Noname manuscript No.
(will be inserted by the editor)

EfficientPose: Scalable single-person pose estimation

Daniel Groos[1] · Heri Ramampiaro[2] · Espen Ihlen[1]

Received: date / Accepted: date

Abstract
Single-person human pose estimation facilitates markerless movement analysis in sports, as well as in clinical applications. Still, state-of-the-art models for human pose estimation generally do not meet the requirements of real-life applications. The proliferation of deep learning techniques has resulted in the development of many advanced approaches. However, with the progresses in the field, more complex and inefficient models have also been introduced, which have caused tremendous increases in computational demands. To cope with these complexity and inefficiency challenges, we propose a novel convolutional neural network architecture, called EfficientPose, which exploits recently proposed EfficientNets in order to deliver efficient and scalable single-person pose estimation. EfficientPose is a family of models harnessing an effective multi-scale feature extractor and computationally efficient detection blocks using mobile inverted bottleneck convolutions, while at the same time ensuring that the precision of the pose configurations is still improved. Due to its low complexity and efficiency, EfficientPose enables real-world applications on edge devices by limiting the memory footprint and computational cost. The results from our experiments, using the challenging MPII single-person benchmark, show that the proposed EfficientPose models substantially outperform the widely-used OpenPose model both in terms of accuracy and computational efficiency. In particular, our top-performing model achieves state-of-the-art accuracy on single-person MPII, with low-complexity ConvNets.

Keywords Human pose estimation · Model scalability · High precision · Computational efficiency · Openly available

Daniel Groos
daniel.groos@ntnu.no

Heri Ramampiaro
heri@ntnu.no

Espen Ihlen
espen.ihlen@ntnu.no

[1] Department of Neuromedicine and Movement Science, Norwegian University of Science and Technology, Trondheim, Norway

[2] Department of Computer Science, Norwegian University of Science and Technology, Trondheim, Norway

1 Introduction

Single-person human pose estimation (HPE) refers to the computer vision task of localizing human skeletal keypoints of a person from an image or video frames. Single-person HPE has many real-world applications, ranging from outdoor activity recognition and computer animation to clinical assessments of motor repertoire and skill practice among professional athletes. The proliferation of deep convolutional neural networks (ConvNets) has advanced HPE and further widen its application areas. ConvNet-based HPE with its increasingly complex network structures, combined with transfer learning, is a very challenging task. However, the availability of high-performing ImageNet [9] backbones, together with large tailor-made datasets, such as MPII for 2D pose estimation [1], has facilitated the development of new improved methods to address the challenges.

An increasing trend in computer vision has driven towards more efficient models [11,38,46]. Recently, EfficientNet [47] was released as a scalable ConvNet archi-

tecture, setting benchmark record on ImageNet with a more computationally efficient architecture. However, within human pose estimation, there is still a lack of architectures that are both accurate and computationally efficient at the same time. In general, current state-of-the-art architectures are computationally expensive and highly complex, thus making them hard to replicate, cumbersome to optimize, and impractical to embed into real-world applications.

The OpenPose network [6] (OpenPose for short) has been one of the most applied HPE methods in real-world applications. It is also the first open-source real-time system for HPE. OpenPose was originally developed for multi-person HPE, but has in recent years been frequently applied to various single-person applications within clinical research and sport sciences [15, 32, 34]. The main drawback with OpenPose is that the level of detail in keypoint estimates is limited due to its low-resolution outputs. This makes OpenPose less suitable for precision-demanding applications, such as elite sports and medical assessments, which all depend on high degree of precision in the assessment of movement kinematics. Moreover, by spending 160 billion floating-point operations (GFLOPs) per inference, OpenPose is considered highly inefficient. Despite these issues, OpenPose seems to remain a commonly applied network for single-person HPE performing markerless motion capture from which critical decisions are based upon [2, 56].

In this paper, we stress the lack of publicly available methods for single-person HPE that are both computationally efficient and effective in terms of estimation precision. To this end, we exploit recent advances in ConvNets and propose an improved approach called EfficientPose. Our main idea is to modify OpenPose into a family of scalable ConvNets for high-precision and computationally efficient single-person pose estimation from 2D images. To assess the performance of our approach, we perform two separate comparative studies. First, we evaluate the EfficientPose model by comparing it against the original OpenPose model on single-person HPE. Second, we compare it against the current state-of-the-art single-person HPE methods on the official MPII challenge, focusing on accuracy as a function of the number of parameters. The proposed EfficientPose models aim to elicit high computational efficiency, while bridging the gap in availability of high-precision HPE networks.

In summary, the main contributions of this paper are the following:

- We propose an improvement of OpenPose, called EfficientPose, that can overcome the shortcomings of the popular OpenPose network on single-person HPE with improved level of precision, rapid convergence during optimization, low number of parameters, and low computational cost.
- With EfficientPose, we suggest an approach providing scalable models that can suit various demands, enabling a trade-off between accuracy and efficiency across diverse application constraints and limited computational budgets.
- We propose a new way to incorporate mobile ConvNet components, which can address the need for computationally efficient architectures for HPE, thus facilitating real-time HPE on the edge.
- We perform an extensive comparative study to evaluate our approach. Our experimental results show that the proposed method achieves significantly higher efficiency and accuracy in comparison to the baseline method, OpenPose. In addition, compared to existing state-of-the-art methods, it achieves competitive results, with a much smaller number of parameters.

The remainder of this paper is organized as follows: Section 2 describes the architecture of OpenPose and highlights research which it can be improved from. Based on this, Section 3 presents our proposed ConvNet-based approach, EfficientPose. Section 4 describes our experiments and presents the results from comparing EfficientPose with OpenPose and other existing approaches. Section 5 discusses our findings and suggests potential future studies. Finally, Section 6 summarizes and concludes the paper.

For the sake of reproducibility, we will make the EfficientPose models available at `https://github.com/daniegr/EfficientPose`.

2 Related work

The proliferation of ConvNets for HPE following the success of DeepPose [54] has set the path for accurate HPE. With OpenPose, Cao et al. [6] made HPE available to the public. As depicted by Figure 1, OpenPose comprises a multi-stage architecture performing a series of detection passes. Provided an input image of 368×368 pixels, OpenPose utilizes an ImageNet pre-trained VGG-19 backbone [41] to extract basic features (step 1 in Figure 1). The features are supplied to a DenseNet-inspired detection block (step 2) arranged as five dense blocks [23], each containing three 3×3 convolutions with PReLU activations [20]. The detection blocks are stacked in a sequence. First, four passes (step 3a-d in Figure 1) of part affinity fields [7] map the associations between body keypoints. Subsequently, two detection passes (step 3e and 3f) predict keypoint

heatmaps [53] to obtain refined keypoint coordinate estimates. In terms of level of detail in the keypoint coordinates, OpenPose is restricted by its output resolution of 46×46 pixels.

The OpenPose architecture can be improved by recent advancements in ConvNets, as follows: First, automated network architecture search has found backbones [47, 48, 62] that are more precise and efficient in image classification than VGG and ResNets [21, 41]. In particular, Tan and Le [47] proposed compound model scaling to balance the image resolution, width (number of network channels), and depth (number of network layers). This resulted in scalable convolutional neural networks, called EfficientNets [47], with which the main goal was to provide lightweight models with a sensible trade-off between model complexity and accuracy across various computational budgets. For each model variant EfficientNet-Bϕ, from the least computationally expensive one being EfficientNet-B0 to the most accurate model, EfficientNet-B7 ($\phi \in [0, 7] \in \mathbb{Z}^{\geq}$), the total number of FLOPs increases by a factor of 2, given by

$$(\alpha \cdot \beta^2 \cdot \gamma^2)^{\phi} \approx 2^{\phi}. \tag{1}$$

Here, α, β and γ denote the coefficients for depth, width, and resolution, respectively, and are set as

$$\alpha = 1.2, \beta = 1.1, \gamma = 1.15. \tag{2}$$

Second, parallel multi-scale feature extraction has improved the precision levels in HPE [25, 33, 44, 57], emphasizing both high spatial resolution and low-scale semantics. However, existing multi-scale approaches in HPE are computationally expensive, both due to their large size and high computational requirements. For example, a typical multi-scale HPE approach has often a size of $16 - 58$ million parameters and requires $10 - 128$ GFLOPS [8, 33, 36, 44, 49, 57, 61]. To cope with this, we propose cross-resolution features, operating on high- and low-resolution input images, to integrate features from multiple abstraction levels with low overhead in network complexity and with high computational efficiency. Existing works on Siamese ConvNets have been promising in utilizing parallel network backbones [17, 18]. Third, mobile inverted bottleneck convolution (MBConv) [38] with built-in squeeze-and-excitation (SE) [22] and Swish activation [37] integrated in EfficientNets has proven more accurate in image classification tasks [47, 48] than regular convolutions [21, 23, 45], while substantially reducing the computational costs [47]. The efficiency of MBConv modules stem from the depthwise convolutions operating in a channel-wise manner [40]. With this approach, it is possible to reduce the computational cost by a factor proportional to the number of channels [48]. Hence, by replacing the regular 3×3 convolutions with up to 384 input channels in the detection blocks of OpenPose with MBConvs, we can obtain more computationally efficient detection blocks. Further, SE selectively emphasizes discriminative image features [22], which may reduce the required number of convolutions and detection passes by providing a global perspective on the estimation task at all times. Using MBConv with SE may have the potential to decrease the number of dense blocks in OpenPose. Fourth, transposed convolutions with bilinear kernel [30] scale up the low-resolution feature maps, thus enabling a higher level of detail in the output confidence maps.

By building upon the work of Tan and Le [47], we present a pool of scalable models for single-person HPE that is able to overcome the shortcomings of the commonly adopted OpenPose architecture. This enables trading off between accuracy and efficiency across different computational budgets in real-world applications. The main advantage of this is that we can use ConvNets that are small and computationally efficient enough to run on edge devices with little memory and low processing power, which is impossible with OpenPose.

3 The EfficientPose approach

In this section, we explain in details the EfficientPose approach. This includes a detailed description of the EfficientPose architecture in light of the OpenPose architecture, and a brief introduction to the proposed variants of EfficientPose.

3.1 Architecture

Figure 1 and Figure 2 depict the architectures of OpenPose and EfficientPose, respectively. As can be observed in these two figures, although being based on OpenPose, the EfficientPose architecture is different from the OpenPose architecture in several aspects, including 1) both high and low-resolution input images, 2) scalable EfficientNet backbones, 3) cross-resolution features, 4) and 5) scalable Mobile DenseNet detection blocks in fewer detection passes, and 6) bilinear upscaling. For a more thorough component analysis of EfficientPose, see Appendix A.

The input of the network consists of high and low-resolution images (1a and 1b in Figure 2). To get the low-resolution image, the high-resolution image is downsampled into half of its pixel height and width, through an initial average pooling layer.

The feature extractor of EfficientPose is composed of the initial blocks of EfficientNets [47] pretrained on

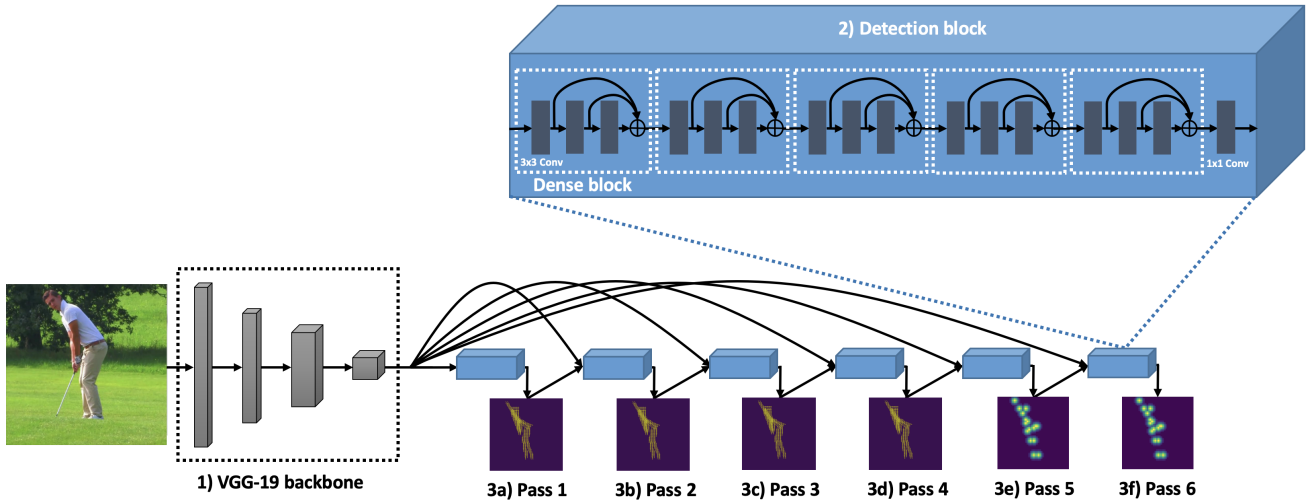

Fig. 1 OpenPose architecture utilizing 1) VGG-19 feature extractor, and 2) 4+2 passes of detection blocks performing 4+2 passes of estimating part affinity fields (3a-d) and confidence maps (3e and 3f)

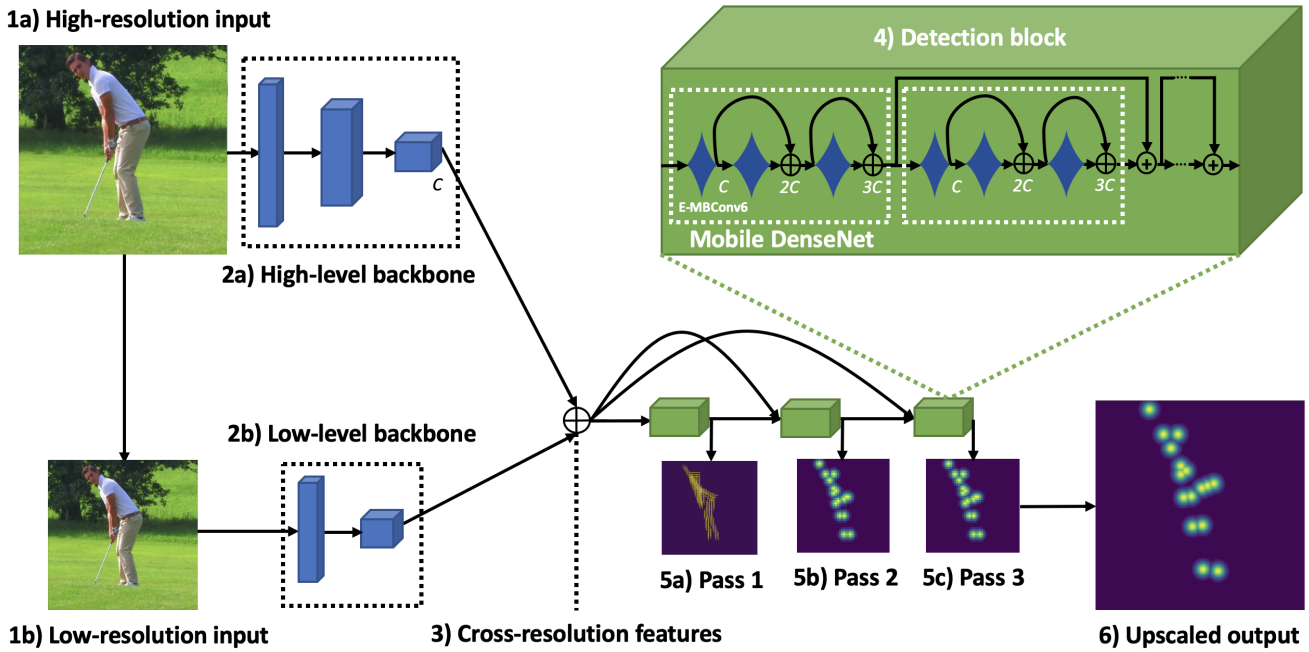

Fig. 2 Proposed architecture comprising 1a) high-resolution and 1b) low-resolution inputs, 2a) high-level and 2b) low-level EfficientNet backbones combined into 3) cross-resolution features, 4) Mobile DenseNet detection blocks, 1+2 passes for estimation of part affinity fields (5a) and confidence maps (5b and 5c), and 6) bilinear upscaling

ImageNet (step 2a and 2b in Figure 2). High-level semantic information is obtained from the high-resolution image using the initial three blocks of a high-scale EfficientNet with $\phi \in [2, 7]$ (see Equation 1), outputting C feature maps (2a in Figure 2). Low-level local information is extracted from the low-resolution image by the first two blocks of a lower-scale EfficientNet-backbone (2b in Figure 2) in the range $\phi \in [0, 3]$. Table 1 provides an overview of the composition of EfficientNet backbones, from low-scale B0 to high-scale B7. The first block of EfficientNets utilizes the MBConvs shown in Figure 3a and 3b, whereas the second and third blocks comprise the MBConv layers in Figure 3c and 3d.

The features generated by the low-level and high-level EfficientNet backbones are concatenated to yield cross-resolution features (step 3 in Figure 2). This enables the EfficientPose architecture to selectively emphasize important local factors from the image of inter-

Table 1 The architecture of the initial three blocks of relevant EfficientNet backbones. For $Conv(K \times K, N, S)$, $K \times K$ denotes filter size, N is number of output feature maps, and S is stride. BN denotes batch normalization. I defines input size, corresponding with image resolution on ImageNet, whereas α^{ϕ} refers to the depth factor as determined by Equation 1

Block	B0	B1	B2	B3	B4	B5	B7
1	$Conv(3 \times 3, 32, 2)$ BN $Swish$			$Conv(3 \times 3, 40, 2)$ BN $Swish$	$Conv(3 \times 3, 48, 2)$ BN $Swish$		$Conv(3 \times 3, 64, 2)$ BN $Swish$
	$MBConv1$ $(3 \times 3, 16, 1)$			$MBConv1$ $(3 \times 3, 24, 1)$			$MBConv1$ $(3 \times 3, 32, 1)$
	–	$MBConv1^*$ $(3 \times 3, 16, 1)$		$MBConv1^*$ $(3 \times 3, 24, 1)$		$[MBConv1^*$ $(3 \times 3, 24, 1)] \times 2$	$[MBConv1^*$ $(3 \times 3, 32, 1)] \times 3$
2	$MBConv6$ $(3 \times 3, 24, 2)$			$MBConv6$ $(3 \times 3, 32, 2)$		$MBConv6$ $(3 \times 3, 40, 2)$	$MBConv6$ $(3 \times 3, 48, 2)$
	$MBConv6^*$ $(3 \times 3, 24, 1)$	$[MBConv6^*$ $(3 \times 3, 24, 1)] \times 2$		$[MBConv6^*$ $(3 \times 3, 32, 1)] \times 2$	$[MBConv6^*$ $(3 \times 3, 32, 1)] \times 3$	$[MBConv6^*$ $(3 \times 3, 40, 1)] \times 4$	$[MBConv6^*$ $(3 \times 3, 48, 1)] \times 6$
3	$MBConv6$ $(5 \times 5, 40, 2)$		$MBConv6$ $(5 \times 5, 48, 2)$		$MBConv6$ $(5 \times 5, 56, 2)$	$MBConv6$ $(5 \times 5, 64, 2)$	$MBConv6$ $(5 \times 5, 80, 2)$
	$MBConv6^*$ $(5 \times 5, 40, 1)$	$[MBConv6^*$ $(5 \times 5, 40, 1)] \times 2$	$[MBConv6^*$ $(5 \times 5, 48, 1)] \times 2$		$[MBConv6^*$ $(5 \times 5, 56, 1)] \times 3$	$[MBConv6^*$ $(5 \times 5, 64, 1)] \times 4$	$[MBConv6^*$ $(5 \times 5, 80, 1)] \times 6$
I	224×224	240×240	260×260	300×300	380×380	456×456	600×600
C	40		48		56	64	80
α^{ϕ}	$1.2^0 = 1.0$	$1.2^1 = 1.2$	$1.2^2 = 1.4$	$1.2^3 = 1.7$	$1.2^4 = 2.1$	$1.2^5 = 2.5$	$1.2^7 = 3.6$

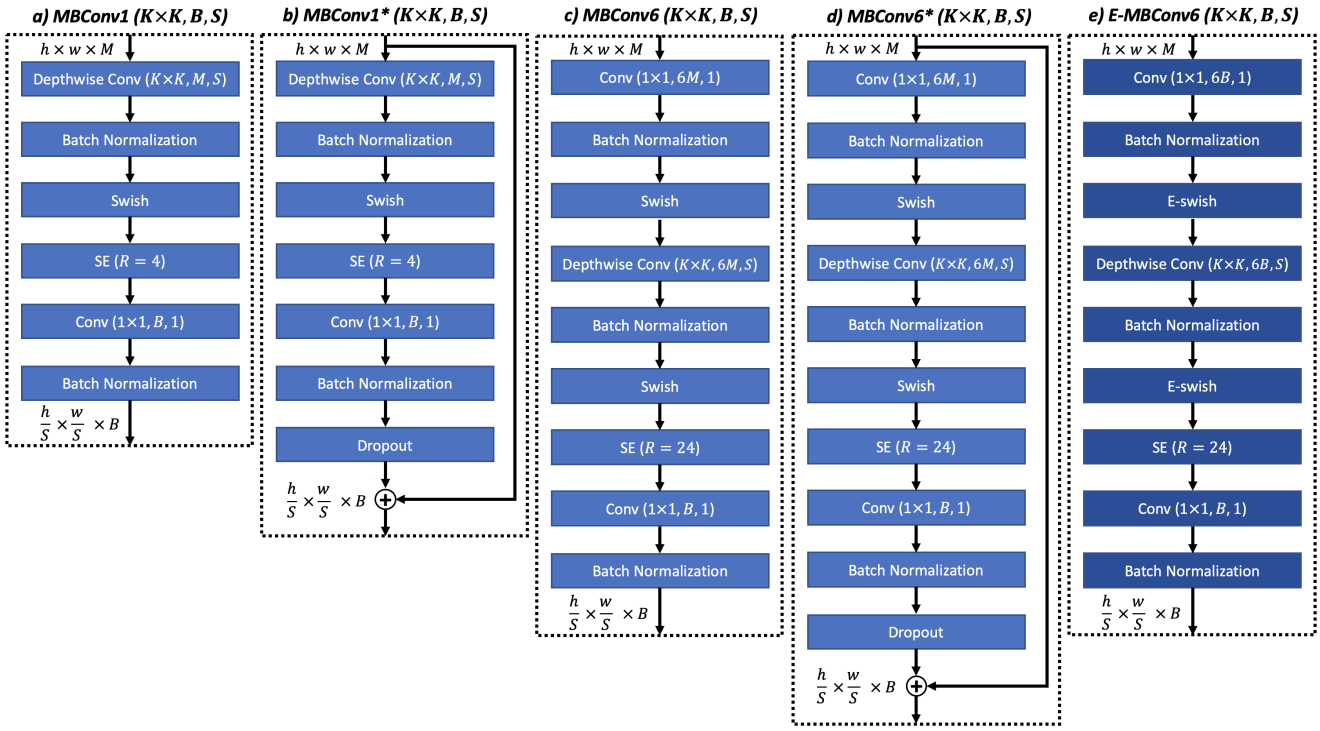

Fig. 3 The composition of MBConvs. From left: a-d) $MBConv(K \times K, B, S)$ in EfficientNets performs depthwise convolution with filter size $K \times K$ and stride S, and outputs B feature maps. $MBConv^*$ (b and d) extends regular MBConvs by including dropout layer and skip connection. e) $E\text{-}MBConv6(K \times K, B, S)$ in Mobile DenseNets adjusts $MBConv6$ with E-swish activation and number of feature maps in expansion phase as $6B$. All MBConvs take as input M feature maps with spatial height and width of h and w, respectively. R is the reduction ratio of SE

est and the overall structures that guide high-quality pose estimation. In this way, we enable an alternative simultaneous handling of different features at multiple abstraction levels.

From the extracted features, the desired keypoints are localized through an iterative detection process, where each detection pass performs supervised prediction of output maps. Each detection pass comprises a detection block and a single 1×1 convolution for output prediction. The detection blocks across all detection passes elicit the same basic architecture, comprising Mobile DenseNets (see step 4 in Figure 2). Data from Mobile DenseNets are forwarded to subsequent layers of the detection block using residual connections. The Mobile DenseNet is inspired by DenseNets [23] supporting reuse of features, avoiding redundant layers, and MBConv with SE, thus enabling low memory footprint. In our adaptation of the MBConv operation ($E\text{-}MBConv6(K \times K, B, S)$ in Figure 3e), we consistently utilize the highest performing combination from [46], i.e., a kernel size ($K \times K$) of 5×5 and an expansion ratio of 6. We also avoid downsampling (i.e., $S = 1$) and scale the width of Mobile DenseNets by outputting number of channels relative to the high-level backbone ($B = C$). We modify the original $MBConv6$ operation by incorporating E-swish as activation function with β value of 1.25 [16]. This has a tendency to accelerate progression during training compared to the regular Swish activation [37]. We also adjust the first 1×1 convolution to generate a number of feature maps relative to the output feature maps B rather than the input channels M. This reduces the memory consumption and computational latency since $B \leq M$, with $C \leq M \leq 3C$. With each Mobile DenseNet consisting of three consecutive $E\text{-}MBConv6$ operations, the module outputs $3C$ feature maps.

EfficientPose performs detection in two rounds (step 5a-c in Figure 2). First, the overall pose of the person is anticipated through a single pass of skeleton estimation (5a). This aims to facilitate the detection of feasible poses and to avoid confusion in case of several persons being present in an image. Skeleton estimation is performed utilizing part affinity fields as proposed in [7]. Following skeleton estimation, two detection passes are performed to estimate heatmaps for keypoints of interest. The former of these acts as a coarse detector (5b in Figure 2), whereas the latter (5c in Figure 2) refines localization to yield more accurate outputs.

Note that in OpenPose, the heatmaps of the final detection pass are constrained to a low spatial resolution, which are incapable of achieving the amount of details that are normally inherent in the high-resolution input [6]. To improve this limitation of OpenPose, a series of three transposed convolutions performing bilinear upsampling are added for $8\times$ upscaling of the low-resolution heatmaps (step 6 in Figure 1). Thus, we project the low-resolution output onto a space of higher resolution in order to allow an increased level of detail. To achieve the proper level of interpolation while operating efficiently, each transposed convolution increases the map size by a factor of 2, using a stride of 2 with a 4×4 kernel.

3.2 Variants

Following the same principle as suggested in the original EfficientNet [47], we scale the EfficientPose network architecture by adjusting the three main dimensions, i.e., input resolution, network width, and network depth, using the coefficients of Equation 2. The results from this scaling are five different architecture variants that are given in Table 2, referred to as EfficientPose I to IV and RT). As can be observed in this table, the input resolution, defined by the spatial dimensions of the image ($H \times W$), is scaled utilizing the high and low-level EfficientNet backbones that best match the resolution of high and low-resolution inputs (see Table 1). Here, the network width refers to the number of feature maps that are generated by each $E\text{-}MBConv6$. As described in Section 3.1, width scaling is achieved using the same width as the high-level backbone (i.e., C). The scaling of network depth is achieved in the number of Mobile DenseNets (i.e., $MD(C)$ in Table 2) in the detection blocks. Also, this ensures that receptive fields across different models and spatial resolutions have similar relative sizes. For each model variant, we select the number (D) of Mobile DenseNets that best approximates the original depth factor α^{ϕ} in the high-level EfficientNet backbone (Table 1). More specifically, the number of Mobile DenseNets are determined by Equation 3, rounding to the closest integer. In addition to EfficientPose I to IV, the single-resolution model EfficientPose RT is formed to match the scale of the smallest EfficientNet model, providing HPE in extremely low latency applications.

$$D = \lfloor \alpha^{\phi} + 0.5 \rfloor \tag{3}$$

3.3 Summary of proposed framework

As can be inferred from the discussion above, the EfficientPose framework comprises a family of five ConvNets (i.e., EfficientPose I-IV and RT) that are constructed by compound scaling [47]. With this, EfficientPose exploits the advances in computationally efficient

Table 2 Variants of EfficientPose obtained by scaling resolution, width, and depth. Mobile DenseNets $MD(C)$ computes $3C$ feature maps. P and Q denotes the number of 2D part affinity fields and confidence maps, respectively. $Conv^T(K \times K, O, S)$ defines transposed convolutions with kernel size $K \times K$, output maps O, and stride S

Stage	EfficientPose RT	EfficientPose I	EfficientPose II	EfficientPose III	EfficientPose IV
High-resolution input	224×224	256×256	368×368	480×480	600×600
High-level backbone	B0 (Block 1-3)	B2 (Block 1-3)	B4 (Block 1-3)	B5 (Block 1-3)	B7 (Block 1-3)
Low-resolution input	–	128×128	184×184	240×240	300×300
Low-level backbone	–	B0 (Block 1-2)	B0 (Block 1-2)	B1 (Block 1-2)	B3 (Block 1-2)
Detection block	$MD(40)$	$MD(48)$	$[MD(56)] \times 2$	$[MD(64)] \times 3$	$[MD(80)] \times 4$
Prediction pass 1	$Conv(1 \times 1, 2P, 1)$				
Prediction pass 2-3	$Conv(1 \times 1, Q, 1)$				
Upscaling	$[Conv^T(4 \times 4, Q, 2)] \times 3$				

ConvNets for image recognition to construct a scalable network architecture that is capable of performing single-person HPE across different computational constraints. More specifically, EfficientPose utilizes both high and low-resolution images to provide two separate viewpoints that are processed independently through high and low-level backbones, respectively. The resulting features are concatenated to produce cross-resolution features, enabling selective emphasis on global and local image information. The detection stage employs a scalable mobile detection block to perform detection in three passes. The first pass estimates person skeletons through part affinity fields [7] to yield feasible pose configurations. The second and third passes estimate keypoint locations with progressive improvement in precision. Finally, the low-resolution prediction of the third pass is scaled up through bilinear interpolation to further improve the precision level.

4 Experiments and results

4.1 Experimental setup

We evaluate EfficientPose and compare it with OpenPose on the single-person MPII dataset [1], containing images of mainly healthy adults in a wide range of different outdoor and indoor everyday activities and situations, such as sports, fitness exercises, housekeeping activities, and public events (Figure 4a). All models are optimized on MPII using stochastic gradient descent (SGD) on the mean squared error (MSE) of the model predictions relative to the target coordinates. More specifically, we applied SGD with momentum and cyclical learning rates (see Appendix B for more information and further details on the optimization procedure). The learning rate is bounded according to the model-specific value of which it does not diverge during the first cycle (λ_{max}) and $\lambda_{min} = \frac{\lambda_{max}}{3000}$. The model backbones (i.e., VGG-19 for OpenPose, and EfficientNets for EfficientPose) are initialized with pretrained ImageNet weights, whereas the remaining layers employ random weight initialization. Supported by our experiments on training efficiency (see Appendix A), we train the models for 200 epochs, except for OpenPose, which requires a higher number of epochs to converge (see Figure 5 and Table 5).

The training and validation portion of the dataset comprises 29K images, and by adopting a standard random split, we obtain 26K and 3K instances for training and validation, respectively. We augment the images during training using random horizontal flipping, scaling ($0.75-1.25$), and rotation ($+/-$ 45 degrees). We utilize a batch size of 20, except for the high-resolutional EfficientPose III and IV, which both require a smaller batch size to fit into the GPU memory, 10 and 5, respectively. The experiments are carried out on an NVIDIA Tesla V100 GPU.

The evaluation of model accuracy is performed using the $PCK_h@\tau$ metric. $PCK_h@\tau$ is defined as the fraction of predictions residing within a distance τl from the ground truth location (see Figure 4b). l is 60% of the diagonal d of the head bounding box, and τ the accepted percentage of misjudgment relative to l. $PCK_h@50$ is the standard performance metric for MPII but we also include the stricter $PCK_h@10$ metric for assessing models' ability to yield highly precise keypoint estimates. As commonly done in the field, the final model predictions are obtained by applying multi-scale testing procedure [44, 49, 57]. Due to the restriction in the number of attempts for official evaluation on MPII, we only used the test metrics on the OpenPose baseline, and the most efficient and most accurate models, EfficientPose RT and EfficientPose IV, respectively. To measure model efficiency, both FLOPs and number of parameters are supplied.

4.2 Results

Table 3 shows the results of our experiments with OpenPose and EfficientPose on the MPII validation dataset.

Fig. 4 The MPII single-person pose estimation challenge. From left: a) 10 images from the MPII test set displaying some of the variation and difficulties inherent in this challenge. b) The evaluation metrics PCK_h@50 and PCK_h@10 define the average of predictions within τl distance ($l = 0.6d$) from the ground-truth location (e.g., left elbow), with τ being 50% and 10%, respectively

As can be observed in this table, EfficientPose consistently outperformed OpenPose with regards to efficiency, with $2.2-184\times$ reduction in FLOPs and $4-56\times$ fewer number of parameters. In addition to this, all the model variants of EfficientPose achieved better high-precision localization, with a $0.8-12.9\%$ gain in PCK_h@10 as compared to OpenPose. In terms of PCK_h@50, the high-end models, i.e., EfficientPose II-IV, managed to gain $0.6-2.2\%$ improvements against OpenPose. As Table 4 depicts, EfficientPose IV achieved state-of-the-art results (a mean PCK_h@50 of 91.2) on the official MPII test dataset for models with number of parameters of a size less than 10 million.

Compared to OpenPose, EffcientPose also exhibited rapid convergence during training. We optimized both approaches on similar input resolution, which defaults to 368×368 for OpenPose, corresponding to EfficientPose II. The training graph shown in Figure 5 demonstrates that EfficientPose converges early, whereas OpenPose requires up to 400 epochs before achieving proper convergence. Nevertheless, OpenPose benefited from this prolonged training in terms of precision, with a 2.6% improvement in PCK_h@50 during the final 200 epochs, whereas EfficientPose II had a minor gain of 0.4% (see Table 5).

5 Discussion

In this section, we discuss several aspects of our findings and possible avenues for further research.

5.1 Improvements over OpenPose

The precision of HPE methods is a key success factor for analyses of movement kinematics, like segment positions and joint angles, for assessment of sport performance in athletes, or motor disabilities in patients. Facilitated by cross-resolution features and upscaling of output (see Appendix A), EfficientPose achieved a higher precision than OpenPose [6], with a 57% relative improvement in PCK_h@10 on single-person MPII (Table 3). What this means is that the EfficientPose architecture is generally more suitable in performing precision-demanding single-person HPE applications, like medical assessments and elite sports, than OpenPose.

Another aspect to have in mind is that, for some applications (e.g., exercise games and baby monitors), we might be more interested in the latency of the system and its ability to respond quickly. Hence, the degree of correctness in keypoint predictions might be less crucial. In such scenarios, with applications that demand high-speed predictions, the 460K parameter model, EfficientPose RT, consuming less than one GFLOP, would be suitable. Nevertheless, it still manages to provide higher precision level than current approaches in the high-speed regime, e.g., [5, 50]. Further, the scalability of EfficientPose enables flexibility in various situations and across different types of hardware, whereas OpenPose suffers from its large number of parameters and computational costs (FLOPs).

5.2 Strengths of the EfficientPose approach

The use of MBConv in HPE is to the best of our knowledge an unexplored research area. This has also been partly our main motivation for exploring the use of

Table 3 Performance of EfficientPose compared to OpenPose on the MPII validation dataset, as evaluated by efficiency (number of parameters and FLOPs, and relative reduction in parameters and FLOPs compared to OpenPose) and accuracy (mean PCK_h@50 and mean PCK_h@10)

Model	Parameters	Parameter reduction	FLOPs	FLOP reduction	PCK_h@50	PCK_h@10
OpenPose [6]	$25.94M$	1×	$160.36G$	1×	87.60	22.76
EfficientPose RT	$0.46M$	56×	$0.87G$	184×	82.88	23.56
EfficientPose I	$0.72M$	36×	$1.67G$	96×	85.18	26.49
EfficientPose II	$1.73M$	15×	$7.70G$	21×	88.18	30.17
EfficientPose III	$3.23M$	8.0×	$23.35G$	6.9×	89.51	30.90
EfficientPose IV	$6.56M$	4.0×	$72.89G$	2.2×	89.75	35.63

Table 4 State-of-the-art results in PCK_h@50 (both for individual body parts and overall mean value) on the official MPII test dataset [1] compared to the number of parameters

Model	Parameters	Head	Shoulder	Elbow	Wrist	Hip	Knee	Ankle	Mean
Pishchulin et al., ICCV'13 [35]	–	74.3	49.0	40.8	32.1	36.5	34.4	35.2	44.1
Tompson et al., NIPS'14 [53]	–	95.8	90.3	80.5	74.3	77.6	69.7	62.8	79.6
Lifshitz et al., ECCV'16 [28]	$76M$	97.8	93.3	85.7	80.4	85.3	76.6	70.2	85.0
Tang et al., BMVC'18 [50]	$10M$	97.4	96.2	91.8	87.3	90.0	87.0	83.3	90.8
Newell et al., ECCV'16 [33]	$26M$	98.2	96.3	91.2	87.1	90.1	87.4	83.6	90.9
Zhang et al., CVPR'19 [60]	$3M$	98.3	96.4	91.5	87.4	90.9	87.1	83.7	91.1
Bulat et al., FG'20 [5]	$9M$	98.5	96.4	91.5	87.2	90.7	86.9	83.6	91.1
Yang et al., ICCV'17 [57]	$27M$	98.5	96.7	92.5	88.7	91.1	88.6	86.0	92.0
Tang et al., ECCV'18 [49]	$16M$	98.4	96.9	92.6	88.7	91.8	89.4	86.2	92.3
Sun et al., CVPR'19 [44]	$29M$	98.6	96.9	92.8	89.0	91.5	89.0	85.7	92.3
Zhang et al., arXiv'19 [61]	$24M$	98.6	97.0	92.8	88.8	91.7	89.8	86.6	92.5
OpenPose [6]	$25.94M$	97.7	94.7	89.5	84.7	88.4	83.6	79.3	88.8
EfficientPose RT	$0.46M$	97.0	93.3	85.0	79.2	85.9	77.0	71.0	84.8
EfficientPose IV	$6.56M$	98.2	96.0	91.7	87.9	90.3	87.5	83.9	91.2

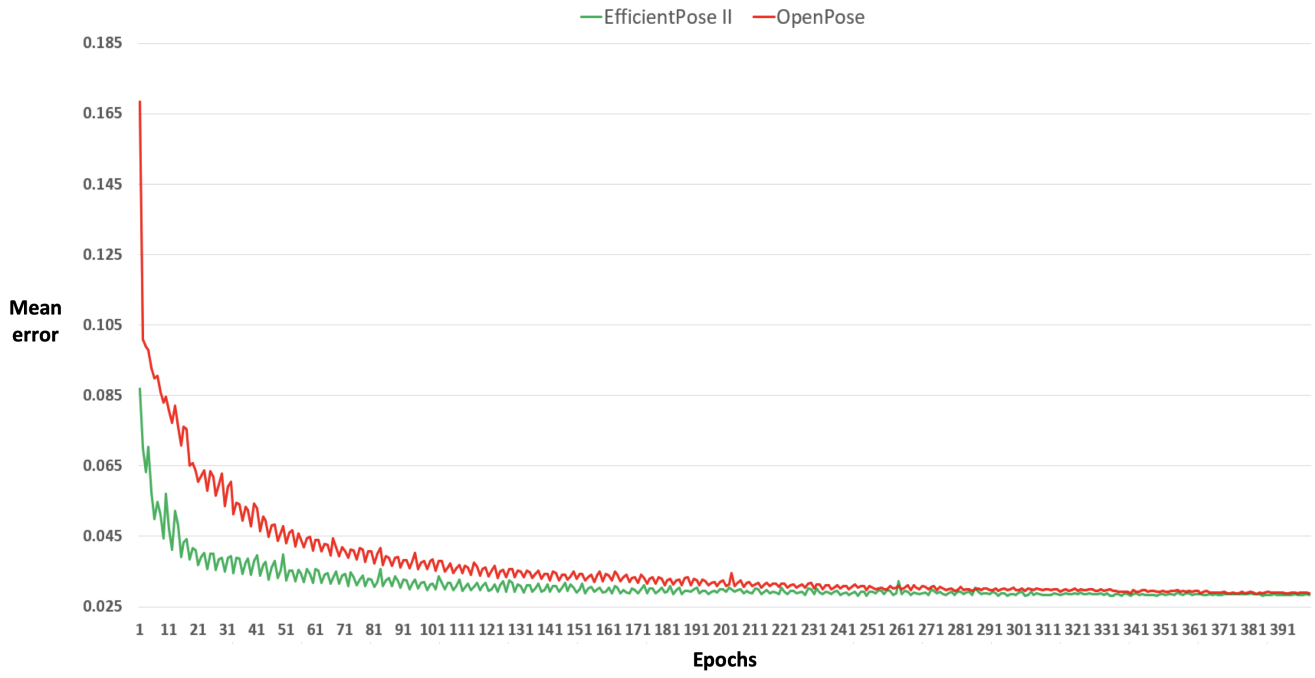

Fig. 5 The progression of the mean error of EfficientPose II and OpenPose on the MPII validation set during the course of training

Table 5 Model accuracy on the MPII validation dataset in relation to the number of training epochs

Model	Epochs	$\mathbf{PCK_h@50}$
OpenPose [6]	100	80.47
OpenPose [6]	200	85.00
OpenPose [6]	400	87.60
EfficientPose II	100	87.05
EfficientPose II	200	88.18
EfficientPose II	400	88.56

MBConv in our EfficientPose approach, recognizing its success in image classification [47]. Our experimental results showed that EfficientPose approached state-of-the-art performance on the single-person MPII benchmark despite a large reduction in the number of parameters (Table 4). This means that the parameter-efficient MBConvs provide value in HPE as with other computer vision tasks, such as image classification and object detection. This, in turns, makes MBConv a very suitable component for HPE networks. For this reason, it would be interesting to investigate the effect of combining it with other novel HPE architectures, such as Hourglass and HRNet [33, 44].

Further, the use of EfficientNet as a backbone, and the proposed cross-resolution feature extractor combining several EfficientNets for improved handling of basic features, are also interesting avenues to explore further. From the present study, it is reasonable to assume that EfficientNets could replace commonly used backbones for HPE, such as VGG and ResNets, which would reduce the computational overheads associated with these approaches [21, 41]. Also, a cross-resolution feature extractor could be useful for precision-demanding applications by providing an improved performance on PCK_h@10 (Table 6).

We also observed that EfficientPose benefited from compound model scaling across resolution, width and depth. This benefit was reflected by the increasing improvements in PCK_h@50 and PCK_h@10 from EfficientPose RT through EfficientPose I to EfficientPose IV (Table 3). To conclude, we can exploit this to further examine scalable ConvNets for HPE, and thus obtain insights into appropriate sizes of HPE models (i.e., number of parameters), required number of FLOPs, and obtainable precision levels.

In this study, OpenPose and EfficientPose were optimized on the general-purpose MPII Human Pose Dataset. For many applications (e.g., action recognition and video surveillance) the variability in MPII may be sufficient for directly applying the models on real-world problems. Nonetheless, there are other particular scenarios that deviate from the setting addressed in this paper. The MPII dataset comprises mostly healthy adults in a variety of every day indoor and outdoor activities [1]. In less natural environments (e.g., movement science laboratories or hospital settings) and with humans of different anatomical proportions such as children and infants [39], careful consideration must be taken. This could include a need for fine-tuning of the MPII models on more specific datasets related to the problem at hand. As mentioned earlier, our experiments showed that EfficientPose was more easily trainable than OpenPose (Figure 5 and Table 5). This trait of rapid convergence suggests that exploring the use of transfer learning on the EfficientPose models on other HPE data could provide interesting results.

5.3 Avenues for further research

The precision level of pose configurations provided by EfficientPose in the context of target applications is a topic considered beyond the scope of this paper and has for this reason been left for further studies. We can establish the validity of EfficientPose for robust single-person pose estimation already by examining whether the movement information supplied by the proposed framework is of sufficiently good quality for tackling challenging problems, such as complex human behavior recognition [12, 29]. To assess this, we could, for example, compare the precision level of the keypoint estimates supplied by EfficientPose with the movement information provided by body-worn movement sensors. Moreover, we could combine the proposed image-based EfficientPose models with body-worn sensors, such as inertial measurement unit (IMU) [27], or physiological signals, like electrical cardiac activity and electrical brain activity [14], to potentially achieve improved precision levels and an increased robustness. Our hypothesis is that using body-worn sensors or physiological instruments could be useful in situations where body parts are extensively occluded, such that camera-based recognition alone may not be sufficient for accurate pose estimation.

Another path for further study and validation is the capability of EfficientPose to perform multi-person HPE. The improved computational efficiency of Effi-

cientPose compared to OpenPose has the potential to also benefit multi-person HPE. State-of-the-art methods for multi-person HPE are dominated by top-down approaches, which require computation that is normally proportional to the number of individuals in the image [13, 59]. In crowded scenes, top-down approaches are highly resource demanding. Similar to the original OpenPose [6], and few other recent works on multi-person HPE [19,24], EfficientPose incorporates part affinity fields, which would enable the grouping of keypoints into persons, and thus allowing to perform multi-person HPE in a bottom-up manner. This would reduce the computational overhead into a single network inference per image, and hence yield more computationally efficient multi-person HPE.

Further, it would be interesting to explore the extension of the proposed framework to perform 3D pose estimation as part of our future research. In accordance with recent studies, 3D pose projection from 2D images can be achieved, either by employing geometric relationships between 2D keypoint positions and 3D human pose models [58], or by leveraging occlusion-robust pose-maps (ORPM) in combination with annotated 3D poses [3, 31].

The architecture of EfficientPose and the training process can be improved in several ways. First, the optimization procedure (see Appendix B) was developed for maximum PCK_h@50 accuracy on OpenPose, and simply reapplied to EfficientPose. Other optimization procedures might be more appropriate, including alternative optimizers (e.g., Adam [26] and RMSProp [52]), and other learning rate and sigma schedules.

Second, only the backbone of EfficientPose was pretrained on ImageNet. This could restrict the level of accuracy on HPE because large-scale pretraining not only supplies robust basic features but also higher-level semantics. Thus, it would be valuable to assess the effect of pretraining on model precision in HPE. We could, for example, pretrain the majority of ConvNet layers on ImageNet, and retrain these on HPE data.

Third, the proposed compound scaling of EfficientPose assumes that the scaling relationship between resolution, width, and depth, as defined by Equation 2, is identical in HPE and image classification. However, the optimal compound scaling coefficients might be different for HPE, where the precision level is more dependent on image resolution, than for image classification. Based on this, a topic for further studies could be to conduct neural architecture search across different combinations of resolution, width, and depth in order to determine the optimal combination of scaling coefficients for HPE. Regardless of the scaling coefficients, the scaling of detection blocks in EfficientPose could be improved. The block depth (i.e., number of Mobile DenseNets) slightly deviates from the original depth coefficient in EfficientNets based on the rigid nature of the Mobile DenseNets. A carefully designed detection block could address this challenge by providing more flexibility with regards to the number of layers and the receptive field size.

Fourth, the computational efficiency of EfficientPose could be further improved by the use of teacher-student network training (i.e., knowledge distillation) [4] to transfer knowledge from a high-scale EfficientPose teacher network to a low-scale EfficientPose student network. This technique has already shown promising results in HPE when paired with the stacked hourglass architecture [33, 60]. Sparse networks, network pruning, and weight quantization [11, 55] could also be included in the study to facilitate the development of more accurate and responsive real-life systems for HPE. Finally, for high performance inference and deployment on edge devices, further speed-up could be achieved by the use of specialized libraries such as NVIDIA TensorRT and TensorFlow Lite [10, 51].

In summary, EfficientPose tackles single-person HPE with an improved degree of precision compared to the commonly adopted OpenPose network [6]. In addition to this, the EfficientPose models have the ability to yield high performance with a large reduction in number of parameters and FLOPs. This has been achieved by exploiting the findings from contemporary research within image recognition on computationally efficient ConvNet components, most notably MBConvs and EfficientNets [38,47]. Again, for the sake of reproducibility, we have made the EfficientPose models publicly available for other researchers to test and possibly further development.

6 Conclusion

In this work, we have stressed the need for a publicly accessible method for single-person HPE that suits the demands for both precision and efficiency across various applications and computational budgets. To this end, we have presented a novel method called EfficientPose, which is a scalable ConvNet architecture leveraging a computationally efficient multi-scale feature extractor, novel mobile detection blocks, skeleton estimation, and bilinear upscaling. In order to have model variants that are able to flexibly find a sensible trade-off between accuracy and efficiency, we have exploited model scalability in three dimensions: input resolution, network width, and network depth. Our experimental results have demonstrated that the proposed approach has the

capability to offer computationally efficient models, allowing real-time inference on edge devices. At the same time, our framework offers flexibility to be scaled up to deliver more precise keypoint estimates than commonly used counterparts, at an order of magnitude less parameters and computational costs (FLOPs). Taking into account the efficiency and high precision level of our proposed framework, there is a reason to believe that EfficientPose will provide an important foundation for the next-generation markerless movement analysis.

In our future work, we plan to develop new techniques to further improve the model effectiveness, especially in terms of precision, by investigating optimal compound model scaling for HPE. Moreover, we will deploy EfficientPose on a range of applications to validate its applicability, as well as feasibility, in real-world scenarios.

Acknowledgements

The research is funded by RSO funds from the Faculty of Medicine and Health Sciences at the Norwegian University of Science and Technology. The experiments were carried out utilizing computational resources provided by the Norwegian Open AI Lab.

References

1. Andriluka, M., Pishchulin, L., Gehler, P., Schiele, B.: 2d human pose estimation: New benchmark and state of the art analysis. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2014)
2. Barra, P., Bisogni, C., Nappi, M., Freire-Obregón, D., Castrillón-Santana, M.: Gait analysis for gender classification in forensics. In: International Conference on Dependability in Sensor, Cloud, and Big Data Systems and Applications, pp. 180–190. Springer (2019)
3. Benzine, A., Luvison, B., Pham, Q.C., Achard, C.: Single-shot 3d multi-person pose estimation in complex images. Pattern Recognition p. 107534 (2020)
4. Buciluǎ, C., Caruana, R., Niculescu-Mizil, A.: Model compression. In: Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining, pp. 535–541 (2006)
5. Bulat, A., Kossaifi, J., Tzimiropoulos, G., Pantic, M.: Toward fast and accurate human pose estimation via soft-gated skip connections. In: FG (2020)
6. Cao, Z., Hidalgo Martinez, G., Simon, T., Wei, S., Sheikh, Y.A.: Openpose: Realtime multi-person 2d pose estimation using part affinity fields. IEEE Transactions on Pattern Analysis and Machine Intelligence (2019)
7. Cao, Z., Simon, T., Wei, S.E., Sheikh, Y.: Realtime multi-person 2d pose estimation using part affinity fields. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 7291–7299 (2017)
8. Chu, X., Yang, W., Ouyang, W., Ma, C., Yuille, A.L., Wang, X.: Multi-context attention for human pose estimation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1831–1840 (2017)
9. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: 2009 IEEE conference on computer vision and pattern recognition, pp. 248–255. Ieee (2009)
10. Developer, N.: NVIDIA TensorRT (2020 (accessed February 23, 2020)). `https://developer.nvidia.com/tensorrt`
11. Elsen, E., Dukhan, M., Gale, T., Simonyan, K.: Fast sparse convnets. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 14629–14638 (2020)
12. Fernando, T., Denman, S., Sridharan, S., Fookes, C.: Tracking by prediction: A deep generative model for mutli-person localisation and tracking. In: 2018 IEEE Winter Conference on Applications of Computer Vision (WACV), pp. 1122–1132. IEEE (2018)
13. Fieraru, M., Khoreva, A., Pishchulin, L., Schiele, B.: Learning to refine human pose estimation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, pp. 205–214 (2018)
14. Fiorini, L., Mancioppi, G., Semeraro, F., Fujita, H., Cavallo, F.: Unsupervised emotional state classification through physiological parameters for social robotics applications. Knowledge-Based Systems **190**, 105217 (2020)
15. Firdaus, N.M., Rakun, E.: Recognizing fingerspelling in sibi (sistem isyarat bahasa indonesia) using openpose and elliptical fourier descriptor. In: Proceedings of the International Conference on Advanced Information Science and System, pp. 1–6 (2019)
16. Gagana, B., Athri, H.U., Natarajan, S.: Activation function optimizations for capsule networks. In: 2018 International Conference on Advances in Computing, Communications and Informatics (ICACCI), pp. 1172–1178. IEEE (2018)
17. Gao, P., Yuan, R., Wang, F., Xiao, L., Fujita, H., Zhang, Y.: Siamese attentional keypoint network for high performance visual tracking. Knowledge-Based Systems p. 105448 (2019)
18. Gao, P., Zhang, Q., Wang, F., Xiao, L., Fujita, H., Zhang, Y.: Learning reinforced attentional representation for end-to-end visual tracking. Information Sciences **517**, 52–67 (2020)
19. Guan, C.z.: Realtime multi-person 2d pose estimation using shufflenet. In: 2019 14th International Conference on Computer Science & Education (ICCSE), pp. 17–21. IEEE (2019)
20. He, K., Zhang, X., Ren, S., Sun, J.: Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In: Proceedings of the IEEE international conference on computer vision, pp. 1026–1034 (2015)
21. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 770–778 (2016)
22. Hu, J., Shen, L., Sun, G.: Squeeze-and-excitation networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 7132–7141 (2018)
23. Huang, G., Liu, Z., Van Der Maaten, L., Weinberger, K.Q.: Densely connected convolutional networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 4700–4708 (2017)
24. Huang, Y., Shum, H.P., Ho, E.S., Aslam, N.: High-speed multi-person pose estimation with deep feature transfer.

Computer Vision and Image Understanding p. 103010 (2020)
25. Ke, L., Chang, M.C., Qi, H., Lyu, S.: Multi-scale structure-aware network for human pose estimation. In: Proceedings of the European Conference on Computer Vision (ECCV), pp. 713–728 (2018)
26. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. In: ICLR (2015)
27. Kundu, A.S., Mazumder, O., Lenka, P.K., Bhaumik, S.: Hand gesture recognition based omnidirectional wheelchair control using imu and emg sensors. Journal of Intelligent & Robotic Systems **91**(3-4), 529–541 (2018)
28. Lifshitz, I., Fetaya, E., Ullman, S.: Human pose estimation using deep consensus voting. In: European Conference on Computer Vision, pp. 246–260. Springer (2016)
29. Liu, L., Wang, S., Hu, B., Qiong, Q., Wen, J., Rosenblum, D.S.: Learning structures of interval-based bayesian networks in probabilistic generative model for human complex activity recognition. Pattern Recognition **81**, 545–561 (2018)
30. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 3431–3440 (2015)
31. Mehta, D., Sotnychenko, O., Mueller, F., Xu, W., Sridhar, S., Pons-Moll, G., Theobalt, C.: Single-shot multi-person 3d pose estimation from monocular rgb. In: 2018 International Conference on 3D Vision (3DV), pp. 120–130. IEEE (2018)
32. Nakai, M., Tsunoda, Y., Hayashi, H., Murakoshi, H.: Prediction of basketball free throw shooting by openpose. In: JSAI International Symposium on Artificial Intelligence, pp. 435–446. Springer (2018)
33. Newell, A., Yang, K., Deng, J.: Stacked hourglass networks for human pose estimation. In: European conference on computer vision, pp. 483–499. Springer (2016)
34. Noori, F.M., Wallace, B., Uddin, M.Z., Torresen, J.: A robust human activity recognition approach using openpose, motion features, and deep recurrent neural network. In: Scandinavian Conference on Image Analysis, pp. 299–310. Springer (2019)
35. Pishchulin, L., Andriluka, M., Gehler, P., Schiele, B.: Poselet conditioned pictorial structures. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 588–595 (2013)
36. Rafi, U., Leibe, B., Gall, J., Kostrikov, I.: An efficient convolutional network for human pose estimation. In: BMVC, vol. 1, p. 2 (2016)
37. Ramachandran, P., Zoph, B., Le, Q.V.: Searching for activation functions. In: 6th International Conference on Learning Representations, ICLR 2018 (2018)
38. Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., Chen, L.C.: Mobilenetv2: Inverted residuals and linear bottlenecks. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 4510–4520 (2018)
39. Sciortino, G., Farinella, G.M., Battiato, S., Leo, M., Distante, C.: On the estimation of children's poses. In: International Conference on Image Analysis and Processing, pp. 410–421. Springer (2017)
40. Sifre, L., Mallat, S.: Rigid-motion scattering for image classification. Ph. D. thesis (2014)
41. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. In: ICLR (2015)
42. Smith, L.N.: Cyclical learning rates for training neural networks. In: 2017 IEEE Winter Conference on Applications of Computer Vision (WACV), pp. 464–472. IEEE (2017)
43. Smith, L.N., Topin, N.: Super-convergence: Very fast training of neural networks using large learning rates. In: Artificial Intelligence and Machine Learning for Multi-Domain Operations Applications, vol. 11006, p. 1100612. International Society for Optics and Photonics (2019)
44. Sun, K., Xiao, B., Liu, D., Wang, J.: Deep high-resolution representation learning for human pose estimation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 5693–5703 (2019)
45. Szegedy, C., Ioffe, S., Vanhoucke, V., Alemi, A.A.: Inception-v4, inception-resnet and the impact of residual connections on learning. In: Thirty-first AAAI conference on artificial intelligence (2017)
46. Tan, M., Chen, B., Pang, R., Vasudevan, V., Sandler, M., Howard, A., Le, Q.V.: Mnasnet: Platform-aware neural architecture search for mobile. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2820–2828 (2019)
47. Tan, M., Le, Q.V.: Efficientnet: Rethinking model scaling for convolutional neural networks. In: ICML (2019)
48. Tan, M., Le, Q.V.: Mixconv: Mixed depthwise convolutional kernels. In: BMVC (2019)
49. Tang, W., Yu, P., Wu, Y.: Deeply learned compositional models for human pose estimation. In: Proceedings of the European Conference on Computer Vision (ECCV), pp. 190–206 (2018)
50. Tang, Z., Peng, X., Geng, S., Zhu, Y., Metaxas, D.N.: Cu-net: coupled u-nets. In: BMVC (2018)
51. TensorFlow: Deploy machine learning models on mobile and IoT devices (2020 (accessed February 23, 2020)). `https://www.tensorflow.org/lite`
52. Tieleman, T., Hinton, G.: Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude. COURSERA: Neural networks for machine learning **4**(2), 26–31 (2012)
53. Tompson, J.J., Jain, A., LeCun, Y., Bregler, C.: Joint training of a convolutional network and a graphical model for human pose estimation. In: Advances in neural information processing systems, pp. 1799–1807 (2014)
54. Toshev, A., Szegedy, C.: Deeppose: Human pose estimation via deep neural networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 1653–1660 (2014)
55. Tung, F., Mori, G.: Clip-q: Deep network compression learning by in-parallel pruning-quantization. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 7873–7882 (2018)
56. Vitali, A., Regazzoni, D., Rizzi, C., Maffioletti, F.: A new approach for medical assessment of patient's injured shoulder. In: International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, vol. 59179, p. V001T02A049. American Society of Mechanical Engineers (2019)
57. Yang, W., Li, S., Ouyang, W., Li, H., Wang, X.: Learning feature pyramids for human pose estimation. In: proceedings of the IEEE international conference on computer vision, pp. 1281–1290 (2017)
58. Yuan, H., Li, M., Hou, J., Xiao, J.: Single image-based head pose estimation with spherical parametrization and 3d morphing. Pattern Recognition p. 107316 (2020)
59. Zhang, F., Zhu, X., Dai, H., Ye, M., Zhu, C.: Distribution-aware coordinate representation for human pose estimation. In: Proceedings of the IEEE/CVF Conference on

Computer Vision and Pattern Recognition, pp. 7093–7102 (2020)
60. Zhang, F., Zhu, X., Ye, M.: Fast human pose estimation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3517–3526 (2019)
61. Zhang, H., Ouyang, H., Liu, S., Qi, X., Shen, X., Yang, R., Jia, J.: Human pose estimation with spatial contextual information. arXiv preprint arXiv:1901.01760 (2019)
62. Zoph, B., Vasudevan, V., Shlens, J., Le, Q.V.: Learning transferable architectures for scalable image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 8697–8710 (2018)

Appendices

A Ablation study

To determine the effect of different design choices in the EfficientPose architecture, we carried out component analysis.

Training efficiency

We assessed the number of training epochs to determine the appropriate duration of training, avoiding demanding optimization processes. Figure 5 suggests that the largest improvement in model accuracy occurs until around 200 epochs, after which training saturates. Table 5 supports this observation with less than 0.4% increase in PCK_h@50 with 400 epochs of training. From this, it was decided to perform the final optimization of the different variants of EfficientPose over 200 epochs. Table 5 also suggests that most of the learning progress occurs during the first 100 epochs. Hence, for the remainder of the ablation study 100 epochs were used to determine the effect of different design choices.

Cross-resolution features

The value of combining low-level local information with high-level semantic information through a cross-resolution feature extractor was evaluated by optimizing the model with and without the low-level backbone. Experiments were conducted on two different variants of the EfficientPose model. On coarse prediction (PCK_h@50) there is little to no gain in accuracy (Table 6), whereas for fine estimation (PCK_h@10) some improvement ($0.6 - 0.7\%$) is displayed taking into account the negligible cost of $1.02 - 1.06\times$ more parameters and $1.03 - 1.06\times$ increase in FLOPs.

Skeleton estimation

The effect of skeleton estimation through the approximation of part affinity fields was assessed by comparing the architecture with and without the single pass of skeleton estimation. Skeleton estimation yields improved accuracy with $1.3-2.4\%$ gain in PCK_h@50 and $0.2 - 1.4\%$ in PCK_h@10 (Table 7), while only introducing an overhead in number of parameters and computational cost of $1.3 - 1.4\times$ and $1.2 - 1.3\times$, respectively.

Number of detection passes

We also determined the appropriate comprehensiveness of detection, represented by number of detection passes. EfficientPose I and II were both optimized on three different variants (Table 8). Seemingly, the models benefit from intermediate supervision with a general trend of increased performance level in accordance with number of detection passes. The major benefit in performance is obtained by expanding from one to two passes of keypoint estimation, reflected by $1.6 - 1.7\%$ increase in PCK_h@50 and $1.8 - 1.9\%$ in PCK_h@10. In comparison, a third detection pass yields only $0.5 - 0.8\%$ relative improvement in PCK_h@50 compared to two passes, and no gain in PCK_h@10 while increasing number of parameters and computation by $1.3\times$ and $1.2\times$, respectively. From these findings, we decided a beneficial trade-off in accuracy and efficiency would be the use of two detection passes.

Upscaling

To assess the impact of upscaling, implemented as bilinear transposed convolutions, we compared the results of the two respective models. Table 9 reflects that upscaling yields improved precision on keypoint estimates by large gains of $9.2 - 12.3\%$ in PCK_h@10 and smaller improvements of $0.5 - 1.1\%$ on coarse detection (PCK_h@50). As a consequence of increased output resolution upscaling slightly increases number of FLOPs ($1.04 - 1.1\times$) with neglectable increase in number of parameters.

B Optimization procedure

Most state-of-the-art approaches for single-person pose estimation are extensively pretrained on ImageNet [44, 61], enabling rapid convergence for models when adapted to other tasks, such as HPE. In contrast to these approaches, few models, including OpenPose [6] and EfficientPose, only utilize the most basic pretrained features. This facilitates construction of more efficient network architectures but at the same time requires careful design of optimization procedures for convergence towards reasonable parameter values.

Training of pose estimation models is complicated due to the intricate nature of output responses. Overall, optimization is performed in a conventional fashion by minimizing the MSE of the predicted output maps Y with respect to ground truth values $\hat{Y}$ across all output responses N.

The predicted output maps should ideally have higher values at the spatial locations corresponding to body part positions, while punishing predictions farther away from the correct location. As a result, the ground truth output maps must be carefully designed to enable proper convergence during training. We achieve this by progressively reducing the circumference from the true location that should be rewarded, defined by the σ parameter. Higher probabilities $T \in [0, 1]$ are assigned for positions P closer to the ground truth position G (Equation 4).

$$T_i = \exp\left(-\frac{\|P_i - G\|_2^2}{\sigma^2}\right) \tag{4}$$

The proposed optimization scheme (Figure 6) incorporates a stepwise σ scheme, and utilizes SGD with momentum of 0.9 and a decaying triangular cyclical learning rate (CLR)

Table 6 Model accuracy on the MPII validation dataset in relation to the use of cross-resolution features

Model	Cross-resolution features	Parameters	FLOPs	PCK_h@50	PCK_h@10
EfficientPose I	✓	$0.72M$	$1.67G$	83.56	26.35
EfficientPose I		$0.68M$	$1.58G$	83.64	25.79
EfficientPose II	✓	$1.73M$	$7.70G$	87.05	29.87
EfficientPose II		$1.69M$	$7.50G$	86.93	29.16

Table 7 Model accuracy on the MPII validation dataset in relation to the use of skeleton estimation

Model	Skeleton estimation	Parameters	FLOPs	PCK_h@50	PCK_h@10
EfficientPose I	✓	$0.72M$	$1.67G$	83.56	26.35
EfficientPose I		$0.54M$	$1.37G$	81.13	25.00
EfficientPose II	✓	$1.73M$	$7.70G$	87.05	29.87
EfficientPose II		$1.27M$	$6.03G$	85.75	29.67

Table 8 Model accuracy on the MPII validation dataset in relation to the number of detection passes

Model	Detection passes	Parameters	FLOPs	PCK_h@50	PCK_h@10
EfficientPose I	1	$0.52M$	$1.33G$	81.85	24.51
EfficientPose I	2	$0.72M$	$1.67G$	83.56	26.35
EfficientPose I	3	$0.92M$	$2.02G$	84.35	26.42
EfficientPose II	1	$1.24M$	$5.92G$	85.42	28.01
EfficientPose II	2	$1.73M$	$7.70G$	87.05	29.87
EfficientPose II	3	$2.22M$	$9.49G$	87.55	29.61

policy [42]. The σ parameter is normalized according to the output resolution. As suggested by Smith and Topin [43], the large learning rates in CLR provides regularization in network optimization. This makes training more stable and may even increase training efficiency. This is valuable for network architectures, such as OpenPose and EfficientPose, less heavily concerned with pretraining (i.e., having larger portions of randomized weights). In our adoption of CLR, we utilize a cycle length of 3 epochs. The learning rate (λ) converges towards λ_∞ (Equation 5), where λ_{max} is the highest learning rate for which the model does not diverge during the first cycle and $\lambda_{min} = \frac{\lambda_{max}}{3000}$, whereas σ_0 and σ_∞ are the initial and final sigma values, respectively.

$$\lambda_\infty = 10^{\frac{\log(\lambda_{max})+\log(\lambda_{min})}{2}} \cdot 2^{\sigma_0-\sigma_\infty} \tag{5}$$

Table 9 Model accuracy on the MPII validation dataset in relation to the use of upscaling

Model	Upscaling	Parameters	FLOPs	$PCK_h@50$	$PCK_h@10$
EfficientPose I	✓	$0.72M$	$1.67G$	83.56	26.35
EfficientPose I		$0.71M$	$1.52G$	82.42	14.02
EfficientPose II	✓	$1.73M$	$7.70G$	87.05	29.87
EfficientPose II		$1.73M$	$7.37G$	86.56	20.66

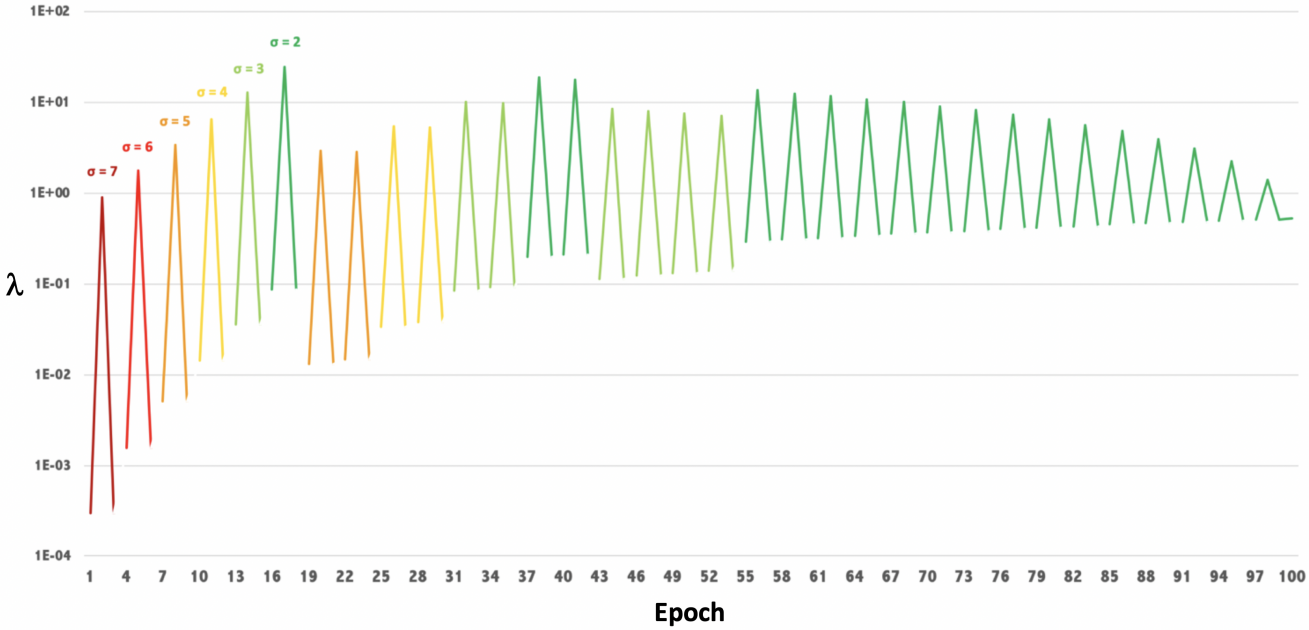

Fig. 6 Optimization scheme displaying learning rates λ and σ values corresponding to the training of EfficientPose II over 100 epochs