Tracking hand articulations: Relying on 3D visual hulls versus relying on multiple 2D cues

Iason Oikonomidis*†, Nikolaos Kyriazis*†, Konstantinos Tzevanidis*†, and Antonis A. Argyros*†
*Institute of Computer Science - FORTH
and
†Computer Science Department, University of Crete
Heraklion, Crete, Greece

***Abstract*—We present a method for articulated hand tracking that relies on visual input acquired by a calibrated multi-camera system. A state-of-the-art result on this problem has been presented in [12]. In that work, hand tracking is formulated as the minimization of an objective function that quantifies the discrepancy between a hand pose hypothesis and the observations. The objective function treats the observations from each camera view in an independent way. We follow the same general optimization framework but we choose to employ the visual hull [10] as the main observation cue, which results from the integration of information from all available views prior to optimization. We investigate the behavior of the resulting method in extensive experiments and in comparison with that of [12]. The obtained results demonstrate that for low levels of noise contamination, regardless of the number of cameras, the two methods perform comparably. The situation changes when noisy observations or as few as two cameras with short baselines are employed. In these cases, the proposed method is more accurate than that of [12]. Thus, the proposed method is preferable in real-world scenarios with noisy observations obtained from easy-to-deploy, stereo camera setups.**

I. Introduction

Tracking of articulated objects is an important problem in computer vision with significant theoretical interest and diverse applications [11], [5]. An instance of this problem, the 3D tracking of human hands, has recently been the subject of intense inquiry [4], [13], [18], [9]. Methods attempting to solve the problem have to overcome a number of interacting complicating factors such as the high dimensionality of the problem, the chromatically uniform appearance of a hand and the severe self-occlusions that occur while a hand is in action. To overcome some of these problems, notable methods employ specialized hardware for motion capture [15] or the use of visual markers as in [19]. However, such methods require a complex and costly hardware setup, interfere with the observed scene, or both, thus limiting their applicability. Recent state-of-the-art methods [13], [9] rely on markerless observations of a hand from an RGBD camera. Despite their success, such methods are not operational in outdoor environments where RGBD sensors cannot provide reliable depth information.

The approach we propose in this paper assumes that a markerless hand is observed by a set of conventional RGB cameras (see Fig. 1). Approaches that only use this kind of input, namely visual markerless data, can be categorized in two main categories [5], appearance-based and model-based ones. Appearance-based methods use a pre-computed map from the space of visual features to that of hand poses to accomplish the task and hence they are essentially limited by their training set. To cope with such limitations, recent works [16], [14], [9] employ training sets in the order of thousands or even million samples. On the other hand, model-based methods employ a model of the human hand from which visual features are computed on-the-fly and compared to the observations. This makes them potentially more accurate but, at the same time, more computationally demanding.

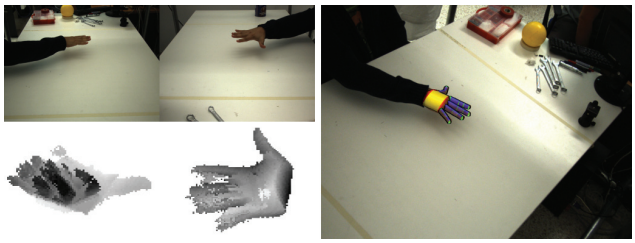

Fig. 1. We investigate the use of visual hull to track a human hand observed by multiple cameras. Top-left: two views of a human hand. Bottom-left: two views of the recovered visual hull, using skin color for segmentation. Right: fitted hand model on another input image.

In this paper we build upon the model-based approach presented in [12]. In that work, we proposed to use skin color and edges as visual features for hand pose estimation and tracking. These cues are estimated in each camera that observes the scene. Given a hand hypothesis, we produce comparable features by means of rendering. Then, hand pose estimation is formulated as an optimization problem whose objective function is the sum of discrepancies between observed and rendered features, across views. In this work we adopt the same framework, altering the objective function. Since we are using a fully calibrated multicamera system, we propose to employ the visual hull [10] of skin color silhouettes as a way to fuse the existing information. Having computed the visual hull of the hand silhouettes, the objective function heavily depends on the discrepancy between this actual hull and the synthesized visual hulls from candidate hand solutions. This choice is justified by the fact that comparisons between visual hulls can be meaningfully quantified, whereas this is not as intuitive for the case of 2D cues across different views.

Previous methods for hand tracking have used various visual features to solve the problem such as skin color, edges, optical flow, however, to the best of our knowledge, the visual hull has not been employed so far. Computing visual hulls is computationally more expensive compared to the 2D

cues used in [12]. However, with careful exploitation of the computational power of modern GPUs [17] it is possible to achieve interactive frame rates.

Extensive experimental results demonstrate that the two approaches behave comparably for ideal cases without noise or in low levels of noise. It is shown that this happens regardless of the number of employed cameras. However, in cases with high noise we observed that the proposed approach can still keep track of a hand while [12] fails. Furthermore, we present experimental evidence in real world data from a narrow baseline stereo system, a scenario that cannot be adequately handled by the approach of [12]. The fact that our approach is more tolerant to noise and performs better from smaller/simpler camera configurations makes it much preferable to [12] in real-world applications.

II. Hand Tracking using 3D Visual Hulls

It is assumed that a set of synchronized and fully calibrated cameras observes the scene (see Fig. 1). The system captures a set of images from all views and skin colored segmentation is employed to segment the hand in each view. The visual hull of these silhouettes is computed and stored as the main visual cue for this frame. Hand hypotheses are then obtained with the help of Particle Swarm Optimization (PSO) [8]. The visual hull of each such hypothesis is computed and compared to the observed hull. Based on this comparison, PSO iteratively drives the process of optimizing the new hypotheses. The result of this optimization process is the output of the method for the given frame. Temporal continuity is exploited to track the hand articulation in a sequence of frames. The remainder of this section describes these algorithmic steps in more detail.

A. Observations

The proposed method operates on sequences of synchronized views acquired by a set of intrinsically and extrinsically calibrated cameras. A set of images acquired from these cameras at the same moment in time is called a *multiframe*. Let $M_i = \{I_1, I_2, \ldots\}$ denote a multiframe of a sequence $S = \{M_1, M_2, \ldots\}$, with I_j denoting the image from the j-th camera/view at the i-th time step. For each image I of a multiframe M, a skin color map $o_s(I)$ is computed using the method presented in [2], and an edge map $o_e(I)$ is computed using the Canny edge detector [3]. The set of skin color maps is used to efficiently compute their visual hull [17] which is kept as the reference volume map $o_v(M)$ of the observed hand. Similarly to [12], we compute the distance transform $o_d(I)$ of the edge map $o_e(I)$ in order to accelerate computations in subsequent steps. As a convention, in occupancy maps a value of 0 denotes absence whereas a value of 1 denotes presence of the respective property.

The visual hull of a 3D object S is the maximal object silhouette-equivalent to S, i.e., which can be substituted for S without affecting any silhouette, as defined in [10]. It can be equivalently defined as the intersection of the generalized cones that are formed by back-projecting each silhouette from the respective camera center to the scene. For details on efficiently computing visual hulls using GPUs, see [17].

The employed hand model is adopted from [12] and has 26 degrees of freedom. Each finger has four degrees of freedom, two at the base and two more in the remaining joints. The joint angle limits are based on anatomical studies [1].

B. Formulation of the Objective Function

Having a parametric 3D model of a hand, the goal is to estimate the model parameters h that are most compatible to the visual observations (Sec. II-A). Towards this end we formulate an error function $E(h, O)$ that quantifies the discrepancy between a hand pose parametrization h and the observation O. More specifically we compute

$$E(h, O) = D(O, h, \mathrm{C}) + \lambda_k \cdot kc(h). \tag{1}$$

In Eq.(1), D quantifies the discrepancy between observed and hypothesized hand volumes and is computed as follows. Given a hand pose hypothesis h and camera calibration information c_i, skin occupancy maps $r_s(h, c_i)$ and edge maps $r_e(h, c_i)$ for each synthetic view i are generated by means of rendering. The volume reconstruction methodology of [17] is then once again employed to produce an occupancy volume $r_v(h)$ that can be directly compared with the observation o_v. The comparison between these occupancy maps quantifies the discrepancy between the observed and the hypothesized hand pose. This is achieved by computing

$$D(O, h, C) = 1 - \frac{2\sum o_v \wedge r_v}{(\sum o_v \wedge r_v) + (\sum o_v \vee r_v)} + \lambda \frac{\sum o_d(I) \cdot r_e(h, c_i)}{\sum r_e(h, c_i) + \epsilon}, \tag{2}$$

where, for the sake of notational simplicity, o_v denotes $o_v(O)$ and r_v denotes $r_v(h)$. The logical operators $\wedge$ and $\vee$ denote per-voxel operations of the respective maps and the summation is taken over the entire maps.

The function kc adds a penalty to kinematically implausible hand configurations and is defined as in [12] to penalize adjacent finger inter-penetration. In all experiments, the value of λ_k was set to 0.1 and the value of λ was set to 0.01.

The solution for each frame is obtained by optimizing the objective function E with the Particle Swarm Optimizer [7]. The "nearest point" policy [6] was adopted to handle collisions with the search space boundaries. The randomization variant proposed in [13], originally seen in [20], was also adopted here since it proved experimentally beneficial for the accuracy of the estimation of the finger pose.

As in [12], we exploit the inherent parallelism of the involved computations by computing multiple values of the objective function in parallel. This can be exploited in PSO, since the evaluations of the objective function for the particles of one generation are independent of each other. Furthermore, the employed hand model is built out of appropriately transformed cylinders and spheres. This exposes data parallelism in the resulting computations.

III. Experimental Evaluation

A number of quantitative experiments was conducted, designed to compare the behavior of the proposed method to that of [12]. These experiments analyzed the behavior of the objective functions of the methods, investigated the parametrization of PSO, assessed the effect of segmentation

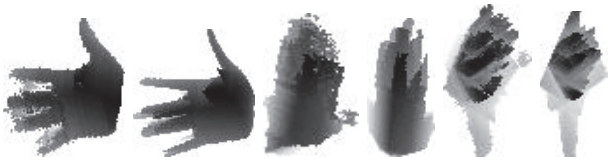

Fig. 2. Different depth map views of the visual hull of the observed skin color (frames 1, 3, 5) and that of a manually fitted hand model (frames 2, 4, 6). Each pair corresponds to a different view of the reconstructed volumes.

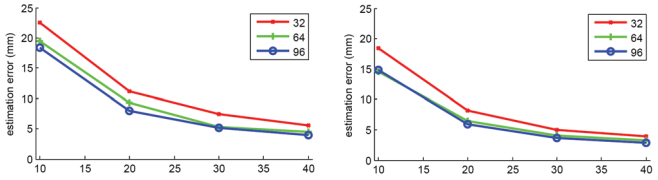

Fig. 3. Investigation of the PSO parametrization for the proposed method (left) and our implementation of the method [12] (right). Different curves in both plots correspond to different populations of PSO particles.

noise and also explored the behavior for different numbers of available views of the scene. Qualitative results in real-world data are also presented. In all the experiments with visual hulls we used a parallelepiped reconstruction space of 128^3 voxels, centered around the previously estimated position of the observed hand. The physical dimensions of this space were $240mm$ along each edge, resulting in a voxel size slightly larger than $2mm$.

A. *Visual hull of hypothesis and observation*

We present a qualitative comparison between a visual hull obtained from real world data and one from synthetically produced silhouettes. The pose of the synthetic hand model was manually adjusted to match as best as possible that of the observed hand. In both cases, we produced silhouette maps from eight views of the scene. For the real world data we used the same procedure used to obtain $o_s(I)$ in Section II-A. For the synthetic maps we used the procedure that generates silhouettes of hypotheses $r_s(h, c_i)$, with the pose h manually adjusted as described above, and the known calibrations c_i for each virtual view. Depth maps of the resulting visual hulls are depicted in Fig. 2. Evidently, even though we are using eight views of the scene, the resulting visual hulls still contain reconstruction artifacts. However, and although the real world data are obviously more noisy, one can observe that the artifacts between the observed and hypothesized hulls are consistent and reproducible. This justifies why comparisons are performed between *observed visual hulls and hypothesized visual hulls* and not between *observed visual hulls and actual hand models*. The second option would be considerably less time consuming since it does not require the visual hull computation for hypothesized hand poses. We chose to follow the first option since visual hulls differ considerably to the actual hand models, especially when fewer cameras are employed in their computation.

B. *Quantitative Evaluation*

We investigated the accuracy of the method for different parameters of PSO. The computational budget of PSO is determined by the number of particles and generations,

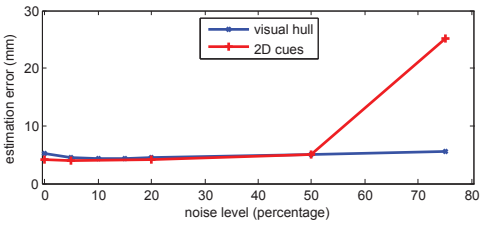

Fig. 4. Investigation of the effect of noise in the two compared methods. The horizontal axis denotes percentage of corrupted pixels in the synthetic input and the vertical denotes average distance from the ground truth.

the product of which yields the total number of objective function evaluations. We selected a set of values for these two parameters and computed the accuracy of the proposed method, as well as that of [12]. In order to quantitatively evaluate the pose estimation accuracy we resort to datasets with available ground truth. More specifically, for all experiments, we used a sequence of 245 multiframes depicting a hand that performs simple everyday motions such as palm flipping and finger bending. In order to compare the estimated pose, we adopt a metric similar to the one used in [12]: 21 landmark points are placed on the model hand, 3 on each finger and the remaining 6 on the palm. The average distance of all these landmarks between the estimated and the actual, ground truth pose is measured, and the average over all the frames of the sequence is computed. We perform this experiment multiple times and compute the median of these values because of the stochastic nature of PSO.

The obtained results are shown in Fig. 3. The upper plot shows the performance of the system using the visual hull as the main observation cue whereas the plot at the bottom shows the results obtained from [12]. The horizontal axis corresponds to the number of generations used for the experiment whereas the vertical axis denotes the measured pose estimation error as described above. Different graphs in the plots correspond to different particle count configurations. Both methods benefit when using more generations or particles, with the number of generations playing a more important role in the performance. The performance of both methods is comparable. Indicatively, the average error of the proposed method for 64 particles and 30 generations is around $5.3mm$ whereas that of [12] achieves $4.1mm$. For both methods, the performance improvement for more particles or generations is small. The additional computational budget for this improvement is disproportionate, so for the remaining experiments we fixed these parameters to $(64, 30)$ for both the proposed method and that of [12].

We conducted another experiment, that was designed to assess the effect on the estimation accuracy of the number of available views, again in the case of ideal data. We varied the number of cameras from one to eight. It should be noted that the two cameras setup in this experiment had a large baseline and almost perpendicular optical axes. The comparison demonstrated that the error for both methods is almost identical.

The quantitative performance experiments were concluded by an experiment investigating the effect of noise on both methods. We employed a noise model similar to that of [14]. More specifically, small disks of randomly selected positions and radii were chosen in the synthetic input images and the

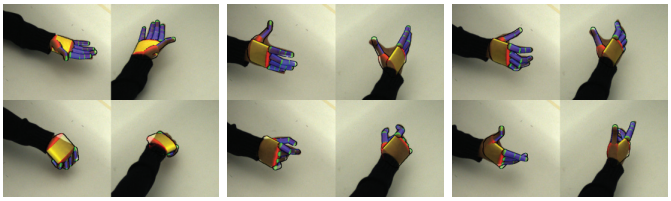

Fig. 5. Results of the proposed method in real-world data. Each pair of images illustrates the same pose from different views.

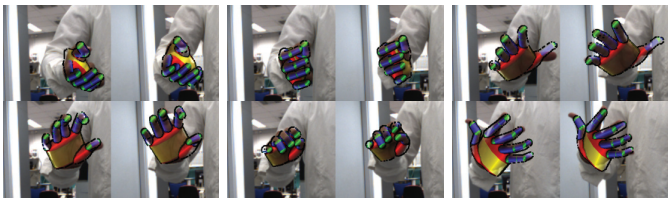

Fig. 6. Sample results of the proposed method on data acquired from a narrow baseline stereo system. Each pair illustrates the same pose.

pixels in them were flipped. Figure 4 illustrates the obtained results. Both methods behave comparably for low and moderate levels of noise, however the proposed method manages to keep track for the large noise level of 75% whereas [12] fails.

C. Real-world Input

A network of eight cameras recorded a human hand that performed simple hand motions such as palm flipping, pinching and grasping. The sequence contains 390 frames. The hand model was manually initialized for the first frame, and the method successfully tracked all the sequence. Sample results are shown in Figure 5. Evidently, the fitted hand model closely matches the observations.

We conducted another experiment in data acquired from a narrow baseline stereo camera system. This scenario is of special practical importance since it is easy to setup and can be employed by, e.g., robotic systems operating outdoors where RGBD cameras currently fail. Sample results from this sequence are shown in Figs. 6 (proposed method), and 7 ([12]). As it can be observed, the proposed method manages to track the hand correctly in this dataset, whereas the method in [12] fails.

IV. Conclusions

We investigated the use of the visual hull as the main observation cue for hand pose tracking using a multicamera system. We compared this approach to that of [12], which uses only 2D cues. The comparison showed that these two methods perform comparably in close-to-ideal data, regardless of the number of employed cameras. However, the proposed approach performs better when the noise level in the observations increase, especially for the case of short baseline stereo. This suggests that the proposed method is of higher practical significance since it is more robust with noisy input and requires simpler camera configurations compared to [12].

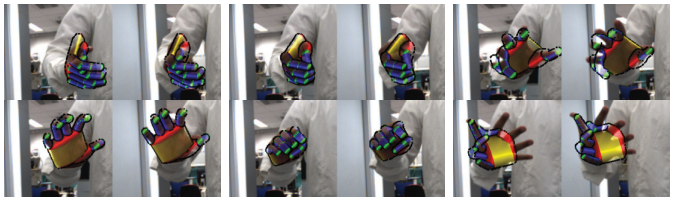

Fig. 7. Results on the same sequence as Figure 6 using the method of [12].

Acknowledgements

This work was partially supported by the EU IST-FP7-IP-288533 project RoboHow.Cog and by the EU FP7-ICT-2011-9-601165 project WEARHAP.

References

[1] I. Albrecht, J. Haber, and H. Seidel. Construction and animation of anatomically based human hand models. In *Proceedings of the 2003 ACM SIGGRAPH/Eurographics symposium on Computer animation*, page 109. Eurographics Association, 2003.

[2] A. Argyros and M. Lourakis. Real-time tracking of multiple skin-colored objects with a possibly moving camera. In *ECCV*, 2004.

[3] J. Canny. A computational approach to edge detection. *PAMI*, 8(6):679–698, 1986.

[4] M. de La Gorce, D. Fleet, and N. Paragios. Model-based 3d hand pose estimation from monocular video. *IEEE Trans. on PAMI*, pages 1–15, Feb. 2011.

[5] A. Erol, G. Bebis, M. Nicolescu, R. D. Boyle, and X. Twombly. Vision-based hand pose estimation: A review. *CVIU*, 108(1-2):52–73, Oct. 2007.

[6] S. Helwig and R. Wanka. Particle Swarm Optimization in High-Dimensional Bounded Search Spaces. In *Swarm Intelligence Symposium*, pages 198–205. Ieee, 2007.

[7] J. Kennedy and R. Eberhart. Particle swarm optimization. In *ICNN*, volume 4, pages 1942–1948. IEEE, Jan. 1995.

[8] J. Kennedy, R. Eberhart, and Y. Shi. *Swarm Intelligence*. Morgan Kaufmann Publishers, 2001.

[9] C. Keskin, F. Kirac, Y. E. Kara, and L. Akarun. Real time hand pose estimation using depth sensors. *ICCV Workshop*, pages 1228–1234, 2011.

[10] A. Laurentini. The visual hull concept for silhouette-based image understanding. *IEEE Trans. on PAMI*, pages 150–162, 1994.

[11] T. B. Moeslund, A. Hilton, and V. Kru. A survey of advances in vision-based human motion capture and analysis. *Computer Vision and Image Understanding*, 104:90–126, 2006.

[12] I. Oikonomidis, N. Kyriazis, and A. A. Argyros. Markerless and Efficient 26-DOF Hand Pose Recovery. In *ACCV*, pages 744–757. Springer, 2010.

[13] I. Oikonomidis, N. Kyriazis, and A. A. Argyros. Efficient Model-based 3D Tracking of Hand Articulations using Kinect. In *BMVC*, Dundee, UK, Aug. 2011.

[14] J. Romero, H. Kjellstrom, and D. Kragic. Monocular real-time 3D articulated hand pose estimation. *2009 9th IEEE-RAS International Conference on Humanoid Robots*, pages 87–92, Dec. 2009.

[15] M. Schneider and C. Stevens. Development and testing of a new magnetic-tracking device for image guidance. *SPIE Medical Imaging*, pages 65090I–65090I–11, 2007.

[16] J. Shotton, A. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman, and A. Blake. Real-Time Human Pose Recognition in Parts from Single Depth Images. In *CVPR*. IEEE, 2011.

[17] K. Tzevanidis, X. Zabulis, T. Sarmis, P. Koutlemanis, N. Kyriazis, and A. A. Argyros. From multiple views to textured 3d meshes: a gpu-powered approach. In *ECCV Workshops*, pages 5–11, 2010.

[18] R. Wang, S. Paris, and J. Popović. 6d hands: markerless hand-tracking for computer aided design. In *Proc. of the 24th annual ACM symposium on UIST*, pages 549–558. ACM, 2011.

[19] R. Y. Wang and J. Popović. Real-time hand-tracking with a color glove. *ACM Transactions on Graphics*, 28(3):1, July 2009.

[20] T. Yasuda, K. Ohkura, and Y. Matsumura. Extended pso with partial randomization for large scale multimodal problems. In *World Automation Congress*, number 1, pages 1–6. IEEE, Apr. 2010.