Generative 3D Hand Tracking with Spatially Constrained Pose Sampling

Konstantinos Roditakis[12]
croditak@ics.forth.gr
Alexandros Makris[1]
amakris@ics.forth.gr
Antonis A. Argyros[12]
argyros@ics.forth.gr

[1] Computational Vision and Robotics Laboratory
Institute of Computer Science FORTH
Greece
[2] Computer Science Department
University of Crete
Greece

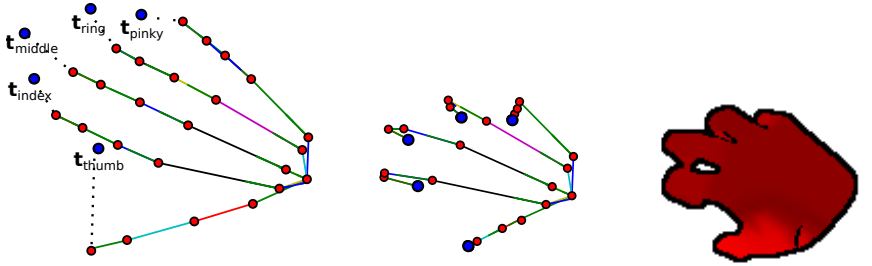
Abstract

We present a method for 3D hand tracking that exploits spatial constraints in the form of end effector (fingertip) locations. The method follows a generative, hypothesize-and-test approach and uses a hierarchical particle filter to track the hand. In contrast to state of the art methods that consider spatial constraints in a soft manner, the proposed approach enforces constraints during the hand pose hypothesis generation phase by sampling in the *Reachable Distance Space* (RDS). This sampling produces hypotheses that respect both the hands' dynamics and the end effector locations. The data likelihood term is calculated by measuring the discrepancy between the rendered 3D model and the available observations. Experimental results on challenging, ground truth-annotated sequences containing severe hand occlusions demonstrate that the proposed approach outperforms the state of the art in hand tracking accuracy.

1 Introduction

Tracking a human hand either in free motion or in interaction with objects is a challenging computer vision problem. Challenges arise due to its several degrees of freedom, hard to avoid visibility limitations (e.g. self-occlusions, occlusions from the interacting objects), and fast motion. Despite these difficulties, several works attempt to address the problem and have pushed significantly the performance boundaries over the last few years. These efforts are motivated by the impact that reliable hand tracking may have in areas such as human computer interaction, virtual reality, human robot interaction and robot control, in medical applications and many others.

Regardless of whether they treat the case of a single hand, a hand interacting with object(s) or multiple hands, previous approaches fall into three main categories: generative, discriminative and hybrid. In order to estimate the hand pose, discriminative approaches [3, 7, 12, 17, 25, 26, 33, 35, 36, 43, 46] learn a mapping between image features and the pose space. Discriminative methods require training on large training sets. At run time they are fast and able to perform single shot hand pose estimation. However, the output pose granularity is relatively coarse.



(a) (b) (c)

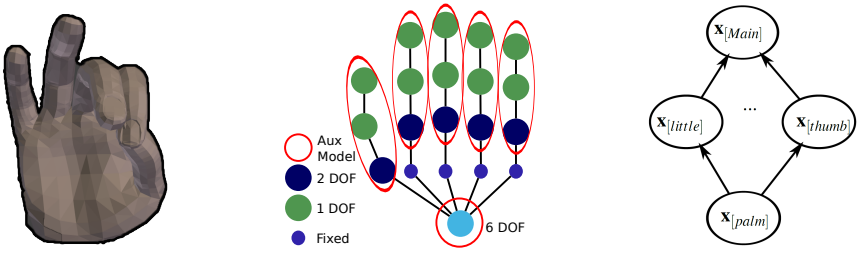
Figure 1: Tracking by spatially constrained sampling illustration: (a) Hand articulation hypothesis from the previous time step and curent step fingertip targets, (b) Articulation hypothesis respecting the target constraints, (c) Corresponding rendered hand model.

Generative methods for tracking single hands [2, 8, 9, 13, 15, 16, 18, 19, 20, 27, 30, 37, 44], hand object interactions [5, 6, 10, 11, 22] or two hands [21] use parametric hand models and filters or optimizers to estimate the state that best explains/fits the available observations. Typically, local optimization is performed, seeded by the solution to the problem in the last frame. This raises the requirement of temporal continuity and, thus, prohibits single shot pose estimation. The hypothesize and test methodology that is followed by the Particle Swarm Optimization (PSO) algorithm [19] and the Particle Filters (PF) [2] has proven particularly suitable for the problem. To tackle the high dimensionality, certain methods [15, 36] create hypotheses hierarchically by exploiting the kinematic structure of the hand.

Hybrid methods [1, 23, 24, 28, 29, 31, 32, 34, 38, 39, 40, 42] attempt to retain the advantages of both the discriminative and generative strategies. Typically, they employ a discriminative component to arrive at a coarse solution which is then refined by a generative component. The discriminative component detects hand parts relying on a set of image features. The detected parts are then either incorporated in the objective function of the generative component as soft constraints [23] or as a seed to the optimization [39].

Hand tracking and pose estimation can benefit a lot from prior information in the form of spatial constraints. For example, if a fingertip detector provides 3D positions for the fingertips of the hand, this provides important constraints on the pose of the hand. Similar constraints can be defined if a hand interacts with a rigid object and hand-object contact points do not change. Existing generative and hybrid methods are able to incorporate such priors. However, they do so in a soft manner. More specifically, this is achieved by introducing an error term in the objective function they optimize, which quantifies how far a candidate solution is from satisfying these constraints. The contribution of this error term is then aggregated with all other error terms during optimization. This has two important, negative implications: (a) At the end of the optimization, it is not guaranteed that the solution sought satisfies the given constraints and, (b) during hypothesize and test, a lot of computational effort is wasted in evaluating hypotheses that do not satisfy the available constraints.

In this work, we address the aforementioned problems of existing methods. We present a generative hand tracking method that exploits efficiently available spatial constraints by considering them during the hypotheses generation stage (Fig.1). The particular type of constraints that we consider is knowledge of the 3D positions of end effectors (fingertips). Scenarios where such positions are available are quite common in practice. For free hand



(a) (b) (c)

Figure 2: Hand modeling: (a) hand geometry, (b) hand kinematics, (c) C-HMF hierarchy.

motion, fingertip detectors can reliably provide them when the fingers are visible [24]. When manipulating rigid objects without changing the contact points, these positions can be inferred by tracking the object, even when the fingers are fully occluded [41]. In order to best exploit the information about the end effector locations, we rely on the concept of *Reachable Distance Space* (RDS) [45]. RDS provides a fast method to generate hypotheses that respect the constraints. This way, we can significantly narrow the search on the high dimensional pose space. RDS-based sampling is used to extend the Hierarchical Model Fusion particle filter (HMF) [14, 15] to estimate the hand pose. HMF decomposes the hand's state according to the kinematic hierarchy (palm plus five fingers) and thus integrates nicely with the RDS provided hypotheses that also concern specific hand parts (fingertips).

Our contribution: Considering the presented literature review, the main contributions of this work are the following:

- We employ RDS to consider explicitly spatial and kinematic constraints at the hand pose hypothesis generation phase. In that direction, we propose a simple and fast method to consider the finger joint limits, extending the original RDS formulation [45] and, thus, rendering it suitable for the real-time performance requirenments of the hand tracking problem.

- We adapt the HMF framework [15] by tightly integrating our RDS-based, constraints-aware sampling strategy and propose the Constrained-HMF (C-HMF) method. This is shown to achieve state of the art hand tracking accuracy, while requiring the evaluation of much less hand hypotheses, all of which satisfy the given constraints.

2 Method description

We use a parametric 3D hand model (Fig. 2(a)) that can be articulated in 3D space. A given hypothesis about the hand configuration provides a hypothesis about the 3D location of every point of the hand model. The hand model is a skinned, anatomically consistent and visually realistic 3D mesh (1597 vertices) animated using a skeleton consisting of 20 bones. The configuration of each hand is represented by 27 parameters: three for the hand 3D position, four for the quaternion representation of the hand rotation, and four for the articulation angles for each of the five fingers.

2.1 C-HMF framework

The state of the hand model is estimated using an adapted version of the HMF tracking framework [14, 15] denoted as C-HMF. C-HMF follows the hypothesize and test approach. The generated hypotheses satisfy both the hand's kinematic constrains (motion model, joint limits) and the available end-effector target constraints. This way, all hypotheses are valid and sampling efficiency is greatly enhanced, therefore less particles are required to achieve the same tracking accuracy. An end-effector target can be either a specific 3D point or a 3D region in the case of uncertainty in the detection. In the later case, we randomly pick a specific 3D point within this region. Not all finger end-effectors are required to be associated with target constraints. For unconstrained fingers, we generate pose hypotheses that only respect the hand's kinematic constrains (motion model, joint limits).

The C-HMF framework follows the divide and conquer strategy to update the high dimensional hand state $\mathbf{x}_t$ at each frame, using several auxiliary models and one main model. Each of the auxiliary models estimates the state of a hand part. We use one auxiliary model for the palm (with 6-DOFs for its 3D position and orientation) and one for each finger (with 4-DOFs for the joint angles), as shown in Fig.2(b). The purpose of the main model is to combine and fine tune the poses estimated by the auxiliary models. The auxiliary models are organized in a hierarchy so that each one is able to provide information on the state of its parents in this hierarchy. We use a hierarchy with 3 levels. The top level contains the main model, the middle level contains the finger auxiliary models, and the bottom level contains the palm auxiliary model, as shown in Fig.2(c). We define the full state $\mathbf{x}_t$ at a time step t as the concatenation of the sub-states that correspond to the M auxiliary models and the main model $\mathbf{x}_{[0:M]t}$ and by $\mathbf{z}_t$ we denote the observations.

The C-HMF framework follows the Bayesian approach for tracking. By $\mathbf{x}_{0:t}$ we denote the state sequence $\{\mathbf{x}_0 ... \mathbf{x}_t\}$ and by $\mathbf{z}_{1:t}$ the set of all measurements $\{\mathbf{z}_1 ... \mathbf{z}_t\}$ from time step 1 to t. Tracking amounts to calculating the posterior $p(\mathbf{x}_{0:t}|\mathbf{z}_{1:t})$ at every step, given the measurements up to that step and a prior, $p(\mathbf{x}_0)$. Using the state decomposition, the solution is expressed as:

$$p(\mathbf{x}_{0:t}|\mathbf{z}_{1:t}) \propto p(\mathbf{x}_{0:t-1}|\mathbf{z}_{1:t-1}) \prod_i p(\mathbf{z}_t|\mathbf{x}_{[i]t}) p(\mathbf{x}_{[i]t}|Pa(\mathbf{x}_{[i]t})), \tag{1}$$

where $Pa(\mathbf{x}_{[i]t})$ denotes the parent nodes of $\mathbf{x}_{[i]t}$ (see Fig. 2(c)). In Eq.(1) we make the approximation that the observation likelihood is proportional to the product of individual model likelihoods $p(\mathbf{z}_t|\mathbf{x}_{[i]t})$.

To efficiently approximate the posterior given the above state decomposition, we use a particle filter that updates the sub-states. The algorithm approximates this posterior by propagating a set of particles for each model (auxiliary and main) using the importance sampling technique. The basic components of the filter are the state evolution dynamic model, the observation likelihood, and the proposal distribution that is used to sample from. The dynamic model that we consider for each sub-model $p(\mathbf{x}^{(n)}_{[i]t}|Pa(\mathbf{x}^{(n)}_{[i]t}))$ is a simple Gaussian model.

The observation likelihood has two components:

1. The rendering component $p(\mathbf{z}_{[ren]t}|\mathbf{x}^{(n)}_{[i]t})$ compares a hypothesized, rendered hand model and the RGB-D image as in [15]. The result of that comparison is a distance D_{ren} (normalized in [0, 1]) that takes into account the silhouette and depth match of the rendered hypothesis and the actual observations. The rendering likelihood is calculated as an

Algorithm 1 C-HMF Hand tracking Algorithm

Input: $\{\mathbf{x}^{(n)}_{[0:M]t-1}, \mathbf{w}^{(n)}_{t-1}\}_{n=1}^{N}$, $\mathbf{z}_t$.
for each model $i = 0$ to M **do**
  for each particle $n = 1$ to N **do**
    Constraints Aware Sample $\mathbf{x}^{(n)}_{[i]t}$ from $p(\mathbf{x}_{[i]t}|Pa(\mathbf{x}_{[i]t})^{(n)})p(\mathbf{z}_{[trg]t}|\mathbf{x}^{(n)}_{[i]t})$ (Section 2.2).
    Update its weight $\mathbf{w}^{(n)}_t$ using $p(\mathbf{z}_{[ren]t}|\mathbf{x}^{(n)}_{[i]t})$.
  end for
  Normalize the particle weights.
  Resample the particle set according to its weights.
end for
Output: $\{\mathbf{x}^{(n)}_{[0:M]t}, \mathbf{w}^{(n)}_t\}_{n=1}^{N}$.

exponential function of D_{ren}:

$$p(\mathbf{z}_{[ren]}|\mathbf{x}) = \exp\left\{-\frac{D^2_{ren}(\mathbf{z}_{[ren]}, \mathbf{x})}{2\sigma^2_{ren}}\right\} \tag{2}$$

2. The target likelihood component $p(\mathbf{z}_{[trg]t}|\mathbf{x}^{(n)}_{[i]t})$ is an exponential function of the average distance D_{trg} between the end-effector targets and the hypothesized end-effector positions with standard deviation σ_{trg}.

The total likelihood is given as the product of these two components.

The proposal distribution, described in detail in Sec. 2.2, generates particles that respect the dynamic model, and the end-effector position constraints when available. The state estimate for each frame $\bar{\mathbf{x}}_{[M]t}$ is given by the main model particle with highest weight. The steps of the algorithm are summarized in Alg. 1.

2.2 Constraints-aware hypotheses generation

Several techniques are integrated to generate constraints-aware hypotheses for each C-HMF sub-model (auxiliary and main) at each time step t.

The palm auxiliary model is updated first, according to the C-HMF hierarchy (Fig. 2(c)). We sample each palm particle from a Gaussian distribution centered at its position at the previous frame $t-1$. Subsequently, we apply rigid least-squares fitting to position the end-effectors close to their corresponding targets. To perform the fitting, given that the pose of the fingers is not yet updated, we augment the palm particle with the fingers pose as they were estimated at $t-1$. The two sets of points that we register in the least squares sense are the particle end-effector positions and their corresponding target positions. To avoid transformations that exceedingly relocate the hand's root joint, we append it in both sets.

For the particles of the finger auxiliary models, we sample from a proposal distribution $q(\mathbf{x}_{[finger]t}|\mathbf{x}_{[palm]t}, \mathbf{z}_{[trg]t}) = p(\mathbf{x}_{[finger]t}|\mathbf{x}_{[palm]t})p(\mathbf{z}_{[trg]t}|\mathbf{x}^{(n)}_{[i]t})$ which is conditioned on the updated palm sub-state and the finger target likelihood. This proposal generates valid kinematic samples that satisfy the end-effector target of that finger if available (see Section 2.2.1). In this step, only the finger joint angles are modified and fingertip targets can be reached

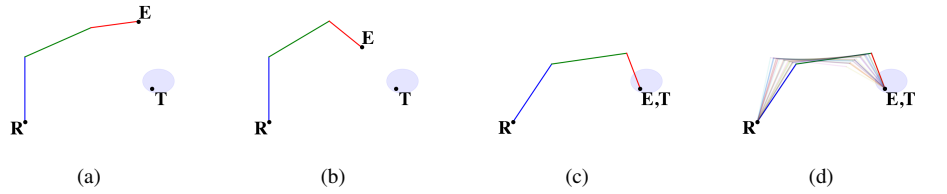


Figure 3: An illustration of the RDS-based sampling process. (a) A simple model of a finger, consisting of three links. R denotes the base of the finger, E the end effector and T the finger end effector target position picked from a target region (blue area). (b) RDS sampling defines the hinge joint angles so that $|RE| = |RT|$. (c) A rotation at the joint base brings E at T. (d) Different solutions in step (b) result in different finger configurations that respect the end effector constraints.

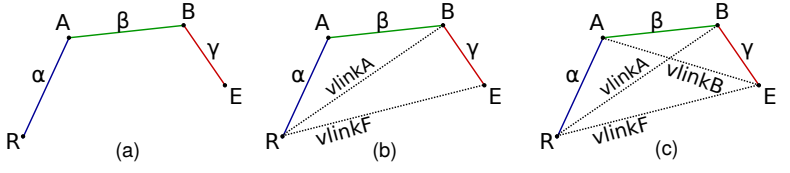


Figure 4: RD-Tree construction example on a 3-link chain (finger): (a) Initial chain, (b) RD-Tree, (c) RD-Tree augmented with *vlinkB*.

only if the corresponding palm pose is appropriate. For fingers with no associated end effector constraints, we sample from the palm-conditioned term of the proposal distribution $p(\mathbf{x}_{[finger]t}|\mathbf{x}_{[palm]t})$.

Finally, the main model samples are generated from a proposal distribution that is conditioned on the updated palm and finger pose provided by the auxiliary models and takes into account all the available end-effector target constraints (see Section 2.2.1).

2.2.1 Sampling in the Reachable Distance Space (RDS)

We present the method that we follow to generate samples for the finger joints given the palm position and orientation. The generated samples respect the hand dynamics and the end effector target constraints. The procedure has two steps which are illustrated in Fig. 3:

1. Sample a finger articulation (proximal inter-phalangeal joint, and distal inter-phalangeal joint) in the Reachable Distance Space which satisfies the target distance constraint. This step is detailed in the rest of the section.

2. Orient the finger by modifying its base (metacarpophalangeal) joint so that its end-effector lies in the line defined by the base-joint and the target. In this step, we consider the joint-angle limits of the finger base.

RDS description: The RDS sampling method [45] can efficiently sample serial kinematic chains with 1-DOF planar joints. Therefore, it is suitable for sampling the pose of each finger. The original RDS sampling scheme operates as follows. Considering a kinematic chain with several links (bones) we define the *virtual link* (*vlink*) as a sub-chain that joins two consecutive *vlinks* or links (see Fig. 4). The *reachable distances* or *Reachable Range*

(RR) of a *vlink* is the range of possible distances between its endpoints. When considering joints with angle limits, the RR of a *vlink* that is comprised of two actual links is given by the distance between its endpoints for the cases of minimum and maximum joint angle. RDS sampling is based on the reachable distance hierarchy denoted as *RD-Tree*. The *RD-Tree* is constructed by recursively joining the links of a chain into *vlinks* until a single root *vlink* is constructed (Fig. 4(b)). RDS sampling is performed by recursively sampling the lengths of the *vlinks* of the RD-Tree, starting from the root *vlink* and descending the hierarchy. After a *vlink* length is sampled, the available RRs of its sibling and children *vlinks* are restricted and have to be recalculated. Given the sampled *vlink* lengths, the configuration angles can be calculated by the law of cosines.

Finger RD-tree construction: The RD-Tree of a finger is visualized in Fig. 4. We construct *vlinkA* from the actual links α, β and we calculate its initial RR_A from triangle RAB and joint *A* angle limits. We construct the root vlink, *vlinkF*, from *vlinkA* and γ. The minimum and maximum joint A and B angles define the RR_F of *vlinkF*.

Assuming that the end-effector target and the finger base position are set, the length of the root *vlinkF* is determined. Therefore, the direct application of the recursive RDS sampling procedure reduces to sampling a single distance for *vlinkA*. Sampling *vlinkA* will always satisfy the limits of joint *A* since it is comprised of actual links. However, the joint limits of joint *B* are not guaranteed. In practice, the majority samples in RD-space that try to satisfy target distances near the minimum *RR* of the root *vlink* violate the limits of joint *B*.

Incorporating joint limits: To remedy this problem, we propose an alternative sampling procedure. For a target root *vlinkF* distance, we seek to restrict the RR_A of *vlinkA* to a range that sampled distances will not force joint *B* to violate its joint limits. Todo so, we augment the RD-Tree with an additional vlink. Specifically, *vlinkB* is constructed from the actual links β, γ and its initial RR_B is calculated from the triangle ABE and joint *B* angle limits (see Fig. 4(c)). Given this configuration, samples are drawn by the following steps: (a) Update the RR_A of *vlinkA* from the lengths of link γ and root *vlinkF* (triangle RBE), (b) update the RR_B of *vlinkB* from the lengths of link α and root *vlinkF* (triangle RAE), (c) update the RR_A of *vlinkA* from minimum and maximum RR_B lengths since these lengths uniquely determine *vlinkA* length, (d) sample the updated RR_A and, finally, (e) compute hinge joint angles from vlink distances.

This process guarantees that sampling in this updated Reachable Distance Space will result to configurations that do not violate any of the finger's hinge-joints limits since by construction *RRs* respect the limits and the subsequent steps do not expand them.

3 Experiments

We performed extensive experiments to assess the performance of the proposed approach. We evaluate the following methods: (a) **C-HMF**, the proposed approach, (b) **HMF**, the original **HMF** method [15] that tracks a hand without considering spatial constraints and, (c) **HMF-SP**, the **HMF** method with augmented likelihood that considers target positions as soft constraints. More specifically, the likelihood is defined as the weighted average of the rendering and the target constraint likelihoods:

$$p(\mathbf{z}|\mathbf{x}) = l\,p(\mathbf{z}_{[trg]}|\mathbf{x}) + (1-l)\,p(\mathbf{z}_{[ren]}|\mathbf{x}) \tag{3}$$

By experimentation we set l to 0.2. The standard deviation parameters for both likelihood components σ_{ren}, σ_{trg} are set to 0.005.

3.1 Datasets

For the qualitative evaluation of the methods we used real data obtained by an RGB-D sensor. For quantitative evaluations we used synthetic data since real world annotated data are difficult to obtain. We followed a common practice in the field [18, 19], that is, to first track real sequences and then use the tracking result as the basis for generating ground-truth annotated synthetic sequences by means of rendering.

End-effector target constraints are available in various hand tracking scenarios. Our datasets cover two such scenarios, one involving tracking a hand with known contact points to a planar surface and another with free hand motion.

Known contact points: In this scenario, it is assumed that a hand moves while some of the fingertips lie at known points on a planar surface. We provide three such sequences: **ALLFNG** where all the fingertips are constrained, **IDXMDL** where the index and middle finger are constrained, and **IDXTHM** where the index and the thumb are constrained.

Free hand motion: In this scenario, a detector provides the fingertip positions at each frame. Therefore, the number of the detected fingertips as well as the accuracy of the detections vary. One sequence is provided for this scenario, **FREEHM**.

3.2 Quantitative results on synthetic data

The synthetic dataset we used consists of the aforementioned 4 sequences containing a total of 1526 frames. The initialization of the methods is performed using the ground truth position for the first frame.

Evaluation criteria: Several error metrics are calculated. E_j measures the average distance between corresponding phalanx endpoints over a sequence. E_{ee} measures the average distance between corresponding end-effectors. E_{trg} measures the average distance only for the end-effectors that have been associated with constraints. Finally, the success rate C is defined as the ratio of the frames of the sequence for which the maximum position error of phalanx endpoints is below a certain threshold. For each experiment and method we measure and report the mean error of five individual runs.

Results for the known contact points scenario: Figure 5 plots the obtained results for all error metrics (columns) and sequences (rows). In all cases, the proposed **C-HMF** method (red curve) outperforms the baseline HMF variant as well as **HMF-SP**. **HMF-SP** performs better **HMF**, but the performance gain is not that significant. The discrepancy in accuracy between the proposed and the rest of the evaluated methods increases if we consider only the end effectors with constraints (3rd column) compared to all end effectors (2nd column) and all hand joints (1st column). However, the results of the 1st column suggest that **C-HMF** does not only improve the estimation of the 3D hand end effectors alone, but the full articulation of the hand.

Results for the free hand motion scenario: We simulated the limitations of a fingertip detector, that is, inaccurate detection of positions and missed detections. In a first experiment, we assessed the tolerance of **C-HMF** to errors in the estimation of the target constraints. To do so, in each frame we added Gaussian noise to the true positions of the fingertips. We considered the performance of **C-HMF** running with 40 particles, as well as of the baseline **HMF** method for two different computational budgets, that is 40 particles (**HMF-40**) and 200 particles (**HMF-200**). Figure 6 plots the obtained results for all error metrics (columns). It can be verified that for noise-free data, the **C-HMF** has 4 times smaller error that **HMF** when they both run with 40 particles. Even if the budget of **HMF** is increased to 200 particles

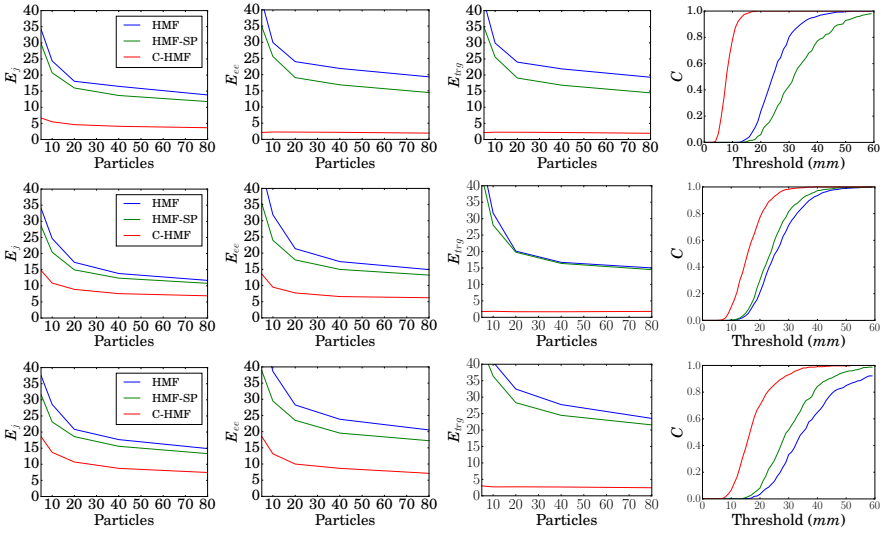


Figure 5: Error plots for the **C-HMF** (proposed, red) in comparison to **HMF** and **HMF-SP**. Figure rows correspond to different sequences, from top to bottom: **ALLFNG**, **IDXMDL**, **IDXTHM**. Columns correspond to the different error metrics.

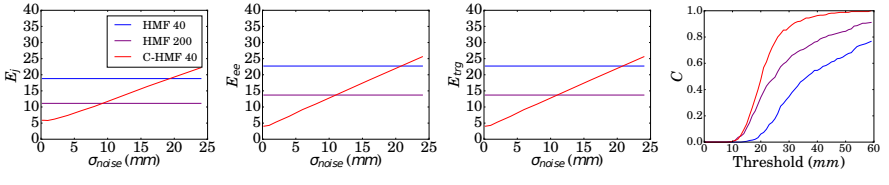


Figure 6: Error plots for the **C-HMF** (proposed, red) in comparison to **HMF** with 40 and 200 particles, for different levels of noise on the 3D position of the end effectors for the **FREEHM** dataset. Columns correspond to the different error metrics.

(5× budget of **C-HMF**), **C-HMF** maintains half the error. In order to match the performance of **HMF-40** and **HMF-200**, the standard deviation of the error in the estimation of the constraints should reach 20*mm* and 10*mm*, respectively.

In a second experiment, we constrained 2, 3, 4 and 5 of the 5 fingers. In each frame of the sequence, the actual ids of constrained fingers were selected randomly. We added Gaussian noise of standard deviation 8*mm* to the true positions of the fingertips. We run **C-HMF** five times for each different number of constrained fingers using 40 particles and measured E_j. E_j varied between 7.5 (5 constrained fingers) and 10.0mm (2 constraint fingers), showing that as the number of constraints increase, the accuracy in hand tracking also increases.

3.3 Qualitative results on real data

We evaluated our method qualitatively using real RGB-D data. Sample results are shown in Fig. 7. The results concern the **IDXMDL** and **IDXTHM** sequences and compare the

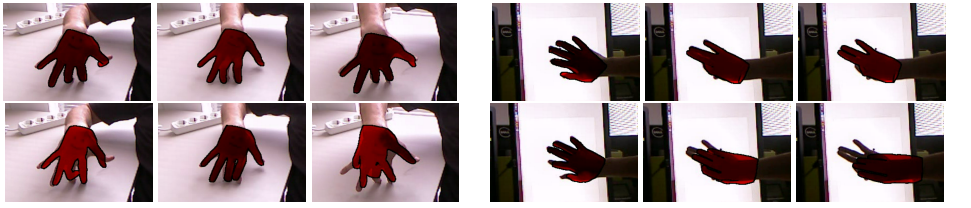


Figure 7: Qualitative results on datasets with known fixed contact points. Rows 1, 2: C-HMF (proposed) and HMF results, respectively. Left: results of the **IDXMDL** sequence, right: results on the **IDXTHM** sequence. Both methods use 80 particles.

proposed **C-HMF** to the **HMF** method. Both utilize 80 particles. In these sequences two fingers have known contact points with a planar surface while the rest can move freely. The results show that **C-HMF** estimates accurately the articulation of the constrained fingers even when they are partially, or even almost fully occluded. Furthermore, the constrained fingers provide anchor points for the palm whose pose is, therefore, better approximated. The state estimation for the rest of the fingers benefits from the better palm pose estimation. Further qualitative results are presented in the supplementary material accompanying this paper [1]. As can be noticed from these results the method fails mostly in cases of severe self-occlusions of fingers without constraints.

4 Summary and conclusions

In this paper, we proposed a novel 3D hand tracking method that explicitly considers constraints on the 3D locations of fingertips. Such constraints arise often, both in free hand motion and in hands interacting with other objects. Existing 3D hand tracking methods exploit such constraints in a soft manner, i.e., by considering them in the objective function they optimize. To the best of our knowledge, our approach is the first hypothesize-and-test method that constructs and evaluates candidate hand poses that are guaranteed to satisfy the available constraints. Extensive experiments on ground truth annotated data sets have shown that hand tracking accuracy is very much improved in comparison to methods that either use soft constraints or no constraints at all. Moreover, the proposed constraints-aware sampling explores more densely the space of feasible solutions. As a result, increased hand tracking accuracy is achieved with a lower number of candidate solution evaluations. Future research will focus on extending the type of employed constraints beyond end effectors/fingertips as well as on exploiting the developed tracking framework in the tracking of other articulated objects such as human bodies.

5 Acknowledgments

This work was supported by the EU FP7-ICT-2011-9 project WEARHAP and H2020-731869 project Co4Robots. The contributions of Iason Oikonomidis and Aggeliki Tsoli, members of CVRL are gratefully acknowledged.

[1] Supplementary material: https://youtu.be/DdXA-fslgpI

References

[1] Luca Ballan, Aparna Taneja, Jürgen Gall, Luc Van Gool, and Marc Pollefeys. Motion capture of hands in action using discriminative salient points. In *ECCV*, 2012.

[2] Matthieu Bray, Esther Koller-Meier, and Luc Van Gool. Smart particle filtering for high-dimensional tracking. *CVIU*, 2007.

[3] Teófilo Emídio de Campos and David W Murray. Regression-based hand pose estimation from multiple cameras. In *CVPR*, 2006.

[4] Shachar Fleishman, Mark Kliger, Alon Lerner, and Gershom Kutliroff. Icpik: Inverse kinematics based articulated-icp. In *CVPRW*, 2015.

[5] Henning Hamer, Konrad Schindler, Esther Koller-Meier, and Luc Van Gool. Tracking a hand manipulating an object. In *ICCV*, 2009.

[6] Henning Hamer, Juergen Gall, Thibaut Weise, and Luc Van Gool. An object-dependent hand pose prior from sparse training data. In *CVPR*, 2010.

[7] Cem Keskin, Furkan Kıraç, Yunus Emre Kara, and Lale Akarun. Hand pose estimation and hand shape classification using multi-layered randomized decision forests. In *ECCV*, 2012.

[8] Sameh Khamis, Jonathan Taylor, Jamie Shotton, Cem Keskin, Shahram Izadi, and Andrew Fitzgibbon. Learning an efficient model of hand shape variation from depth images. In *CVPR*, 2015.

[9] David Kim, Otmar Hilliges, Shahram Izadi, Alex D Butler, Jiawen Chen, Iason Oikonomidis, and Patrick Olivier. Digits: freehand 3d interactions anywhere using a wrist-worn gloveless sensor. In *ACM User interface software and technology*, 2012.

[10] Nikolaos Kyriazis and Antonis Argyros. Physically plausible 3d scene tracking: The single actor hypothesis. In *CVPR*, 2013.

[11] Nikolaos Kyriazis and Antonis Argyros. Scalable 3d tracking of multiple interacting objects. In *CVPR*, 2014.

[12] Peiyi Li, Haibin Ling, Xi Li, and Chunyuan Liao. 3D Hand Pose Estimation Using Randomized Decision Forest with Segmentation Index Points. In *ICCV*, 2015.

[13] Alexandros Makris and Antonis A Argyros. Model-based 3d hand tracking with on-line shape adaptation. In *BMVC*, 2015.

[14] Alexandros Makris, Dimitrios Kosmopoulos, Stavros Perantonis, and Sergios Theodoridis. A hierarchical feature fusion framework for adaptive visual tracking. *Image and Vision Computing*, 2011.

[15] Alexandros Makris, Nikolaos Kyriazis, and Antonis A. Argyros. Hierarchical particle filtering for 3D hand tracking. In *CVPRW*, 2015.

[16] Stan Melax, Leonid Keselman, and Sterling Orsten. Dynamics based 3d skeletal hand tracking. In *Graphics Interface*, 2013.

[17] Markus Oberweger, Paul Wohlhart, and Vincent Lepetit. Training a feedback loop for hand pose estimation. In *ICCV*, 2015.

[18] I. Oikonomidis, M.I.A. Lourakis, and A.A. Argyros. Evolutionary Quasi-Random Search for Hand Articulations Tracking. In *CVPR*, 2014.

[19] Iason Oikonomidis, Nikolaos Kyriazis, and Antonis A Argyros. Efficient model-based 3d tracking of hand articulations using kinect. In *BMVC*, 2011.

[20] Iason Oikonomidis, Nikolaos Kyriazis, and Antonis A Argyros. Full dof tracking of a hand interacting with an object by modeling occlusions and physical constraints. In *ICCV*, 2011.

[21] Iasonas Oikonomidis, Nikolaos Kyriazis, and Antonis A Argyros. Tracking the articulated motion of two strongly interacting hands. In *CVPR*, 2012.

[22] Paschalis Panteleris, Nikolaos Kyriazis, and Antonis A Argyros. 3d tracking of human hands in interaction with unknown objects. In *BMVC*, 2015.

[23] Georg Poier, Konstantinos Roditakis, Samuel Schulter, Damien Michel, Horst Bischof, and Antonis A. Argyros. Hybrid One-Shot 3D Hand Pose Estimation by Exploiting Uncertainties. In *BMVC 2015*, 2015.

[24] Chen Qian, Xiao Sun, Yichen Wei, Xiaoou Tang, and Jian Sun. Realtime and robust hand tracking from depth. In *CVPR*, 2014.

[25] Grégory Rogez, James S Supancic, and Deva Ramanan. First-person pose recognition using egocentric workspaces. In *CVPR*, 2015.

[26] Grégory Rogez, James S Supancic, and Deva Ramanan. Understanding everyday hands in action from rgb-d images. In *ICCV*, 2015.

[27] Tanner Schmidt, Richard Newcombe, and Dieter Fox. Dart: Dense articulated real-time tracking. *RSS*, 2014.

[28] Toby Sharp, Cem Keskin, Duncan Robertson, Jonathan Taylor, Jamie Shotton, David Kim, Christoph Rhemann, Ido Leichter, Alon Vinnikov, Yichen Wei, et al. Accurate, robust, and flexible real-time hand tracking. In *ACM Human Factors in Computing Systems*, 2015.

[29] Srinath Sridhar, Antti Oulasvirta, and Christian Theobalt. Interactive markerless articulated hand motion tracking using rgb and depth data. In *ICCV*, 2013.

[30] Srinath Sridhar, Helge Rhodin, Hans-Peter Seidel, Antti Oulasvirta, and Christian Theobalt. Real-time hand tracking using a sum of anisotropic gaussians model. In *3DV*, 2014.

[31] Srinath Sridhar, Franziska Mueller, Antti Oulasvirta, and Christian Theobalt. Fast and robust hand tracking using detection-guided optimization. In *CVPR*, 2015.

[32] Srinath Sridhar, Franziska Mueller, Michael Zollhöfer, Dan Casas, Antti Oulasvirta, and Christian Theobalt. Real-time joint tracking of a hand manipulating an object from rgb-d input. In *ECCV*, 2016.

[33] Xiao Sun, Yichen Wei, Shuang Liang, Xiaoou Tang, and Jian Sun. Cascaded hand pose regression. In *CVPR*, 2015.

[34] Andrea Tagliasacchi, Matthias Schröder, Anastasia Tkach, Sofien Bouaziz, Mario Botsch, and Mark Pauly. Robust articulated-icp for real-time hand tracking. In *Computer Graphics Forum*, 2015.

[35] Danhang Tang, Tsz-Ho Yu, and Tae-Kyun Kim. Real-time articulated hand pose estimation using semi-supervised transductive regression forests. In *ICCV*, 2013.

[36] Danhang Tang, Jonathan Taylor, Pushmeet Kohli, Cem Keskin, Tae-Kyun Kim, and Jamie Shotton. Opening the black box: Hierarchical sampling optimization for estimating human hand pose. In *ICCV*, 2015.

[37] Jonathan Taylor, Richard Stebbing, Varun Ramakrishna, Cem Keskin, Jamie Shotton, Shahram Izadi, Aaron Hertzmann, and Andrew Fitzgibbon. User-specific hand modeling from monocular depth sequences. In *CVPR*, 2014.

[38] Jonathan Taylor, Lucas Bordeaux, Thomas Cashman, Bob Corish, Cem Keskin, Toby Sharp, Eduardo Soto, David Sweeney, Julien Valentin, Benjamin Luff, et al. Efficient and precise interactive hand tracking through joint, continuous optimization of pose and correspondences. *ACM TOG*, 2016.

[39] Jonathan Taylor, Benjamin Luff, Arran Topalian, Erroll Wood, Sameh Khamis, Pushmeet Kohli, Shahram Izadi, Richard Banks, Andrew Fitzgibbon, Jamie Shotton, Lucas Bordeaux, Thomas Cashman, Bob Corish, Cem Keskin, Toby Sharp, Eduardo Soto, David Sweeney, and Julien Valentin. Efficient and precise interactive hand tracking through joint, continuous optimization of pose and correspondences. *ACM Transactions on Graphics*, 2016.

[40] Jonathan Tompson, Murphy Stein, Yann Lecun, and Ken Perlin. Real-time continuous pose recovery of human hands using convolutional networks. *ACM TOG*, 2014.

[41] Tu-Hoa Pham, Abderrahmane Kheddar, Ammar Qammaz, and Antonis A. Argyros. Towards force sensing from vision: Observing hand-object interactions to infer manipulation forces. In *CVPR*, 2015.

[42] Dimitrios Tzionas, Luca Ballan, Abhilash Srikantha, Pablo Aponte, Marc Pollefeys, and Juergen Gall. Capturing hands in action using discriminative salient points and physics simulation. *IJCV*, 2015.

[43] Chengde Wan, Angela Yao, and Luc Van Gool. Direction matters: hand pose estimation from local surface normals. *arXiv:1604.02657 [cs]*, 2016.

[44] Yangang Wang, Jianyuan Min, Jianjie Zhang, Yebin Liu, Feng Xu, Qionghai Dai, and Jinxiang Chai. Video-based hand manipulation capture through composite motion control. *ACM TOG*, 2013.

[45] Xinyu Tang, Shawna Thomas, Phillip Coleman, Nancy M. Amato, Xinyu Tang, Shawna Thomas, Phillip Coleman, and Nancy M. Amato. Reachable Distance Space: Efficient Sampling-Based Planning for Spatially Constrained Systems. *The International Journal of Robotics Research*, 2010.

[46] Chi Xu and Li Cheng. Efficient hand pose estimation from a single depth image. In *ICCV*, 2013.