Convolutional Neural Networks for Pose Recognition in Binary Omni-directional Images

S. V. Georgakopoulos, K. Kottari, K. Delibasis, V. P. Plagianakos, I. Maglogiannis

Convolutional Neural Networks for Pose Recognition in Binary Omni-Directional Images

S.V. Georgakopoulos[1], K. Kottari[1], K. Delibasis[1], V.P. Plagianakos[1], I. Maglogiannis[2]

[1] Dept. of Computer Science and Biomedical Informatics, Univ. of Thessaly, Greece
[2]Department of Digital Systems, University of Piraeus, Greece
{spirosgeorg,vpp}@dib.uth.gr
{kdelibasis, kottarikonstantina}@gmail.com
imaglo@unipi.gr

Abstract. In this work, we present a methodology for pose classification of silhouettes using convolutional neural networks. The training set consists exclusively from the synthetic images that are generated from three-dimensional (3D) human models, using the calibration of an omni-directional camera (fish-eye). Thus, we are able to generate a large volume of training set that is usually required for Convolutional Neural Networks (CNNs). Testing is performed using synthetically generated silhouettes, as well as real silhouettes. This work is in the same realm with previous work utilizing Zernike image descriptors designed specifically for a calibrated fish-eye camera. Results show that the proposed method improves pose classification accuracy for synthetic images, but it is outperformed by our previously proposed Zernike descriptors in real silhouettes. The computational complexity of the proposed methodology is also examined and the corresponding results are provided.

Keywords: Computer Vision, Convolutional Neural Networks (CNNs), omni-directional image, fish-eye camera calibration, pose classification, synthetic silhouette

1 Introduction

Several computer vision and Artificial Intelligence applications require classification of segmented objects in digital images and videos. The use of object descriptors is a conventional approach for object recognition though a variety of classifiers. Recently, many reports have been published supporting the ability of automatic feature extraction by Convolutional Neural Networks (CNNs) that achieve high classification accuracy in many generic object recognition tasks, without the need of user-defined features. This approach is often referred to as deep learning.
More specifically, CNNs are state of the art classification methods in several problems of computer vision. They have been suggested for pattern recognition [2], object localization [3], object classification in large-scale database of real world images [4], and malignancy detection on medical images [5], [6], [7]. Several reports exist in

literature for the problem of human pose estimation and can be categorized in two approaches. The first approach relies on leveraging images local descriptors (HoG [8], SHIFT [9], Zernike [1], [10], [11], [11]) to extract features and subsequently constructing a model for classification. The second approach is based on model fitting processes [13, 14].

CNNs are trainable multistage architectures that belong to the first approach of classification methods [15]. Basically each stage comprises of three types of layers, the convolution, the pooling and the classic neural network layer, which is commonly refereed as fully-connected layer. Each stage consists of one of the previous layer type or an arbitrary combination of them. The trainable components of a convolutional layer are mapped as a batch of kernels and perform the convolution operator on the previous layer's output. The pooling layer performs a subsampling to its input with most commonly used pooling function to be the max-pooling, taking the maximum value of the local neighborhoods. Finally, the fully-connected layer can be treated as a special case of kernel with size 1×1. To train this network, the Stochastic Gradient Descent is usually utilized with the usage of mini-batches [16]. However, a drawback of the CNNs is the extensive training time required because of the amount of trainable parameters. Due to the inherent parallelism in their structure, the usage of the graphics processing units (GPUs) has been established to perform the training phase [4]. To achieve high quality results, the CNNs require training dataset of large size.

Very recently, methods that use deep neural networks in order to tackle the problem of human pose estimation have started to appear in the literature. In [17] the use the CNN as a regressor for rough joint locator hand-annotations of the Frame Labeled In Cinema (FLIC) dataset is reported. While the results are very promising, the existence of hand-annotations on training set is needed.

In this work, we test the suitability of a well-established CNN methodology for pose recognition in synthetic binary silhouette images. To the best of our knowledge CNNs have not been utilized to deal with binary images problems. These types of problems are distinct, because information is limited by the lack of RGB data. Furthermore, the silhouettes of this work are rendered through an omni-directional camera – fisheye. Fisheye cameras are dioptric omni-directional cameras, increasingly used in computer vision applications [18], [19], due to their 180 degree field of view (FoV). In [20], [21], [22] the calibration of fish-eye camera is reported to emulate the strong deformation introduced by the fish-eye lens. In [23] a methodology for correcting the distortions induced by the fish-eye lens is presented. The high volume of artificial silhouettes required for training the CNN, is produced by using 3D human models rendered by a calibrated fish-eye camera. Comparisons are provided for synthetic and real data with our recent method proposed in [1].

2 Methodology

2.1 Overview of the method

The main goal of this work is the assessment of the popular CNN technique ability to recognize different poses of binary human silhouettes from indoor images acquired by

a roof-based omnidirectional camera. An extensive dataset of binary silhouettes is created using 3D models ([24] and [25]) of a number of subjects in 5 different standing positions. The 3D models are placed in different positions and at different rotations round the Z-axis in the real world room. Then they are rendered through the calibration of the fish-eye camera, generating binary silhouettes. The dataset of binary silhouettes is separated into training and testing subsets. Classification results are calculated using the testing subset, as well as real segmented silhouettes of approximately the same poses. The aforementioned steps of the proposed methodology are shown in the block diagram of Fig. 1.

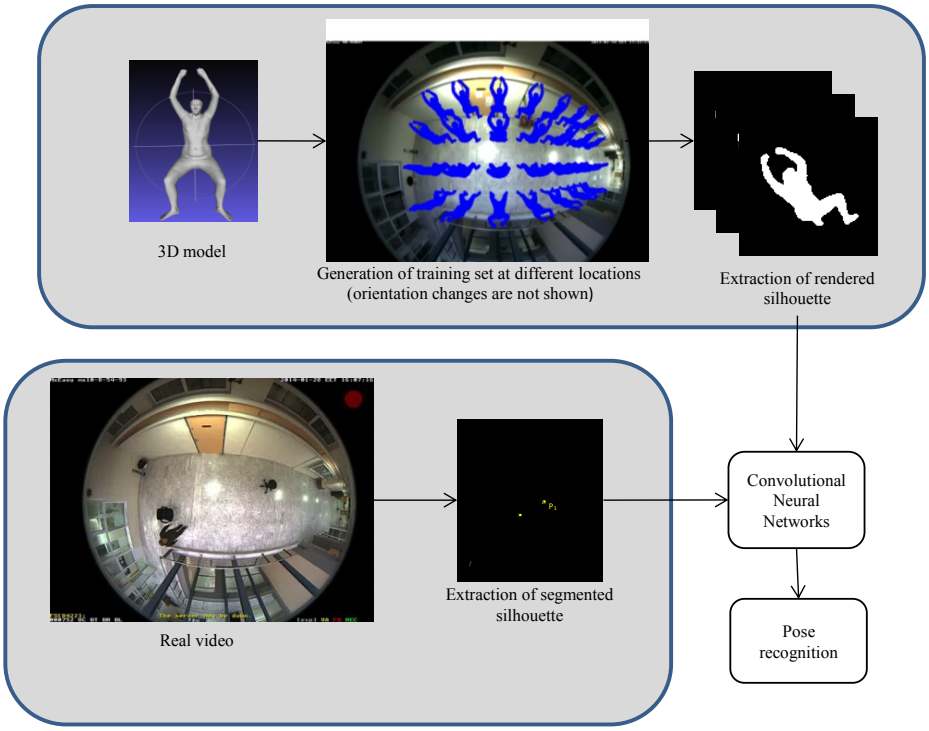

Fig. 1. The steps of the proposed methodology

2.2 Calibration of the fish-eye camera

The fish-eye camera is calibrated using a set of manually provided points, as described in detail in [26]. The achieved calibration compares favourably to other state of the art methodologies [27] in terms of accuracy. In abstract level, the calibration process defines a function F that maps a real world point $\mathbf{x_{real}} = (x_{real}, y_{real}, z_{real})$ to coordinates (i, j) of an image frame:

$$F(\mathbf{x}_{\mathbf{real}}) = (i, j) \tag{1}$$

The resulting calibration is visualized in Fig. 2 for a grid of points virtually placed on the floor and on the walls of the room.
Let B be the binary frame. If we apply the above equation for each $\mathbf{x}_{\mathbf{real}}$ point of a 3D model and set $B(i, j) = 1$, then B will contain the silhouette as imaged by the fish-eye camera.

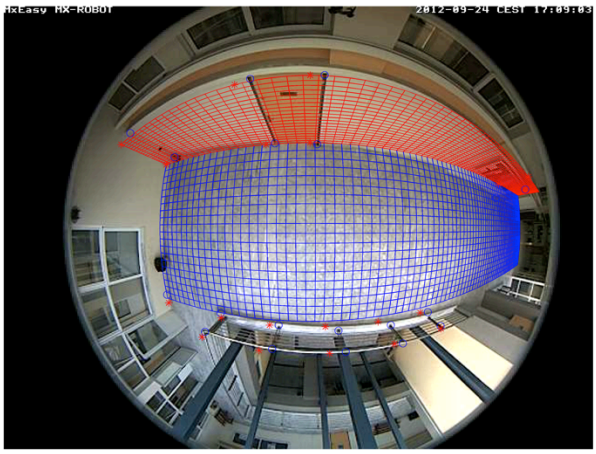

Fig. 2. Visualization of the resulting fish-eye model calibration, on the FoV of the indoor environment in which experiments have taken place. The landmark points defined by the user are shown as circles and their rendered position on the frame marked by stars.

2.3 Synthetically generated silhouettes

In this work, a number of 3D models (Fig. 3), obtained from [24], [25], are utilized. These models have known real-world coordinates stored in the form of triangulated surfaces. However, only the coordinates of the vertices are used for rendering the binary silhouette frames. Rendering of the models and generation of a silhouette in a binary frame B was achieved by using the parameterized camera calibration as following:

$$B\left(F\left(\mathbf{x}_{\mathbf{real},k}\right)\right) = 1 \tag{2}$$

where $\left\{\mathbf{x}_{\mathbf{real},k}, k = 1,2,\ldots,N\right\}$ are the points of a 3D model. Every model was placed in the viewed room at different locations, having different orientations (rotation round the Z-axis).

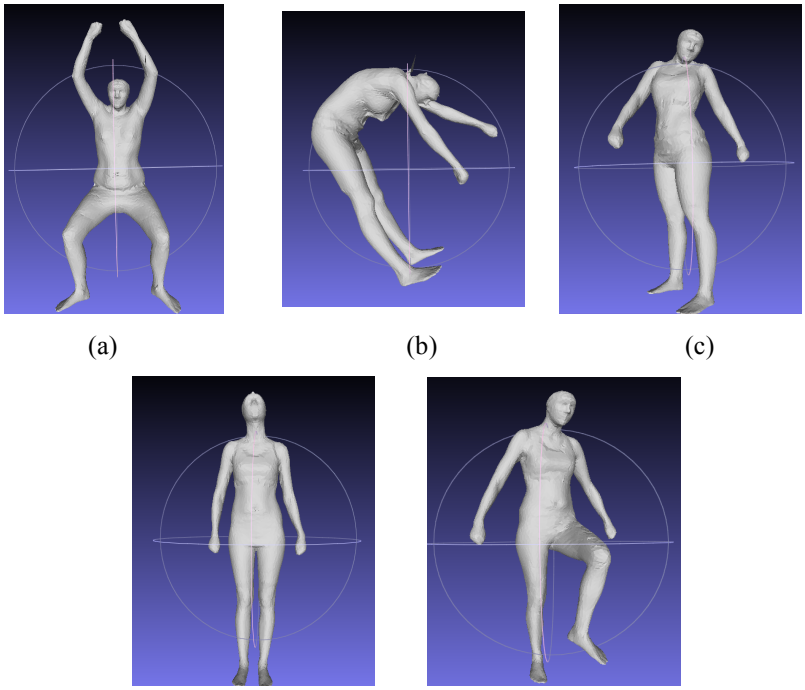

Fig. 3. The 3D human model poses used in this work.

2.4 Convolutional Neural Networks

The CNN was implemented with four convolutional layers; each one of the first three is followed by a max-pooling layer, while the fourth convolutional layer is followed by a fully-connected feed-forward Neural Network with two hidden layers (see Fig. 4). There exist four convolutional layers, each one consists of a n × n, filter, for n=5, 4, 3 and 2 respectively. Pooling filters of a 2 × 2 dimension exists between each two successive convolutional layers. The convolutional layers consist of 16, 16, 32, and 32 filters, respectively, while the max-pooling layers utilize 16, 16, and 32 filters. Finally, the Feed Forward Neural Network consists of one layer with 64 neurons, followed by 20 hidden neurons and one output layer of the generalization of logistic regression function, the softmax functions, where maps the CNN's input into the class probabilities and is commonly used in CNNs. In order to exploit the power of the CNNs that relies on the depth of their layers and at the same time considering the limitations of the GPU memory, we feed the network with binary images sized to 113 × 113 pixels that contain the synthetic silhouettes. The structure of the CNN used in this work is constructed as following: The inputs of the network are binary image

silhouettes. The network consists of four convolutional layers, two fully connected layers and the output layer, which are succeeded by a max-pooling layer. The aforementioned construction is illustrated in Fig. 4.

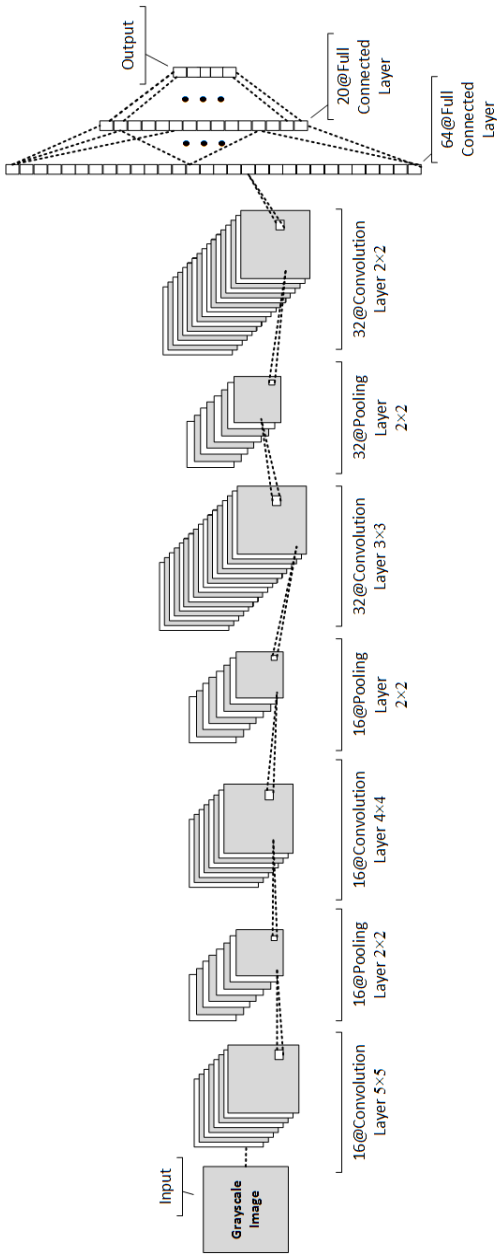

Fig. 4. An illustration of the employed CNN architecture.

3 Results

The generation of an extensive dataset is very important for the successful application of CNNs. We used the dataset that we generated for our previous work involving evaluation of geodetically corrected Zernike moments [1]. Thus, for each one of the 5 poses, the 3D model of each available subject was placed at 13x8 different positions defined on a grid with constant step of 0.5 meters. For each position the model is rotated round the Z-axis every $\pi/5$ radians. Positions with distance less than 1 meter from the trail of the camera were excluded. Silhouettes were rendered using the camera calibration (Subsection 2.2).The total number of data for all 5 poses equals to 32142, as described in Table 1. The 50% of the silhouettes was used as training test and the rest as test set.

Table 1. Dataset for the classification of 5 standing poses.

Pose (Class)	Data
1	7446
2	5096
3	7056
4	6272
5	6272
Sum	32142

The CNN learning algorithm was implemented using the Stochastic Gradient Descent (SGD) with learning rate 0.01 and momentum 0.9 with 30000 iterations on the 50% of the whole dataset and a mini-batch of 50 images. The training of the CNN was performed using the GPU NVIDIA GeForce GTX 970 with 4 GB GPU-RAM and the Convolutional Architecture for Fast Feature Embedding (CAFFE) library [2]. The confusion matrix for the 5 synthetic datasets is shown in Table 2. The achieved accuracy was 98.08 %, which is marginally better than the accuracy obtained with the geodetically corrected Zernike moments (95.31 % as reported in [1]).

Table 2. The confusion matrix achieved for the synthetic data.

	Pose 1	Pose 2	Pose 3	Pose 4	Pose 5
Pose 1	3663	12	17	21	10
Pose 2	17	2490	13	11	17
Pose 3	13	12	3473	17	13
Pose 4	18	17	16	3074	11
Pose 5	13	11	16	10	3086

In order to further test the proposed CNN-based methodology, 4 short video sequences were acquired using the indoor, roof-based fish-eye camera, during which the subjects assumed two generic poses: “standing” and “fallen”. Each frame of the

real video was manually labelled. The training set that was used for training the CNN and the other classifiers in the case of ZMI [1] for the two generic poses (standing and fallen) was obtained as following: The dataset for the generic “standing” pose consists of the union of the 5 standing poses (of Fig. 3). The dataset for the generic "fallen" pose consists of prone/back and side falling, generated from the generic standing pose by interchanging Z-axis with Y-axis coordinate and Z-axis with X-axis, respectively. This dataset (which does not contain any real silhouettes) was used in order to train the CNN and the other classifiers to evaluate the 2 generic poses and recognise them in the real silhouettes. Comparative results with our previous approach (geodetically corrected Zernike moments [1]) are shown in Table 3. It can be seen that in real data, our previously proposed geodesically-corrected ZMI (GZMI) have better discriminating power compared to the CNN. Based on our results so far, it appears that the GZMI are more immune to imperfect segmentation than the CNN. Possible explanations on this observation are suggested in the next section.

Table 3. The classification results for the standing/fallen generic classes.

	Accuracy	Sensitivity	Specificity
real video, CNN	0.6715	0.5281	0.6948
real video, GZMI	0.8173	0.8854	0.7864

The achieved accuracy of the CNN compares favorably with the accuracy achieved using the recently proposed GZMI [1] for different sizes of the training subset. Fig. 5 shows, comparatively, the accuracies achieved by the two methods for training set equal to 10% up to 50% of the total dataset of the synthetically generated poses.

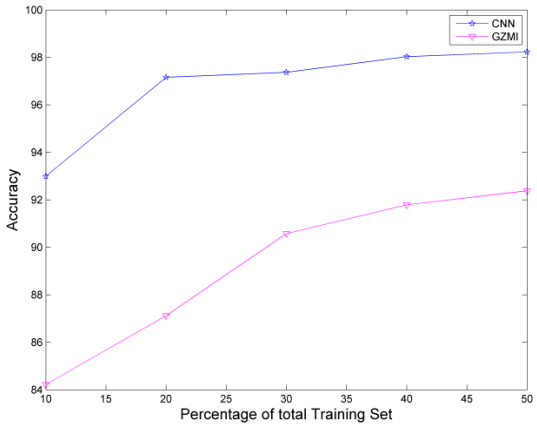

Fig. 5. The accuracy of the test set as a function of the size of the training set comparatively for the two methods. The sizes of the indicated training sets are equal to 10% up to 50% (with a step of 10%) of the total dataset. It can be observed that the accuracy of the proposed CNN (upper blue line) compares favorably to the accuracy of the GZMI (lower magenta line) of [1].

4 Conclusions and further work

The purpose of this work was to assess the ability of convolutional neural networks – CNNs – to correctly classify synthetically generated, as well as real silhouettes, using an extensive synthetic training set. The training set consists exclusively from the synthetic images that are generated from three-dimensional (3D) human models, using a calibrated omni-directional camera (fish-eye). Our results show that the proposed CNN approach is marginally better than the geodesic Zernike moment invariants (GZMI) proposed in our recent work [1], but appears to be outperformed in the problem of real silhouette classification. The GZMI features were adapted from their classic definition, using geodesic distances and angles defined by the camera calibration. On the other hand, the CNN generates features that minimize classification error during the training phase, but do not correspond directly to physical aspects of omni-directional image formation. The results that are reported in this work indicate that the proposed GZMI features appear to be more robust to noise induced by imperfect segmentation, than the features generated by CNN. However, more experimentation is required to draw more definite conclusions. Thus, it is in our future steps to further investigate the structure and the parameters of the utilized CNN in order to improve the performance.

References

1. Delibasis, K. K., Georgakopoulos, S. V., Kottari, K., Plagianakos, V. P., & Maglogiannis, I. (2016). Geodesically-corrected Zernike descriptors for pose recognition in omni-directional images. Integrated Computer-Aided Engineering, (Preprint), 1-15.
2. Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R., Guadarrama, S., & Darrell, T. (2014, November). Caffe: Convolutional architecture for fast feature embedding. In Proceedings of the ACM International Conference on Multimedia (pp. 675-678). ACM.
3. Oquab, M., Bottou, L., Laptev, I., & Sivic, J. (2015). Is object localization for free?-weakly-supervised learning with convolutional neural networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 685-694).
4. Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In Advances in neural information processing systems (pp. 1097-1105).
5. Cheng, J. Z., Ni, D., Chou, Y. H., Qin, J., Tiu, C. M., Chang, Y. C., Huang, C. S., Shen, D., & Chen, C. M. (2016). Computer-Aided Diagnosis with Deep Learning Architecture: Applications to Breast Lesions in US Images and Pulmonary Nodules in CT Scans. Scientific reports, 6.
6. Li, Q., Cai, W., Wang, X., Zhou, Y., Feng, D. D., & Chen, M. (2014, December). Medical image classification with convolutional neural network. In Control Automation Robotics & Vision (ICARCV), 2014 13th International Conference on (pp. 844-848). IEEE.
7. Suk, H. I., Lee, S. W., Shen, D., & Alzheimer's Disease Neuroimaging Initiative. (2014). Hierarchical feature representation and multimodal fusion with deep learning for AD/MCI diagnosis. NeuroImage, 101, 569-582.

8. Junior, O. L., Delgado, D., Gonçalves, V., & Nunes, U. (2009, October). Trainable classifier-fusion schemes: an application to pedestrian detection. In Intelligent Transportation Systems (Vol. 2).
9. Tamimi, H., Andreasson, H., Treptow, A., Duckett, T., & Zell, A. (2006). Localization of mobile robots with omnidirectional vision using particle filter and iterative sift. Robotics and Autonomous Systems, 54(9), 758-765.
10. Hwang, S. K., Billinghurst, M., & Kim, W. Y. (2008, May). Local descriptor by zernike moments for real-time keypoint matching. In Image and Signal Processing, 2008. CISP'08. Congress on (Vol. 2, pp. 781-785). IEEE.
11. Zhu, H., Shu, H., Xia, T., Luo, L., & Coatrieux, J. L. (2007). Translation and scale invariants of Tchebichef moments. Pattern Recognition, 40(9), 2530-2542.
12. Shutler, J. D., & Nixon, M. S. (2001, September). Zernike Velocity Moments for Description and Recognition of Moving Shapes. In BMVC (pp. 1-10).
13. Yang, Y., & Ramanan, D. (2011, June). Articulated pose estimation with flexible mixtures-of-parts. In Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on (pp. 1385-1392). IEEE.
14. Kottari, K., Delibasis, K., Plagianakos, V., & Maglogiannis, I. (2014). Fish-eye camera video processing and trajectory estimation using 3d human models. In Artificial Intelligence Applications and Innovations (pp. 385-394). Springer Berlin Heidelberg.
15. LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. Proceedings of the IEEE, 86(11), 2278-2324.
16. Bottou, L. (2012). Stochastic gradient descent tricks. In Neural Networks: Tricks of the Trade (pp. 421-436). Springer Berlin Heidelberg.
17. Toshev, A., & Szegedy, C. (2014). Deeppose: Human pose estimation via deep neural networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 1653-1660).
18. Kemmotsu, K., Tomonaka, T., Shiotani, S., Koketsu, Y., Iehara, M.: Recognizing human behaviors with vision sensors in a Network Robot System. In: Proceedings of IEEE Int. Conf. on Robotics and Automation, pp. l274-1279. (2006).
19. Zhou, Z., Chen, X., Chung, Y., He, Z., Han, T. X. Keller, M.: Activity Analysis, Summarization and Visualization for Indoor Human Activity Monitoring. IEEE Trans. on Circuit and systems for Video Technology 18(II), 1489-1498 (2008).
20. Mei, C., Rives, P.: Single View Point Omnidirectional Camera Calibration from Planar Grids. In: IEEE International Conference on Robotics and Automation, pp.3945-3950, (ICRA), Rome, Italy, IEEE, (2007).
21. Li, H., Hartley, R.: Plane-Based Calibration and Auto-calibration of a Fish-Eye Camera. In P.J. Narayanan et al. (Eds.): ACCV 2006, LNCS 3851, pp. 21–30, Springer-Verlag Berlin Heidelberg (2006).
22. Shah, S., Aggarwal, J.: Intrinsic parameter calibration procedure for a high distortion fish-eye lens camera with distortion model and accuracy estimation. Pattern Recognition 29(11), 1775-1788 (1996).
23. Wei, J., Li, C. F., Hu, S. M., Martin, R. R., & Tai, C. L. (2012). Fisheye video correction. Visualization and Computer Graphics, IEEE Transactions on, 18(10), 1771-1783.
24. Hasler, N., Ackermann H., Rosenhahn B., Thormahlen T. Seidel H.P.: Multilinear pose and body shape estimation of dressed subjects from image sets. In: IEEE Conf. on Computer Vision and Pattern Recognition (CVPR 2010), pp. 1823–1830. (2010).
25. http://resources.mpi-inf.mpg.de/scandb/

26. Delibasis, K., Plagianakos V., Maglogiannis, I.: Refinement of human silhouette segmentation in omni-directional indoor videos. Computer Vision and Image Understanding 128, 65-83 (2014).
27. Rufli, M., Scaramuzza, D., Siegwart, R.: Automatic Detection of Checkerboards on Blurred and Distorted Images. In: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2008), pp. 3121-3126. Nice, France, (2008).