Camera parameters estimation from pose detections

E.A. Shalimova [1], E.V. Shalnov [1], A.S. Konushin [1,2]

[1] Samsung-MSU Laboratory, Lomonosov Moscow State University, Leninskie Gory 1-52, Moscow, Russia

[2] Samsung AI Center, 5c Lesnaya str., Moscow, Russia

Abstract

Some computer vision tasks become easier with known camera calibration. We propose a method for camera focal length, location and orientation estimation by observing human poses in the scene. Weak requirements to the observed scene make the method applicable to a wide range of scenarios. Our evaluation shows that even being trained only on synthetic dataset, the proposed method outperforms known solution. Our experiments show that using only human poses as the input also allows the proposed method to calibrate dynamic visual sensors.

Keywords: camera calibration, dynamic vision sensor, video surveillance.

Citation: Shalimova EA, Shalnov EV, Konushin AS. Camera parameters estimation from pose detections. Computer Optics 2020; 44(3): 385-392. DOI: 10.18287/2412-6179-CO-600.

1. Introduction

Video surveillance is an essential part of modern urban infrastructure. It makes cities safer, simplifies traffic control and urban planning. Thus surveillance systems have to see and automatically analyse as much as possible through hundreds of thousands of high dimensional artificial eyes. Understanding of scene geometry is one of the basic tasks that should be solved as it simplifies the further data processing stages ([9]). This task has to be solved automatically without any interaction with an operator even for static cameras because a camera orientation may change unexpectedly from time to time due to other people's actions.

A common scene in the CCTV scenario is composed of a static camera, a single ground plane, people and obstacles such as buildings, benches, trash bins etc. In this case camera location, orientation and internal parameters provide sufficient information for retrieving scene geometry and, for example, filtering detection hypotheses.

Looking at the images of pedestrians, a human can roughly estimate camera position and rotation (fig. 1), since the approximate human height distribution is known, and feet of the people in the image are usually located close to the ground plane. Similarly to this, we propose an algorithm that estimates camera location and orientation (and also some of the intrinsic parameters) from human pose detections. To make it robust to the variety of possible detections (due to walking directions, location in the scene, height, gait specifics, etc.) we suggest training neural networks that learn the mapping from the detected human poses to the camera parameters. Analytical solutions are usually based on some models of the observed world and require the input data to be measured precisely. Moreover, an analytical solution that takes into account the great variability of the detected human poses would be very sophisticated. Unlike analytical solutions, the neural networks do not rely on any restrictions, rules or features explicitly; instead they directly reveal the dependencies between the (possibly noisy) input data and the target values. This allows us to avoid modeling complex dependencies between human pose keypoints locations and camera parameters, filtering outliers and dealing with incorrect detections, as neural network can learn to do this from the large amount of training data.

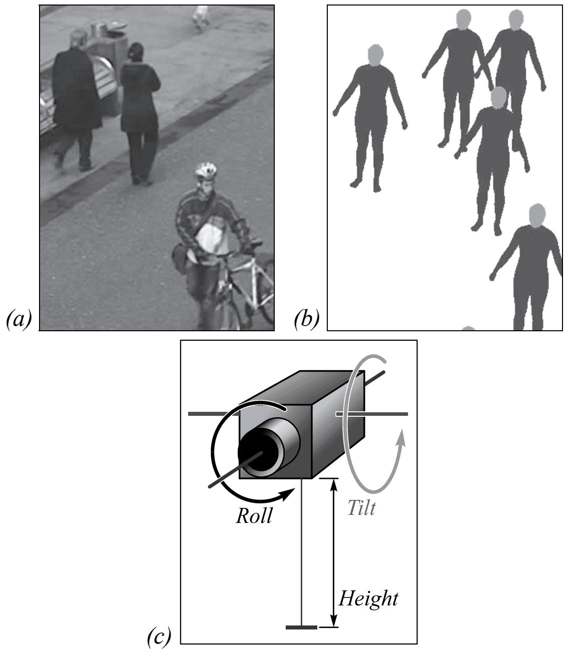

Fig. 1. (a) A part of the CCTV scene; (b) scene geometry can be estimated just from detected human positions and scale without any background information; (c) extrinsic camera parameters

The proposed algorithm automatically estimates both camera extrinsic parameters (location and view direction) and focal length. The only information it requires is a set of human poses specific to people in the observed scene. The algorithm makes no additional assumptions about the input data. Moreover, in contrast to the traditional calibration methods, it does not require any interaction with an operator (such as showing checkerboard pattern in a camera field of view or explicitly specifying size of some object present in the scene) which is crucial for the applicability of the method to the calibration of modern surveillance systems. Experiments show that it can also be applied to calibrate event-based cameras ([6]).

Our main contributions are:

1) An algorithm that estimates both focal length and extrinsic parameters of a static camera;

2) A method to calibrate dynamic vision sensor even without specialized pose detectors;
3) A way to tune parameters of the proposed method without real labeled data.

2. Related work

Recent camera calibration methods can be split into two groups. The first group focuses on localization of three orthogonal vanishing points, which makes such methods applicable to the Manhattan world scenes.

A fundamental relation between camera focal length and location of three orthogonal vanishing points (TOVPs) was discovered in [5]. Li et al. [13] suggest a method that estimates focal length by parallel lines detection, which is mostly applicable to scenes with buildings, roads or other manmade static structures. Sochor et al. [20] and Dubska et al. [7] solve the task of traffic surveillance camera calibration by extracting parallel lines from trajectories of cars on different lanes or fitting 3D models to detected cars. Lv et al. [16] and Li et al. [12] extract vertical lines from person silhouettes. Huang et al. [11] extract parallel lines from human feet detections, assuming the humans move along the line with steps of equal length. The quality of these methods depends on the presence and behavior of objects in the scene.

The second group skips TOVPs localization step and estimates camera parameters directly from the input image. Several works (Workman et al. [23]; Workman et al. [24]; Hold-Geoffroy et al. [10]; Yan et al. [25]) estimate focal length or horizon line from raw pixel intensities using Convolutional Neural Networks (CNNs). However the parameters estimated by these methods are insufficient to recover the scene geometry. Shalnov et al. [19] use CNN that takes focal length and human head detections as input and estimates camera location and orientation. The latter approach suits well the typical surveillance scenarios, however it requires focal length to be known. On the other hand, this approach does not use intensity values, thus can be applied in event-based vision [17].

3. Proposed algorithm

3.1. Pinhole camera model

We use a simple pinhole camera model. Under the assumption of zero skew and unit sensor aspect ratio, the matrix of intrinsic parameters is given by

$$\begin{bmatrix} f & 0 & p_x \\ 0 & f & p_y \\ 0 & 0 & 1 \end{bmatrix}, \qquad (1)$$

where f is the focal length value evaluated at pixel scale and (p_x, p_y) is the principal point. Moreover, we assume that the principal point is located in the center of the image, and that the distortion is negligible (note that images with known distortion can be automatically corrected as in [26] and then used for calibration). These assumptions are common for similar works in the field (i.e. [14, 15]). Therefore the focal length is the only unknown intrinsic parameter.

Camera orientation includes three angles known as tilt, roll and yaw. Yaw angle does not affect the scene geometry, thus assumed to have zero value. Camera location is fully specified by height above the ground plane.

3.2. Algorithm overview

The proposed algorithm estimates camera parameters from location of people joints on the image plane. We construct this mapping in form of convolutional neural network trained on the synthetic dataset.

The proposed algorithm consists of four stages (Fig. 2). At the first stage we apply OpenPose detector ([4, 22]) to estimate human poses on each frame. The first CNN (Coarse Focal length Network, or CFN) uses the constructed set of human poses to produce an initial coarse estimate of the camera focal length. Both the estimate and the set of poses are fed to the Location and Orientation Networks (LONs) that compute the camera location and orientation parameters separately. At the last stage focal length is refined with another CNN named Refinement Focal length Network (RFN).

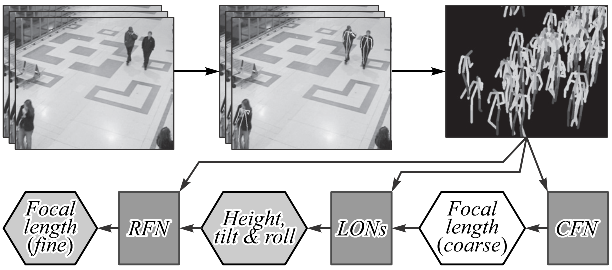

Fig. 2. The scheme of the proposed method. Rectangular boxes depict neural networks. Arrows show CNNs inputs and outputs

All the forementioned networks for camera parameters estimation have similar architectures. They consist of 2 convolutional and 2 fully connected layers. Since 64 pose detections are used for camera parameters estimation, the input is a 8×8×28 tensor, where the detections are placed in a 8×8 grid, while the third dimension represents a single pose (there are 14 keypoints, each of which is determined by 2 coordinates). The previously estimated camera parameters are concatenated with the computed features just before the first fully connected layer in Extrinsics and Focal length refinement CNNs (Fig. 3).

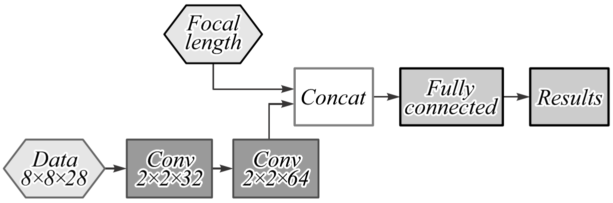

Fig. 3. An architecture of the proposed network estimating extrinsic parameters (LON)

3.3. Training dataset

A large labeled dataset of surveillance videos with diverse known camera parameters is required to train the CNNs. The sophisticated calibration process makes construction of such a dataset an extremely challenging task. However, only a set of human poses and calibration parameters are required at training stage, so the algorithm can be trained on any set of realistic human poses.

We define a unique set of camera parameters as a scene. Our synthetic dataset consists of 200000 such scenes, each of which contains 64 human figures standing on a ground plane. The algorithm uses only the human pose detector results, thus the dataset containscamera calibration parameters and 3D locations of human body keypoints in the scene.

We augment the set of human poses by adding a slight noise to the 3D position of each joint.

We choose camera parameters at random from uniform distribution with supported range presented in Tab. 1.

Tab. 1. Camera parameters description and limits

Parameter	Supported range
Focal length(pixels)	(10, 5000)
Tilt (degrees)	(0, 90)
Roll (degrees)	(–15, 15)
Height (cm)	(10, 2000)

3.4. Human pose model

We use CMU Graphics Lab Motion Capture Database [1] that contains 3D positions of joints of walking and standing people to simulate results of a human pose detector. We choose disjoint sets of actors to generate train and test subsets of our dataset. Human pose is defined by locations of 14 joints corresponding to knees, elbows, shoulders and so on.

In real world scenario person has similar poses on neighbouring frames. Thus to reduce the gap between real and synthetic data, we use not only single poses, but also pose sequences from the motion capture database mentioned above. Therefore the synthetic samples usually have several isolated detections and several detection sequences, which approximates typical detector result on a real surveillance video. The ratio of the number of isolated detections and detection sequences varies randomly between the samples. An example of synthetic data is shown below (Fig.4). It also can be seen that synthetic data contain keypoints of poses of people walking in various directions.

4. Architecture choices

This section outlines our architecture explorations. The lack of labeled real surveillance videos does not allow us to choose the network architecture without overfitting. Thus we use a validation part of the constructed synthetic dataset to approve the choice of the architecture.

4.1. Linear layers

We begin by constructing simple networks to estimate camera parameters from pose detections. The aim of this experiment was to find out whether convolutional layers usage is beneficial for camera parameters estimation. Therefore we trained several networks with and without convolutional layers and compared the results. For the sake of brevity we describe just two of them, each best in its class.

The first network, Fully-Connected network, consists of 4 dense layers. Each layer has 256 neurons. The second network has two convolutional layers of 32 and 64 filters, which are followed by two dense layers. The networks were trained separately on the same synthetic data until their validation losses stopped decreasing, which took around 200 epochs in both cases. As Table 2 (columns 2 and 3) shows, Convolutional network significantly outperforms Fully-Connected. Therefore we use convolutional layers in all networks in subsequent experiments.

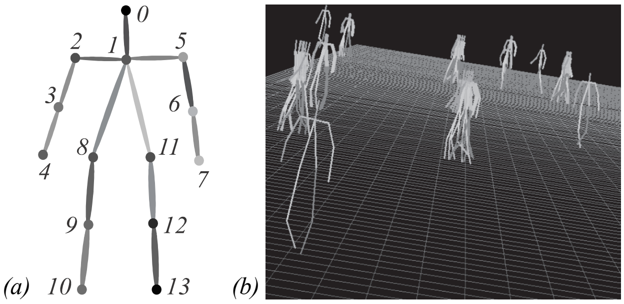

Fig. 4. (a) Body keypoints scheme; (b) visualization of synthetic data example

Tab. 2. RMSE error on test synthetic data

Parameter	Fully-Connected network	Convolutional network	Separate convolutional networks
Focal length (pixels)	670.13	445.11	**439.50**
Tilt (degrees)	8.59	3.72	**3.32**
Roll (degrees)	3.17	2.41	**2.18**
Height (cm)	288.16	210.77	**204.79**

4.2. Number of networks

We then explore whether it is better to estimate all the parameters with one network or with independent networks for every parameter.

We compare the following options: (1) a network which output is Dense layer with 4 neurons (since there are 4 estimated camera parameters); (2) a set of 4 independent networks. The results (*Tab.* 2, columns 3 and 4) show that the independent networks perform slightly better.

4.3. Pipeline

We found that the overall quality of the model can be improved by resolving visual ambiguity of focal length and camera height. Joint parameter estimation can be difficult due to visual ambiguity of focal length and camera height: for instance, simultaneous increase of focal length and camera height can result in a scene looking very similar to the original one, except for some fine details (i.e. objects looking more or less flattened).

We then conducted experiments on how the prior estimate of focal length influences extrinsics estimation quality, and vice versa. We studied not only the case of the ground truth focal length/extrinsics as an additional input, but also of its coarse estimate. For instance, as previous experiments show, it is possible to predict focal length from pose detections with RMSE about 440 pixels. To find out whether this estimate can be useful, we added to ground truth focal length random noise with mean value of 500 and gave it as an additional input to the extrinsics estimation networks. For focal length estimation with extrinsics prior, we used the following mean noise values: 10 degrees for tilt angle, 5 degrees for roll angle and 200 cm for height.

Our experiments show that prior knowledge of other camera parameters is indeed beneficial for both focal length (Table 3) and extrinsics estimation (Table 4).

Tab. 3. RMSE for focal length est. network with and without extrinsics estimate as an additional input

Parameters	Detections only	+ noisy extrinsics	+ ground truth extrinsics
Focal length	439.5	341.7	215.7

Tab. 4. RMSE for extrinsics est. network with and without focal length estimate as an additional input

Parameters	Detections only	+ noisy focal length	+ ground truth focal length
Tilt	3.32	3.07	2.07
Roll	2.18	2.11	2.07
Height	204.79	169.04	144.13

We use these findings in the following way: initially we have only pose detections, so we get a coarse focal length estimate from them. It is then given to the extrinsics nets, which produce relatively good extrinsics estimation. At the last stage, the extrinsics estimation is used to get refined focal length estimation.

4.4. Number of pose detections

The plane can be defined with three points, so there are no theoretical limitations to number of people used for calibration, as every human pose detection we consider consists of 14 keypoints which carry sufficient information. However human poses vary greatly, as well as human height and shape, and pose detector results aren't perfect. Therefore relying on a small number of detections can be error-prone.

We trained our networks on different number of input detections from 4 to 64. As can be seen in Fig. 5, RMSE errors gradually decrease with input detection number increasing, and after detection number reaches 16, this decrease becomes more gradual.

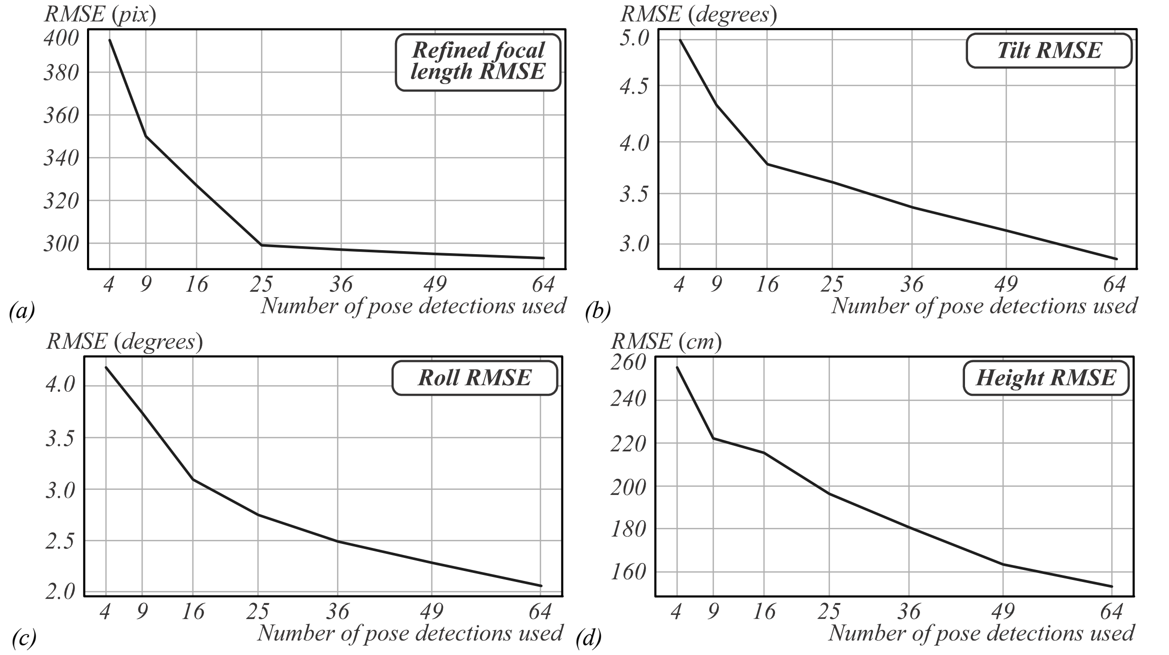

Fig. 5. RMSE for various input detections number: (a) focal length RMSE; (b) tilt RMSE; (c) roll RMSE; (d) height RMSE

5. Evaluation and results

We trained every network for 300 epochs using Adam optimizer on the constructed synthetic dataset with *L2* loss.

5.1 Dealing with false positive detections

While a lot of person images can be found in the real surveillance video, the proposed algorithm uses just 64 of them to make a single prediction. Thus a problem of choosing in some sense best person observations arises. One possible solution is to use the pose detector confidence as a quality criterion. On the one hand, greater confidence leads to exclusion of false detections that can possibly confuse the algorithm. On the other hand, it should be easier to infer scene geometry from a set of detections that are scattered all over the scene. Experiments show that high-quality detections tend to be located very close to each other. Therefore some trade-off between detection confidence and positions has to be found.

Our experiments show that the best results can be achieved from the following procedure:

1) Construct set of all presented poses;
2) Filter out the poses whose confidence is less than 0.1;
3) Sample at random 300 subsets of size 64 of the remaining pose detections;
4) Estimate camera parameters for each subset separately;
5) Compute average on each parameter separately.

5.2. Evaluation datasets

5.2.1. RGB data

We evaluate the proposed method on 31 video sequences from PETS 2006 (4 different cameras), PETS

2007 (4 different cameras), PETS 2009 (8 different cameras), EPFL Campus & Terrace [3, 8] (7 cameras) and 3DPeS [2] (8 cameras) datasets. Some of these sequences violate the assumption of a single ground plane (see Fig. 6 for example).

The calibration parameters used as ground truth were actually obtained by the authors of the respective datasets with Tsai calibration method [21]. Therefore the comparison to the ground truth is effectively a comparison with well-known classical method [21]. Since the goal of our work was to develop a method for fully-automatic calibration of hardly accessible cameras (and DVS sensors that cannot be calibrated at all with traditional methods) rather than to achieve superior calibration precision, we considered it possible to use Tsai calibration results as ground truth.

5.2.2. Event-based data

Dynamic Vision Sensor (DVS) is a type of sensor that records events of brightness change at every pixel at a very high rate (thousands frames per second). These sensors completely ignore static objects in the scene, which makes most of existing calibration methods inapplicable for them.

To our knowledge, there are no DVS data with known camera calibration suitable for out method evaluation. Therefore we simulate such data from the same datasets we use for RGB evaluation using the method from [17].

We convert an event stream back to "images" by summing up event polarities in a temporal window. This approach suppresses noise and some features of the detected objects, but the silhouettes are usually visible (Fig. 7).

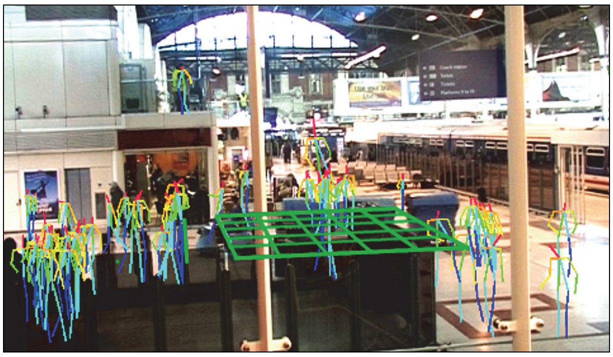

Fig. 6. Example of visualized pose detection used for camera parameters estimation (PETS 2006, camera 2). The ground plane computed with the predicted calibration parameters is shown in green (each square side is 1 m). The assumption of a single ground plane is violated by the human detection on the balcony. False detections are also present (note the big figure in the middle and the figure on the balcony), however the estimated ground plane is quite plausible

Our experiments show that OpenPose [4] detector can be applied to these visualized data even despite being designed for ordinary RGB data. It finds less poses in event-based versions than in RGB, and the quality of such detections is noticeably lower, but our algorithm is robust enough to get calibration results for event-based data close to those for RGB data (Tab. 5).

We have simulated event-based stream for all the abovementioned RGB datasets.

5.3. Visual evaluation of the results

One of the possible applications of calibration is ground plane estimation, which can be useful for filtering the false detections based on their size and location. Figure 8 shows that the calibration parameters predicted by our method produce rather plausible ground plane.

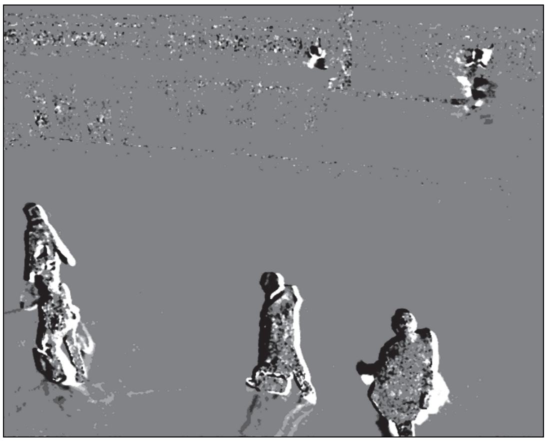

Fig. 7. A visualized frame of simulated event-based stream for one of PETS 2006 sequences

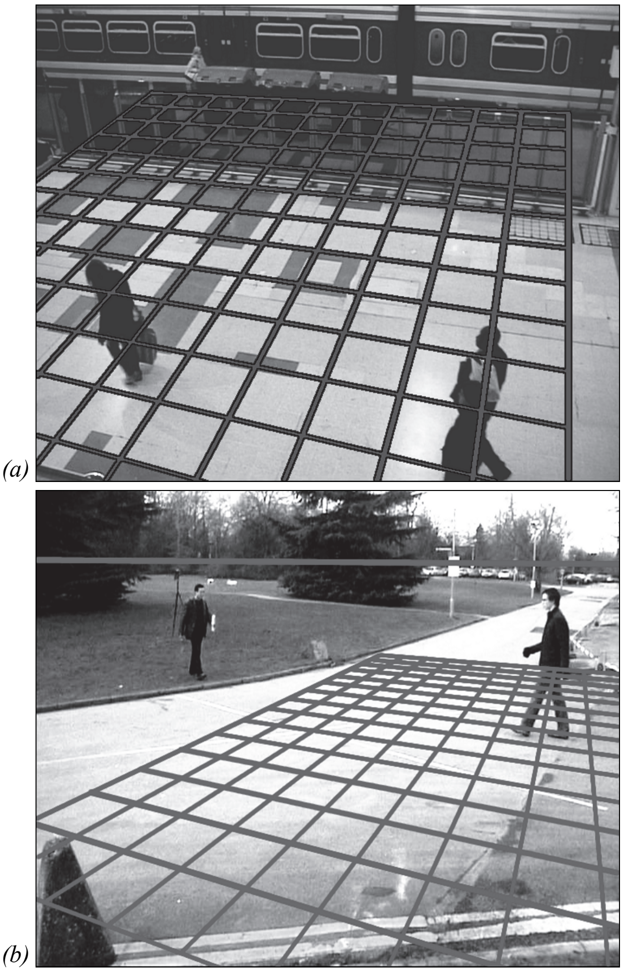

Fig. 8. Visualization of the ground plane (shown as a grid; each square side is 0.5 m) computed with the predicted calibration parameters for two PETS videos

We also analyzed how the method works if an important assumption of a single ground plane on which all the people in the scene stand is violated. We ran our method on several videos with escalators and stairways and found that in most cases the people that were not on the ground plane were occluded either by escalator railings or other people located on the ground plane closer to the camera. Since the algorithm uses only the detections for which all pose keypoints are visible, these detections didn't contribute to the results. Moreover, in many such videos the detections that violate single ground plane assumption are rare (see fig. 6 with only one such detection for example). Since our algorithm averages the results obtained on random subsets of all detections, as described in section 5.1, these detections have a decent chance to be chosen only a few times and thus contribute to the result less than the normal detections. This may be why roll angle estimation in fig. 6 is plausible despite the detection on the balcony. However in scenes similar to the one in fig. 9b, where the steps are wide and close to the camera, tilt angle estimation seems to be incorrect. This is somehow expected because the neural network tries to find a plane that all the people in the scene, including those on the stairs, could be staying on.

5.4. Comparison with another detection-based calibration method

Comparative evaluation of our approach is slightly complicated, mostly due to input data limitations. Our algorithm is targeted on the CCTV scenario, in which the cameras can be difficult to access physically, and there can be hundreds or thousands of cameras needing calibration. The traditional methods of camera calibration require presence of the checkerboard pattern and also need human interaction to recover the scale of the scene (i.e. via specifying size of some known object). Therefore it is difficult to obtain a large amount of data suitable to our scenario with ground truth camera calibration known. Moreover, the calibration patterns usually do not appear in the surveillance videos, so we cannot test the traditional methods on data suitable for our method without access to the actual cameras used to record it. Therefore we are limited to the comparison to another methods of automatical surveillance camera calibration, of which the most (i.e. [11], [13], [16]) neither have an available implementation nor report more than a few results on the real videos (in some cases the videos themselves are unavailable); an exception is [19]. There also are several traffic surveillance camera calibration methods (i.e. [7]) that require cars presence, while our methods needs pedestrian detections, so it's difficult to find a video that both [7] and our algorithm can be tested on.

Shalnov et al. [19] estimate extrinsic camera parameters from human head detections obtained with [18], hence it's possible to test our and their methods on the same data. Fortunately the head detector also proves to be applicable for visualized event-based data. Therefore both methods are tested on RGB and event-based data.

The results of evaluation are presented in Table 5. The proposed method outperforms [19] even though the latter gets camera focal length as additional input. The results of our method on event-based data are close to the results on RGB data. Our results are not exactly close to Tsai (ground truth) results, but they're still applicable, for example, for tasks that require imprecise ground plane estimation to filter detections based on their size. The pictures in section 5.3 show that the estimated ground plane is quite close to the real ground plane.

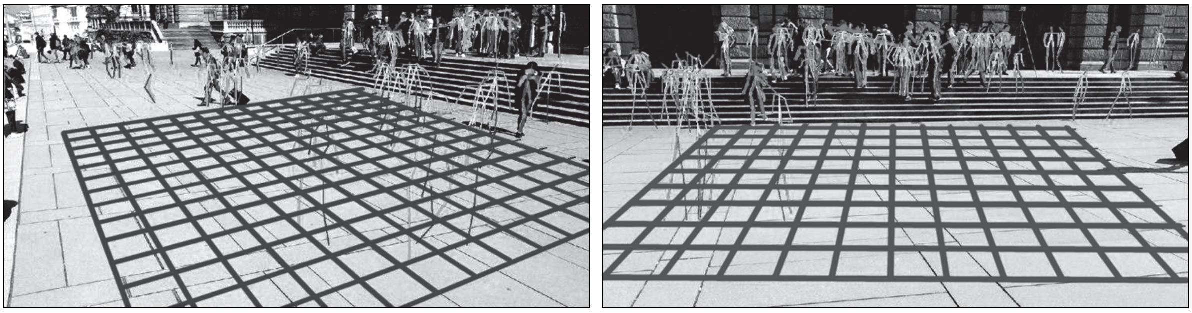

Fig. 9. Visualization of the ground plane (shown as a grid; each square side is 0.5 m) computed with the predicted calibration parameters for videos with detections that violate the assumption of a single ground plane

Tab. 5. Comparison of [19] and the proposed method on real data

	Method	RGB data				Event-based data			
		Focal length (pix)	Tilt (degrees)	Roll (degrees)	Height (cm)	Focal length (pix)	Tilt (degrees)	Roll (degrees)	Height (cm)
Mean error	[19]	–	17.37	5.51	327.10	–	16.52	4.57	307.50
	Ours	354.5	**4.02**	**1.04**	**131.2**	330.1	**4.87**	**1.48**	**151**
Median error	[19]	–	18.99	5.76	190.18	–	14.55	3.47	256.9
	Ours	128.8	**3.7**	**0.58**	**43**	163.4	**5.03**	**1.17**	**44**

6. Conclusion

In this work we describe an algorithm that estimates camera location, view direction and focal length using only human pose detections. We also show that the algorithm is applicable to Dynamic Vision Sensor parameters estimation. The future work can include several aspects:

1) integration with specialized object detectors for event-based data;
2) real data collection for performance evaluation on a larger dataset;
3) improvement of parameter estimation quality.

References

[1] CMU graphics lab motion capture database. Source: ⟨http://mocap.cs.cmu.edu/⟩.

[2] Baltieri D, Vezzani R, Cucchiara R. 3dPES: 3D people dataset for surveillance and forensics. Proceedings of the 1st International ACM Workshop on Multimedia access to 3D Human Objects 2011: 59-64. DOI: 10.1145/2072572.2072590.

[3] Berclaz J, Fleuret F, Turetken E, Fua P. Multiple object tracking using k-shortest paths optimization. IEEE Trans Pattern Anal Machine Intell 2011; 33(9): 1806-1819. DOI: 10.1109/TPAMI.2011.21.

[4] Cao Z, Simon T, Wei S-E, Sheikh Y. Realtime multi-person 2D pose estimation using part affinity fields. Proc IEEE Conf Comp Vis Patt Recogn 2017: 7291-7299. DOI: 10.1109/CVPR.2017.143.

[5] Caprile B, Torre V. Using vanishing points for camera calibration. Int J Comp Vis 1990; 4(2): 127-139. DOI: 10.1007/BF00127813.

[6] Delbruck T. Frame-free dynamic digital vision. Proceedings of International Symposium on Secure-Life Electronics, Advanced Electronics for Quality Life and Society 2008: 21-26.

[7] Dubská M, Herout A, Sochor J. Automatic camera calibration for traffic understanding. Proceedings of the British Machine Vision Conference 2014; 4: 8. DOI: 10.5244/C.28.42.

[8] Fleuret F, Berclaz J, Lengagne R, Fua P. Multi-camera people tracking with a probabilistic occupancy map. IEEE Trans Pattern Anal Machine Intell 2008; 30(2): 267-282. DOI: 10.1109/TPAMI.2007.1174.

[9] Hoiem D, Efros AA, Hebert M. Putting objects in perspective. Int J Comp Vis 2008; 80(1): 3-15. DOI: 10.1109/CVPR.2006.232.

[10] Hold-Geoffroy Y, Sunkavalli K, Eisenmann J, Fisher M, Gambaretto E, Hadap S, Lalonde J-F. A perceptual measure for deep single image camera calibration. Proc IEEE Conf Comp Vis Pattern Recogn 2018: 2354-2363. DOI: 10.1109/CVPR.2018.00250.

[11] Huang S, Ying X, Rong J, Shang Z, Zha H. Camera calibration from periodic motion of a pedestrian. Proc IEEE Conf Comp Vis Pattern Recogn 2016: 3025-3033. DOI: 10.1109/CVPR.2016.330.

[12] Li B, Peng K, Ying X, Zha H. Simultaneous vanishing point detection and camera calibration from single images. In Book: Bebis G, Boyle R, Parvin B, Koracin D, Chung R, Hammound R, Hussain M, Kar-Han T, Crawfis R, Thalmann D, Kao D, Avila L, eds. Advances in visual computing. Berlin, Heidelberg: Springer-Verlag; 2010: 151-160. DOI: 10.1007/978-3-642-17274-8_15.

[13] Li S, Nguyen VH, Ma M, Jin C-B, Do TD, Kim H. A simplified nonlinear regression method for human height estimation in video surveillance. EURASIP Journal on Image and Video Processing 2015; 2015(1): 32. DOI: 10.1186/s13640-015-0086-1.

[14] Liu J, Collins RT, Liu Y. Surveillance camera autocalibration based on pedestrian height distributions. British Machine Vision Conference (BMVC) 2011; 2: 117. DOI: 10.5244/C.25.117.

[15] Liu J, Collins RT, Liu Y. Robust autocalibration for a surveillance camera network. 2013 IEEE Workshop on Applications of Computer Vision (WACV) 2013: 433-440. DOI: 10.1109/WACV.2013.6475051.

[16] Lv F, Zhao T, Nevatia R. Self-calibration of a camera from video of a walking human. Object Recognition Supported by User Interaction for Service Robots 2002; 1: 562-567. DOI: 10.1109/ICPR.2002.1044793.

[17] Mueggler E. Event-based vision for high-speed robotics. PhD thesis. University of Zurich; 2017.

[18] Pricasariu V, Reid I. FastHOG-a real-time GPU implementation of HOG. Source: ⟨http://www.robots.ox.ac.uk/~lav/Papers/prisacariu_reid_tr2310_09/prisacariu_reid_tr2310_09.pdf⟩.

[19] Shalnov E, Konushin A. Convolutional neural network for camera pose estimation from object detections. International Archives of the Photogrammetry, Remote Sensing & Spatial Information Sciences 2017: 1-6. DOI: 10.5194/isprs-archives-XLII-2-W4-1-2017.

[20] Sochor J, Juránek R, Herout A. Traffic surveillance camera calibration by 3D model bounding box alignment for accurate vehicle speed measurement. Computer Vision and Image Understanding 2017; 161: 87-98. DOI: 10.1016/j.cviu.2017.05.015.

[21] Tsai RY. An efficient and accurate camera calibration technique for 3D machine vision. Proc IEEE Conf Comp Vis Pattern Recogn 1986: 364-374.

[22] Wei S-E, Ramakrishna V, Kanade T, Sheikh Y. Convolutional pose machines. Proc IEEE Conf Comp Vis Pattern Recogn 2016: 4724-4732. DOI: 10.1109/CVPR.2016.511.

[23] Workman S, Greenwell C, Zhai M, Baltenberger R, Jacobs N. DEEPFOCAL: A method for direct focal length estimation. IEEE Int Conf Image Process (ICIP) 2015: 1369-1373. DOI: 10.1109/ICIP.2015.7351024.

[24] Workman S, Zhai M, Jacobs N. Horizon lines in the wild. Proc British Machine Vis Conf 2016: 20. DOI: 10.5244/c.30.20.

[25] Yan H, Zhang Y, Zhang S, Zhao S, Zhang L. Focal length estimation guided with object distribution on FocaLens dataset. Journal of Electronic Imaging 2017; 26(3): 033018. DOI: 10.1117/1.JEI.26.3.033018.

[26] Zhang Z. A flexible new technique for camera calibration. IEEE Trans Pattern Anal Machine Intell 2000; 22: 1330-1334. DOI: 10.1109/34.888718.

Author's information

Ekaterina Alekseevna Shalimova (b. 1996) graduated from Moscow State University in 2018, majoring in Applied Mathematics and Informatics. She works at Samsung-MSU Laboratory. Research interests include computer vision, machine learning and neural networks. E-mail: *ekaterina.shalilmova@graphics.cs.msu.ru* .

Evgeny Vadimovich Shalnov (b. 1991) graduated in 2013 from Lomonosov Moscow State University and defended PhD thesis in Computer Science in 2019. He works as a junior research associate at Lomonosov Moscow State Uni-

versity. Primary research interests include computer vision, deep learning, bayesian methods in machine learning. E-mail: eshalnov@graphics.cs.msu.ru .

Anton Sergeevich Konushin (b. 1980) graduated from Lomonosov Moscow State University in 2002. In 2005 he successfully defended his PhD thesis in M.V. Keldysh Institute for Applied Mathematics RAS. He is currently associate professor at Lomonosov Moscow State University and associate professor at National Research University Higher School of Economics. Research interests are computer vision and machine learning.
E-mail: anton.konushin@graphics.cs.msu.ru .

Received July 19, 2019. The final version – February 18, 2020.