# 3D Skeletal reconstruction from low-resolution multi-view images

Mayank Rana Graham Taylor Ian Spiro
Christoph Bregler
Courant Institute of Mathematical Sciences, New York University
New York, USA
{mayank,graham,ian,chris}@movement.nyu.edu

## Abstract

*This paper demonstrates how 3D skeletal reconstruction can be performed by using a pose-sensitive embedding technique applied to multi-view video recordings. We apply our approach to challenging low-resolution video sequences. Usually skeletal reconstruction can be only achieved with many calibrated high-resolution cameras, and only blob detection can be achieved with such low-resolution imagery. We show that with this embedding technique (a metric learning technique using a deep convolutional architecture), we achieve very good 3D skeletal reconstruction on low-resolution outdoor scenes with many challenges.*

## 1. Introduction

Human pose reconstruction and human activity recognition has been a very active field over the past decades. Techniques range from high-resolution multi-camera inputs such as systems that use 12 cameras in uncluttered green screen environments [38] or those that work on the HumanEVA indoor lab dataset [35] to low-resolution scenarios that can detect pedestrians or find approximate body part areas in single views [14, 5]. A more detailed overview of related approaches is provided in section 2.

Our system differs from these two extremes: it uses the multi-camera approach, but instead of using 12 high-resolution cameras, we use only 3 views and resolutions comparable to the imagery of the low-resolution systems previously mentioned. High-resolution multi-camera systems can use a kinematic body model, but low-resolution systems do not work with kinematic chains. The number of degrees of freedom is not in balance with the lower pixel resolution. Our system follows a different approach: It uses a metric learning technique that recently has been successfully applied to very challenging cluttered domains for upper body 2D pose matching [40]. We demonstrate how this technique can be extended to a 3D skeletal model, and how it can be applied to low-resolution camera views that are recorded from two cameras on the 12th floor in a city building facing down to a busy side walk, and from another camera on the 2nd floor. The system is first trained with a crowd-sourced technique using amazon mechanical turk, and then used for 3D reconstruction, using a hybrid of the learned pose-embedding technique and a 3D structure from motion technique. No pre-calibration of the cameras and no prior skeletal model is necessary. We demonstrate surprisingly good 3D skeletal motion reconstruction given the challenges of this outdoor low-resolution domain.

## 2. Related Work

Activity in human pose estimation has a long tradition in computer vision reaching back to milestone papers in model-based approaches over the past three decades [27, 19, 46, 15, 30, 29, 6, 22, 12, 34, 36] and reaching a paper count of over 350 significant contributions between 2001 and 2006 [25]. Most recently, 3D pose estimation has reached new heights through the use of much better cameras, many more cameras, higher resolution mesh models [23, 8, 11, 38], human motion priors both machine learning [41, 47] and physics-based [7, 44], and through the availability of the HumanEVA database [35].

All of these techniques mainly work with many cameras in a laboratory setting. Approaches that work on arbitrary outdoor footage in cluttered environments have emerged over the past decade, not by using a kinematic model, but by using extensive training data, or new features (learned or hand coded) [26, 9, 1, 18, 14, 5, 2, 31, 32, 28, 13].

More closely related to the method proposed in this paper are nearest-neighbour and locally-weighted regression-based techniques. One family of techniques has performed 2D pose estimation in the absence of a kinematic model by learning pose-sensitive hash functions [33, 20]. These approaches use edge histogram features similar to HOG [10]. They rely on good background subtraction or recordings with clear backgrounds. Our domain contains clutter, lighting variations and low resolution such that it is impossible to separate body features from background successfully. We

Figure 1. Example views from our the cameras facing the sidewalk. We used three cameras programmed with different aspect ratios to attain the best coverage of the sidewalk. Two overhead cameras faced out the window of our 12th floor lab, and one camera was mounted 12 feet high on the side of the sidewalk.

instead learn relevant features directly from pixels (instead of pre-coded edge or gradient histogram features), and learn background invariance from training data.

## 3. Our Approach

Our system can be positioned as a solution that lies somewhere between a high-end indoor studio capture setup, and a single video surveillance camera. In the past, we applied a similar method to very low-resolution single view videos. In this paper, we show how we can perform 3D skeletal tracking of people in an outdoor scene (a crowded walkway in the downtown area of a bigger city), using 3 camera views that are placed at distance. Figure 1 shows several example frames from our dataset.

### 3.1. Preprocessing

We first located people on the scene by running on the overhead views an adaptive background subtraction technique (figure 2) followed by a graph matching-based blob tracking [45, 48] technique. This technique tracks very robustly objects from aerial views [45], and in our case, people who are close to a pure overhead view. Side views are more difficult for background subtraction, especially with traffic in the background and significant occlusion between pedestrians. Knowing the camera angles and the ground plane allows us to estimate regions of interest (ROI) for all 3 views, starting with the blob estimation from the overhead view, and reconstructing the same position in other views.

Once we have calculated the region of interest corresponding to potential people, we normalize all cropped ROIs to 128 $\times$ 128 pixels. Frequently the original resolution of the cropped region is less than 128 $\times$ 128, in which

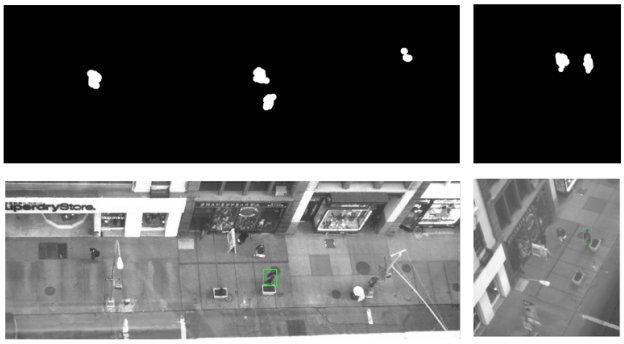

Figure 2. Example background subtraction and ROI tracking.

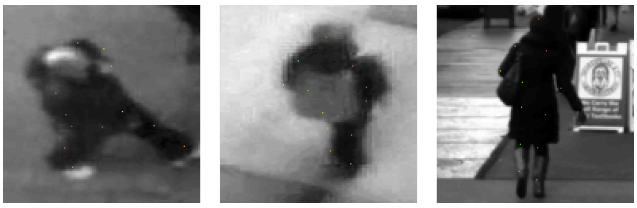

Figure 3. Example images of pedestrians.

case we upsample (e.g. the overhead camera). In other cases (e.g. from the sidewalk camera 12 feet high) it is down sampled. Figure 3 shows a representative example set for all views.

### 3.2. Pose Estimation Architecture

Once we have estimated the normalized ROI of the person, we learn a mapping from the 128 $\times$ 128 pixel domain into a latent feature space that captures 2D or 3D pose configuration. In this specific example, the pose is defined by 13 points at body joints, head, hands, and feet position. In the 2D domain it is a 26 dimensional vector, and in the 3D domain it is a 39 dimensional vector. The mapping is not made directly into the pose domain. Instead we learn a mapping into a 32 dimensional "latent space" or metric space that defines "closeness" to be images that contain people in "similar" pose, but potentially quite different in appearance. The technique is similar to recent hashing techniques, that maintain a large database of images and aim to match novel "query" images to semantically similar images in the database. The non-linear embedding that we learn is based on several extensions to the Neighborhood Component Analysis (NCA) framework [16]. Our method is convolutional, enabling it to scale to realistically-sized images. By cheaply labeling a large set of example images with different poses, we can afford to generate large example databases very quickly for new domains, like this specific set of cameras pointing at a specific corner of the street. In the case of 2D body pose, we have already demonstrated

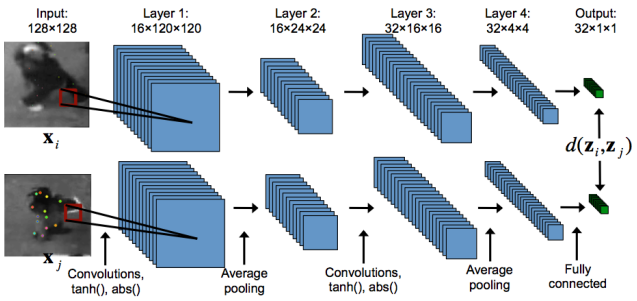


Figure 4. Pose Sensitive Embedding.

the success of this method in many different domains, including hand gesture analysis of lecturers during a conference [40] and learning the poses of music fans [42].

Our pose-embedding technique which we call Pose Sensitive Embedding (PSE) is based on a standard convolutional architecture [24, 21]: alternating local contrast normalization, convolution, and subsampling layers followed by a single fully-connected layer (see Fig. 4). The "deep learning" community has argued that low-level and mid-level features in such an architecture achieve similar or superior performance to other established features, such as HOG and SIFT and can automatically adapt to different domains [4]. Our architecture differs from typical convolutional nets in the objective function with which it is trained (i.e. minimizing a loss function). Because the loss is defined on pairs of examples, we use a siamese network [3] whereby pairs of frames are processed by separate networks with shared weights. The loss is then computed on the output of both networks. If two images are similar in terms of their pose vectors, then we aim to produce outputs that are close in a Euclidean sense. If the images are not similar in their pose vectors, then we aim to produce outputs that are far apart.

Figure 4 shows such a network. Images are preprocessed using LCN (Local Contrast Normalization). Convolutions are followed by pixel-wise $\tanh$ and absolute value rectification. The abs prevents cancellations in local neighbourhoods during average downsampling [21]. Our architectural parameters (size of filters, number of filter banks, etc.) are chosen to produce a 32-dimensional output.[1]

The loss function that is minimized in this network is based on a "soft" notion of similarity in pose-space of 2 sets of pose vectors $y_i$ and $y_j$ coding the joint locations:

$$\hat{\gamma}_{ij} = \frac{\exp(-||\mathbf{y}_i - \mathbf{y}_j||_2^2)}{\sum_{k \neq i} \exp(-||\mathbf{y}_i - \mathbf{y}_j||_2^2)}. \quad (1)$$

[1] We have found empirically that the performance of our method is not overly sensitive to the choice of latent dimensions. 32D not only works reasonably well but is well-suited for GPU implementations. See [40] for further discussion.

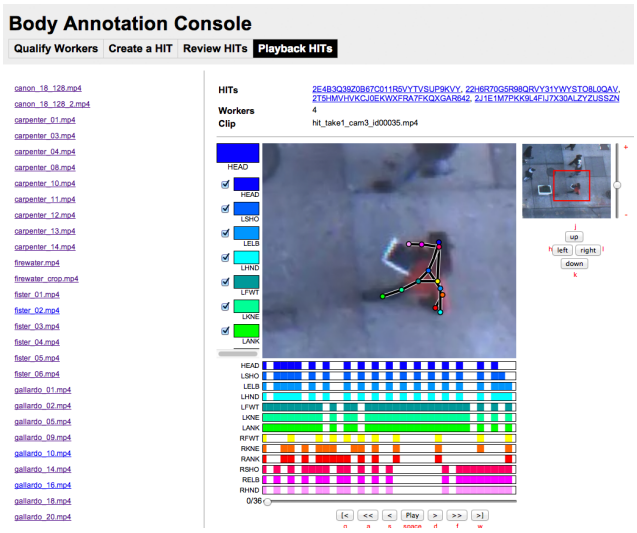


Figure 5. Example interface for amazon mechanical turk.

Learning is performed by back-propagating the error through the remaining layers of the network for all possible pairs of training images, or randomly selected pairs. In [40] we have shown that these techniques show improved performance over other embedding techniques using a publicly available database [17].

### 3.3. Labeling the Training Set

Our nearest neighbor technique relies on having enough training data to learn a "pose-sensitive" mapping. We heavily depend on crowd-sourced labeling of training data for many domains. Instead of building a generic human pose estimator, we follow the strategy that for each new domain (lectures, music fans, basketball stadium, or in this case pedestrians), we quickly and cheaply collect training data with an Amazon Mechanical Turk (AMT) interface we previously developed [37]. This allows us, in some cases, near real-time data collection. For instance, several times during conferences or workshops, we could collect and analyze the gestures of speakers prior to our slot and then demonstrate the results as part of our presentation. Recently it was exposed very explicitly that every database has a strong bias [43]. This includes large, popular collections like PASCAL or ImageNet. We subscribe to this philosophy and, in response, build very biased datasets for each new domain quickly and cheaply.

Figure 5 shows our interface for mechanical turk crowd sourcing applied to the domain of this paper. All results in this paper are based on collecting only 40 videos of 60 frames each (2,400 poses) for the total amount of 80 USD. Each HIT (human intelligence task) asks a worker to label 13 joint locations on one view in a region of interest. We asked 3 online workers to work on the same task for redun-

dancy and error correction.

### 3.4. Training Pose Embedding

The $2,400$ images with pose annotations where split into $75\%$ training data and $25\%$ test data. For both the training data and the test data we enlarged the number of images by a factor of 16 (total of $38,400$ image / pose label pairs) in warping the images by random offsets up to $20\%$ . This is a common heuristic to increase generalization performance, and make the pose embedding more robust to inaccurate tracking estimates.

Figure 6 shows several examples of unseen input images fed into the embedding, and it's 16 nearest neighbors from the training data, including the labeled pose annotations. The average pixel error for pose using the nearest neighbor estimate is 2.13 in $128 \times 128$ pixel image resolution ($1.6\%$ error). Using the top 2 nearest neighbors, the pixel error increases to 2.23.

### 3.5. 3D Pose Estimation

We are currently investigating several options how to compute 3D pose estimation:

- *Algo-1*: Compute just the nearest neighbor for each camera angle and reconstruct in 3D the skeleton in applying a standard structure from motion technique [39] on the entire space-time data of the predicted multi-view 2D poses.

- *Algo-2*: 3D reconstruct the training data first with the same technique [39] and use as input/label pairs the multi-view image and its 3D reconstruction.

At the time of this workshop submission, we only have good results for *Algo-1*, as seen in figure 7.

## 4. Conclusions

We have presented a method for 3D skeletal reconstruction from low-resolution, noisy, inexpensive cameras. Our technique falls somewhere in-between an expensive high-performance indoor multi-camera setup and a flexible but inadequate single-camera configuration. Paramount to our approach is a data-driven non-parametric nearest neighbor algorithm that relies on quickly crowd-sourcing domain-specific datasets via Amazon Mechanical Turk. Our results show that we can obtain accurate 3D body pose in unstructured and challenging environments. Here we have focused on the problem of analyzing pedestrians on a crowded and busy city street. Future work will examine at the active learning scenario, specifically exploiting AMT and the fact that our learned embedding can be incrementally updated.

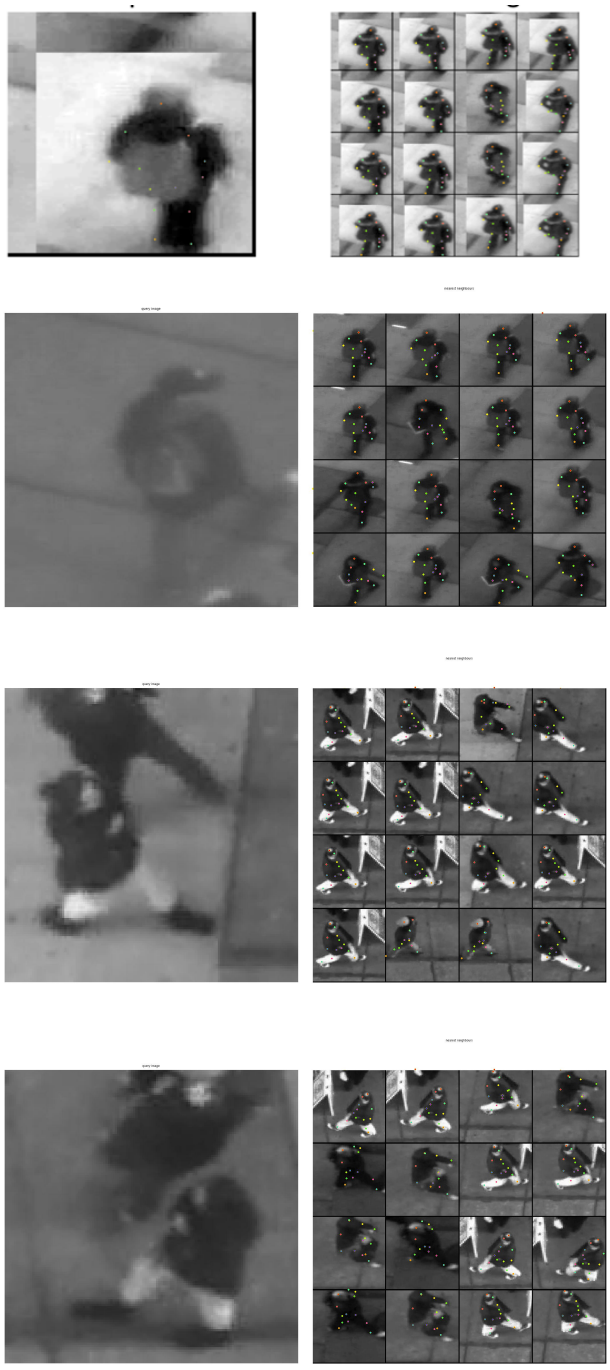

Figure 6. Examples of 16 nearest neighbors in the learned embedded space for a selection of unseen input images.

## 5. Acknowledgements.

This research has been partially funded by the US Office of Naval Research Grant N000141210327 and the US Air Force Contract FA8650-11-C-6227 via a subcontract

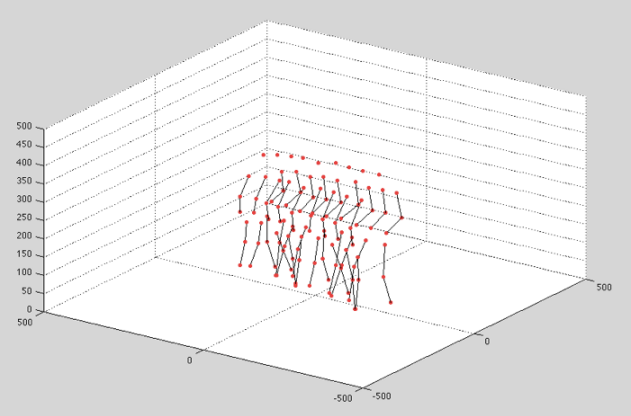


Figure 7. An example walk cycle of a person using our multi-view pose embedding and *Algo-1* to 3D reconstruct the pose sequence.

from ObjectVideo. We also like to thank Atul Kanaujia and Thomas Huston for their help in this research.

## References

[1] A. Agarwal and B. Triggs. 3d human pose from silhouettes by relevance vector regression. In *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*, volume 2, pages II–882. IEEE, 2004. 1

[2] M. Andriluka, S. Roth, and B. Schiele. Monocular 3d pose estimation and tracking by detection. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 623–630. IEEE, 2010. 1

[3] S. Becker and G. Hinton. Self-organizing neural network that discovers surfaces in random-dot stereograms. *Nature*, 355(6356):161–163, 1992. 3

[4] Y. Bengio. Learning deep architectures for AI. *Foundations & Trends in Mach. Learn.*, 2(1):1–127, 2009. 3

[5] L. Bourdev and J. Malik. Poselets: Body part detectors trained using 3d human pose annotations. In *Computer Vision, 2009 IEEE 12th International Conference on*, pages 1365–1372. IEEE, 2009. 1

[6] C. Bregler and J. Malik. Tracking people with twists and exponential maps. In *Computer Vision and Pattern Recognition, 1998. Proceedings. 1998 IEEE Computer Society Conference on*, pages 8–15. IEEE, 1998. 1

[7] M. A. Brubaker, D. J. Fleet, and A. Hertzmann. Physics-Based Person Tracking Using the Anthropomorphic Walker. *IJCV*, 87(1):140–155, 2010. 1

[8] S. Corazza, L. Mündermann, E. Gambaretto, G. Ferrigno, and T. Andriacchi. Markerless motion capture through visual hull, articulated icp and subject specific model generation. *International journal of computer vision*, 87(1):156–169, 2010. 1

[9] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 1, pages 886–893. IEEE, 2005. 1

[10] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *Proc. International Conference on Computer Vision and Pattern Recognition*, pages 886–893, 2005. 1

[11] E. De Aguiar, C. Stoll, C. Theobalt, N. Ahmed, H. Seidel, and S. Thrun. Performance capture from sparse multi-view video. In *ACM Transactions on Graphics (TOG)*, volume 27, page 98. ACM, 2008. 1

[12] J. Deutscher, A. Blake, and I. Reid. Articulated body motion capture by annealed particle filtering. In *Computer Vision and Pattern Recognition, 2000. Proceedings. IEEE Conference on*, volume 2, pages 126–133. IEEE, 2000. 1

[13] P. Dollár, C. Wojek, B. Schiele, and P. Perona. Pedestrian detection: A benchmark. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 304–311. Ieee, 2009. 1

[14] P. Felzenszwalb, D. McAllester, and D. Ramanan. A discriminatively trained, multiscale, deformable part model. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–8. Ieee, 2008. 1

[15] D. Gavrila and L. Davis. 3-d model-based tracking of humans in action: a multi-view approach. In *Computer Vision and Pattern Recognition, 1996. Proceedings CVPR'96, 1996 IEEE Computer Society Conference on*, pages 73–80. IEEE, 1996. 1

[16] J. Goldberger, S. Roweis, G. Hinton, and R. Salakhutdinov. Neighbourhood components analysis. In *NIPS*, 2004. 2

[17] G. W. Graham Taylor, Ian Spiro and C. Bregler. The snowbird dataset (v. 2011-05-24). Downloaded from http://movement.nyu.edu/snowbird/, Mar. 2012. 3

[18] K. Grauman and T. Darrell. The pyramid match kernel: Efficient learning with sets of features. *The Journal of Machine Learning Research*, 8:725–760, 2007. 1

[19] D. Hogg. Model-based vision: a program to see a walking person. *Image and vision computing*, 1(1):5–20, 1983. 1

[20] P. Jain, B. Kulis, and K. Grauman. Fast image search for learned metrics. In *CVPR*, Los Alamitos, CA, USA, 2008. 1

[21] K. Jarrett, K. Kavukcuoglu, M. Ranzato, and Y. LeCun. What is the best multi-stage architecture for object recognition? In *ICCV*, 2009. 3

[22] L. Kakadiaris and D. Metaxas. Model-based estimation of 3d human motion. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 22(12):1453–1459, 2000. 1

[23] A. Kanaujia, C. Sminchisescu, and D. Metaxas. Semi-supervised hierarchical models for 3d human pose reconstruction. In *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*, pages 1–8. IEEE, 2007. 1

[24] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proc. IEEE*, 86(11):2278–2324, 1998. 3

[25] T. Moeslund, A. Hilton, and V. Krüger. A survey of advances in vision-based human motion capture and analysis. *Computer vision and image understanding*, 104(2):90–126, 2006. 1

[26] G. Mori and J. Malik. Estimating human body configurations using shape context matching. *Computer VisionECCV 2002*, pages 150–180, 2002. 1

[27] J. O'Rourke and N. Badler. Model-based image analysis of human motion using constraint propagation. *IEEE TRANS. PATTERN ANALY. AND MACH. INTELLIG.*, 2(6):522–536, 1980. 1

[28] D. Ramanan, D. Forsyth, and A. Zisserman. Strike a pose: Tracking people by finding stylized poses. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 1, pages 271–278. IEEE, 2005. 1

[29] J. Rehg and T. Kanade. Model-based tracking of self-occluding articulated objects. In *Computer Vision, 1995. Proceedings., Fifth International Conference on*, pages 612–617. IEEE, 1995. 1

[30] K. Rohr. Towards model-based recognition of human movements in image sequences. *CVGIP-Image Understanding*, 59(1):94–115, 1994. 1

[31] P. Sabzmeydani and G. Mori. Detecting pedestrians by learning shapelet features. In *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*, pages 1–8. IEEE, 2007. 1

[32] E. Seemann, M. Fritz, and B. Schiele. Towards robust pedestrian detection in crowded image sequences. In *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*, pages 1–8. IEEE, 2007. 1

[33] G. Shakhnarovich, P. Viola, and T. Darrell. Fast pose estimation with parameter-sensitive hashing. In *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*, pages 750–757. IEEE, 2003. 1

[34] H. Sidenbladh, M. Black, and D. Fleet. Stochastic tracking of 3d human figures using 2d image motion. *Computer VisionECCV 2000*, pages 702–718, 2000. 1

[35] L. Sigal, A. Balan, and M. Black. 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, 2010. 1

[36] C. Sminchisescu and B. Triggs. Covariance scaled sampling for monocular 3d body tracking. In *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, volume 1, pages I–447. IEEE, 2001. 1

[37] I. Spiro, G. Taylor, G. Williams, and C. Bregler. Hands by hand: Crowd-sourced motion tracking for gesture annotation. In *IEEE CVPR Workshop on Advancing Computer Vision with Humans in the Loop (ACVHL)*, 2010. 3

[38] C. Stoll, N. Hasler, J. Gall, H. Seidel, and C. Theobalt. Fast articulated motion tracking using a sums of gaussians body model. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pages 951–958. IEEE, 2011. 1

[39] P. Sturm and B. Triggs. A factorization based algorithm for multi-image projective structure and motion. *Computer VisionECCV'96*, pages 709–720, 1996. 4

[40] G. Taylor, R. Fergus, G. Williams, I. Spiro, and C. Bregler. Pose-sensitive embedding by nonlinear nca regression. In *Advances in Neural Information Processing Systems 23*, pages 2280–2288. 2010. 1, 3

[41] G. Taylor, L. Sigal, D. Fleet, and G. Hinton. Dynamical binary latent variable models for 3d human pose tracking. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 631–638. IEEE, 2010. 1

[42] G. W. Taylor, I. Spiro, C. Bregler, and R. Fergus. Learning invariance through imitation. *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on*, pages 2729–2736, 2011. 3

[43] A. Torralba and A. Efros. Unbiased look at dataset bias. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 1521–1528. IEEE, 2011. 3

[44] M. Vondrak, L. Sigal, and O. C. Jenkins. Dynamical simulation priors for human motion tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 99(PrePrints), 2012. 1

[45] J. Xiao, H. Cheng, H. Sawhney, and F. Han. Vehicle detection and tracking in wide field-of-view aerial video. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 679–684. IEEE, 2010. 2

[46] M. Yamamoto and K. Koshikawa. Human motion analysis based on a robot arm model. In *Computer Vision and Pattern Recognition, 1991. Proceedings CVPR'91., IEEE Computer Society Conference on*, pages 664–665. IEEE, 1991. 1

[47] A. Yao, J. Gall, L. V. Gool, and R. Urtasun. Learning probabilistic non-linear latent variable models for tracking complex activities. In J. Shawe-Taylor, R. Zemel, P. Bartlett, F. Pereira, and K. Weinberger, editors, *Advances in Neural Information Processing Systems 24*, pages 1359–1367. NIPS Society, 2011. 1

[48] R. Zass and A. Shashua. Probabilistic graph and hypergraph matching. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–8. Ieee, 2008. 2