Error Compensation Heatmap Decoding for Human Pose Estimation

YANG FEIYU1,2, SONG ZHAN1, XIAO ZHENZHONG2, MO YAOYANG2, CHEN YU2, PAN ZHE2, ZHANG MIN2, ZHANG YAO2, QIAN BEIBEI2, AND JIN WU3

1 Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Shenzhen 518055, China
2 Orbbec Inc., Shenzhen 518062, China
3 College of Energy and Power Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing 210016, China

Corresponding author: Yang Feiyu (feiyuyang@vip.sina.com)

This work was supported in part by the National Natural Science Foundation of China under Grant 51906106, in part by the Key-Area Research and Development Program of Guangdong Province under Grant 2019B010149002, and in part by the Dongguan City Core Technology Research Frontier Project under Grant 2019622101001.

ABSTRACT As a fundamental component of heatmap-based human pose estimation methods, heatmap decoding is to transform heatmaps into joint coordinates. We found that previous heatmap decoding methods generally ignored the effect of systematic errors introduced by the resolution increasing operations in the network decoder. This work fills the gap by taking the systematic errors in heatmap decoding into consideration. We proposed a fast method to reduce the systematic and random errors in one shot by error compensation. The proposed method outperforms the previous best method on the COCO and the MPII datasets while being over 2 times faster. Extensive experiments with different networks, resolutions, metrics and datasets have proved the rationality of the proposed idea.

INDEX TERMS Human pose estimation, heatmap, decoding, error compensation.

I. INTRODUCTION

In recent years, the rapid advance in deep learning [1] has tremendously boosted human pose estimation. Among different learning-based approaches, the heatmap-based methods perform the best [2]–[6]. Heatmap decoding is to estimate joint coordinates from the predicted heatmaps. Different from other computer vision tasks, like image classification [7]–[9], object detection [10]–[12] and semantic segmentation [13], [14], human pose estimation is more sensitive to heatmap decoding because it employs metrics that conduct point to point comparison of human joints. Besides, human pose estimation generally require real-time performance (30 FPS) on embedded devices. Thus, developing accurate and fast heatmap decoding methods is of significant importance.

However most research is network-design related while only limited studies are focused on heatmap decoding, which is found also significant recently. The earliest heatmap decoding method is the standard method. It simply takes coordinate of the maximal as the joint. It is straightforward but susceptible to errors. To be more error resistant, [15] expressed the joint location as an empirical function of the coordinates of the maximal and the second maximal. As a result, it achieves higher accuracy. [16] took a step further. It calculates the first and second order derivatives of the entire heatmap to solve the joint location. Solving derivatives is slow but it leads to better performance. Combined with the flip operation, it achieves the state-of-the-art performance on the COCO dataset [23]. However, none of these methods takes the systematic errors into consideration, which limited their performance to a certain extent.

The heatmap-based network generally contains two parts: the encoder and the decoder. The encoder part extracts semantic information by reducing the feature resolution. The decoder part restores the features to higher resolution to generate heatmaps. Common operations to increase resolution are upsampling, unpooling and deconvolution. These operations can lead to biased joint locations thus introducing systematic errors. However, those systematic errors are ignored by previous methods. In view of that, we propose a method to decode the heatmap with error compensation. This method compensates not only the systematic but the random errors in one shot. We define an optimal error compensation factor...
$\Delta_{opt}$ to describe the error intensity and theoretically derive how $\Delta_{opt}$ can be used to compensate errors. Our contributions include:

1) We revealed that heatmap-based methods suffer from significant systematic errors, which nevertheless was ignored by existing methods.
2) We proposed a method to compensate the systematic and random errors in one shot. Taking systematic errors into consideration, it outperforms previous methods.
3) The proposed method is lightweight. It proceeds at over 2 times the speed of the previous best method [16].

II. RELATED WORK

In this section, we first briefly introduce the learning-based pose estimation methods and then review previous heatmap decoding approaches.

A. POSE ESTIMATION

Benefiting from the dramatic advance in neural network technique [1], pose estimation has entered a new era of rapid development. Human pose estimation is commonly split into single and multi-person tasks. Without learning joint connection knowledge, single person pose estimation detects only human joints and, as a consequence, achieves relatively high performance [2], [3], [17], [18]. Multi-person pose estimation further falls into two categories: the top-down methods [4] and the bottom-up methods [19]–[21].

The top-down methods are essentially integrating person detection with single person pose estimation. Bounding boxes of person instances are first detected by a person detector, such as YOLO [11] and Mask RCNN [10], and then those persons are cropped and fed to a single person pose estimator like CPN [4] and HRNet [2]. Such two-stage processes, to a certain extent, suffer from inference speed, while brings relatively better performance in return. Most state-of-the-art performance on multi-person pose estimation challenges is achieved with the top-down frameworks [3].

The bottom-up methods detect multi-person in one shot by learning not only the joints coordinates but also the limb connections [19]–[21]. Representative connection learning designs include learning the part affinity fields [19], grouping human joints by associate embedding [20] and learning connections by probabilities [21]. Generally, the bottom-up methods is less accurate but faster than the top-down methods, making it more practical for application.

B. HEATMAP DECODING

Unlike the extensive studies on network, only limited investigations on heatmap decoding are available. To our best knowledge, there are 3 heatmap decoding methods: the standard method, the shifting method [15] and the DARK method [16]. The standard, as well as the most widely used, method simply extracts the maximal coordinates after smoothing the heatmap with a Gaussian filter [22]. Because it relays on only one point (the maximal) on the heatmap, the standard method is extremely sensitive to errors. To alleviate this issue, Newell et al. [15] proposed the shifting method that locates joint coordinates by shifting the maximal towards the second maximal by 1/4 of the distance between them. Involving two instead of one maximal of the heatmap, the shifting method outperforms the standard method significantly. To take a step further, Zhang et al. [16] proposed the DARK method, which took first and second order derivatives on the entire heatmap to solve the mean value of Gaussian distribution, which is also the joint location. By further exploiting the heatmap, the DARK method presents better performance. However, none of the 3 methods take the systematic errors into consideration, making their ideal upper bounds limited.

III. METHODOLOGY

In this section, we first briefly review the heatmap encoding process (i.e. transforming coordinates into heatmaps). Then we theoretically derive the proposed error compensation factor of our heatmap decoding method. Last we describe how the error compensation factor is determined with network and dataset.

A. HEATMAP ENCODING

For heatmap-based human pose estimation, human coordinates are encoded into heatmaps proportionally by.

$$ p' = \lambda p $$

where $p$ and $p'$ denote the point coordinates in the image and the heatmap respectively; $\lambda$ denotes the output stride of neural networks. The most straightforward encoding is to set the very joint pixel as one with others as zero (one-hot encoding). However, such discrete distribution makes heatmap prediction error-prone. By contrast, learning continuous probability distributions is more robust for neural networks. In light of that, encoding human joint coordinates with a 2D-Gaussian distribution has become a de facto standard. Thus, we only focus on decoding heatmaps encoded with 2D-Gaussian distribution in this work. The 2D-Gaussian distribution can be expressed as

$$ G(p; \mu, \Sigma) = \frac{1}{2\pi|\Sigma|^{0.5}} e^{-\frac{1}{2}(p-\mu)^T \Sigma^{-1}(p-\mu)} $$

where $\mu$ is the mean value and $\Sigma$ is the covariance matrix. The encoded heatmap subject to 2D-Gaussian distribution and the heatmap decoding process is to estimate mean value $\mu$ from a predicted heatmap.

B. ERROR COMPENSATION DECODING

This section introduce our error compensation strategy and formulate the proposed error compensation factor. Figure 1 decomposes the predicted heatmap from neural networks into 2 parts: the ideal Gaussian distribution and errors. The predict heatmap can be represented as

$$ f(x) = g(x) + h(x) $$

where $g(x)$ is the ideal Gaussian distribution and $h(x)$ is the error term. Our error compensation factor $\lambda$ is determined via network and the dataset to optimize the mean value of the predicted heatmap $p'$.

The probability density function of the 2D-Gaussian distribution can be represented as

$$ G(p; \mu, \Sigma) = \frac{1}{2\pi|\Sigma|^{0.5}} e^{-\frac{1}{2}(p-\mu)^T \Sigma^{-1}(p-\mu)} $$

and the integral of the 2D-Gaussian distribution can be represented as

$$ \int G(p; \mu, \Sigma) dp = 1 $$

The error compensation factor $\lambda$ can be used to compensate errors. Our contributions include:

1) We revealed that heatmap-based methods suffer from significant systematic errors, which nevertheless was ignored by existing methods.
2) We proposed a method to compensate the systematic and random errors in one shot. Taking systematic errors into consideration, it outperforms previous methods.
3) The proposed method is lightweight. It proceeds at over 2 times the speed of the previous best method [16].
where \( f(x) \) denotes the predicted heatmap which is subject to noises, \( g(x) \) denotes the ideal Gaussian distribution with a mean value of \( \mu \) and \( h(x) \) denote errors (or noises) introduced by the method. The objective of heatmap decoding is to estimate the mean value \( \mu \) of \( g(x) \) given \( f(x) \). For simplicity, we illustrate distributions in 1D and the conclusions is applicable to 2D scenario. For Gaussian distribution \( g(x) \), its mean value \( \mu \) can be determined by

\[
\mu = \frac{x_1 + x_2}{2} = \frac{\int_{x_1}^{x_2} x g(x) dx}{\int_{x_1}^{x_2} g(x) dx}
\]

Similarly, if we define a small quantity \( \Delta \), the mean value \( \nu \) of noisy Gaussian distribution \( f(x) \) over range \([x_1, x_2 - \Delta]\) can be expressed as a function of \( \Delta \)

\[
\nu(\Delta) = \frac{\int_{x_1}^{x_2-\Delta} x f(x) dx}{\int_{x_1}^{x_2-\Delta} f(x) dx}
= \frac{\int_{x_1}^{x_2-\Delta} x g(x) dx + \int_{x_1}^{x_2-\Delta} h(x) dx}{\int_{x_1}^{x_2-\Delta} g(x) dx + \int_{x_1}^{x_2-\Delta} h(x) dx}
\]

For a pretrained network, it is well converged on the training dataset, suggesting that its signal noise ratio should be small. As a result, the integral of noise \( h(x) \) over range \([x_1, x_2 - \Delta]\) should be negligible compared with that of the signal \( g(x) \)

\[
\int_{x_1}^{x_2-\Delta} g(x) dx \gg \int_{x_1}^{x_2-\Delta} h(x) dx
\]

By substituting (6) into (5), the mean value \( \nu(\Delta) \) can be expressed as

\[
\nu(\Delta) \approx \frac{\int_{x_1}^{x_2-\Delta} x g(x) dx + \int_{x_1}^{x_2-\Delta} x h(x) dx}{\int_{x_1}^{x_2-\Delta} g(x) dx + \int_{x_1}^{x_2-\Delta} h(x) dx}
\]

Since \( \Delta \) is small and the mean value of Gaussian distribution \( g(x) \) is at \( \frac{x_1 + x_2}{2} \), the integral of \( g(x) \) over range \([x_1, x_2]\) should be much larger than that over range \([x_2 - \Delta, x_2]\). That is

\[
\int_{x_1}^{x_2} g(x) dx \gg \int_{x_2-\Delta}^{x_2} g(x) dx
\]

Therefore, by substituting (8) into (7), the mean value \( \nu(\Delta) \) can be further simplified as

\[
\nu(\Delta) \approx \frac{\int_{x_1}^{x_2-\Delta} x g(x) dx + \int_{x_1}^{x_2-\Delta} x h(x) dx}{\int_{x_1}^{x_2-\Delta} g(x) dx + \int_{x_1}^{x_2-\Delta} h(x) dx}
\]

Here we define variable \( \delta(\Delta) \) as

\[
\delta(\Delta) = \frac{\int_{x_1}^{x_2-\Delta} x g(x) dx + \int_{x_1}^{x_2-\Delta} x h(x) dx}{\int_{x_1}^{x_2-\Delta} g(x) dx + \int_{x_1}^{x_2-\Delta} h(x) dx}
\]

Equation (10) builds the bridge between the to be determined variable \( \mu \) and the mean value function \( \nu(\Delta) \) of the given noisy Gaussian distribution \( f(x) \). Assuming that if there exists a value \( \Delta_{\text{opt}} \), which satisfies

\[
\delta(\Delta_{\text{opt}}) = \frac{\int_{x_1}^{x_2-\Delta_{\text{opt}}} x g(x) dx + \int_{x_1}^{x_2-\Delta_{\text{opt}}} x h(x) dx}{\int_{x_1}^{x_2-\Delta_{\text{opt}}} g(x) dx + \int_{x_1}^{x_2-\Delta_{\text{opt}}} h(x) dx} = 0
\]

Then the mean value \( \mu \) can be estimated with

\[
\mu \approx \nu(\Delta_{\text{opt}})
\]

When \( \Delta_{\text{opt}} \) is obtained, according to the definition of \( \nu(\Delta) \) in (5), the mean value \( \mu \) can be calculated as

\[
\mu \approx \frac{\int_{x_1}^{x_2-\Delta_{\text{opt}}} x f(x) dx}{\int_{x_1}^{x_2-\Delta_{\text{opt}}} f(x) dx}
\]

We call variable \( \Delta \) as the Error Compensation Factor and \( \Delta_{\text{opt}} \) as the Optimal Error Compensation Factor. As analyzed above, if the optimal error compensation factor \( \Delta_{\text{opt}} \) is obtained, the mean value \( \mu \) of \( g(x) \) can be estimated with (13), suggesting that \( \mu \) can be approximated as the mean value of \( f(x) \) over range \([x_1, x_2 - \Delta_{\text{opt}}]\). Intuitively, the error compensation factor \( \Delta \) indicates how significant the error is. If there is no error at all, then \( \Delta \) should equals 0. On the other hand, if the error is large, then the absolute value of \( \Delta \) should be relatively larger, which means the integral region should be shortened or expanded more to compensate the error introduced by the method. The sign of \( \Delta \) indicates the biased direction of the error. Positive \( \Delta \) mean the error is biased to the right, thus the upper bound of the integral region should be reduced to compensate the bias. When \( \Delta = \Delta_{\text{opt}} \), the effect of error is minimized by compensation. The determination of \( \Delta_{\text{opt}} \) is addressed in the following subsection.
C. ESTIMATION OF THE OPTIMAL ERROR COMPENSATION FACTOR

As analyzed above, given the optimal error compensation factor $\Delta_{opt}$, the mean value $\mu$ of Gaussian distribution $g(x)$ can be determined with equation (13). However, since the error function $h(x)$ can be arbitrary functions, there may not exist a value $\Delta_{opt}$ that satisfies equation (11). As a result, we can only approximate $\Delta_{opt}$ by finding the $\Delta$ that minimizes the absolute value of $\delta(\Delta)$, which can be expressed as

$$\Delta_{opt} \approx \arg \min_{\Delta} |\delta(\Delta)| \quad (14)$$

Unfortunately, the optimal error compensation factor $\Delta_{opt}$ still cannot be estimated because the error function $h(x)$ is unknown which means function $\delta(\Delta)$ is unknown as well. To tackle this problem, we try to estimated $\Delta_{opt}$ from another perspective that $\Delta_{opt}$ is the error compensation factor which maximizes the accuracy of the network on its training dataset. That is given a network $\mathcal{N}$ and its training dataset $\mathcal{D}$, the accuracy $A$ of $\mathcal{N}$ over $\mathcal{D}$ can be expressed as a function of the mean value $\nu$

$$A = A_\nu(\nu; \mathcal{D}, \mathcal{N}) \quad (15)$$

According to equation (5), $\nu$ is a function of the error compensation factor $\Delta$, thus $A$ is also a function of $\Delta$

$$A = A_\nu(\Delta; \mathcal{D}, \mathcal{N}) = A_\Delta(\Delta; \mathcal{D}, \mathcal{N}) \quad (16)$$

The accuracy $A$ achieves its maximum value at $\Delta = \Delta_{opt}$, which means $\Delta_{opt}$ can be estimated by

$$\Delta_{opt} = \arg \max_{\Delta} \{A_\Delta(\Delta; \mathcal{D}, \mathcal{N})\} \quad (17)$$

Equation (17) can be applied to calculate the optimal error compensation factor $\Delta_{opt}$ in practice. It is noteworthy that heatmap has two dimensions $x$ and $y$, they undergo the same process of heatmap encoding, network inference and heatmap decoding. As a result, according to the symmetry property, $x$ and $y$ subject to the same error distribution, suggesting that the same optimal error compensation factor is applicable for both axis.

$$\Delta_{opt} = \Delta_{opt}^x = \Delta_{opt}^y \quad (18)$$

Note that we did not distinguish systematic error and random error in the derivation. The error function $h(x)$ contains both type of errors. Previous methods use Gaussian-smoothing to preprocess the heatmap before decoding to minimize the effect of random errors. By contrast, the propose heatmap decoding approach is Gaussian-smoothing free, which makes it fast and robust. The effect of Gaussian-smoothing and decoding speed will be discussed in other sections.

IV. EXPERIMENTAL SETTINGS

In this section we conduct experiments to compare the proposed method with previous ones. As discussed above, the optimal error compensation factor $\Delta_{opt}$ is estimated with equation (17). The specific procedure is described in Fig.2. (1) First, find point $A$, the coordinate of the maximal, as the location of the mean value of Gaussian distribution. (2) Second, set the integral region as $3\sigma + 3$, where $\sigma$ is the standard deviation of the Gaussian distribution. Since the $[-3\sigma, 3\sigma]$ region already covers 99.7% of the Gaussian distribution, we expand a pixel further to make sure the entire distribution is considered. Note that the integral region can be set as any size as long as it covers the entire distribution. We use $6\sigma + 3$ to minimize unnecessary computations. (3) Third, the upper bond of integral region is shifted by $\Delta$ (only $x$ direction is shown in Fig.2) to calculate the accuracy the mean value $\nu$ with equation (5) and the accuracy $A$ with equation (16). Here $A$ can be different metrics for different datasets. (4) Last, the $\Delta$ leads to the maximum $A$ is the optimal error compensation factor $\Delta_{opt}$.

Two widely used human pose estimation datasets: COCO-2017 and MPII are employed for testing. The COCO keypoint dataset [23] contains 200,000 images of more than 250,000 person samples with various body scales, background environments and occlusion patterns. Each person instance is labeled with 17 joints. The MPII human pose dataset [24] contains 20,000 images with more than 40,000 person samples, each labeled with 16 keypoints. For accuracy $A$, the Average Precision (AP) metric and the Percentage of Correct Points w.r.t. head (PCKh) metric are respectively used for the COCO and MPII datasets to evaluate model performance.

We employ 14 models for evaluation, the High-Resolution network group (HR-W32 and HR-W48) [2] and the Simple-Baseline network group (ResNet-50, ResNet-101 and ResNet-152) [17] with 3 input resolutions (256 × 192, 256 × 256 and 384 × 288). We use the pertained weights of those models and decode their heatmaps with different methods. We compare our method with the standard, the shifting [15] and the DARK [16] methods. Note that the original DARK [16] method use flip operation to improve performance. Specifically, the prediction of the original and the
flipped (usually right-to-left flip) images are averaged as the final prediction. However adding flip operation requires 2 times of network inference and heatmap decoding making it inapplicable for practical application, especially when speed matters. Therefore, we do not employ flip operation for all the methods to mimic practical application scenario.

V. RESULTS AND DISCUSSIONS
In this section, we first present the performance of different methods on the COCO [23] and MPII [24] datasets. Then the systematic and random errors are discussed respectively. Last we conduct the speed comparison of different approaches.

A. RESULTS ON COCO
The COCO dataset [23] is one of the most widely used datasets for human pose estimation. We compared the performance of different network and decoding method combinations. The results are listed in Table. 1. As can be noted, the proposed approach outperforms other methods for most metrics. The AP (average precision) metric is considered as the most representative and comprehensive metric among the metrics listed, thus we plot the AP values of different methods in Fig. 3 to present a more intuitively comparison. As can be seen, our method outperforms the DARK [16] method by considerable accuracy gains. Take the input resolution of 256 × 192 an example, the Simple-Baseline ResNet-50, ResNet-101 and ResNet-152 networks gain 2.24, 2.68, and 2.59 AP, respectively. Moreover, the accuracies of the HR-W32 and HR-W48 networks are increased by 2.73 and 2.86 AP.

The performance of the four methods can be sorted in the order of ours > DARK [16] > shifting [15] > standard. As mentioned above, the first 3 methods are built on the basis of the standard method, so they unsurprisingly perform better. The shifting method [15] involves only two maximum points of the heatmap to estimate joint location. The second largest point assists to reduce the impact of random error on the maximum. The DARK method [16] takes a step further to involve the entire heatmap to determine the joint location. Therefore, it suffers from less random errors than the shifting method [15] resulting in better performance. However the DARK method [16] is still less accurate than the proposed method because it still suffers from systematic errors. Fig. 4 presents qualitative comparison between the DARK [16] and the proposed methods. As can be seen, error compensation leads to more reasonable joint predictions.

B. RESULTS ON MPII
The MPII dataset [24] is a specialized dataset for Human pose estimation. It uses the PCKh (Percentage of Correct Points w.r.t. head) metric to evaluate model performance. PCKh means when the distance between the predicted and the ground truth joint is less than x% of that between the head and the neck joint, the prediction is accounted as correct. Table. 2 lists the overall and joint-specific PCKh values of different networks evaluated on the MPII dataset. As can be noted, the DARK method [16] predicts better head joints, while the proposed method predicts better shoulder, elbow, wrist, hip and knee joints. Fig. 5 compares the PCKh0.1 and PCKh0.5 metrics of different methods. The method performance can still be sorted as ours > DARK [16] > shifting [15] > standard. Take the PCKh0.1 metric as an example, the proposed method contributes 8.4% and 7.8% accuracy gains for the HR-W32-256 × 256 and ResNet-152-256 × 256 networks, respectively. Note that the evaluations conducted with different networks, resolutions, metrics and datasets validate the generality of the proposed approach.

C. SYSTEMATIC ERRORS
An algorithm basically suffers from two types of errors: the random error and the systematic error. The random error is caused by stochastic process and fluctuates around zero. Therefore it can be reduced by averaging multiple predictions. As an example, apply Gaussian-smoothing on heatmaps is an effective way to reduce random error. By contrast, systematic is more troublesome because it introduces a certain nonzero bias. It cannot be reduced without the knowledge of ground truth. The process of estimating systematic error with ground truth is also known as calibration. The proposed heatmap decoding method provides a perspective to look into the systematic error of heatmap decoding.
Table 1 lists the $\Delta_{opt}$ values of different networks calculated following the steps described above. As can be noted, the $\Delta_{opt}$ values are positive ($\Delta_{opt} = 4.5$) for all the tested networks, which means the bottom-right corner of the integral region is cut to compensate errors, suggesting that those networks suffer from systematic errors which universally bias to the bottom-right corner of the heatmap. These systematic errors can result from the operations that increase resolution.

A heatmap-based pose estimation network generally consists of two parts, an encoder for semantic feature extraction and...
Y. Feiyu et al.: Error Compensation Heatmap Decoding for Human Pose Estimation

**FIGURE 5.** Comparison of the $PCK_h^{0.1}$ (left) and $PCK_h^{0.5}$ (right) metric between the proposed method with the standard, shifting [15] and DARK [16] methods on the MPII [24] validation dataset. Dash lines denote the ResNet group. Flip operation is not employed to mimic practical application scenario.

**TABLE 3.** The optimal error compensation factor $\Delta_{opt}$ of different models. As can be noted, all $\Delta_{opt}$ values are positive, suggesting that systematic errors are bias to the bottom-right corner.

| Model       | Input         | $\Delta_{opt}$ |
|-------------|---------------|----------------|
| ResNet-50   | 256x192       | 4              |
| ResNet-50   | 256x256       | 4              |
| ResNet-101  | 384x288       | 5              |
| ResNet-101  | 256x192       | 4              |
| ResNet-101  | 256x256       | 4              |
| ResNet-152  | 384x288       | 5              |
| ResNet-152  | 256x192       | 4              |
| ResNet-152  | 256x256       | 4              |
| IIR-W32     | 384x288       | 5              |
| HR-W32      | 256x192       | 4              |
| HR-W32      | 256x256       | 4              |
| HR-W32      | 384x288       | 5              |
| IIR-W48     | 256x192       | 4              |
| HR-W48      | 384x288       | 5              |

a decoder for resolution restoration. Operations mapping lower resolution feature maps to higher ones like upsampling, unpooling, deconvolution are commonly involved in a decoder. Those operation tends to introduce biases. Take upsampling a $2 \times 2$ feature map to a $4 \times 4$ one as an example, the original $2 \times 2$ pixels are first anchored to corresponding pixels of the $4 \times 4$ feature map, then extra pixels are calculated by interpolation (e.g. bilinear interpolation). The anchor pixels can be chosen as top-left, top-right, bottom-left or bottom-right. Note that none of them are located in the center of the region, suggesting that this process substantially introduce biases, also known as systematic errors.

Generally, it is unlikely to reduced those systematic error by the heatmap itself. That is why the standard, shifting [15] and DARK [16] methods all severely suffer from systematic errors. Alternatively, working with the flip operation, their performance can be improved significantly. Because the systematic error of the original prediction biases to one direction and the flipped one biases to the opposite direction, by adding them up, the systematic error is readily reduced. However flip operation doubles the computation making it inapplicable in practice. As mentioned above, systematic error can be reduced by the using the ground truth to estimate its pattern and reduce it in the post-process. The proposed method follows this strategy by using the model performance on the training dataset as priors to reduce systematic errors by error compensation.

**D. RANDOM ERRORS**

Since random error fluctuates around zero, it can be reduced by averaging. Gaussian-smoothing is the most widely applied method. Both the shifting [15] and the DARK [16] methods employ Gaussian-smoothing as a preprocess. Differently, since the proposed method compensates all errors in one shot, it reduces not only the systematic errors but also the random errors simultaneously. Table. 4 compares the accuracy evaluated on the COCO dataset [23] with and without Gaussian-smoothing for the DARK [16] and the proposed method. As can been noted, the DARK [16] method deteriorates without Gaussian-smoothing while the proposed method is insensitive to Gaussian-smoothing. Take the ResNet-50-256 $\times$ 192 network as an example, without Gaussian-smoothing, the accuracy decreased by 0.52 $AP$ when using the DARK [16] method, while decreased by only 0.06 $AP$ when using the propose method. For the ResNet-50-384 $\times$ 288 network, the absence of Gaussian-smoothing even results in 0.08 $AP$ increase in accuracy using the proposed method. In general, the proposed method shows comparable performance with and without Gaussian-smoothing, suggesting that this method is smoothing-free. The DARK method [16] requires a smooth heatmap to get more accurate first and second order derivatives. While the proposed method is more noise resistant because it employs integrals instead of derivatives.

**E. SPEED COMPARISON**

Human pose estimation generally requires real-time performance on embedded devices, thus it is of significant importance to develop lightweight algorithms. The shifting [15], the DARK [16] and the proposed methods are all
In this work, we revealed that heatmap-based human pose estimation networks suffer from systematic errors caused by the resolution increasing operations, which were generally ignored by previous methods. A novel error compensation heatmap decoding method is proposed to reduce systematic and random errors simultaneously. The proposed method outperforms previous state-of-the-art method while being over 2 times faster. Extensive experiments with different networks, resolutions, metrics and datasets have validated the reasonability and generality of proposed method. In addition, in-depth analyses on both systematic and random errors are conducted. Hopefully this work brings potential inspirations to future research on human pose estimation.

REFERENCES

[1] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, “Learning internal representations by error propagation,” Jolla Inst. Cogn. Sci., California Univ., San Diego, CA, USA, 1985.

[2] K. Sun, B. Xiao, D. Liu, and J. Wang, “Deep high-resolution representation learning for human pose estimation,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2019, pp. 5893–5903.

[3] W. Li, Z. Wang, B. Yin, Q. Peng, Y. Du, T. Xiao, G. Yu, H. Lu, Y. Wei, and J. Sun, “Rethinking on multi-stage networks for human pose estimation,” 2019, arXiv:1901.00148. [Online]. Available: http://arxiv.org/abs/1901.00148

[4] Y. Chen, Z. Wang, Y. Peng, Z. Zhang, G. Yu, and J. Sun, “Cascaded pyramid network for multi-person pose estimation,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., Jun. 2018, pp. 7103–7112.

[5] I. Lifshitz, E. Fetaya, and A. Rabinovich, “Flex for pose estimation using deep consensus voting,” in Proc. Eur. Conf. Comput. Vis., Oct. 2016, pp. 246–260.

[6] X. Zhou, D. Wang, and P. Krahenbuhl, “Objects as points,” 2019, arXiv:1904.07850. [Online]. Available: https://arxiv.org/abs/1904.07850

[7] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, “ImageNet: A large-scale hierarchical image database,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Jun. 2009, pp. 211–212.

[8] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2016, pp. 770–777.

[9] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, “Going deeper with convolutions,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Jun. 2015, pp. 1–9.

[10] K. He, G. Gkioxari, P. Dollar, and R. Girshick, “Mask R-CNN,” in Proc. IEEE Int. Conf. Comput. Vis. (ICCV), Oct. 2017, pp. 2980–2988.

[11] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, “You only look once: Unified, real-time object detection,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2016, pp. 779–788.

[12] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C. Y. Fu, and A. C. Berg, “SSD: Single shot MultiBox detector,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Oct. 2016, pp. 21–37.

[13] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, “DeepLab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 40, no. 4, pp. 834–848, Apr. 2018.

[14] O. Ronneberger, P. Fischer, and T. Brox, “U-Net: Convolutional networks for biomedical image segmentation,” in Medical Image Computing and Computer-Assisted Intervention—MICCAI, 2015, pp. 234–241.

[15] A. Newell, K. Yang, and J. Deng, “Stacked hourglass networks for human pose estimation,” in Proc. Eur. Conf. Comput. Vis. Cham, Switzerland: Springer, 2016, pp. 483–499.

[16] F. Zhang, X. Zhu, H. Dai, M. Ye, and C. Zhu, “Distribution-aware coordinate representation for human pose estimation,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2020, pp. 7093–7102.

[17] B. Xiao, H. Wu, and Y. Wei, “Simple baselines for human pose estimation and tracking,” in Proc. Eur. Conf. Comput. Vis., Sep. 2018, pp. 472–487.
Y. Feiyu et al.: Error Compensation Heatmap Decoding for Human Pose Estimation

[18] S.-E. Wei, V. Ramakrishna, T. Kanade, and Y. Sheikh, “Convolutional pose machines,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2016, pp. 4724–4732.

[19] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh, “Realtime multi-person 2D pose estimation using part affinity fields,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jul. 2017, pp. 1302–1310.

[20] A. Newell, Z. Huang, and J. Deng, “Associative embedding: End-to-end learning for joint detection and grouping,” in Proc. Adv. Neural Inf. Process. Syst., 2017, pp. 2277–2287.

[21] T. Sekii, “Pose proposal networks,” in Proc. Eur. Conf. Comput. Vis., Sep. 2018, pp. 350–366.

[22] J. Tompson, A. Jain, Y. Lecun, and C. Bregler, “Joint training of a convolutional network and a graphical model for human pose estimation,” in Proc. Adv. Neural Inf. Process. Syst., 2014, pp. 1799–1807.

[23] T. Y. Lin, M. Maire, S. Belongie, J. Hays, and C. L. Zitnick, “Microsoft COCO: Common objects in context,” in Proc. Eur. Conf. Comput. Vis., Sep. 2014, pp. 740–755.

[24] M. Andriluka, L. Pishchulin, P. Gehler, and B. Schiele, “2D human pose estimation: New benchmark and state of the art analysis,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Jun. 2014, pp. 3686–3693.

YANG FEIYU was born in Sichuan, China. He received the B.S., M.S., and Ph.D. degrees from Xi’an Jiaotong University, in 2019. He is currently holding a postdoctoral position with Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences. He is also a Senior Algorithm Engineer working with Orbbec Inc. His research interests include renewable fuel combustion, quantum chemistry, fluid mechanics, artificial intelligence, and computer vision.

SONG ZHAN was born in Shandong, China. He received the Ph.D. degree from The Chinese University of Hong Kong, in 2008. He is currently a Ph.D. Supervisor with Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences. His research interests include 3D reconstruction, computer graphics, artificial intelligence, and computer vision.

XIAO ZHENZHONG was born in Shandong, China, in 1980. He received the Ph.D. degree from Xi’an Jiaotong University, in 2008. He is currently one of the founders and the Chief Technology Officer of Orbbec Inc. His research interests include 3D perception, artificial intelligence, and computer vision.

MO YAOYANG was born in Guangdong, China, in 1995. He received the B.S. and M.S. degrees from the South China University of Technology, in 2019. He is currently working as an Algorithm Engineer with Orbbec Inc. His research interests include human pose estimation, image matting, and semantic segmentation.

CHEN YU was born in Hunan, China, in 1988. He received the B.S. and M.S. degrees from Shenzhen University. He is currently working as an Algorithm Engineer with Orbbec Inc. His research interests include machine learning, human pose estimation, and 3D human pose construction.

SONG ZHAN was born in Shandong, China. He received the Ph.D. degree from The Chinese University of Hong Kong, in 2008. He is currently working as an Algorithm Engineer with Orbbec Inc. His research interests include deep learning, human pose estimation, and 3D human pose construction.

ZHANG YAO was born in Jiangsu, China. He is currently working as a Senior Algorithm Architect with Orbbec Inc. His research interests include human pose estimation and 3D human pose construction.

ZHANG MIN was born in Shaanxi, China, in 1994. She received the B.S. and M.S. degrees from Xi’an Electronic Technology University. She is currently working as an Algorithm Engineer with Orbbec Inc. Her research interests include deep learning, human pose estimation, and 3D human pose construction.

JIN WU was born in Jiangxi, China, in 1991. He received the B.S., M.S., and Ph.D. degrees from Xi’an Jiaotong University. He is currently working as an Algorithm Engineer with Orbbec Inc. His research interests include combustion, deep learning, and aircraft design.