 results highlight the importance of DriPE to improve the  
 559 performance of current models on monitoring people in the  
 560 consumer car context. But they also bring attention to open  
 561 challenges on keypoint prediction that cannot be solved by  
 562 simply finetuning the current models on a dataset-specific  
 563 task. Astonishingly, Simple Baseline ranks higher than more  
 564 recent methods according to mAPK. This can be observed  
 565 on both datasets and it is especially true for precision val-  
 566 ues. It reveals that Simple Baseline has a lower number of  
 567 false positives, which shows a better ability to not predict  
 568 non-annotated keypoints.

## 569 7. Conclusion and Perspectives

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

579 The DriPE dataset is the first publicly available dataset  
 580 depicting images of drivers in real-world conditions. We  
 581 have shown that it may contribute to further improve the per-  
 582 formance of deep neural networks on the driver monitoring  
 583 task. Moreover, the mAPK metric indicates that simply fine-  
 584 tuning current methods on the DriPE dataset is insufficient to  
 585 fully address the driver monitoring task. These results imply  
 586 that more precise methods must be developed to tackle the  
 587 existing challenges.

588 Future work will investigate how to include other evalua-  
 589 tion aspects in the proposed metric. For instance, the impact  
 590 of the confidence threshold on the measured performance.  
 591 Also, the proposed metric ignores the varying difficulty of  
 592 predicting keypoints of different limbs and treats equally  
 593 keypoints of different levels of visibility. Predicting the visi-  
 594 bility of keypoints could provide interesting information for  
 595 a spatial understanding of the interactions of the person with  
 596 the scene.

## 597 Acknowledgements

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

| AP OKS (%) | AP        | AP <sup>50</sup> | AP <sup>75</sup> | AP <sup>L</sup> | AR        | AR <sup>50</sup> | AR <sup>75</sup> | AR <sup>L</sup> |
|------------|-----------|------------------|------------------|-----------------|-----------|------------------|------------------|-----------------|
| SBI [39]   | 72        | 92               | 80               | 77              | 76        | 93               | 82               | 80              |
| MSPN [22]  | <b>77</b> | 94               | 85               | 82              | <b>80</b> | 95               | 87               | 85              |
| RSN [5]    | 76        | 94               | 84               | 81              | 79        | 94               | 85               | 84              |

Table 2: HPE on the COCO 2017 validation set.

| AP OKS (%) | AP | AP <sup>50</sup> | AP <sup>75</sup> | AP <sup>L</sup> | AR | AR <sup>50</sup> | AR <sup>75</sup> | AR <sup>L</sup> |
|------------|----|------------------|------------------|-----------------|----|------------------|------------------|-----------------|
| SBI [39]   | 75 | 99               | 91               | 75              | 81 | 99               | 94               | 81              |
| MSPN [22]  | 81 | 99               | 97               | <b>81</b>       | 85 | 99               | 97               | <b>85</b>       |
| RSN [5]    | 75 | 99               | 93               | 75              | 79 | 99               | 95               | 79              |

Table 3: HPE on the DriPE test set.

| AP OKS (%) | AP | AP <sup>50</sup> | AP <sup>75</sup> | AP <sup>L</sup> | AR | AR <sup>50</sup> | AR <sup>75</sup> | AR <sup>L</sup> |
|------------|----|------------------|------------------|-----------------|----|------------------|------------------|-----------------|
| SBI [39]   | 97 | 100              | 80               | <b>97</b> ↑     | 97 | 100              | 99               | 99              |
| MSPN [22]  | 97 | 100              | 99               | <b>97</b> ↑     | 98 | 100              | 99               | <b>98</b>       |
| RSN [5]    | 91 | 99               | 98               | 91 ↑            | 94 | 100              | 99               | 94              |

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

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

Table 5: HPE on the COCO 2017 validation set.

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

Table 6: HPE on the DriPE test set.

|    | Head       | Should. | Elbow | Wrist | Hip  | Knee | Ankle       | Total |
|----|------------|---------|-------|-------|------|------|-------------|-------|
| GT | 17k        | 25k     | 21k   | 26k   | 26k  | 26k  | 11k         | 152k  |
| TP | <b>16k</b> | 21k     | 20k   | 23k   | 23k  | 18k  | <b>2.8k</b> | 124k  |
| FP | <b>50k</b> | 5.7k    | 6.4k  | 3.1k  | 3.1k | 8.4k | <b>24k</b>  | 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.

| mAPK (%) |           | Head | Sho. | Elb. | Wri. | Hip | Knee | Ank. | Mean        |
|----------|-----------|------|------|------|------|-----|------|------|-------------|
| AP       | SBI [39]  | 24   | 90   | 79   | 94   | 98  | 98   | 40   | <b>75</b> ↑ |
|          | MSPN [22] | 25   | 89   | 79   | 91   | 97  | 94   | 38   | 73 ↓        |
|          | RSN [5]   | 25   | 88   | 78   | 91   | 95  | 86   | 30   | 70 ↓        |
| AR       | SBI [39]  | 93   | 97   | 98   | 98   | 98  | 98   | 94   | <b>97</b> ↑ |
|          | 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.

## 601 References

- 602 [1] Mykhaylo Andriluka, Leonid Pishchulin, Peter Gehler, and  
 603 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, 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 Katroliya, 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, Ihssen 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, 8
- [23] Tsung-Yi Lin, Piotr Dollár, 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 Stiefelwagen. 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

- 717 [26] Alejandro Newell, Zhiao Huang, and Jia Deng. Associative 775  
718 embedding: End-to-end learning for joint detection and group- 776  
719 ing. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. 777  
720 Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in* 778  
721 *Neural Information Processing Systems 30*, pages 2277–2287. 779  
722 Curran Associates, Inc., 2017. 3 780
- 723 [27] Alejandro Newell, Kaiyu Yang, and Jia Deng. Stacked Hour- 781  
724 glass Networks for Human Pose Estimation. In Bastian Leibe, 782  
725 Jiri Matas, Nicu Sebe, and Max Welling, editors, *Computer* 783  
726 *Vision – ECCV 2016*, pages 483–499, Cham, 2016. Springer 784  
727 International Publishing. 3, 4
- 728 [28] Chao Peng, Tete Xiao, Zeming Li, Yuning Jiang, Xiangyu 781  
729 Zhang, Kai Jia, Gang Yu, and Jian Sun. Megdet: A large mini- 782  
730 batch object detector. In *Proceedings of the IEEE Conference*  
731 *on Computer Vision and Pattern Recognition*, pages 6181–  
732 6189, 2018. 6
- 733 [29] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun.  
734 Faster r-cnn: Towards real-time object detection with region  
735 proposal networks. In C. Cortes, N. D. Lawrence, D. D. Lee,  
736 M. Sugiyama, and R. Garnett, editors, *Advances in Neural*  
737 *Information Processing Systems 28*, pages 91–99. Curran  
738 Associates, Inc., 2015. 3
- 739 [30] Benjamin Sapp and Ben Taskar. Modec: Multimodal decom-  
740 posable models for human pose estimation. In *Proceedings*  
741 *of the IEEE Conference on Computer Vision and Pattern*  
742 *Recognition (CVPR)*, 2013. 1, 2
- 743 [31] Ke Sun, Bin Xiao, Dong Liu, and Jingdong Wang. Deep high-  
744 resolution representation learning for human pose estimation.  
745 In *Proceedings of the IEEE/CVF Conference on Computer*  
746 *Vision and Pattern Recognition (CVPR)*, June 2019. 3
- 747 [32] Xiao Sun, Jiaxiang Shang, Shuang Liang, and Yichen Wei.  
748 Compositional human pose regression. In *Proceedings of the*  
749 *IEEE International Conference on Computer Vision (ICCV)*,  
750 Oct 2017. 2, 6
- 751 [33] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet,  
752 Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent  
753 Vanhoucke, and Andrew Rabinovich. Going deeper with  
754 convolutions. In *Proceedings of the IEEE Conference on*  
755 *Computer Vision and Pattern Recognition (CVPR)*, June 2015.  
756 2
- 757 [34] Wei Tang and Ying Wu. Does learning specific features for  
758 related parts help human pose estimation? In *Proceedings of*  
759 *the IEEE/CVF Conference on Computer Vision and Pattern*  
760 *Recognition (CVPR)*, June 2019. 3, 5
- 761 [35] Wei Tang, Pei Yu, and Ying Wu. Deeply learned composi-  
762 tional models for human pose estimation. In *Proceedings*  
763 *of the European Conference on Computer Vision (ECCV)*,  
764 September 2018. 3, 4
- 765 [36] A. Toshev and C. Szegedy. Deeppose: Human pose estima-  
766 tion via deep neural networks. In *2014 IEEE Conference on*  
767 *Computer Vision and Pattern Recognition*, pages 1653–1660,  
768 2014. 2
- 769 [37] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P  
770 Simoncelli. Image quality assessment: from error visibility to  
771 structural similarity. *IEEE transactions on image processing*,  
772 13(4):600–612, 2004. 4
- 773 [38] Shih-En Wei, Varun Ramakrishna, Takeo Kanade, and Yaser  
774 Sheikh. Convolutional pose machines. In *Proceedings of the*  
*IEEE Conference on Computer Vision and Pattern Recogni-*  
*tion (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, 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