Evolutionary Quasi-random Search for Hand Articulations Tracking

Iason Oikonomidis[1,2] , Manolis I.A. Lourakis[1] , Antonis A. Argyros[1,2]
[1] Computational Vision and Robotics Laboratory, Institute of Computer Science, FORTH, Greece
[2] Computer Science Department, University of Crete
{oikonom,lourakis,argyros}@ics.forth.gr

Abstract

We present a new method for tracking the 3D position, global orientation and full articulation of human hands. Inspired by recent advances in model-based, hypothesize-and-test methods, the high-dimensional parameter space of hand configurations is explored with a novel evolutionary optimization technique. The proposed method capitalizes on the fact that the quasi-random samples of the Sobol sequence have low discrepancy and exhibit a more uniform coverage of the sampled space compared to random samples obtained from the uniform distribution. The method has been tested for the problems of tracking the articulation of a single hand ($27D$ parameter space) and two hands ($54D$ space). Extensive experiments have been carried out with synthetic and real data, in comparison with state of the art methods. The quantitative evaluation shows that for cases of limited computational resources, the new approach achieves a speed-up of four (single hand tracking) and eight (two hands tracking) without compromising tracking accuracy. Interestingly, the proposed method is preferable compared to the state of the art either in the case of limited computational resources or in the case of more complex (i.e., higher dimensional) problems, thus improving the applicability of the method in a number of application domains.

1. Introduction

Articulated motion estimation and tracking is an important problem in computer vision with significant theoretical interest and numerous and diverse applications. The instances regarding the human body and the human hand are of particular interest since their solution can support the development of a number of important applications in the fields of human-computer interaction (HCI), robot learning by demonstration, etc.

Significant efforts have been devoted in solving these problems [11, 6]. Despite the large body of related work, these problems are still attracting the attention of the research community because of the numerous difficulties that one has to overcome to solve them in the general case. As an example, for the problem of hand tracking, the uniform color appearance of the hand makes it difficult to segment its parts. Its high versatility and dexterity yields an infinite number of configurations in most of which the hand is considerably self-occluded. This limits the applicability of approaches that rely on the detection of a few recognizable poses. Finally, the high speed of the human hand and fingers challenge most of the contemporary tracking approaches.

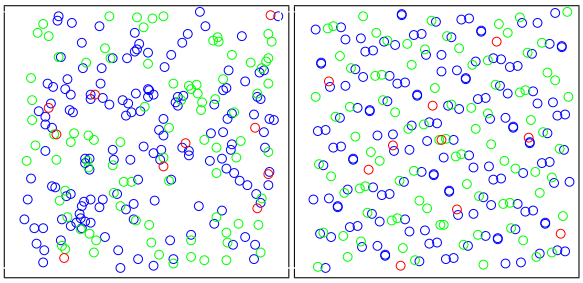


Figure 1. Left: 256 points on the $2D$ plane obtained from a pseudo-random number generator. Right: the first 256 samples of the Sobol sequence. Samples 1 to 10, 11 to 100 and 101 to 256 are in red, blue and green colors, respectively. The Sobol sequence covers the space more evenly. In our problem formulation, the Sobol sequence is used to form quasi-random hypotheses of hand configurations in the $27D$ (single hand) and $54D$ (two hands) configuration spaces. Example inspired from http://en.wikipedia.org/wiki/Sobol_sequence.

There are two broad categories of 3D hand articulation tracking methods, the *appearance-based* and the *model-based* ones (see Sec. 2). Appearance-based methods are more computationally efficient but can only identify a limited number of known hand configurations. Model-based methods explore the continuous space of hand configurations at the cost of solving online an optimization problem in a high dimensional space. Recent formulations of the hand tracking problem, together with the utilization of parallel hardware such as the GPUs, yielded model-based approaches that perform in near real time [15]. Still, the computational requirements of model-based methods at runtime remain much larger than those of appearance-based ones.

In this paper we present a novel model-based approach for tracking the articulated motion of human hand(s) as observed by an RGB-D sensor. The main contribution of this work is the proposal of a novel optimization method that explores very efficiently and effectively the high-dimensional configuration space of the human hand. The new optimization technique capitalizes on quasi-random sampling [24] to select a limited number of candidate problem solutions that span uniformly (see Fig. 1) the configuration space of the human hand. The advantage of quasi-random over pseudo-random sampling is a more uniform coverage of the sampled space, because of the low-discrepancy property exhibited by quasi-random samples. Articulated structures such as the human body and the human hand are parameterized in spaces of tens of dimensions that need to be sampled by hypothesize-and-test methods. Thus, it is expected that the low-discrepancy property of quasi-random sampling is beneficial to the effective exploration of such high dimensional spaces. Quasi-random sampling has been employed for human silhouette tracking [17] but not for human body or human hand tracking.

Evolutionary optimization algorithms have also been successfully applied before in human body tracking [26] and human hand tracking [16, 15], but never combined with quasi-random sampling. The optimization method also takes advantage of knowledge in the problem domain, namely the kinematic chain of the human hand and the way this structure affects the scoring of neighboring hand poses. The resulting optimization method has a number of free parameters whose values are systematically determined through meta-optimization.

The proposed method has been employed to track the articulation (a) of a single hand and (b) of two strongly interacting hands. The parameter spaces of these problems are 27D and 54D, respectively. For both problems, the proposed approach has been evaluated quantitatively and qualitatively in data sets annotated with ground truth and in comparison with state of the art model-based methods [16, 15] that rely on the Particle Swarm Optimization (PSO) [9] method. The obtained results reveal that the proposed approach can achieve a speed-up of up to $4\times$ for the case of a single hand and up to $8\times$ for the case of two hands, without compromising tracking accuracy. The speed-up gains are more significant in the hard cases, i.e., in the case of limited computational budgets. This is because lower budgets are associated with a sparser sampling of the parametric space of the problem, in which case the low-discrepancy property of the quasi-random sampling becomes more important. Thus, besides its theoretical significance, the proposed optimization method has a great practical importance, as it bridges the gap in computational performance with appearance-based methods while maintaining the advantages of the hypothesize-and-test methods.

2. Related work

Hand articulation tracking methods can be classified [5, 15] based on the way candidate solutions are generated and tested against the observations. For the so called appearance-based approaches [2, 21, 27, 20, 10], a large set of hand configurations is generated off-line. Relevant features are extracted for each of the generated poses, resulting in a database where each pose is associated with image features. At runtime, comparable features are extracted from the acquired image(s) and searched for in the precomputed database. The solution reported is one or more of the stored poses that match the computed features.

Model-based methods [19, 25, 5, 7, 15, 3, 16] generate hand poses, extract features and compare them to the observed ones at runtime. Typically, an optimization method is responsible for finding the pose that best explains the available observations.

Appearance-based methods are more computationally efficient than model-based ones, at the cost of having a fixed accuracy that depends on the precomputed data rather than the computational budget devoted to the problem at runtime. Furthermore, they are more difficult to adapt to different problems, since changing the object to be tracked essentially amounts to solving the time consuming problem of generating new off-line data. As an example, moving from a single hand tracker to a two hands tracker requires training to be performed from scratch. In contrast, model-based methods can be more easily adapted to different scenarios, since all that is required is a change in the appearance and kinematics model used. Furthermore, the accuracy usually benefits from more computational budget, or alternatively one can trade accuracy for speed. In general, the computational requirements of model-based methods at runtime is larger than that of appearance-based ones. However, recent formulations of the problem that also exploit parallel hardware such as the GPUs have resulted in methods that perform in near real time [15].

Quasi-random sampling has been applied before in computer vision problems such as silhouette tracking [17], block motion estimation [18] and map estimation [22]. Loosely speaking, the idea behind quasi-random sampling is that a carefully constructed sequence of points in the S-dimensional hypercube $[0, 1]^S$ may cover more uniformly this space than the commonly employed pseudo-random number generators. Specifically, such sequences exhibit the so called low-discrepancy property [12] according to which, any subset of the sampled hypercube has a probability of containing samples that is proportional to its volume. Although this property holds at the limit for standard uniform sampling too, the variance for low numbers of quasi-random samples is lower (see Fig. 1). The low-discrepancy property was first introduced by Sobol in [24], where he also presented the first low-discrepancy family of

sequences. In a later work [23], Sobol provided arithmetic values to construct such sequences up to dimensions of size 51. Even later [12], Niederreiter categorized and systematically studied low-discrepancy sequences. He also presented a sampling-based optimization algorithm [14]. The idea behind that approach was to keep sampling the search space using a low-discrepancy sequence until either a fixed number of objective function evaluations is reached or a fixed objective function score is found.

Evolutionary optimization is widely regarded as a powerful strategy to optimize objective functions with significant amounts of noise, discontinuities and even uncertain values [8, 4]. The model-based formulation of hand tracking results in objective functions that exhibit all these properties. Thus, commonly employed optimization techniques such as gradient descent fail to provide robust solutions. Particle Swarm Optimization (PSO) [9], a widely used evolutionary algorithm was shown very effective in this problem [15, 16]. Inspired from the success of PSO, we propose a novel evolutionary algorithm which takes advantage of the power of quasi-random sampling and is able to surpass PSO in the rate of convergence and estimation accuracy.

3. Method Description

We present an evolutionary algorithm, capable of efficiently searching the high-dimensional parameter space of hand configurations. Briefly stated, the method operates as follows. First, quasi-random sampling is used to define a number of atoms / candidate solutions / model hypotheses[1] in the parametric space of hand configurations. An objective function quantifies the difference between a hand hypothesis and the actual observations. Then, the computation proceeds in iterations called generations. At each such generation, a "center" hand hypothesis is computed as a weighted average of a fixed number of the so far best scoring hypotheses. Quasi-random sampling is again employed to sample effectively the parametric space around this center point, yielding new candidate solutions to the problem. The range of the sampling around the center is diminished exponentially as a function of the number of generations. The hypothesis that yields the best score after a fixed number of generations is termed the solution for a given frame. To achieve tracking over time, temporal continuity is exploited, in the sense that the solution for frame $t-1$ bootstraps the optimization process for the frame t.

3.1. Sobol Sequence

Sobol [24] introduced a low-discrepancy sequence of n samples x_i in the S-dimensional hypercube $[0..1]^S$ with the aim to approximate the integral

$$\int_{[0,1]^S} f(x)\,\mathrm{d}x \tag{1}$$

of an arbitrary function f over $[0,1]^S$ by the limit

$$\lim_{n\to\infty} \sum_{i=1}^{n} f(x_i) \tag{2}$$

with the fastest possible convergence. Loosely speaking, for small sample sizes, in the order of tens or hundreds of samples, the resulting coverage of the sampled space is more even, leaving smaller gaps than that of a set of points sampled from a uniform distribution (see also Fig. 1). The comparison favors quasi-random sampling as the number of space dimensions S is increased. For a more detailed and formal presentation of low-discrepancy and quasi-random sequences, the reader is referred to [13].

[1] In the remainder of the paper, the terms atom, candidate solution and model or hand hypothesis are used interchangeably.

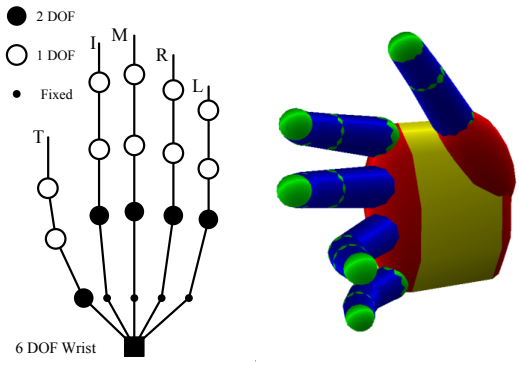


Figure 2. (a) The kinematic structure and (b) a 3D rendering of the hand model used in this work.

3.2. Hand Modeling and Hypothesis Scoring

We adopt the parameterization of hand pose and the modelling of hand kinematics and appearance that are proposed in [15]. In that parameterization, a fixed point and orientation in the palm defines the global pose of the modeled hand. This is six degrees of freedom (DoFs) encoded as seven parameters because of the use of the quaternion representation for 3D rotation. Each of the fingers is fully described by four joint angles, two for the saddle joint at the base and two more, one for each of the two remaining hinge joints. The schematic of this parameterization is illustrated in Fig. 2(a) which is taken from [6] (Fig. 2(b) of that manuscript) and adjusted to the model we use. This results in a total of 26 DoFs encoded as a vector $h \in \mathbb{R}^{27}$. Appropriate limits for the joint angles are determined by anatomical studies [1]. Regarding modelling the appearance of the hand, Oikonomidis *et al*. [15] propose to build the necessary shapes from appropriately transformed instances of

two geometric primitives, cylinders and spheres, resulting in a shape depicted in Fig. 2(b). This strategy enables a high degree of computational parallelism, a fact that can yield significant speed boost with an implementation for modern GPUs.

Using this appearance model and given a specific hand pose h along with the camera calibration information K, it is possible to synthesize feature maps of the hand hypothesis h from the viewpoint of the camera that observes the scene. Specifically, we synthesize the depth map and the skin color segmentation map, two features that can be obtained from the RGB-D data we use as input. A scoring function E is then used to quantify the discrepancy between a hypothesis and the actual observations. We employ the one proposed in [15] (Eq. (1) of that manuscript) which depends on the compatibility of the skin color and depth maps associated with an observed hand and a rendered hand hypothesis. More specifically, the mean of absolute differences between the observed and hypothesized depth maps is computed. A quantity that measures the agreement between the observed and hypothesized skin color silhouettes is also computed. These two values quantify the agreement in appearance between hypothesis and observation. From these values, a single score is computed using experimentally determined weights. An additional term that does not depend on appearance but instead serves to penalize anatomically infeasible poses is also appropriately weighted and added, forming the final definition of E. For a detailed description, the reader is referred to [15].

The task of estimating the hand pose is thus reduced to that of optimizing the objective function E. The computation of E requires as input the hand configuration h, the camera calibration information K and the observation o, however, for notational brevity we use $E(h)$ to denote $E(h, o, K)$. For a given observation, the function E is optimal at the configuration that represents the observed hand pose. However E also exhibits local minima and discontinuities because of aliasing artifacts. On top of this, a graphics/rasterization engine is involved in the computations, so the analytical differentiation of E is impossible.

3.3. Evolutionary Sobol Search for Hand Tracking

As in most evolutionary optimization algorithms, in our approach there is the notion of a population of candidate solutions in the search space. This population evolves in steps, called generations. A high-level outline of the proposed search algorithm is presented in Algorithm 1.

The algorithm maintains the full history H of all atoms identified so far, along with their corresponding fitness scores w, and runs for a fixed number G of generations. In each generation g, $0 \leq g \leq G-1$, a center position h_C is defined. For the first generation, h_C is set equal to the solution h_{t-1} sought for the previous frame. N atoms h_t^i are

Algorithm 1 The proposed evolutionary search algorithm

Input: The solution h_{t-1} for the previous frame.
Output: The solution h_t for the current frame.

$H \leftarrow \emptyset$; $T \leftarrow \emptyset$;
$h_C \leftarrow h_{t-1}$;
for $g = 0 \ldots G-1$ **do**
  // Define atoms h_t^i $(1 \leq i \leq N)$ around h_C (Eq.(3))
  $\{h_t^i\} \leftarrow SobolSequence(h_C, N, g)$;
  // compute E for atoms and store fitness
  $H \leftarrow H \cup \{h_t^i\}$;
  $w(h_t^i) = E(h_t^i)$;
  $T \leftarrow TopScoringAtoms(H, N_T)$;
  $h_C \leftarrow WeightedSum(T)$; (Eq.(4))
end for
$h_t \leftarrow TopScoringAtoms(H, 1)$
return (h_t);

then defined around h_C based on the Sobol sequence. This is done so as to take advantage of the way quasi-random sampling can evenly fill high-dimensional spaces. Each parameter dimension has different units and range, so a vector s of scales is used to adjust the original range $[0, 1]$ of the Sobol sequence to the appropriate one. More specifically,

$$h_t^i = h_C + s \circ c^g \circ (2x_{r+i} - 1). \tag{3}$$

In Eq.(3), i iterates over the population count, "$\circ$" denotes the Hadamard or entry-wise product between vectors, x_n is the n-th sample of the Sobol sequence of appropriate dimensions, and r is a large random integer after which we draw samples from the Sobol sequence. c is a vector of contraction coefficients, with entries in the range $(0..1]$. Raising to g denotes entry-wise power. The goal of this operation is to reduce the size of the search space around h_C as a function of the generation count.

All the identified atoms h_t^i are inserted in the history H. The objective function E is consequently evaluated for each of these atoms h_t^i, resulting in corresponding fitness scores $w(h_t^i) = E(h_t^i)$. Next, from the whole history H, the set T containing n_T atoms with the highest fitness scores is computed. A new center in the search space h_C is computed as a weighted sum of these n_T atoms, as follows:

$$h_C = \frac{1}{\sum_{h \in T} q(w(h))} \sum_{h \in T} q(w(h)) \cdot h. \tag{4}$$

We chose $q(x) = \exp(ax)$ because a can be appropriately chosen to scale the weights so that there is a fixed ratio between the first and second best scoring atoms.

The above procedure is repeated for all G generations. After this computation is completed, the most fit atom among the whole history H is reported as the result h_t of the optimization process.

From a computational complexity point of view, the most expensive part of the algorithm is the evaluation of the objective function for a given atom. N such evaluations are performed in each generation, thus the product $N \cdot G$ determines the computational budget of the method. It should be noted that within each generation, the computations for each atom are independent of the other atoms. This inherent computational parallelism can be exploited to achieve very efficient implementations in GPU architectures.

3.4. Meta-optimization

The algorithm outlined in the previous section has a number of free parameters, namely the scaling vector s, the contraction coefficients c, the weight parameter a and the number N_T of top scoring positions that contribute to the calculation of h_C. In order to determine appropriate values for these parameters in a systematic way, we resorted to meta-optimization, i.e., the use of an optimization algorithm in order to tune the parameters of another.

For the case of tracking a single hand, we first recorded a sequence of 370 frames of a hand waving and performing object grasp like motions. We tracked this sequence using the method of [15] with a high computational budget, to ensure the highest possible tracking accuracy. We then synthesized the same sequence using our hand model. Having a sequence of synthesized images along with corresponding hand poses as ground truth, we were able to quantify the performance of a given parameterization of the search algorithm. For more details on this quantification the reader is referred to Section 4.1. To meta-optimize the two hand tracking problem, we followed a similar approach on a sequence showing two hands in strong interaction.

The parameterization of the meta-optimization problem itself, is as follows. We partitioned the scaling vector s in three types of parameters, namely the positional scale, the rotational scale and the scale associated with finger joint angles. Thus, we reduced the 27 or 54 parameters (single/two hands tracking) to just 3. The intention behind not keeping all the different parameters is to avoid over fitting for the specific sequences we used for meta-optimization.

The contraction coefficients c were partitioned with respect to their distance from the root of the kinematic chain, a choice that reflects the way the scoring function E is affected by each of the problem parameters. The intuition is that the parameters describing the position and orientation of the palm must be fixed in order to measure meaningful values when varying the position of, e.g., a fingertip. We thus identified four different levels, starting with position and rotation in the root(palm), the DoFs of the metacarpophalangeal joints at the next level, the proximal interphalangeal (IP) joints at the third level and the distal IP at the last level (bottom to top in Fig. 2(a)). We did not optimize for N_T, in all experiments we used $N = N_T = 16$ and

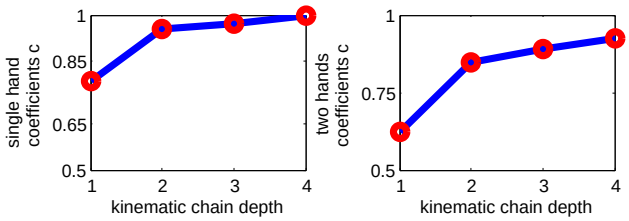


Figure 3. The results of meta-optimization yielded an exponential relation between the kinematic chain depth and the contraction coefficients c for both cases, single hand tracking and two hands tracking.

$G = 25$. The three different scale parameters, the four different contraction coefficients and the weighting parameter a amount to a total of 8 parameters. In order to find optimal values for those parameters we employed Particle Swarm Optimization (PSO) [9].

The separate meta-optimization of the single hand case and the case of two hands, resulted in two different sets for these 8 parameters and in some rather interesting results. For both the single hand and the two hands case, in close agreement with intuition, the optimum contraction coefficients decrease exponentially with the distance from the kinematic root (see Fig. 3). Furthermore, for the single hand case it turned out that the optimal value of a is very close to 0. Thus, the top N_T atoms contribute to the definition of the center h_C with equal weight. This however was not the case for the case of two hands tracking. For that case the optimum value of a was close to 1.6.

4. Experimental Evaluation

We present experiments that assess the effectiveness of the proposed evolutionary quasi-random search approach. The proposed method was employed to track the articulation of (a) a single hand and (b) of two strongly interacting hands. The dimensionality of these two problems is 27 DoFs and 54 DoFs, respectively. The results obtained by the proposed method are compared quantitatively and qualitatively to those obtained by state of the art methods ([15] for (a) and [16] for (b)). For quasi-random sampling we used the default Matlab implementation of the Sobol sequence of appropriate dimensions (either 27 or 54).

4.1. Quantitative evaluation, single hand tracking

We conducted several experiments to quantitatively assess the performance of the proposed method in comparison with the approach presented in [15]. In order to do so, we created synthetic data (i.e., annotated with ground truth) in a way similar to that described in Sec. 3.4.

Specifically, a real-world sequence was tracked with good accuracy using the method in [15] and a large com-

putational budget. This sequence consists of 200 frames depicting a hand performing a variety of motions. It should be stressed that this sequence is different to the one used for the meta-optimization (see Sec. 3.4). The resulting track is a sequence of hand poses, closely resembling the observed hand. We used this track to generate synthetic data, i.e. a sequence of synthetic RGB-D images. Since this sequence is produced from a known hand track, we can use that track as the ground truth for that sequence.

Having a sequence with associated ground truth, we compute the distance of a track from this ground truth. To do so, we select 21 key points on the hand model we use. The first point is at the root of the kinematic chain inside the palm. Each finger has 4 of the remaining 20 points, starting with one at the base, having one at each of the intermediate joints, and with the last placed at the fingertip. This placement of points can be computed for any given hand pose. Given two hand poses, a ground truth and an estimated one, we can compute the Euclidean distances between such corresponding points. The mean value of all these distances is the error measure we use for a pose estimation of a given frame. For a pose sequence we compute the mean value of such mean distances, resulting in a single error estimate for the whole sequence. Due to its stochastic nature, our algorithm does not perform identically in different runs. Thus, for a given configuration we repeat the experiment 11 times, yielding 11 different mean errors. The values we report in all experiments are medians of those mean errors.

As stated in Sec. 3.3, the two parameters determining the computational budget of the algorithm are the number of atoms N and the number of generations G because their product yields the number of objective function evaluations. Particle Swarm Optimization (PSO), the optimization algorithm used in [15] is parameterized similarly by the number of its atoms called particles and generations. We assessed the performance of our algorithm in comparison to [15] as a function of these two parameters. The results of this experiment are visualized in Fig. 4.

It can be verified that the proposed evolutionary method performs better or equal to the method in [15]. This is amenable to dual interpretation: we either get more accuracy with the same computational budget or we get the same accuracy with less computational resources. The differences in favor of the proposed approach become far more striking in small atom and generation counts. This is quite important because it means that higher accuracy can be achieved for small computational budgets. As an example, the accuracy obtained by 16 atoms of the proposed approach running for 10 generations is equal to the accuracy obtained by 64 particles of [15] for the same number of generations. Given that the objective function E is common for both methods, the evaluation of an atom in our approach is identical to the evaluation of a particle in [15]. This means

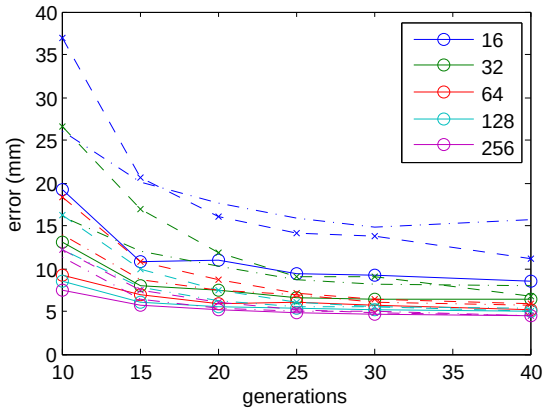


Figure 4. The performance of our method (solid lines) in comparison to that of [15] (dashed) for the problem of single hand tracking and for different particle and generation counts. The dotted lines illustrate the performance of the proposed evolutionary scheme using uniform random instead of quasi-random sampling (best viewed in color).

that the proposed algorithm achieves a $4\times$ speed-up over the state of the art. As an alternative view of the same results, for the same low budget (16 atoms/particles, 10 generations), the proposed approach is almost two times more accurate. In the same graph we assess the importance of using quasi-random sampling instead of uniform. The dotted graphs exhibit the performance of a variant of the proposed method with the only difference being that for this experiment we drew samples from the uniform distribution instead of quasi-random sampling. Evidently the behavior of the proposed method is better than this variant, and again the lower budgets exhibit the bigger differences.

Figure 6 shows sample results from the application of the proposed algorithm on the synthetic sequence used in this quantitative evaluation.

4.2. Quantitative evaluation, two hands tracking

Similarly to the case of a single hand, we recorded a sequence showing two hands in strong interaction. We employed the proposed evolutionary quasi-random search algorithm in a parametric space of 54 DoFs and we experimented with different numbers of atoms and generations. The obtained results are shown in Fig. 5 in comparison with those obtained by the method of [16].

The complex interaction between hands generates more occlusions, so, it is impossible to resolve ambiguity regarding some poses of the sequence. This, in turn, implies that the lowest achievable error for the case of two hands is higher than that of the single hand case. Nevertheless, the advantages of the proposed method are even more prominent in the case of tracking two strongly interacting hands.

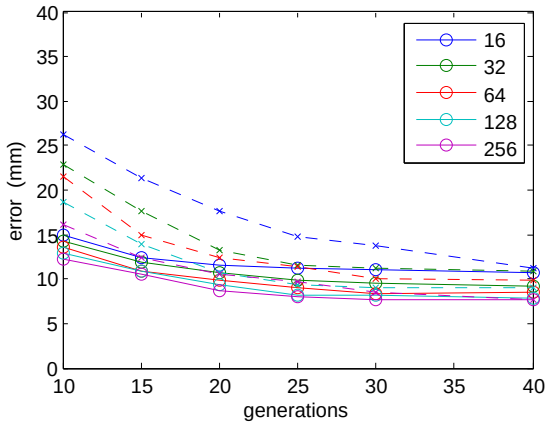


Figure 5. The performance of our method (solid lines) in comparison to that of [16] (dashed) for the problem of tracking two strongly interacting hands and for different particle and generation counts (best viewed in color).

The lowest budget configuration we tested, $N = 16$ and $G = 10$ was able to achieve an average error of $15mm$ whereas the method of [16] achieved for the same budget an average error of $26mm$. The proposed method achieved for the configuration of $N = 64$ and $G = 25$ an error of 8.9, within $1.5mm$ from the largest budget we tested, namely $(N, G) = (256, 40)$ which yielded $7.6mm$ of average error. Thus, for the case of two hands tracking, the proposed solution can achieve a speed-up of almost $8\times$, enabling real-time tracking of two interacting hands.

Figure 6 shows sample results from the application of the proposed algorithm on the synthetic sequence used in this quantitative evaluation.

4.3. Qualitative Evaluation

Figure 7 shows sample results from the application of the proposed method on the real world sequences reported in [15, 16]. The videos of these experiments are available at http://youtu.be/3yvaFuX09xY.

5. Discussion

The model-based, hypothesize-and-test methods for tracking the articulated motion of human hands have a number of important advantages over their appearance-based counterparts. Their Achilles' heel is their computational requirements. Recent progress in the field [15] achieves near real time performance ($20Hz$) in a GPU-powered, high end computer. In this work, we proposed a novel hypothesize-and-test method for this problem. The evolutionary optimization core of the method relies on the fact that low discrepancy sequences like the Sobol sequence are better suited for uniformly sampling high dimensional spaces. The method is tailored to the problem of hand tracking in the sense that it capitalizes on the tree structure of the kinematics of the human hand. Quantitative experimental results for two problems of different dimensionality (single hand tracking, two hands tracking) demonstrated that the proposed approach exhibits a much better performance/accuracy trade-off compared to the state of the art. More specifically, it is demonstrated that single hand tracking can be speeded up by a factor of $4\times$, and two hands tracking can be speeded up by a factor of $8\times$, without sacrificing tracking accuracy. Ongoing research focuses on applying the evolutionary quasi-random search method to other computer vision optimization problems.

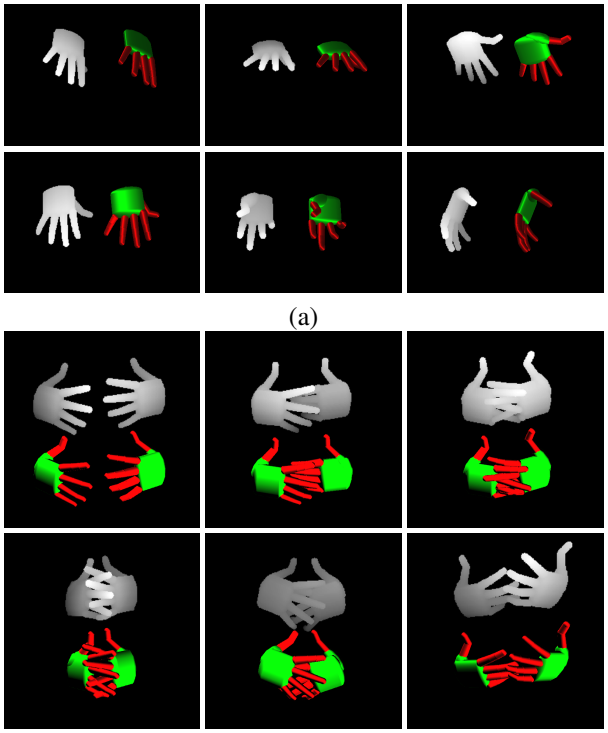


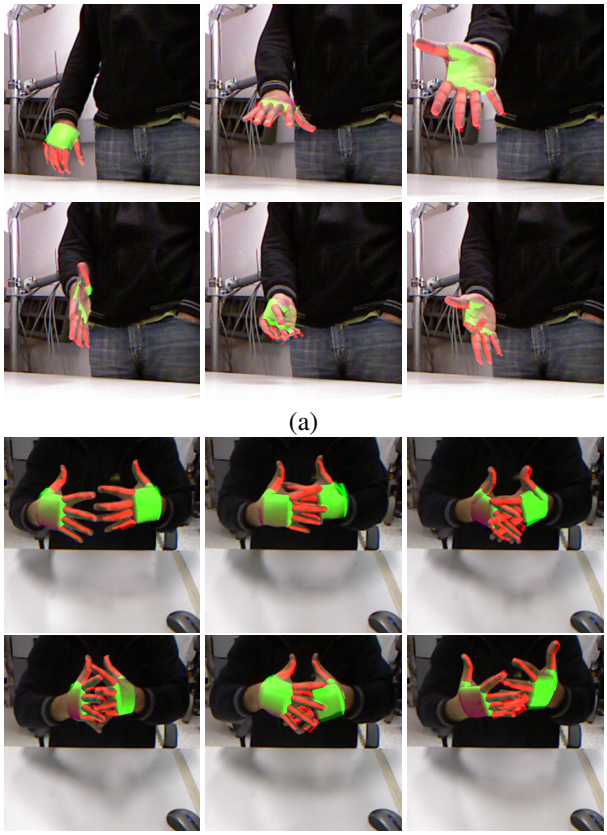
Figure 6. Sample results from the application of the proposed evolutionary quasi-random search method to (a) the single hand tracking and (b) two hands tracking synthetic data sets. For each frame, the rendered depth map and the estimated hand pose is shown.

Acknowledgments

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

References

[1] I. Albrecht, J. Haber, and H. Seidel. Construction and animation of anatomically based human hand models. In *2003 ACM SIGGRAPH/Eurographics symposium on Computer Animation*. Eurographics Association, 2003. 3



(a) (b)

Figure 7. Sample results from the application of the proposed evolutionary quasi-random search method to (a) the single hand tracking and (b) two hands tracking real world sequences reported in [15] and [16], respectively.

[2] V. Athitsos and S. Sclaroff. Estimating 3D Hand Pose From a Cluttered Image. In *CVPR*, pages II–432–9. IEEE, 2003. 2

[3] L. Ballan, A. Taneja, J. Gall, L. Van Gool, and M. Pollefeys. Motion capture of hands in action using discriminative salient points. In *ECCV*, pages 640–653, 2012. 2

[4] K. A. De Jong. *Evolutionary computation: a unified approach*, volume 262041944. MIT press, 2006. 3

[5] 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. 2

[6] 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. 1, 3

[7] H. Hamer, K. Schindler, E. Koller-Meier, and L. V. Gool. Tracking a hand manipulating an object. In *ICCV*, Oct 2009. 2

[8] Y. Jin and J. Branke. Evolutionary optimization in uncertain environmentsa survey. *IEEE Trans. on Evolutionary Computation*, 9(3):303, 2005. 3

[9] J. Kennedy and R. Eberhart. Particle Swarm Optimization. In *International Conference on Neural Networks*, volume 4, pages 1942–1948. IEEE, Jan. 1995. 2, 3, 5

[10] C. Keskin, F. Kirac, Y. E. Kara, and L. Akarun. Real time hand pose estimation using depth sensors. *ICCV Workshop*, pages 1228–1234, 2011. 2

[11] T. B. Moeslund, A. Hilton, and V. Kru. A survey of advances in vision-based human motion capture and analysis. *CVIU*, 104:90–126, 2006. 1

[12] H. Niederreiter. Low-discrepancy and low-dispersion sequences. *Journal of number theory*, 30(1):51–70, 1988. 2, 3

[13] H. Niederreiter. Constructions of (t, m, s)-nets and (t, s)-sequences. *Finite Fields and Their Applications*, 11(3):578–600, 2005. 3

[14] H. Niederreiter and K. McCurley. Optimization of functions by quasi-random search methods. *Computing*, 22(2):119–123, 1979. 3

[15] I. Oikonomidis, N. Kyriazis, and A. A. Argyros. Efficient Model-based 3D Tracking of Hand Articulations using Kinect. In *BMVC*, Dundee, UK, Aug. 2011. 1, 2, 3, 4, 5, 6, 7, 8

[16] I. Oikonomidis, N. Kyriazis, and A. A. Argyros. Tracking the articulated motion of two strongly interacting hands. In *CVPR*, pages 1862–1869. IEEE, June 2012. 2, 3, 5, 6, 7, 8

[17] V. Philomin, R. Duraiswami, and L. S. Davis. Quasi-random sampling for condensation. In *Proceedings of the 6th European Conference on Computer Vision-Part II*, pages 134–149. Springer-Verlag, 2000. 2

[18] D. Quaglia and B. Montrucchio. Sobol partial distortion algorithm for fast full search in block motion estimation. In *Multimedia 2001*, pages 87–94. Springer, 2002. 2

[19] J. Rehg and T. Kanade. Model-based tracking of self-occluding articulated objects. In *ICCV*, page 612, Los Alamitos, CA, USA, 1995. IEEE Computer Society. 2

[20] J. Romero, H. Kjellstrom, and D. Kragic. Monocular real-time 3d articulated hand pose estimation. In *IEEE-RAS Int'l Conf. on Humanoid Robots*. IEEE, Dec 2009. 2

[21] R. Rosales, V. Athitsos, L. Sigal, and S. Sclaroff. 3D hand pose reconstruction using specialized mappings. *ICCV*, pages 378–385, 2001. 2

[22] J. M. Saez and F. Escolano. A global 3d map-building approach using stereo vision. In *ICRA*, volume 2, pages 1197–1202. IEEE, 2004. 2

[23] I. Sobol and Y. L. Levitan. The production of points uniformly distributed in a multidimensional cube. *Preprint IPM Akad. Nauk SSSR*, (40), 1976. 3

[24] I. M. Sobol. On the distribution of points in a cube and the approximate evaluation of integrals. *USSR Computational Mathematics and Mathematical Physics*, 7(4):86–112, 1967. 2, 3

[25] E. Sudderth, M. Mandel, W. Freeman, and A. Willsky. Visual hand tracking using nonparametric belief propagation. In *CVPR Wkshp on Generative Model-based Vision*, 2004. 2

[26] J. Vijay, E. Trucco, and S. Ivekovic. Markerless human articulated tracking using hierarchical particle swarm optimisation. *Image and Vision Computing*, 28(11):1530–1547, 2010. 2

[27] Y. Wu and T. Huang. View-independent recognition of hand postures. In *CVPR*, volume 2, pages 88–94. IEEE, 2000. 2