Vision, Modeling, and Visualization (2011)
Peter Eisert, Konrad Polthier, and Joachim Hornegger (Eds.)

Markerless Motion Capture using multiple Color-Depth Sensors

Kai Berger[†], Kai Ruhl[‡], Yannic Schroeder[§], Christian Bruemmer[¶], Alexander Scholz[‖], & Marcus Magnor[**]

Abstract

With the advent of the Microsoft Kinect, renewed focus has been put on monocular depth-based motion capturing. However, this approach is limited in that an actor has to move facing the camera. Due to the active light nature of the sensor, no more than one device has been used for motion capturing so far. In effect, any pose estimation must fail for poses occluded to the depth camera.

Our work investigates on reducing or mitigating the detrimental effects of multiple active light emitters, thereby allowing motion capture from all angles. We systematically evaluate the concurrent use of one to four Kinects, including calibration, error measures and analysis, and present a time-multiplexing approach.

Categories and Subject Descriptors (according to ACM CCS): I.4.8 [Image Processing and Computer Vision]: Scene Analysis—Range data

1. Introduction

In typical markerless motion capturing scenarios, the actor is placed inside a greenroom and captured with several time-synchronized stationary cameras. A geometric proxy or an articulated skeleton model then is fitted to the captured data, e.g. the projected silhouettes of the actor. In recent years, several new approaches to markerless motion capture have been presented, for example the capturing of human motions with unsynchronized non-stationary camera setups [HRT*09]. With the advent of the Microsoft Kinect in 2010 [SFC*11], motion capturing based on depth data, previously predominantly performed with time-of-flight cameras, has become more feasible, allowing consumer-grade depth-based monocular motion capturing: A monocular depth sensor provides cues for decision forests [SFC*11] which infer a pose of the captured actor. However, monocular motion capturing constrains the actor to perform his movements with respect to the camera. The poses of partially invisible body parts have to approximated with decision forests and a database covering a huge set of poses and shapes. We seek to combine purely silhouette-based multi-camera motion capturing with the active depth sensoring of the Kinect. As the Kinect's depth sensing is based on the emission of an infrared pattern, previous approaches are limited to only use one Kinect for motion capturing because of interference errors. In our work we will reinvestigate the magnitude of interferences in a motion capturing setup with multiple Kinects. We show its suitability for capturing the movement of an actor without the use of a database covering a huge set of poses and shapes.

The merit of this paper can thus be summarized as follows:

- Two new versatile methods for simultaneously calibrating multiple depth and RGB sensors
- An investigation of interference errors for varying number of Kinects in a motion capturing setup featuring different materials
- A hardware solution for time-multiplexing up to four Kinects in order to mitigate interference errors
- A motion capturing method purely based on the depth images of multiple calibrated depth sensors

Note, that we are focusing on motion capturing with depth data only. RGB data is only used for verification.

[†] TU Braunschweig e-mail: berger@cg.cs.tu-bs.de
[‡] TU Braunschweig e-mail: ruhl@cg.cs.tu-bs.de
[§] TU Braunschweig e-mail: schroeder@cg.cs.tu-bs.de
[¶] TU Braunschweig e-mail: bruemmer@cg.cs.tu-bs.de
[‖] TU Braunschweig e-mail: scholz@cg.cs.tu-bs.de
[**] TU Braunschweig e-mail: magnor@cg.cs.tu-bs.de

DOI: 10.2312/PE/VMV/VMV11/317-324

2. Related work

Motion Capturing

While a good overview of the efforts in motion capturing can be found in the work of Moeslund et al. [MHK06], we will mainly focus on Carranza et al. [CTMS03]. They introduced motion capturing based on matching the silhouette of a rendered model to the silhouettes of the captured model, thus reducing pose estimation to solving an optimization problem. Subsequent approaches used deformable meshes [DATSS07] or laser scans [DAST*08]. The latter uses the scanned data only as initialization at the very beginning of recording. While these motion capturing approaches remain purely image-based and thus passive, much research has also gone into active sensors, e.g. time-of-flight sensors. A good overview can be found in a survey by Kolb et al. [KBKL09]. Until recently, active devices have been considered to be specialized and expensive equipment, restricting most experiments to only one device. Although structure-from-motion approaches exist [BKWK07, KBK07], they are restricted to static scenes and are thus unsuitable for a motion capture scenario. With more affordable depth sensors such as the Microsoft Kinect [Mic10], the use of more than one active light sensor becomes a more attractive consideration.

Calibration

The most prominent way to calibrate multiple RGB cameras to a common reference frame is by processing images of a captured checkerboard. A popular approach has been introduced by Bouguet [Bou10], as it provides for a comfortable GUI to find the checkerboard's corner points. Svoboda [SMP05] proposed to solve for point correspondences that are retrieved when capturing a moving point light source in a dark room over time. The captured light positions in each camera frame define the point correspondence for a time instant. Another approach has been proposed by Snavely et al. [SSS06]: A scene containing a multitude of feature locations is captured with multiple cameras. The positions of the captured features define the point correspondences between the cameras and are iteratively optimized.

As the Kinect combines a passive RGB and an active structured light sensor, simultaneous calibration has the same challenges as previous fusion approaches, as e.g. by Gudmundsson et al. or Huhle et al. [GLA*08, HJS08]. However, they use a fixed rig, greatly easing computation of the extrinsic and intrinsic camera parameters. Other approaches use a single-lens device [IY01], where all calibration information is already known at manufacturing time. As we place Kinects in a multi-view setup, the challenge is to engineer methods that simultaneously calibrate RGB and depth sensors using a suitable calibration pattern. On the internet, several interesting approaches have been proposed, ranging from finding texture differences in the IR sensor with an occluded emitter [Eng11] to treating the checkerboard printout as a planar surface [Bur10]. Another approach uses 3D-printouts of cuboids to provide distinguishable values in the depth image [Gaf11]. However, these solutions remain ad-hoc and have not been proven in setups with multiple Kinects. Instead, we found that the simultaneous calibration of a passive and an active sensor can be solved by employing materials with different BRDFs.

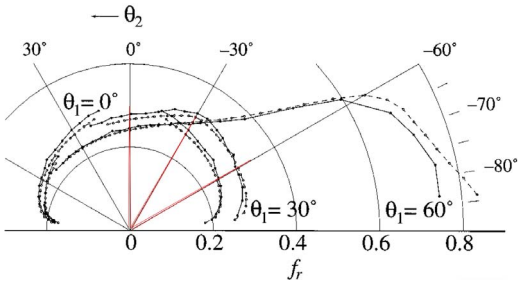

Figure 1: *We provide a distinguishable calibration pattern by using a combination of paper sheet and mirroring foil, as their BRDFs show a clearly distinguishable behavior: the paper diffusely reflects the light (black connected dots; diffuse except for grazing angles), while mirroring foil has a peak at the reflection angle only (red lines) for varying input angles* $\theta_i \in (0^\circ, 30^\circ, 60^\circ)$. *Reproduced from [MWLT00].*

Our solution is to have one material that deflects the emitted light of the depth sensor, thus making it invisible, and one material that diffusely reflects it.

Multi-View Depth

In order to increase the spatial resolution of depth sensors, the use of multiple devices becomes a viable alternative. Wilson et.al. [WB10] use multiple depth sensors to monitor a room, but their setup ensures that the active light does not overlap. Other approaches use different modulation frequencies per camera [KCTT08, GFP08]. A similar approach with the Kinect's structured light sensor would not be technically feasible and was not investigated. In our work, strongly overlapping regions are recorded while successively adding more depth sensors. We investigate it by providing an error analysis for different materials and varying the number of depth sensors, in scenarios featuring both simultaneously and alternatingly emitting Kinects.

3. Setup and Calibration

We conduct our studies in a green room measuring $3m \times 3m \times 2.5m$. At each corner, we placed a Microsoft Kinect at $2.5m$ height and rotated it to focus on a spot in the center of the room at a height of about $1m$. We also provided for diffuse indirect lighting from above the room.

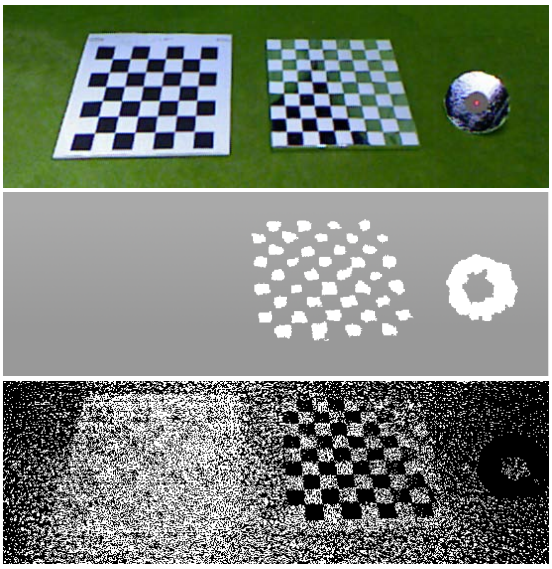

Figure 2: *Our new calibration approaches solve the problem that RGB and depth sensors can not be calibrated simultaneously. We propose binary surface patterns, e.g. a checkerboard consisting of white diffuse and mirroring patches or a point light source with a mirror disc attached. The advantage becomes clear in the depth image (second row), where our patterns become distinguishable, while a classical printed checkerboard pattern becomes invisible. The bottom row shows the captured IR-values (thresholded for better visibility).*

3.1. Checkerboard

In order to simultaneously align a depth sensor with a consumer-grade RGB camera, we introduce a new calibration pattern. Our goal is to provide a clear and distinguishable calibration pattern in both the RGB and the depth sensor. We decided to start with a checkerboard pattern. A normal checkerboard pattern, printed out on paper, however, will uniformly reflect the IR pattern back to the depth sensor and thus the captured image of the checkerboard only consists of indistinguishable depth values, at most a gradient, Fig. 2.

We found that mirroring aluminum foil, placed at the right angle (i.e. every angle except nearly orthogonal to camera's viewing axis), will project the patterns to infinity, Fig. 1 (second row, third row). Therefore, we designed our checkerboard pattern to be white and diffuse on the one hand and mirroring on the other hand. White paper interchanges with aluminum foil in our pattern. Thus, we get a projected checkerboard image in both the RGB and depth camera which can be used for robust alignment. We calibrate the depth sensors using this binary pattern consisting of diffuse and mirroring patches. The reflective patches act as mirrors and deflect the IR pattern to infinity producing invalid values, i.e. the pixel in the depth image is I(x,y)=2047. The diffuse patterns reflect the IR light and provide depth values in the captured image, Fig. 3 (top row). The resulting images are then used for calibration in the Matlab calibration toolbox [Bou10].

3.2. Point light source

An alternative calibration approach is to provide single point correspondences. The idea is that a light spot in a dark room is moved over a defined time period and is recorded by several cameras. The cameras are assumed to be approximately synchronized. Each recorded frame then defines a unique point correspondence, i.e. the position of the light spot in each video at that time instant. The combination of the recorded frames over time then provides the linear system of point correspondences.

A self-calibration toolbox that implements this idea has already been made available [Svo05]. The person holding the point light source either wears green clothing matching the greenroom, and can thus be easily thresholded out of the image, or the person is captured in a darkened room. However for depth sensors, the person is always completely visible in all frames.

We solve this problem by introducing a point light source with an attached disk of $\approx 20cm$ diameter consisting of mirroring material, e.g. aluminum foil, Fig. 2 (right device). The disk is attached to the light source such that it is placed in the center of the disk, visible through the hole. Once again this material deflects the emitted IR light, therefore in the depth images there is a ring with invalid values, i.e. the pixel in the depth image is I(x,y)=2047.

We search for this ring using a connected component search, Alg. 1. Once detected, we compute its midpoint, which coincides with the position of the light spot. After thresholding we can provide point correspondences in both the RGB and the depth images, Fig. 3 (bottom row).

4. Multiplexing

In order to design a motion capturing setup consisting of multiple active sensors, here, the Kinect's depth sensors, we seek to get an evaluation of the overall introduced depth estimation error. Thus, we first measure the depth errors, i.e. the percentage of invalid pixels, for increasing number n of simultaneously running depth sensors, $n \in (1..4)$ for a set of materials with different BRDFs, Fig. 5. The set consists of a diffuse, a specular, a mirroring and a plastic material. Then, we apply a set of steerable hardware shutters to the Kinects in order to block the emitted laser light, thus allowing for time-multiplexing. We investigate a time-multiplexed setup, where we measured two different cycles. The first cycle allows two Kinects to emit light at the same time instant. We

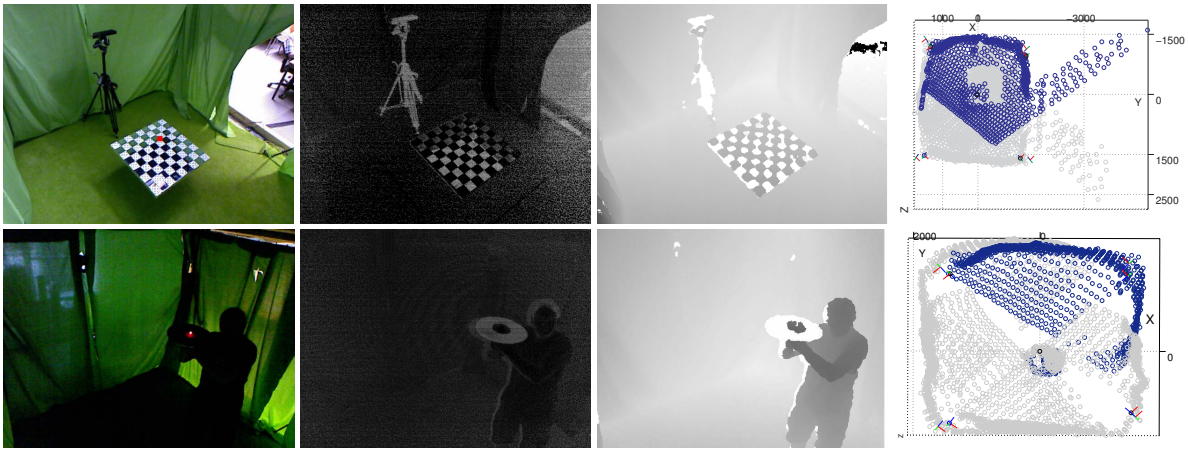

Figure 3: *Our reflective-diffuse checkerboard provides for a robust pattern in both the rgb and the depth sensor. We use the Matlab calibration toolbox [Bou10] and the thresholded infrared image. Alternatively, we calibrate the depth sensors with per frame point correspondences. A small point light with a deflective disk attached provides for point locations in both the RGB and depth image. We use the Matlab self-calibration toolbox [Svo05] and a preprocessed depth image, Alg. 1. A top-view of the reconstruction from the depth values of four Kinect sensors are shown in the rightmost image; the blue points correspond to the depth image.*

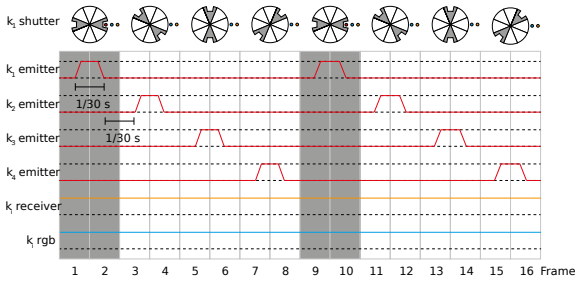

Figure 4: *In our setup with n Kinects, $n \in 1...4$, we employed four simultaneously running Kinects $k_{1..4}$, i.e. four sensors that actively emit light. We constructed a steerable hardware-shutter to simulate different cycles. The diagram shows a cycle, that allows for only one Kinect to emit IR-light at a time instant. Note, that the IR and RGB sensors are not occluded by the shutter.*

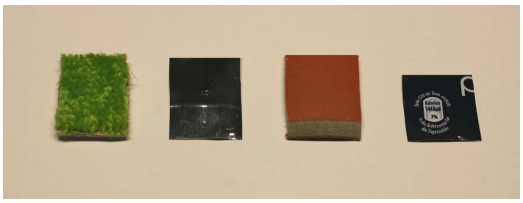

Figure 5: *For our multiple IR emitter interference error measurements, we conducted a study with the following materials: diffuse carpet, mirroring foil, plastic tube and specular cans. We also conducted a study in an empty greenroom.*

Algorithm 1 We find the point light in the depth image based on connected component search. The center of the deflective disk is computed in the depth image.

Finding the pointlight in the depth image
begin $I_{close} = binary_closing(I)$;
  comment: Eliminate background and small objects
  $componentList = findConnectedComponents(I_{close})$;
  $sort(componentList, descending)$;
  $I_{close}(componentList[0]) = 0$;
  $componentList = findConnectedComponents(I_{close})$;
  $sort(componentList, ascending)$;
  while $componentList.size() > 1$ **do**
    $I_{close}(componentList[0]) = 0$;
    $componentList = findConnectedComponents(I_{close})$;
    $sort(componentList, ascending)$;
  od
  comment: The remaining component is the area inside the ring
  $pixel = geometricCenter(I_{close}(componentList[0]))$;
  $I_{close}(pixel) = color$;
end

use cycles with either two or four opening phases, applied to four Kinects with 30 fps (33.33ms per frame) each. With 2 opening phases, a pair of two Kinects receive equal time slices of 16.66ms per frame; With 4 opening phases, each of the Kinects receives an equal time slice of 8.33ms per frame. The procedure is illustrated in Fig. 4.

	μ_{IR}	σ_{IR}	μ_{RGB}	σ_{RGB}
$x_{Chessboard}$	0.5672px	0.2407	0.4870px	0.3306
$y_{Chessboard}$	0.3787px	0.0421	0.3995px	0.2047
$x_{Pointlight}$	1.7996px	0.2080	1.3883px	0.2311
$y_{Pointlight}$	1.6452px	0.1116	1.2430px	0.0598

Table 1: *Average reprojection errors and standard deviations for chessboard and point light reprojection, both in x and y dimensions, in pixels. Surprisingly, the checkerboard calibration is superior to the point light, even though the latter spans the recording room better.*

5. Motion Capture

Our motion capturing algorithm follows Carranza et al. [CTMS03], i.e. we solve an optimization problem regarding the amount of overlapping pixels for the projected silhouettes of the model S_{m_i} to be fitted and the silhouettes of the actor S_{a_i} (input silhouettes), $i \in 1..n$. Here, the input silhouettes are extracted from the n depth images of the multiple Kinect setup, which we assume to be calibrated and temporally aligned (e.g. using the method by Meyer et al. [MSMP08]). We may also guide the algorithm by exploiting depth discontinuities of occluding body parts.

Note, that in a greenroom with no green clothing present, the silhouettes from the depth sensors are essentially equal to ones made with RGB cameras. However, depth sensors can be used for non-greenroom settings as well without quality degradation.

6. Results

Calibration

We conducted a study on the reprojection errors for the two calibration methods in a multicamera setup with four Kinects, as shown in Tab. 1.

The chessboard calibration method yields an average reprojection error of around 0.5 pixel in both the IR and RGB images, with a slight advantage for the RGB camera. The pointlight calibration yields an average reprojection error of around 1.5 pixels in IR and RGB.

Surprisingly, the checkerboard calibration was more accurate than the point light calibration, although the latter spans the room better. Furthermore, the variance of estimated depth values was between 0.5 and 1.5mm, and was not dependent on the distance to the sensor.

Multiplexing

We measured the percentage of error pixels for different amounts of Kinects running simultaneously, Tab. 2 and found that, unsurprisingly, the error increases with the number of simultaneously running Kinects. Also, the amount of error pixels increases with narrower angles, more Kinects and higher specularity of the materials' BRDF. More interestingly, we found that the depth estimation of the remaining pixels which are not labeled as invalid does not degrade with narrower angles, more Kinects, and higher specularity of the materials' BRDF.

As expected, we found that two simultaneously running Kinects should be placed with maximal angle between their viewing axes, i.e. 180°, in order to produce optimal results.

To evaluate the effectiveness of the different setups, we then measured the pixel errors for hardware induced multiplexing, i.e. synchronized shutters, Tab. 2. Surprisingly, even with more Kinects, the depth variance over time was still between 0.5 and 1.5mm. This indicates that even though the number of invalid depth pixels increases, the quality of the remaining pixels stays the same.

Motion Capturing

We also compared our motion capturing based on multiple depth images to the monocular motion capturing based on depth cue inference. We placed the actor so that he is captured from behind, and let the actor fold his arms. It becomes obvious that the motion capturing based on depth cue inference fails tracking the arm movement because the actor is not facing the camera, Fig. 6 (top row).

The same motion sequence is reconstructed with our motion capturing algorithm based on multiple depth images, Fig. 6 (second, third row). We tested the "worst case" with regard to interference: four simultaneously running Kinects. The arm movement could be tracked over the whole sequence, Fig. 6 (fourth, fifth, sixth row). The information degradation due to multiple active sensors does not affect the motion capturing substantially. Fig. 7 shows that the same holds for obstructed body parts. The actor crosses his arms above his head. While depth cue inference fails tracking the arms, our motion capturing algorithm based on multiple depth images succeeds to capture the whole sequence.

We also tested the setup outside the green room and found equivalent results. While the number of pixel errors may increase depending e.g. on the distance or specularity of the background, the silhouette is faithfully preserved.

In summary, the results show that motion capture with multiplexed Kinects is very well feasible. The only drawback is that due to the reduced effective frame rate, faster motions produce ambiguities.

7. Conclusions

We investigated the effects of using multiple Kinects in a motion capturing setup. To evaluate the accuracy, we first introduced two new calibration devices, one based on checkerboard calibration, one based on time-varying point correspondences. We found that the average reprojection error for the checkerboard calibration is under 1px, while the time-

	Concurrently running					Time-multiplexed		
Material vs. Setup	1	2 adjacent	2 opposite	3	4	1	2 adjacent	2 opposite
Blank room	0.1159%	0.5908%	0.4420%	0.0253%	2.8267%	1.8413%	23.2563%	4.3864%
Diffuse Carpet	0.3696%	0.9446%	0.9643%	1.6493%	2.5914%	2.1332%	22.0778%	5.1351%
Mirroring foil	3.3111%	6.0228%	5.6591%	7.8242%	10.2082%	5.9220%	25.1686%	13.2228%
Plastic Pipe	0.5351%	1.1571%	1.1257%	1.8063%	3.1328%	2.6534%	23.1102%	4.3260%
Specular Cans	0.4908%	1.2881%	1.0737%	2.1112%	3.4963%	2.4954%	21.7885%	5.4662%
fps / ms	30 / 33					30 / 33	15 / 66	15 / 66

Table 2: *We conducted an evaluation on the amount of depth pixel errors for varying captured materials with an increasing number of Kinects (left part). We also examined a cycle with one exclusively running and two simultaneously running Kinects (right part). The additional error introduced by the hardware shutter is slight compared to a setup with the same amount of simultaneously running Kinects. Vertical: Different materials. Horizontal: Kinect setups. Values: Percentage of depth pixel errors, averaged over 80 frames. The data clearly shows that narrower angles, more Kinects, a shutter, and higher specularity lead to increased errors.*

varying point correspondences introduce an average reprojection error of between 1 and 2px.

Then, we investigated the effects of the number of simultaneously and successively running Kinects and found that a suitable solution for capturing scenarios can be found if two simultaneously running Kinects are placed with an 180° angle to each other. Furthermore, we found that a 2x2 multiplexing provides sufficient accuracy for motion capturing featuring moderate movements.

We found that the motion capturing based on multiple Kinects succeeds to capture obstructed body parts where the monocular depth cue inference fails. The depth based silhouette construction furthermore proved to be robust against background color, in contrast to RGB camera approaches which are dependent on a clearly distinguishable background.

Acknowledgements

This work has been partially funded by the German Science Foundation, DFG MA2555/5-1, and by the European Research Council ERC under contract No. 256941 “Reality CG”. Their support is gratefully acknowledged.

References

[BKWK07] BARTCZAK B., KOESER K., WOELK F., KOCH R.: Extraction of 3d freeform surfaces as visual landmarks for real-time tracking. *Journal of Real-Time Image Processing 2*, 2 (2007), 81–101. 2

[Bou10] BOUGUET J.: Camera calibration toolbox. *See http://www. vision. caltech. edu/bouguetj/calib_doc* (2010). 2, 3, 4

[Bur10] BURRUS N.:. http://nicolas.burrus.name/index.php/Research/KinectCalibration, November 2010. 2

[CTMS03] CARRANZA J., THEOBALT C., MAGNOR M., SEIDEL H.: Free-viewpoint video of human actors. In *ACM SIGGRAPH 2003 Papers* (2003), ACM, pp. 569–577. 2, 5

[DAST*08] DE AGUIAR E., STOLL C., THEOBALT C., AHMED N., SEIDEL H., THRUN S.: Performance capture from sparse multi-view video. In *ACM Transactions on Graphics (TOG)* (2008), vol. 27, ACM, p. 98. 2

[DATSS07] DE AGUIAR E., THEOBALT C., STOLL C., SEIDEL H.: Marker-less deformable mesh tracking for human shape and motion capture. In *2007 IEEE Conference on Computer Vision and Pattern Recognition* (2007), IEEE, pp. 1–8. 2

[Eng11] ENGELHARD N.:. http://www.informatik.uni-freiburg.de/~engelhar/calibration.html, mai 2011. 2

[Gaf11] GAFFNEY M.:. http://www.thingiverse.com/thing:7793, 2011. 2

[GFP08] GUAN L., FRANCO J., POLLEFEYS M.: 3d object reconstruction with heterogeneous sensor data. In *4th International Symposium on 3D Data Processing, Visualization and Transmission (3DPVT), Atlanta, GA, USA* (2008). 2

[GLA*08] GUDMUNDSSON S., LARSEN R., AANAES H., PARDAS M., CASAS J.: Tof imaging in smart room environments towards improved people tracking. In *Computer Vision and Pattern Recognition Workshops, 2008. CVPRW'08. IEEE Computer Society Conference on* (2008), IEEE, pp. 1–6. 2

[HJS08] HUHLE B., JENKE P., STRASSER W.: On-the-fly scene acquisition with a handy multi-sensor system. *International Journal of Intelligent Systems Technologies and Applications 5*, 3 (2008), 255–263. 2

[HRT*09] HASLER N., ROSENHAHN B., THORMAHLEN T., WAND M., GALL J., SEIDEL H.: Markerless motion capture with unsynchronized moving cameras. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on* (2009), IEEE, pp. 224–231. 1

[IY01] IDDAN G., YAHAV G.: 3d imaging in the studio (and elsewhere). In *Proc. SPIE* (2001), vol. 4298, Citeseer, pp. 48–55. 2

[KBK07] KOESER K., BARTCZAK B., KOCH R.: Robust gpu-assisted camera tracking using free-form surface models. *Journal of Real-Time Image Processing 2*, 2 (2007), 133–147. 2

[KBKL09] KOLB A., BARTH E., KOCH R., LARSEN R.: Time-of-flight sensors in computer graphics. *Eurographics State of the Art Reports* (2009), 119–134. 2

[KCTT08] KIM Y., CHAN D., THEOBALT C., THRUN S.: Design and calibration of a multi-view tof sensor fusion system. In *Computer Vision and Pattern Recognition Workshops, 2008.*

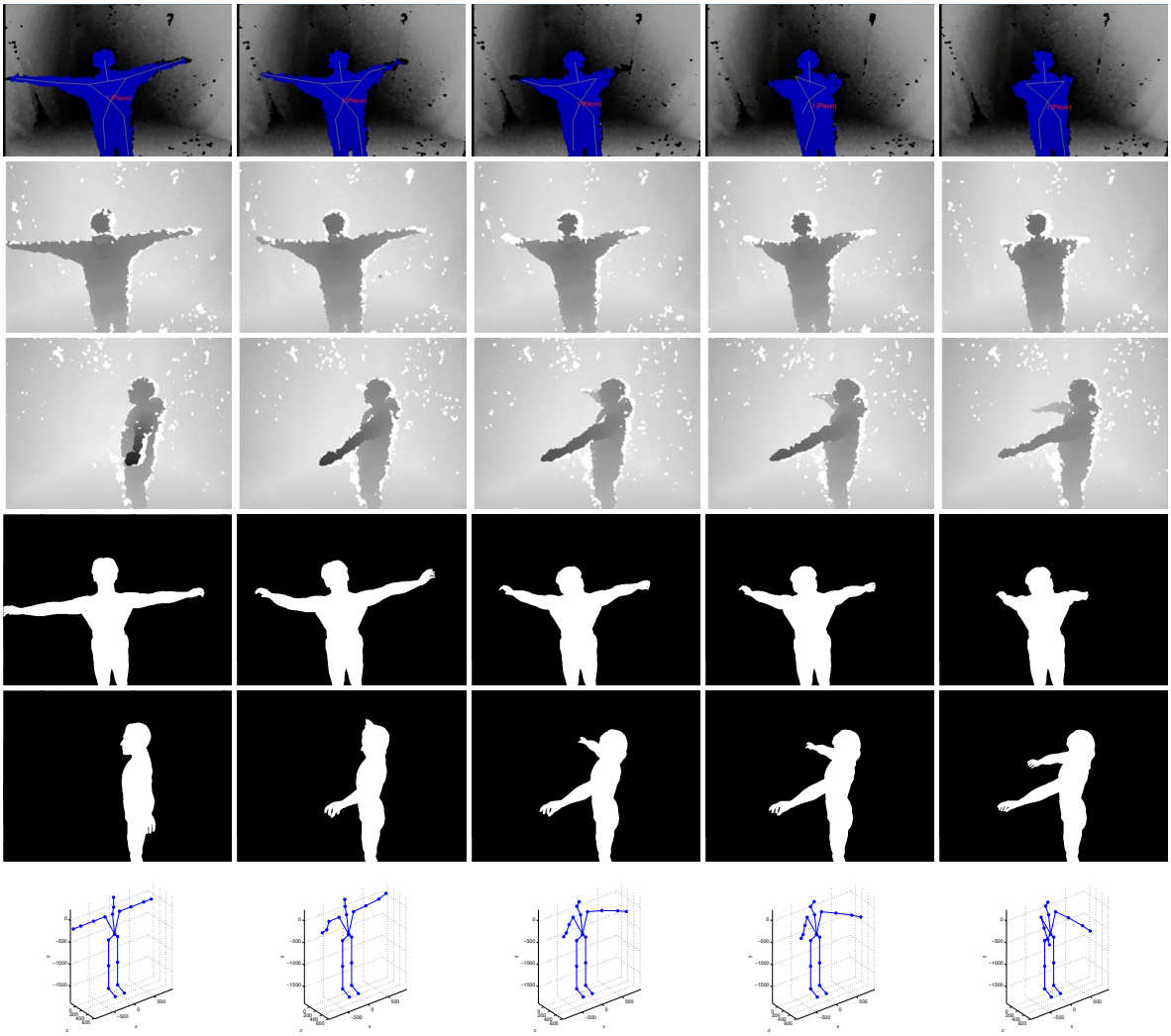

Figure 6: *An arm folding sequence captured with a single Kinect (first row). The pose of the actor, who is captured from behind, is retrieved by depth cue inference [Pri11]. Note that the tracking of the arms (yellow lines) is lost over the time. The same sequence is captured with four Kinects (second and third row show two sensor streams) and a model is fit to the depth silhouettes (fourth and fifth row). Despite the quality degradation from multiple active sensors, the arms are tracked over the entire sequence (sixth row, generated with [Law11]).*

CVPRW'08. IEEE Computer Society Conference on (2008), IEEE, pp. 1–7. 2

[Law11] LAWRENCE N. D.: Mocap toolbox for matlab. http://www.cs.man.ac.uk/~neill/mocap/, 2011. 7, 8

[MHK06] MOESLUND T., HILTON A., KRUGER V.: A survey of advances in vision-based human motion capture and analysis. *Computer vision and image understanding 104*, 2-3 (2006), 90–126. 2

[Mic10] MICROSOFT CORPORATION: Kinect for xbox 360, November 2010. Redmond WA. 2

[MSMP08] MEYER B., STICH T., MAGNOR M., POLLEFEYS M.: Subframe temporal alignment of non-stationary cameras. In *Proc. British Machine Vision Conference (BMVC) 2008* (2008). 5

[MWLT00] MARSCHNER S., WESTIN S., LAFORTUNE E., TORRANCE K.: Image-based bidirectional reflectance distribution function measurement. *Applied Optics 39*, 16 (2000), 2592–2600. 2

[Pri11] PRIMESENSE:. http://www.openni.org/downloadfiles/openni-compliant-middleware-binaries/34-stable, mai 2011. 7, 8

[SFC*11] SHOTTON J., FITZGIBBON A., COOK M., SHARP T., FINOCCHIO M., MOORE R., KIPMAN A., BLAKE A.: Real-time human pose recognition in parts from single depth images.

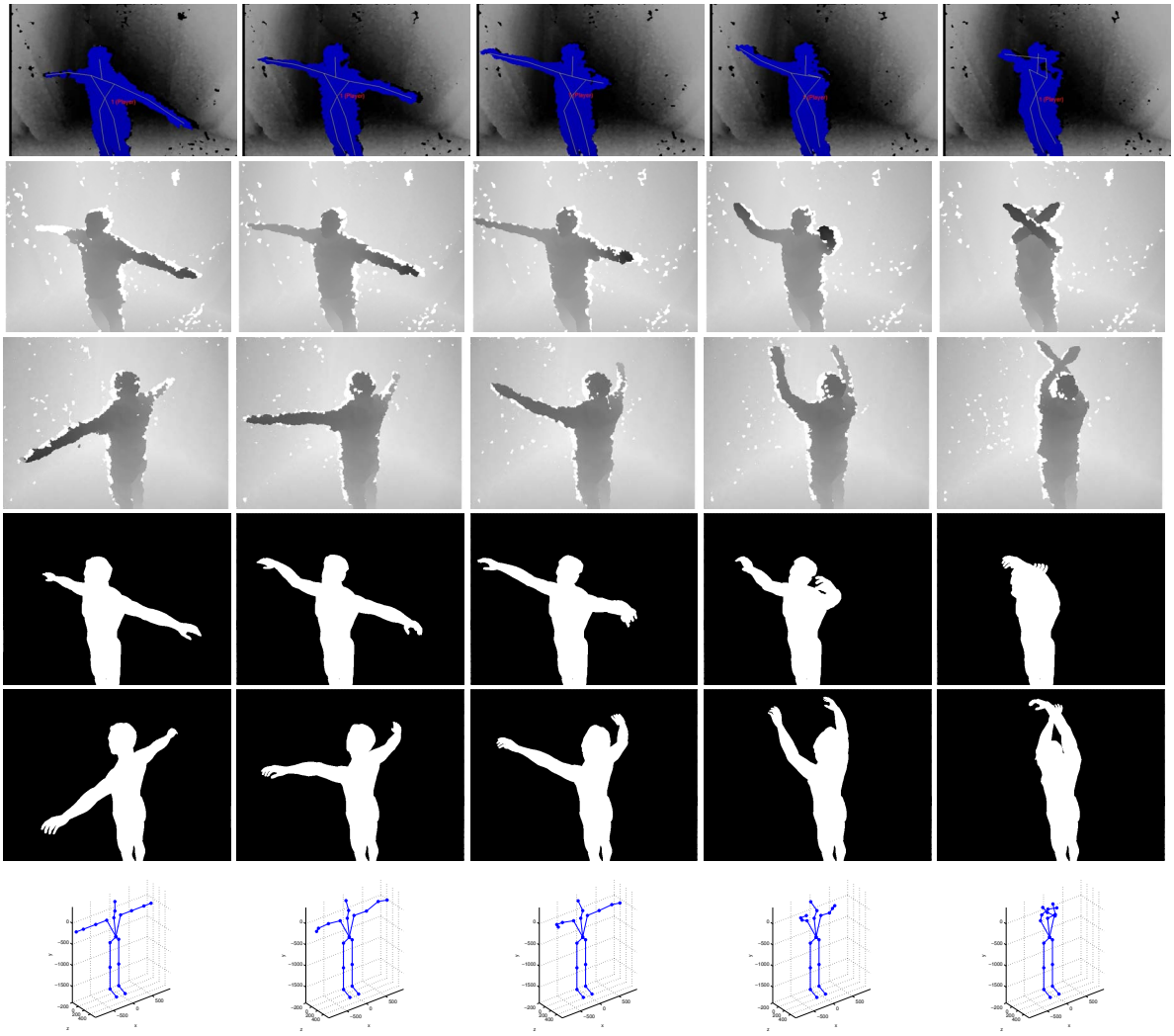

Figure 7: *An arm crossing sequence captured with a single Kinect (first row). The pose of the actor is retrieved by depth cue inference [Pri11]. Note that the crossing of the arms (yellow lines) can not be tracked over the time. The same sequence is captured with four Kinects (second and third row show two sensor streams) and a model is fit to the depth silhouettes (fourth and fifth row). Again, despite the quality degradation from multiple active sensors, the arm crossings are tracked over the entire sequence (sixth row, generated with [Law11]).*

In *In CVPR* (2011). 1

[SMP05] SVOBODA T., MARTINEC D., PAJDLA T.: A convenient multicamera self-calibration for virtual environments. *Presence: Teleoperators & Virtual Environments 14*, 4 (2005), 407–422. 2

[SSS06] SNAVELY N., SEITZ S., SZELISKI R.: Photo tourism: exploring photo collections in 3d. In *ACM Transactions on Graphics (TOG)* (2006), vol. 25, ACM, pp. 835–846. 2

[Svo05] SVOBODA T.: A software for complete calibration of multicamera systems. In *SPIE-IS&T Electronic Imaging, SPIE* (2005), vol. 5658, pp. 115–128. 3, 4

[WB10] WILSON A. D., BENKO H.: Combining multiple depth cameras and projectors for interactions on, above and between surfaces. In *Proceedings of the 23nd annual ACM symposium on User interface software and technology* (New York, NY, USA, 2010), UIST '10, ACM, pp. 273–282. 2