3D Human Pose Estimation based on Multi-Input Multi-Output Convolutional Neural Network and Event Cameras: a proof of concept on the DHP19 dataset

Alessandro Manilii[1], Leonardo Lucarelli[1], Riccardo Rosati[1⋆], Luca Romeo[1,2], Adriano Mancini[1], and Emanuele Frontoni[1]

[1] Department of Information Engineering (DII), Università Politecnica delle Marche, Via Brecce Bianche, 12, 60131 Ancona, Italy
[2] Computational Statistics and Machine Learning and Cognition, Motion and Neuroscience, Istituto Italiano di Tecnologia, Genova, Italy

Abstract. Nowadays Human Pose Estimation (HPE) represents one of the main research themes in the field of computer vision. Despite innovative methods and solutions introduced for frame processing algorithms, the use of standard frame-based cameras still has several drawbacks such as data redundancy and fixed frame-rate. The use of event-based cameras guarantees higher temporal resolution with lower memory and computational cost while preserving the significant information to be processed and thus it represents a new solution for real-time applications. In this paper, the DHP19 dataset was employed, the first and, to date, the only one with HPE data recorded from Dynamic Vision Sensor (DVS) event-based cameras. Starting from the baseline single-input single-output (SISO) Convolutional Neural Network (CNN) model proposed in the literature, a novel multi-input multi-output (MIMO) CNN-based architecture was proposed in order to model simultaneously two different single camera views. Experimental results show that the proposed MIMO approach outperforms the standard SISO model in terms of accuracy and training time.

Keywords: Human Pose Estimation · Event Cameras · Multi-input Multi-Output Convolutional Neural Network.

1 Introduction

Human pose estimation (HPE) is a traditional computer vision challenge aiming at generating 2D or 3D human skeleton from single or multiple view of one or more subjects. This tasks need great computational capacity to achieve good performance since big amount of data need to be processed. Standard approaches rely on RGB[5] or RGB-D camera with several applications in different scenarios including retail [18] [19], people counting, person re-identification [11], clinical

⋆ Corresponding author: E-mail: r.rosati@pm.univpm.it

monitoring [16] and rehabilitation [6] [7]. Event cameras represent a great solution, guaranteeing high temporal resolution, high dynamic range and fewer storage requirements [3]. These cameras, such as the Dynamic Vision Sensor (DVS), reduce the amount of data recording only changes in pixel intensity values: this produces asynchronous streams of events and it avoids the redundancy caused by fixed and insignificant background information. Moreover, they also prevents problems such as fixed frame rate which does not change if the registered object is static or in movement, producing higher quality output. In this scenario, the DVS 3D Human Pose dataset (DHP19) [4], the first DVS dataset for 3D human pose estimation, could represent the benchmark for future works in real-time context in which lightness and speed are crucial features. The aim of this work is to explore the dataset by evolving the single-input single-output (SISO) approach proposed in [4] with a novel multi-input multi-output (MIMO) Convolutional Neural Network (CNN) model able to improve generalization performance by simultaneously modeling two different views. In particular, the model learns simultaneously the human pose from different views (cameras) thus leading to the simultaneous HPE for each different camera. We introduce a detailed and reproducible experimental setup procedure of our proposed model by tuning the optimal hyperparameters and selecting the best confidence threshold strategy. The main contribution of the work is the proposal of a MIMO strategy based on CNN model that allows to (i) improve the generalization performance and (ii) to reduce the computation effort in terms of training time [3].

The paper is organized as follows: Section 2 provides a description of the state-of-the-art about HPE task and CNN approaches applied to event cameras datasets. Section 3 gives details on the DHP19 dataset, the preprocessing steps applied and the proposed MIMO approach, which is the main core of this work. In Section 4, a comparative evaluation of our approach with the state-of-the-art is offered. Finally, in Section 5, discussion and conclusions about future directions for this field of research are drawn.

2 Related Work

2.1 HPE Datasets

Until today, there are multiple existing datasets for 3D HPE recorded using frame-based cameras. Between these, the most used are HumanEva [21], Human3.6M [9] and MPI-INF-3DHP [15]. All of them include a whole-body recording obtained with multiple cameras on different subjects, performing different movements. Moreover, they include ground-truth 3D pose recording from a motion capture system. However, state-of-the-art contributions on HPE through innovative datasets are not many. In literature, there are only two works related to human gestures or body movements recorded through event-based cameras [2, 8], but the only dataset built for HPE purpose with whole-body joints position is the one introduced in [4] and described in section 3.1.

[3] The code to reproduce all results is available at the following link:

2.2 CNN architectures

In the literature, a wide use of CNN has been made to solve the HPE task based on standard RBG images [23, 17, 22]. CNNs have been applied to the output of event-based cameras only to solve classification problems as in [2, 12, 13] or regression problems using independently different input and generating single output in [14]. Differently from the state of the art work we formulate the HPE using a MIMO strategy based on CNN models. As we shall see in the experimental results, our model performs favorably with respect to the SISO strategy proposed in [4].

3 Materials & Methods

DHP19 is the first human pose dataset with data collected from DVS event-based cameras. This specific feature implies a more complex preprocessing step in order to generate a standard frame from a sequence of events, but it introduces great enhancements, i.e. avoiding redundancy and saving space for more information, making the dataset more valuable. For the HPE task, the baseline approach is the same described in [4], in which a single CNN is used for each camera to predict joints' positions in 2D. Instead in the multi-view approach, two frames from different cameras but related to the same time instant are given as input to the network, and multiple outputs are returned as a couple of heatmaps representing estimated 2D joints positions for the two frames. For both methods, the final 3D pose estimation is inferred from 2D predictions using triangulation and knowing the position of the camera. A detailed description of the dataset contents, the instruments used, the preprocessing steps and the network architectures is provided below.

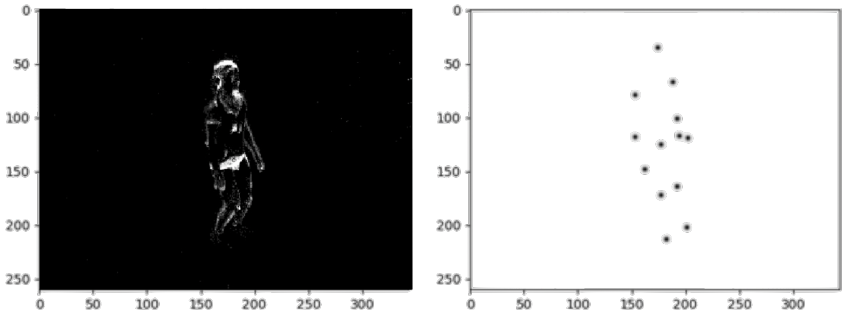

Fig. 1: a) Example frame generated from DVS events for subject 2, movement 9 of DHP19 dataset and b) the relative joint labels obtained as described in 3.2.

3.1 Dataset

DHP19 contains the records of 17 subjects, 12 females and 5 males who perform specific movements in a recording volume of 2x2x2 m^3 in a therapy environment. All 33 movements, grouped in 5 sessions, can be categorized in upper-limb movements (1, 2, 5, 6, 15-20, 27-33), lower-limb movements (3, 4, 7, 8, 23-26), and whole-body movements, and they are performed by each subject 10 times consecutively. The subject position is represented through 13 labeled joints positions corresponding to the head, both shoulder, both elbow, hands, left/right hip, both knees and feet as can be seen in Fig.1. In our experiments, subjects 1-9 (52% of the dataset) were used as training set, subjects 10-12 (18% of the dataset) as validation set and remaining subjects 13-17 (30%) as testing set.

3.2 Preprocessing and frame generation

Data have been acquired with the simultaneous use of 4 Dynamic and Active Pixel Vision Sensor (DAVIS) cameras [10], a complex version of standard DVS with a resolution of 260x344 pixels, and Vicon motion capture system for ground-truth recording, made of 10 Bonita Motion Capture (BMC) infrared (IR) cameras. An event $e = (x, y, t, p, c)$ is made of: position (x, y) in pixel array, time t in microsecond, polarity of the brightness change p and camera ID (0-3) c. In order to make use of pre-existent frame-based deep learning algorithms for event cameras, we applied the same preprocessing steps and transformed event stream into frames as in [4]. For this purpose, a fixed amount of events (about $7.5k$ per camera) are grouped in each frame, which is finally normalized in the range [0,255]. Instead, labels are constructed knowing initial and final event timestamps for each generated frame calculating the average position in that time window. The last step consists of mapping the obtained 3D label position into frame space, rounding to the nearest pixel, making use of projection matrices for each camera view. The projected 2D labels represent the absolute position in pixel space. Finally, a smoothing filter is applied on each heatmap through Gaussian blurring with a sigma of 2 pixels.

3.3 Baseline: Single-Input Single-Output (SISO) architecture

In this approach, a single CNN takes as input one frame at a time and so, for each camera, a different output for the same time instant is obtained. Input image is downsampled and then upsampled to produce a heatmap of the same size: the output of the network is an array of shape (260, 344, 13), where the 13 heatmaps represent the probability for each joint to be in a certain position among all 260x344 (frame size) possibilities. The training points are a sequence of frames picked up from a different subject, session, movement and from the two cameras. To get the final prediction, the position corresponding to the max value is taken and then compared with the label through mean squared error. A detailed description of model architecture and methods used for training and testing is provided in the following paragraphs.

Model architecture CNN is organized in 18 convolutional layers as shown in Table 1. Each layer is followed by Rectified Linear Unit (RELU) activation and it has a variable number of 3x3 filters from 16 to 64 contributing to a total amount of about 220k trainable parameters. A bidimensional max-pooling allows decreasing computational cost and affecting possible overfitting. In order to reproduce the baseline results in [4], we implemented their architecture considering the same setting for each layer parameter (i.e. number of filters, kernel size, strides, dilation rate).

Table 1: SISO model architecture

Layer	**Out dimension**	**Stride**	**Dilatation**
(1) Conv2D	(260, 344, 16)	1	1
MaxPooling2D	(130, 172, 16)	1	1
(2) Conv2D	(130, 172, 16)	1	1
(3) Conv2D	(130, 172, 32)	1	1
(4) Conv2D	(130, 172, 32)	1	1
MaxPooling2D	(65, 86, 32)	1	1
(5) Conv2D	(65, 86, 64)	1	2
(6) Conv2D	(65, 86, 64)	1	2
(7) Conv2D	(65, 86, 64)	1	2
(8) Conv2D	(65, 86, 64)	1	2
(9) Conv2DTc	(130, 172, 32))	2	1
(10) Conv2D	(130, 172, 32)	1	2
(11) Conv2D	(130, 172, 32)	1	2
(12) Conv2D	(130, 172, 32)	1	2
(13) Conv2D	(130, 172, 32)	1	2
(14) Conv2DTc	(260, 344, 16)	2	1
(15) Conv2D	(260, 344, 16)	1	1
(16) Conv2D	(260, 344, 16)	1	1
(17) Conv2D	(260, 344, 16)	1	1
(18) Conv2D	(260, 344, 13)	1	1

3.4 Proposed approach: Multiple-Input Multiple-Output (MIMO) architecture

In order to deeply adapt model architecture to the multi-view case-study, we implemented a multi-input and multi-output (MIMO) CNN, which employs shared layers. In particular, as shown in Fig.2, the network takes as input two frames of the same instant time respectively from cameras 2 and 3 and it outputs 13 heatmaps, one per joint, for each one as in single-input CNN. In single-input approach, the same layers were trained on both cameras and the batches are made up of a casual sequence of frames picked up from a different subject, session, movement. Differently, in this case, the not-shared layers are only trained

on a specific camera view, generating batches as a sequence of a couple of frames always related to two different cameras and randomly put in sequence.

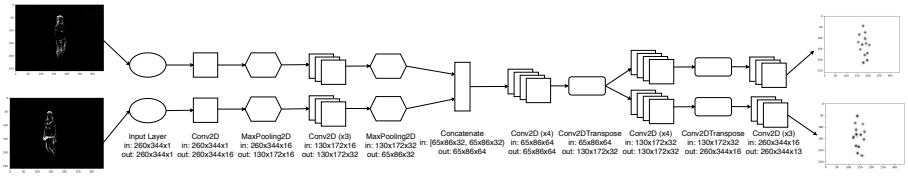

Fig. 2: Multi-Input Multi-Output (MIMO) model architecture.

Model architecture As shown in Fig. 2, CNN is constituted by two separate single input heads and output tails that share a group of central layers.The shared layers are the four convolutional layers (5-8) and the transposed convolution one (9) indicated in Table 1. These layers are updated on frames of both cameras while the others are only related to images from a specific cam. Where the two branches are separated (Conv2DTranspose layer), the bank of filters are duplicated. The increased number of layers due to this new configuration rises the number of trainable parameters from 220k to 310k.

3.5 3D Human Pose Estimation

There are different approaches for reconstructing 3D HPE from 2D HPE belonging to different multi-camera views [1, 20]. The method we have chosen for 3D triangulation is the same described in [4]. Firstly the model predicts 2D joints position, then the triangulation method is applied to project the 2D prediction on the 3D space knowing the position of the camera, as represented in Fig.3

3.6 Experimental procedure

Training setting The SISO model was trained for 20 epochs using a batch size of 64 and a variable learning rate of 10e-3 for the first 10 epochs, of 10e-4 from epoch 11 to 15 and of 10e-5 from 16 to 20. Mean Squared Error (MSE) and RMSProp are used respectively as cost function and optimizer. The training phase took about 80 hours on an NVIDIA RTX-2080 Ti. As regards the MIMO model, the training procedure and network parameters remain the same described for SISO to perform a fair comparison between the two approaches, except for the batch size equal to 32 in order to generate batches with the same number of frame used for the single-input model. In this case, the training time was about 70 hours: this lower execution time with respect to the single-input approach is due to the implementation of multi-input approach which, for each couple of frames given in input, decrease the total amount of training points.

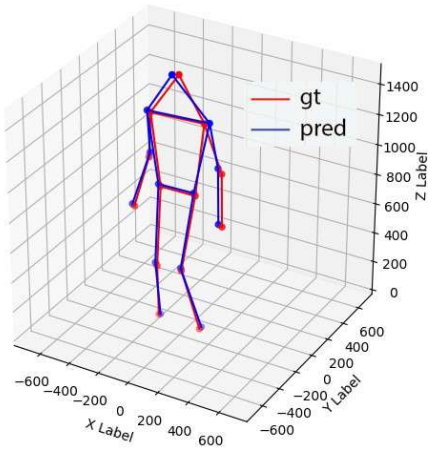

Fig. 3: Ground-truth and prediction overlapping in 3D space.

Testing After the preprocessing step, we tested the models on the same test subjects (13-17) with and without confidence threshold in the range [0.1 - 0.5]. The confidence threshold mechanism works comparing the max value of the heatmap with the set threshold for each joint and, in case the condition isn't satisfied, the last valid joint position is taken. This can lead to taking better or worst prediction depending on which is the average max value in the output heatmaps: this is the reason why using a higher confidence threshold does not improve performance, as shown by the results in following paragraphs. We first tested the model trained by [4] in order to reproduce their own results, then the SISO and MIMO models trained from scratch.

Evaluation metric For evaluation purposes the mean per joint position error (MPJPE) metric is used, expressed with the following formula:

$$MPJPE = \frac{1}{J}\sum_{i}^{J} \left\| x_i - \hat{x}_i \right\| \tag{1}$$

where i represent a different joint at every iteration, ranging in the interval [1,J], J is the number of joints, x is the ground-truth absolute position acquired by the Vicon system and $\hat{x}$ is the predicted absolute position.

4 Results

In Section 4.1 the validation results are reported, while in Section 4.2 and Section 4.3 we reported the results related to 2D pose estimation and 3D pose estimation respectively.

4.1 Validation results

Despite we use MSE as validation loss for both models, it's more explanatory to calculate MPJPE on the validation subjects in order to analyze how training performance evolves during epochs. Figures 4 represents the trend of the MPJPE through the 20 epochs respectively for the single-input model and the multi-input model, calculated with the corresponding optimal confidence threshold identified in the next sections. We can notice how both models converge and the trend follows the learning rate scheduler.

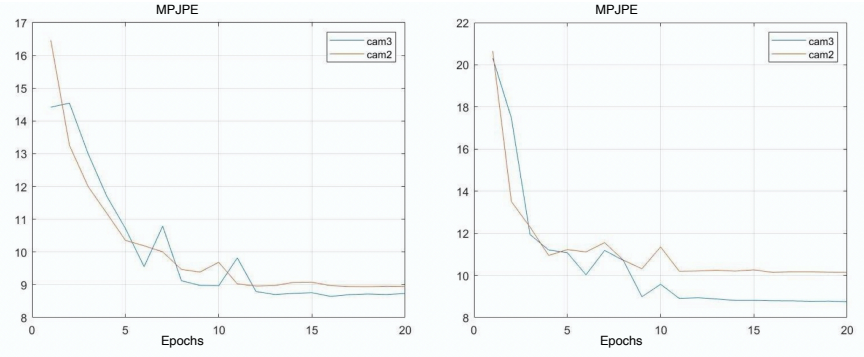

Fig. 4: MPJPE trend for a) SISO model and b) MIMO model for camera 2 and camera 3 through the 20 epochs.

4.2 2D pose estimation

Table 2 shows the testing results in terms of MPJPE values for different confidence thresholds for the pre-trained model introduced in [4] and the models described in sections 3.3 and 3.4. The MPJPE are averaged over all the testing subjects (subject 13, 14, 15, 16 and 17). The optimal threshold for 2D pose estimation is calculated as a function of the average max of the predicted value for each pixel. In fact, a higher threshold means that a large number of predictions are discarded and for each one of them this mechanism takes as an actual prediction the last one (in terms of time) that has satisfied the threshold's condition. Hence, a higher confidence threshold means a higher probability that the temporal gap between the current prediction and the last good one is elevated. On the other hand, a lower confidence threshold may lead to a higher sparsity of the heatmap values and to a higher uncertainty of the prediction. The extracted SISO model results reveals a difference with respect to the baseline results reported in [4]. For the baseline model, the best results correspond to the confidence threshold equal to 0.3, with a relative improvement (compared to no-confidence results) of 6% and 12% for cameras 2 and 3 respectively. Results

of the SISO model are worse compared to the baseline results, with an average MPJPE increment of %30. This may be due to a different set of training parameters or a different type of data normalization made by authors in [4]. However, the MIMO model leads to an average improvement of the 2D HPE results of 3.5% compared to SISO model (referring to the value corresponding to the best confidence threshold of each model).

Table 2: Summary table comparing MPJPE score on test subjects for the baseline pre-trained model, the SISO model and the MIMO model, all trained on the two frontal cameras (camera 2 and 3) and tested with various confidence thresholds. In bold the best-selected confidence threshold used for 3D projection.

Conf. Thr.	None	0.01	0.1	0.3	0.4
Baseline					
Cam 2	7.70	7,55	7.42	**7.22**	7.26
Cam 3	7.92	7.69	7.25	**6.91**	6.98
SISO					
Cam 2	11.47	11.12	**10.56**	11.09	11.65
Cam 3	10.86	10.72	**10.16**	10.36	10,77
MIMO					
Cam 2	10.49	**10.40**	11.82	14.85	16.54
Cam 3	9.64	**9.58**	10.51	13.75	15.93

4.3 3D pose estimation

Table 3 summarises the 3D pose estimation results. For all the models, the best results are obtained for the second section (precisely movements 9-14, 21, 22), which correspond to movements where the human shape is entirely visible in the DVS frames. The pre-trained baseline model, with a confidence threshold of 0.3, reaches the averaged MPJPE of 80,31 mm. The MIMO model obtains an improvement (118.23 mm vs 115.48 mm respectively) of 2.5% in terms of 3D MPJPE with respect to the standard SISO model.

The increased number of layers due to this new configuration rises the number of trainable parameters from 220k (for a single network) to 308k. However, the computation effort in terms of the training time of the MIMO is reduced by 10% compared to the SISO model. This fact can be explained by the lower number of training images (the half compared to SISO) required by MIMO for learning simultaneously the pose from two different cameras instead of aggregating the data acquired from the two cameras.

Table 3: Averaged 3D MPJPE (in mm) of the 5 testing subject through the 5 sessions for different CNNs. In bold the overall mean of the 3D MPJPE for each model.

\	Subj. 13	Subj. 14	Subj. 15	Subj. 16	Subj. 17	Subj. Mean
Baseline						
Session 1	84,67	95,87	86,74	125,37	107,41	90,95
Session 2	40,69	53,29	48,65	72,88	79,43	66,43
Session 3	91,80	134,50	104,70	125,16	134,74	124,79
Session 4	77,57	101,05	105,97	92,53	99,63	80,53
Session 5	75,70	107,24	106,48	147,81	111,46	113,84
Session Mean	60,44	80,91	75,85	92,12	86,22	**80,31**
SISO						
Session 1	175,96	190,79	166,43	202,26	175,70	150,09
Session 2	64,21	73,77	68,12	101,60	102,92	87,17
Session 3	162,35	186,77	171,70	214,22	182.45	209,74
Session 4	129,42	147,47	149,90	135.82	123.36	116,29
Session 5	147,47	198,07	177.16	240,46	180,28	198,40
Session Mean	95,27	121,50	116,74	136,84	116,08	**118,23**
MIMO						
Session 1	147,02	168,42	179,63	173,78	263,26	165,38
Session 2	55,70	61,70	60,24	93,12	104,13	85,06
Session 3	149,17	166,00	150,37	216,58	191,38	183,94
Session 4	131,73	142,51	154,79	126,71	129,93	115,48
Session 5	144,58	161,27	124,75	224,25	177,62	174,22
Session Mean	96,90	101,78	107,61	127,60	126,65	**115,48**

5 Conclusions

In this work we improve the single-input single-output (SISO) approach proposed in [4] with a novel multi-input multi-output (MIMO) Convolutional Neural Networks (CNN) model able to improve the generalization performance of HPE by simultaneously modeling two different views of event cameras. Starting from the experimental procedure presented in [4] we tried to reproduce their experimental results implementing the same SISO CNN based model. However, the baseline results extracted by [4] seems to be not fully reproducible. The incongruences found between the SISO and baseline results on DHP19 dataset reported in the paper may be due to several reasons. The more reasonable one regards the training hyperparameters, especially for SISO model in which CNN architecture has not to be modified. For example, the setting of different batch size or learning rate schedule, according to the evaluation metrics trend among epochs, could

lead to a difference in performance. Other reasons could be linked to the data generator mechanism, which may lead to a different training procedure, or to another kind of output normalization.

Our Experimental results on the DHP19 dataset demonstrated how the novel MIMO approach allows improving the generalization performance of 2D and 3D HPE while reducing the computation effort in terms of training time. This can be explained by considering that the two inputs (event camera views) given to the model share the same discriminative features since they correspond to 2 different points of view of the same instant of time. Thus the use of shared layers may encourage this relatedness by increasing generalization performance, as well as guaranteeing lower training time.

Future works could be related to exploring different training hyperparameters, by also selecting the optimal number of shared layers. Finally, another interesting future direction could be addressed to (i) extend the MIMO strategy by converting the 2D CNN into 3D CNN for obtaining a direct 3D HPE (ii) to impose kinematic constraints to refine the overall 3D HPE. Accordingly, recurrent 3D CNN can be investigated in order to learn spatio-temporal features by modeling sequential temporal relationships among weights.

References

1. Amin, S., Andriluka, M., Rohrbach, M., Schiele, B.: Multi-view pictorial structures for 3d human pose estimation. In: 24th British Machine Vision Conference. pp. 1–12. BMVA Press (2013)
2. Amir, A., Taba, B., Berg, D., Melano, T., McKinstry, J., Di Nolfo, C., Nayak, T., Andreopoulos, A., Garreau, G., Mendoza, M., Kusnitz, J., Debole, M., Esser, S., Delbruck, T., Flickner, M., Modha, D.: A low power, fully event-based gesture recognition system. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 7388–7397 (2017)
3. Brandli, C., Berner, R., Yang, M., Liu, S.C., Delbruck, T.: A 240× 180 130 db 3 μs latency global shutter spatiotemporal vision sensor. IEEE Journal of Solid-State Circuits **49**(10), 2333–2341 (2014)
4. Calabrese, E., Taverni, G., Awai Easthope, C., Skriabine, S., Corradi, F., Longinotti, L., Eng, K., Delbruck, T.: Dhp19: Dynamic vision sensor 3d human pose dataset. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops (June 2019)
5. Cao, Z., Simon, T., Wei, S., Sheikh, Y., et al.: Openpose: Realtime multi-person 2d pose estimation using part affinity fields. IEEE Transactions on Pattern Analysis and Machine Intelligence (2019)
6. Capecci, M., Ceravolo, M.G., D'Orazio, F., Ferracuti, F., Iarlori, S., Lazzaro, G., Longhi, S., Romeo, L., Verdini, F.: A tool for home-based rehabilitation allowing for clinical evaluation in a visual markerless scenario. In: 2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC). pp. 8034–8037. IEEE (2015)
7. Capecci, M., Ceravolo, M.G., Ferracuti, F., Iarlori, S., Monteriù, A., Romeo, L., Verdini, F.: The kimore dataset: Kinematic assessment of movement and clinical scores for remote monitoring of physical rehabilitation. IEEE Transactions on Neural Systems and Rehabilitation Engineering **27**(7), 1436–1448 (2019)

8. Hu, Y., Liu, H., Pfeiffer, M., Delbruck, T.: Dvs benchmark datasets for object tracking, action recognition, and object recognition. Frontiers in Neuroscience **10**, 405 (2016). https://doi.org/10.3389/fnins.2016.00405, https://www.frontiersin.org/article/10.3389/fnins.2016.00405
9. Ionescu, C., Papava, D., Olaru, V., Sminchisescu, C.: Human3.6m: Large scale datasets and predictive methods for 3d human sensing in natural environments. IEEE Transactions on Pattern Analysis and Machine Intelligence **36**(7), 1325–1339 (jul 2014)
10. Lichtsteiner, P., Posch, C., Delbruck, T.: A 128×128 120 db 15μ s latency asynchronous temporal contrast vision sensor. IEEE journal of solid-state circuits **43**(2), 566–576 (2008)
11. Liciotti, D., Paolanti, M., Frontoni, E., Mancini, A., Zingaretti, P.: Person re-identification dataset with rgb-d camera in a top-view configuration. In: Video Analytics. Face and Facial Expression Recognition and Audience Measurement, pp. 1–11. Springer (2016)
12. Liu, H., Moeys, D.P., Das, G., Neil, D., Liu, S., Delbrück, T.: Combined frame- and event-based detection and tracking. In: 2016 IEEE International Symposium on Circuits and Systems (ISCAS). pp. 2511–2514 (2016)
13. Lungu, I., Corradi, F., Delbrück, T.: Live demonstration: Convolutional neural network driven by dynamic vision sensor playing roshambo. In: 2017 IEEE International Symposium on Circuits and Systems (ISCAS). pp. 1–1 (2017)
14. Maqueda, A.I., Loquercio, A., Gallego, G., García, N., Scaramuzza, D.: Event-based vision meets deep learning on steering prediction for self-driving cars. CoRR **abs/1804.01310** (2018), http://arxiv.org/abs/1804.01310
15. Mehta, D., Rhodin, H., Casas, D., Sotnychenko, O., Xu, W., Theobalt, C.: Monocular 3d human pose estimation using transfer learning and improved CNN supervision. CoRR **abs/1611.09813** (2016), http://arxiv.org/abs/1611.09813
16. Moccia, S., Migliorelli, L., Carnielli, V., Frontoni, E.: Preterm infants' pose estimation with spatio-temporal features. IEEE Transactions on Biomedical Engineering (2019)
17. Newell, A., Yang, K., Deng, J.: Stacked hourglass networks for human pose estimation. In: European conference on computer vision. pp. 483–499. Springer (2016)
18. Paolanti, M., Romeo, L., Liciotti, D., Pietrini, R., Cenci, A., Frontoni, E., Zingaretti, P.: Person re-identification with rgb-d camera in top-view configuration through multiple nearest neighbor classifiers and neighborhood component features selection. Sensors **18**(10), 3471 (2018)
19. Paolanti, M., Romeo, L., Martini, M., Mancini, A., Frontoni, E., Zingaretti, P.: Robotic retail surveying by deep learning visual and textual data. Robotics and Autonomous Systems **118**, 179–188 (2019)
20. Rhodin, H., Robertini, N., Casas, D., Richardt, C., Seidel, H., Theobalt, C.: General automatic human shape and motion capture using volumetric contour cues. CoRR **abs/1607.08659** (2016), http://arxiv.org/abs/1607.08659
21. Sigal, L., Balan, A., Black, M.J.: HumanEva: Synchronized video and motion capture dataset and baseline algorithm for evaluation of articulated human motion. International Journal of Computer Vision **87**(1), 4–27 (Mar 2010)
22. 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)
23. 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)