

指导教师： 杨涛

提交时间： 2015.03.29

The task of
Digital Image Processing

数字图像处理

School of Computer Science

No: 1

姓名： 侯瑞兵

学号： 2012302457

班号： 10011203

实时和鲁棒的深度手跟踪技术

摘要

我们呈现了一个使用深度传感器的实时手跟踪系统。它能够在较大的视角之下跟踪一个全关节的手(一个电脑上每秒 25 帧而且不使用一个 GPU 处理器),并且具有很高的精确度(误差不超过 10 毫米)。据我们所知,它是第一个能同时达到鲁棒性,精确性和高速度的系统,并且已经用实时数据证实了。

我们的系统由几个新颖的技术组成。并且仅仅使用了许多球体为一个手建模且定义了一个代价函数。这些都是实时性能的关键。我们提出了一个结合了基于梯度和随机优化的方法来实现快速的收敛和良好的精度。我们提出了新的手指探测和手初始化的方法,可以大大提高跟踪的鲁棒性。

1.引言

手跟踪在许多人类电脑的交互式的应用程序上是重要的,它已经被广泛的研究了几十年了【10,28,4,2,7,14】。尽管如此,由于它的复杂性,它仍然具有挑战性。手是用复杂的手指间的交互结合起来的。它可以在大的角度变换下快速的移动。

尽管这几年来取得了重大的发展,这些使用了最先进技术的方法仍被局限在特定的方面上。有着高精确度的手动动作捕捉工作使用了复杂的网络模型【13】,但是受限于缓慢的局部优化算法。在工程上【8,9】使用了一个简单的多边形模型,并实现了实时的性能,但需要昂贵的 GPU 渲染模式和代价函数评价。这些优化仅仅是局部的优化,而且一旦跟踪错误就不能重新跟踪。全球技术【3,4】搜索了大规模的参数空间为了避免局部最优,但是结果通常是不精准得而且搜索通常很慢。这些方法【21,25】可以达到全局的实时搜索和局部最优,但是依赖于不方便的设备(一个彩色的手套【21】和多台摄像机【25】)。其它实时和鲁棒性的系统仅仅可以辨认出不连续的手势

【31,5,6,29】,并且没有进行优化,这些系统支持少数的自由度【22】,或者在一个固定的视角下【12】。上述的限制是系统的复杂性和既定的目标之间的一个权衡。为了达到较高的精度和速度,以前的作品使用了复杂的模型,复杂的代价函数,昂贵的优化,或者专用的设备。

我们实现了一个最先进的手跟踪系统。它能够在桌面上以很高的精度实时的跟踪自由和复杂的手的移动。我们的工作主要启发于最近的人体跟踪技术的发展

【24,1,15,30,27】。然而,由于上述提到的在手跟踪技术上特有的挑战,直接将现有的身体跟踪技术应用在手跟踪上效果通常会很差。因此,需要仔细调整和改进。

我们遵循了“局部优化+按零件检测初始化”的框架【1,15,30】，采取了现在已有的几个技术。我们采用了一个简单的手模型，该模型使用了一组球体，并且有一个测量模型到稀疏点间距离的快速代价函数。这些简化是实时性能的关键。具体的细节将在第二部分给出。

尽管简单，代价函数仍然是有效的，因为几乎在所有情况下给出的合适的手姿势中，它都达到了全局最小值。然而，它还不是足够的平滑，而且在高维空间中有丰富的局部最优算法。以往的在身体跟踪技术【27】中使用的基于梯度优化的算法和在手跟踪【8,9】技术中使用的随机优化算法并不能将成本降到最低，它们或者对于局部优化太过敏感或者收敛太慢。通过观察这两种方法的结合，我们开发了一种结合了上述两种方法优点的新方法。就像在第三部分描述的那样，它收敛更快，并且更好的抵抗局部最优。

部分检测和基于部分的初始化对身体跟踪技术【18,1,30,24】的鲁棒性是至关重要的，这一点已经被证实了。通过这些作品的启发，对于手指探测，分割和手的初始化，我们在第四部分提出了一些新的和有效的方法。在实时数据下的综合实验验证了我们系统的功能，这部分将在第五部分显示。

2.模型和代价函数

手模型。为了模拟手的运动，在手移动模型中我们采用了通常使用的 26 个自由度【21,9】：其中全局的手姿势用了六个自由度，每个手指用了四个自由度，如图 1(a) 所示。在优化过程中，我们通过有效范围内强力改变关节的角度保证了手的运动学约束（在【23】中被定义）。这 26 个运动参数被定义为 Θ 。

一个简单的集合模型是快速计算距离和交叉点的关键。例如，【27】，该人体被建模为球体和圆柱体的混合物。在这项工作中，我们用简单的球体集表示。特别地，我们使用了 48 个球体构造了一个近似多边形的网络模型，如图 1 (b) 和图 1 (c) 所示。每一部分球体的数量是手动指定的：每个手指六个（大拇指 8 个），手掌 16 个。如图 1 所示，为了在标准的姿势下手模型近似为多边形，球体的尺寸和中心凭经验给定。在这项工作中，我们并不使用任何只适用于个人的模型（除了在全球范围内，如第五部分），并且我们发现我们的球体模型可以在几个不同的对象中跨越）。尽管如此，使用最优化的个人手模型和球体逼近技术可以进一步提高精度。

这个球体的模型用 $M(\Theta) = \{S_i\}_{i=1}^{48}$ 表示。每一个球体 $s = \{c(\Theta), r\}$ 有一个中心 $c(\Theta)$ 和半径 r 。这种记法意味着半径是固定的但是通过正向动力学中心取决与参数 Θ 。为了简洁性我们在这篇论文的剩余部分不再使用标记 Θ 。

数据。我们使用了英特尔的极具创造性的交互式的手势摄像机。深度的分辨率是 320×240 。为了手的分割，我们使用了一个黑色的带绕在手腕上，用来创造深度的空隙，并且找到了距离手区域最近的连通分支，假定手是靠近摄像机的。手的区域被中值滤波算法和形态学开运算进一步的完善，被深度图 D 表示。然后它被转换成一个 3 维的点云，被 P 表示。

代价函数。它测量了手模型和输入深度的差值，以及手模型的有效性。它被定义为：

$$\lambda \cdot \sum_{p \in \text{sub}(P)} D(p, s_{x(p)})^2 + \sum_i B(c_i, D)^2 + \sum_{i,j} L(s_i, s_j)^2. \quad (1)$$

第一项 $D(\cdot)$ 使点云 P 对齐到球体模型 M 。为了减少计算的复杂度，点云被随机的降采样 256 点，被定义为 $\text{sub}(P)$ 。此操作取得了有效性和准确性间的良好权衡。每一个点是它最接近的球体的索引， $D(\cdot)$ 是从这个点到球表面的距离，

$$D(p, s) = \text{abs}(\|p - c\|_2 - r). \quad (2)$$

第二项 $B(\cdot)$ 使模型位于点云里。每一个球体中心以 $j(c)$ 被投影到深度图中。假如 $j(c)$ 中的深度是接近的，球体的中心是位于深度图的前面，可以使用一个使其深度不同的惩罚机制。假如在 $j(c)$ 中没有深度，球体的中心是在深度图剪影的外部，可以使用一个到剪影¹ 距离的惩罚机制，这个距离可以通过剪影的距离转换高效的计算出。正式地，这项被定义为：

$$B(c, D) = \begin{cases} \max(0, D(j(c)) - e_z) & \text{if } D(j(c)) \text{ is valid} \\ \text{dist}(j(c), \text{silhouette of } D) & \text{otherwise} \end{cases} \quad (3)$$

第三项 $L(\cdot)$ 使用了一个模型自碰撞的惩罚机制。碰撞的代价为：

$$L(s_i, s_j) = \max(r_i + r_j - \|c_i - c_j\|_2, 0). \quad (4)$$

我们观测到在优化过程中大多数的碰撞是在相邻手指间的，因此，为了高效性，我们仅仅测量相邻手指间的碰撞。

代价函数简单并且高效。第一项是将可见的球体匹配成点云。它通过加权计算得到，因此和第二项中的梯度相同。第二项使封闭的球体位于深度后并且完善了第一项。它和【27,30】有着相似的原理但是更加简单，因为它仅仅用一些球体评估。

第一项的计算复杂度为 $O(|M| \cdot \text{sub}(P))$ ，因为距离每个点的最近的球体都要被计算。其余两项的复杂度为 $|M|$ 。因为 $|M|$ 和 $\text{sub}(P)$ 都很小，所以代价函数的计算是很快的。这一点对于实时性能是至关重要的，因为任何合适的优化方法都会计算代价函数很多次。

简单起见，我们并不考虑代价函数的时间一致性，这项工作将留在以后来做。

¹ 这个距离用像素来测量。并且用平均输入深度来转换为毫米。

3. ICP-PSO 局部优化

跟踪通过来自初始的手姿势的局部优化 (1) 来进行，手姿势或者来自最后一帧后者来自当前帧的手指分割。

对于点模型调整的任务，迭代最近点 (ICP) 方法被广泛的应用。它采用了交互和基于梯度的优化法，收敛快，适合实时应用系统。然而，它很容易陷入到差的局部优化中，并且不能很好的处理非刚性的物体。各种各样扩展的方法去处理有关节运动的物体已经被提出，而且最近，ICP 已经成功的在人体跟踪技术中采用。然而，在高维的铰接式手跟踪中，特别是在自由的视角下，仍然有不足。更糟糕的是，对于实时性能我们限于使用稀疏子集而不是完整的点云。所有上述的因素导致了代价函数中丰富的局部优化。手的快速移动也经常会使初始化很差。

这些挑战在图 2 中示意。有很多局部最优都会陷入基于梯度的优化中，这些基于梯度的优化来自最后一帧很差的初始化。这个问题由于点云的稀疏采样变得更加糟糕。

随机优化是必须的，用于缓解这个问题。我们使用粒子群算法 (PSO)，在这里简要的回顾一下这个算法。一个粒子是一个高维的参数向量，一个群体是粒子的集合，PSO 是一个粒子互相交互去搜索参数空间的进化过程。在进化过程中，对于群体最知名的全局粒子位置，以及对于每一个粒子最知名的局部位置都被记住。初始化的粒子被随机的采样 (通常会在一个初始化的粒子附近)，它们初始速率都被设置成零。在每一代进化中，一个粒子的速度被更新为它以前速度的随机加权和，这个速度朝向局部最优的位置，并且这个速度也朝向全局最优位置。然后这个粒子以它现在的速度从它以前的位置移动。在所有粒子移动之后，全局最优位置和局部最优位置被更新。

PSO 可以更好的探索参数空间，并通过吸引更多的粒子移向更有可能的领域来避免局部最优。最近，它已经被成功的应用于手跟踪技术中。然而，在我们这种情况下，它并不能很好的工作。可能是因为代价函数需要比较所有点的深度，但是我们的代价函

数仅仅使用了稀疏的点并且也不平滑，如图 2 所示。我们发现即使使用许多粒子，PSO 仍然收敛很慢，并且不能使代价最小化。这个问题被叫做早熟收敛，在高维空间中经常经常被观测到。因为每个粒子都有一个很大的局部搜索空间，PSO 的随机搜索算法并不是足够的高效，一个粒子可能被过早地吸引到一个错误的全局最优位置，即使这时它附近存在着一个好的局部最优位置。

上述分析表明这两种方法互补的性质，ICP 能够快速达到局部最优；PSO 更高效的考察参数空间，但是存在早熟收敛的问题。我们提出了一个混合的优化方法，它结合了这两种方法的优点，克服了它们的缺点。其核心思想是在每一代 PSO 中粒子随机运动之前，都需要一个像梯度下降算法那样的额外的 ICP。在这种方式下，每个粒子可以移动的更快，并且将成本更高效的最小化，就像在 ICP 做的那样。所有的粒子相互作用去对最可能的空间更频繁地采样，一个单一的粒子有更高的机会从局部最优中跳出，就像在 PSO 做的那样。因此，这个相结合的方法收敛更快，并且在抵抗局部最优中做的比其中任何一个都好。

一个说明性的例子示于图 3 的顶部，在零代，在很差的初始环境中随机采样了一些粒子（右下），在 ICP 后最好的一个粒子（固体的绿色十字）被吸引到最可能的区域（左上）。通过 PSO 的更新，整个群体逐渐的被吸引到最可能的区域，最后在第十代达成了正确的解决方案。

我们进一步的扩充了 PSO 来更有效的处理多重局部最优。每一代，用 k-means 算法将粒子分成多簇，以手关节距离均值作为粒子的距离。这个动态粒子将更有效地重新分配粒子，因为一个更好的局部最优通常会吸引更多的粒子。

每个群体中粒子都独立地被随机更新，使这些群体都保持自身的局部最优。从图 3 的底部我们可以清楚的看到。当 $k=2$ 时，这些粒子自动的收敛到两个局部最优的位置。从而进一步的减轻了粒子早熟收敛的问题。

我们的优化方法被叫做 ICP-PSO，它在算法 1 中被概略的叙述。在初始化时，每个粒子都是最后一帧手势的随机扰动。当基于手势的手指探测是可信的，我们根据经验指定 $1/4$ 的粒子用于初始化。在这个随机扰动中，每一个粒子以一个 ID 高斯分布独立地被制图，这个高斯分布的中心是初始化时的数据，标准偏差被手动指定为：每个角度为 5 度（关节角度和全局的旋转角度），全局位置为 15 毫米。在 ICP 这一部分，我们使用了和在【16】，i.e，相似的策略，代替全部使用 LM 算法-所有 26 个参数都执行梯度下降，仅仅随机选择的参数执行梯度下降，并且这一过程仅被重复很少次（特别是令 $m=10$ ）。在【16】中这已经被证明比 LM 算法更具有鲁棒性，在我们的实验中我们也观测到了相似的结果。为了简洁，我们并不细说 PSO 的部分，读者若想了解更多的细节，请参考【9】。

4.手初始化时的手指探测

仅仅根据最后一帧跟踪是脆弱的。人体跟踪技术【24,1,15,30,27】的最新研究已经证实了，在每一帧上重新初始化的能力对于跟踪的鲁棒性是至关重要的。在这个工作中，我们对于手指探测和手的初始化提出了简单而有效的方法。他们源自直观的几何观点，并没有使用像【25】中那样的学习算法。

手指探测。最新的身体部位检测方法发现 3 维云中的极值点使用了 3 维侧地距离。我们发现这个方法并不适用于检测，因为由于手指遮挡和快速移动造成的深度噪音底部图是相当不稳定的。因此，指尖经常不在被检测的极值点的顶部。

我们算法的关键思想是利用手指几何结构的简明性。代替使用 3 维点云，我们分别在 2 维的 XY 平面和一维的 Z 方向上检测了极值点。这种做法更加简单和稳定。为了将极点分为是属于指尖还是不是属于指尖，我们从每一个极点中形成了一个手指分割的提案，并且检查这个几何图形是否和一个手指相似。这个几何检查是直观的，快速的，并且精准的。

XY-Fingers 是和面平面平行的，并且用来自深度图的二进制掩码来检测。我们初始化第一个极值点作为掩码中心，使用距离计算它到所有像素的侧地距离。然后我们重复的使在深度图中最高的点作为新的极值点，并且以递增的方式更新深度图，和【1】相似。从每一个极值点，在当前的深度图上一个分割逐渐扩大直到它的长度和宽度超过了一个手指的大小。标注每段的长度和宽度在 2 维掩码中是以像素表示的，为了对照，根据摄像机的参数和像素深度将其转换为毫米。如果分段的长度，宽度和长宽比与一个真是的手指足够接近，它可以认为是一个手指。重复六次过程，然后发现了手腕和（至多）五个手指。

Z-Fingers 对准 摄像机，它们的尖端作为深度图的局部极小值被检测到。从每个尖端，每个分段通过种子填充逐渐扩大直到它的深度超过了指骨顶部。为了区分开是否为尖端，我们观察到一个好的 Z-Finger 分割应该在一个小的球体中有最多的像素，并且这个球体的中心是一个极点且接近指骨的顶部。球体外部剩余的像素应该在一个小的角度范围。因此，我们检查了在两个小的球体间的环和一个更大的球体（3 倍），发现有着最高密度的扇区（跨越了 60 度）在这个环中，如果在这个环中该扇区有最多的像素（多于百分之九十）则认为这个此分割是正确的。通过使用一个沿着角度标注计算像素的积分直方图，可以高效的在环中发现这个扇区。

² 因为目前为止手的身份还是不知道的，我们简单的使用我们手模型的食指为了比较。

我们的方法和结果如图 4 所示。我们不严格区分 XY 和 Z 上的手指。倾斜的手指通常可以通过双方被有效的检测到。我们在检测 Z 上手指之前检测 XY 上的手指，无论何时当它生长到可以触碰到现有的分割，就立刻丢弃。我们的方法仅仅使用了简单的操作

(增量的距离变化和种子填充)，并且仅仅在很小的领域内执行了几次(在 XY 上六次，Z 上大约五次)，因此非常快。为了高效性，我们进一步下采样使深度图从 320*240 到 160*120。检测平均花费了两毫秒(一毫秒在 XY 和 Z 上)。所有分段参数的增长和几何检查都根据经验值设定以达到高的精度。

手初始化 尽管反向动力学是一个标准的技术根据末端传感器(例如手指头)去估计一个有关节的物体(例如手)的形状，但对于高度的关节人手来说它是不稳定的。同样使用手指分割我们提出了一个更简单和更鲁棒的方法。鉴于被检测的手指的参数 f ，我们假定每个被检测的手指是直的(每个有 2 个自由度)，没被检测到的手指是弯的(自由度可以忽略)。因此，手势的参数被简化为 $2*f+6$ 个自由度，被定义为。

每个指尖 t 提供三个约束。根据每个手指分割我们通过 PCA 估计它的方向 d ，其中方向 d 它提供了两个约束。我们从点云中移除了手指分割，并且通过 PCA 使用剩余的 3 维点去估计手掌的方位 l ，其中该参数提供了三个约束。因此我们有 $5*f+3$ 个约束，可以足够解决 $2*f+6$ 个未知量，当 f 从 1 到 5 时。

从正向动力学中，我们可以导出每一个指尖参数，手指方向参数，手掌定位参数。在那些量中，我们发现了可以使不同最小化的最优手势，

$$\Theta'_{opt} = \arg_{\Theta'} \min \sum_{i=1}^f \|t_i - \hat{t}_i(\Theta')\|_2 + \sum_{i=1}^f \text{angle}(d_i, \hat{d}_i(\Theta')) + \text{angle}(l, \hat{l}(\Theta')). \quad (5)$$

因为手指身份不明，我们列举了几个组合，为每一个运行了优化，就(5)而言选择了有最小代价的解决方案。这种优化问题很小并且花费了不到一毫秒的时间解决。

由于以上做的在上的简化，估算的手姿势通常是粗略的。然而，它通常可以和手指很好的对齐，并且在第三部分很好去初始化局部优化。我们的方法在许多有用的手姿势中都工作的很好，这些姿势有伸长和可见的手指，例如，在手语中的姿势。例如图 4 所示的结果。在有手指弯曲的复杂手姿势中并不是很有效，但是，这些姿势对于任何一个其它的手指检测方法都是一个挑战。

5.实验

在文献中手跟踪技术的评估仍比较原始。它缺乏公共的数据集，协议和度量。这使得交互的方法比较变得十分困难。在这项工作中，我们用手动标记的有监督训练的训练集创造了一个实时的具有挑战的数据集。据我们所知，在相关文献中它的复杂性和规模是最全面的。我们使用了严格的评估流程和度量，这在以前很少使用。我们希望这些能够促进未来工作的实验实践的发展。

数据集和度量。我们让六个对象做大量迅速的手势。每个人有一个四百帧的视频序列被记录。为了 2400 帧我们手动的标记了有监督训练的训练集的手势。为了说明不同手的大小，为每个对象指定一个全局手模型规模（见表 2），但是没有使用进一步的个人适应。

在评估中，我们对六个关节：五个手指头和手腕测量了平均误差 E 。测量是严格的因为这些关节在语义上很重要，并且比其他内部的关节有更大的误差。我们也测量了成功率，它是误差小于 10 毫米的帧所占的百分比。

³ 在 <http://research.microsoft.com/en-us/people/yichenw/> 可用。

优化评估。我们将推荐使用的 ICP-PSO 方法和基线法 ICP 和 PSO 进行比较，其中 ICO 和 PSO 是从算法 1 中简化得到的。ICP 基线法将基线 10 移到了基线 11，并且和多个独立地作业是等价的。PSO 基线法将基线 4 移到了基线 9，并且将参数 k 设置为 1。我们在 PSO，ICP，PSO-ICP 中各自使用了 128,40,32 个粒子，因此每一代它们的运行时间都近似相等。所有的方法在运行 50 代后都趋于相同。

直接比较视频跟踪的结果是不合适的。因为最后一帧的结果高度依赖于早期的那些帧，性能的优越并不能完全归功于方法本身。相反的，在相同初始化下单一帧的比较是可行的。每一帧来看，我们随机扰乱了有监督学习训练集中的手势去产生各种初始化手势。然后把初始的手势分为三个难度层次。所有的层次每一帧中都包含十个初始化的手势。他们的关节平均误差分别在 [15 毫米，25 毫米]，[25 毫米，35 毫米]，[25 毫米，35 毫米]，以及 [35 毫米，45 毫米] 之内。

为了进一步的巩固我们的实验，在每一帧上我们都用有标记的基于监督学习训练集的手势合成了深度图。在图 1(b)中为了使深度更可靠我们使用了多边形模型，而不是球体模型。通过使用真实和合成的数据，表一显示了所有层次上平均精度。它清楚地表明 ICP-PSO 在所有的难度层次上更好，并且使用多个群体 ($k > 1$) 比使用一个群体 ($k = 1$) 更好，特别是当初始错误很多并且局部优化问题严重的时候。这些结论在真实的合成的结果中是一致的，然而有时候真实的结论稍微差一些。在余下的实验中，当我们使用 32 个粒子时令 $k = 4$ ，因为此时是最理想的。

然后我们用难度层次 [15 毫米，25 毫米] 和真实的深度研究了产生几代和粒子数量的影响。和在其他情况下的结论一致。图 5 显示我们的方法减小了代价函数并且更快的提高了精度。图 6 显示使用更多的粒子可以提高精度，但是即使有更多的粒子 ICP 和 PSO 仍然很糟。

跟踪和初始化的评估 在跟踪时，我们使用监督训练产生的训练集初始化了第一帧。我们也在【9】(FORTH)⁴中实现了最先进的方法。使用（被定义为 *）和并不使用基于手指的手初始化，将四种方法比较。表二显示了全部对象在所有帧上关节平均误差。

我们的方法在不用初始化的情况下取得了高精度。使用初始化可以进一步显著地改善所有的方法。图 7 表明第一个对象所有帧上的错误。通过使用初始化可以减少大部分不使用初始化产生的错误。图 8 中显示了所有方法的实例结果。

对于报道了在具有挑战性实时数据上的手跟踪精度的任何工作，我们都不知道。我们的精度大约是 10 毫米，可以跟已经被报道了的在合成数据【9,13】上的精度（大约 5 毫米）进行友好的比较。

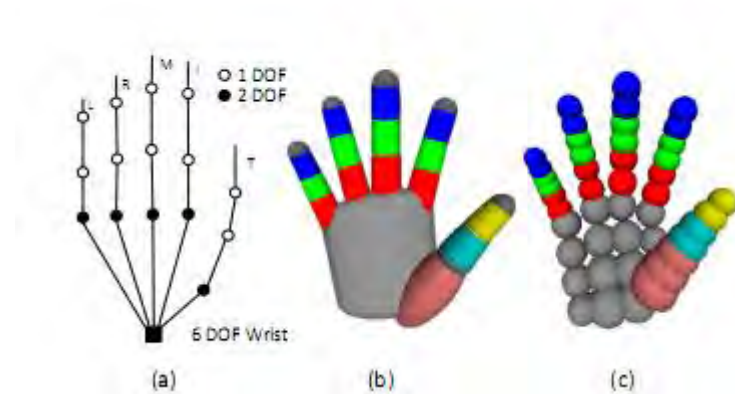
追踪的时序，我们分别使用了 128,40 和 32 个粒子在 PSO，ICP，和 ICP-PSO 上。所有的运行 20 代，并且有相似的速度。在一个 3.4Ghz 的英特尔 i7 处理器上，运行时间是：2 毫秒的预处理，2 毫秒手指探测，1 毫秒手初始化，和 35 毫秒最优化（四个进程）。转换速度为每秒 25 帧（FPS）。标注我们可以通过使用更少的粒子（见图 6）和更少的代数（见图 5）交换得到小的精度损失为了更高的帧速率。

我们在 FOPTH 实现中使用了 64 个粒子和 40 代，导致了在一个 nVidia GeForce 580 的处理器上的每秒 14 帧的速率。更多的代并不能提高精度。

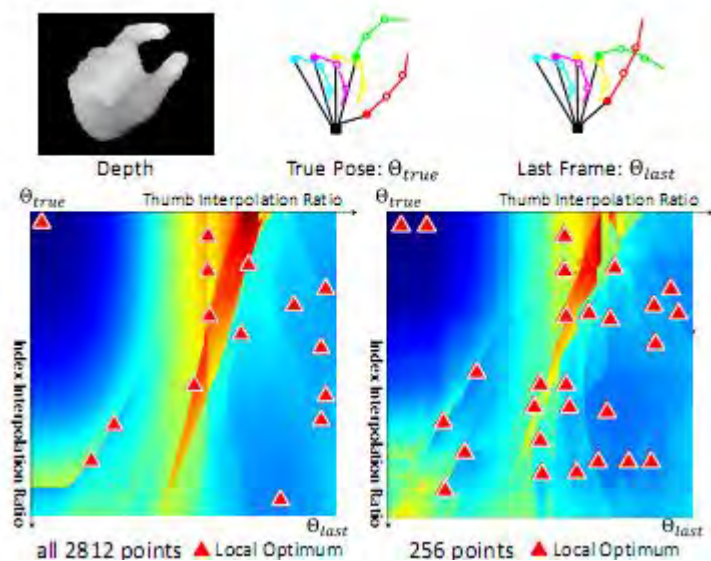
⁴【9】的公共的实现使用颜色基于皮肤分割，他并不适合我们的案例。

6.结论

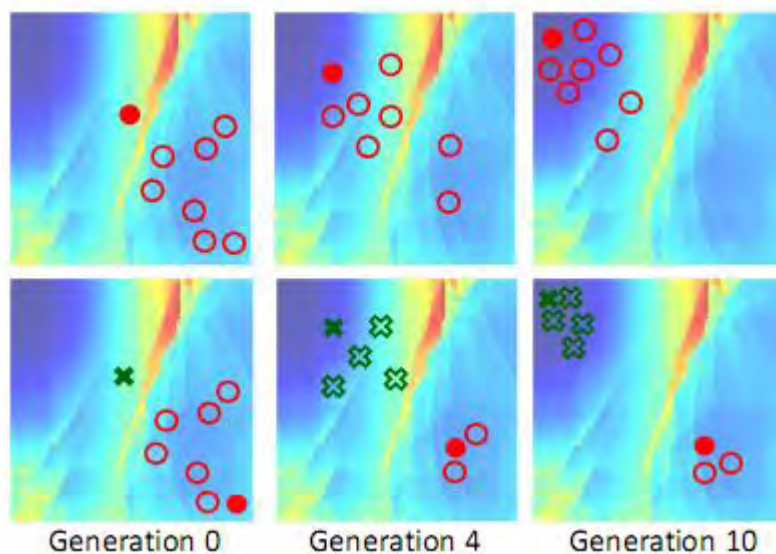
我们呈现了一个新的最先进的手跟踪系统，以一个简化模型的配合，一个快速的代价函数，和最优化和初始化的有效方法实现。在台式机上它的实时性和鲁棒性使它对于很多应用程序，例如用户界面，手语识别和虚拟实时控制都非常有效。



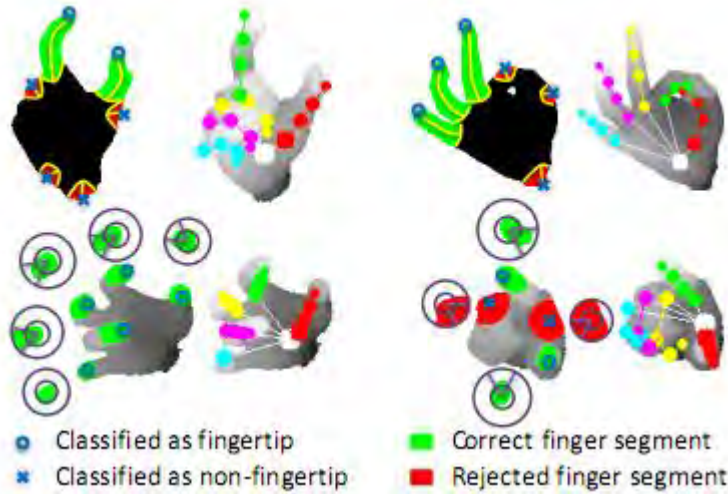
图一：(a) 26 个自由度手运动模型；(b) 手运动模型在【8】；(c) 我们的手运动模型近似 (b) 使用了 48 个球体。



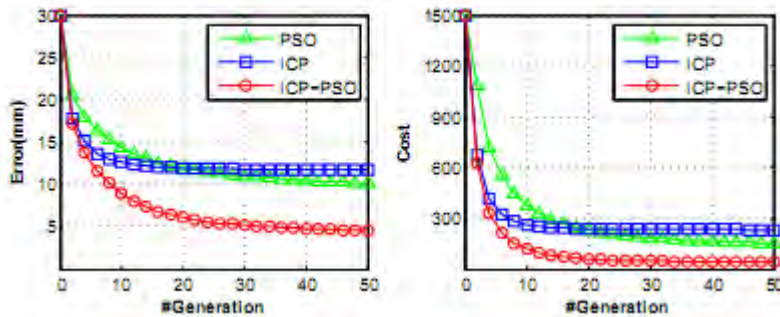
图二（在颜色上可以更好的被观测到）说明一个实时跟踪的例子。顶部：由于拇指和食指的快速移动。来自最后一帧的手势对于当前帧来说是一个差的初始化。底部：我们通过插值最后和真实的手势中拇指和食指参数来生成中间的手势，同时保持其他的参数不变。代价函数可以通过两个插值系数评估和使用密集和稀疏的三维点云，用局部的最优覆盖可视化。



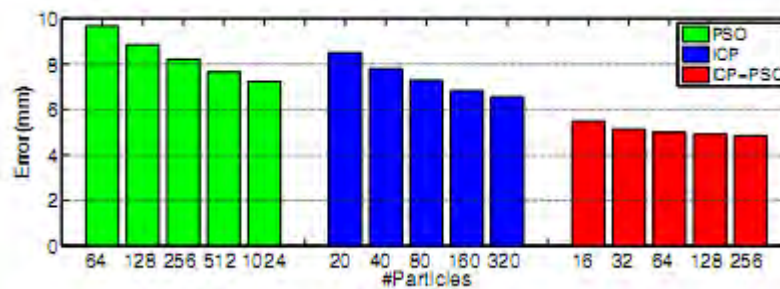
图三：（在颜色上更好的被观测到）使用图二的例子 $k=1$ （顶部）和 $k=2$ （底部）观测到 ICP-PSO 的优化流程，详细的信息请参考文本。



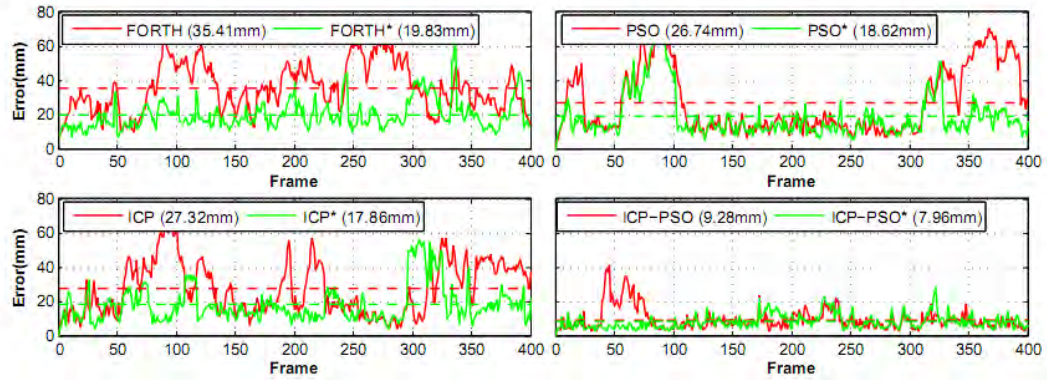
图四（在颜色上更好的被观测到）手指检测和手势的初始化的插图表示。对于每一个例子，左部显示了检测到的极点，手指分割的方案，和指尖与非指尖的分类结果。右部显示了预计的手姿势。顶部的两个例子是 XY 的手指。每一部分将它的用于几何检查的长度和宽度线可视化。底部的两个例子是 Z 上的手指。每一部分都将它的两个球体和用于几何检查的在环上的部分可视化。



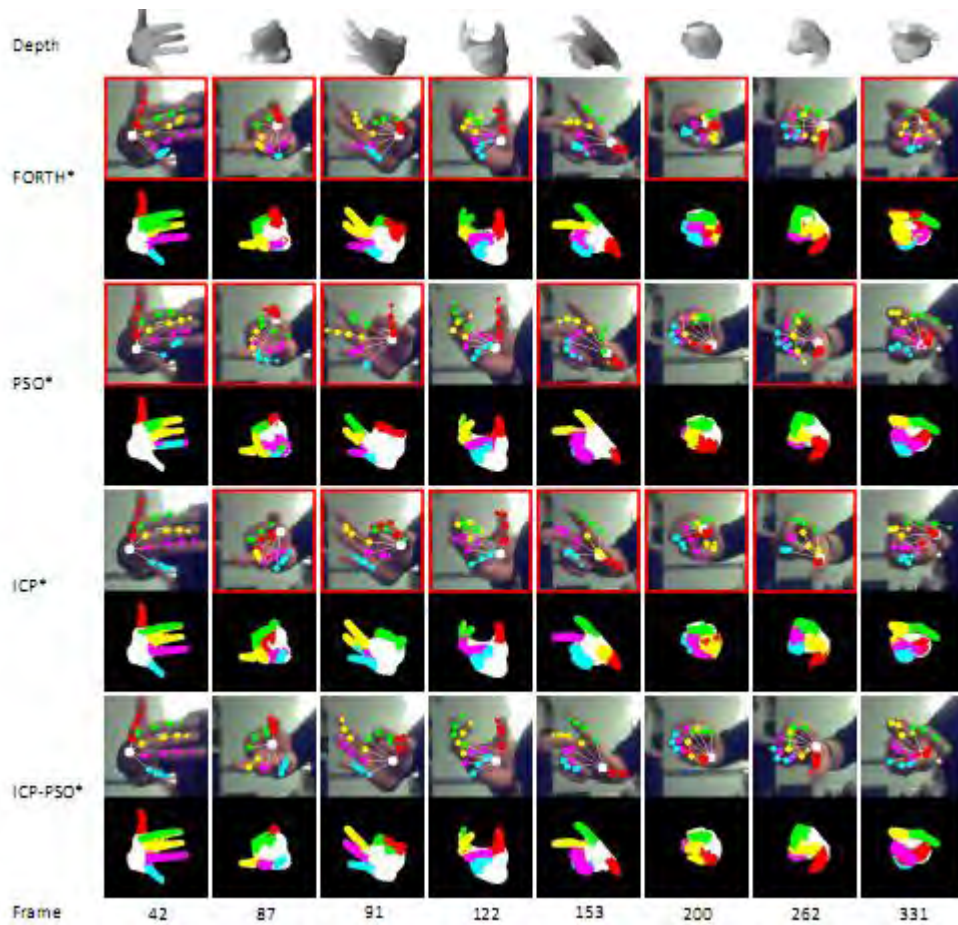
图五。当最优方法通过几代运行之后平均误差和代价函数值减少。



图六.使用不同数目粒子三种方法的平均误差。



图七。第一个试验者的在所有帧上的平均关节误差。每一个图都显示了使用 (*) 和不使用初始化的一个方法的结果。这个水平虚拟线以为这所有帧上每一个方法的误差。



图八。(在颜色上更好的被观测到) 第一个试验者的跟踪结果的例子。有红色帧的那些有很大的错误。每一个结果的对应该图的颜色编码为了更好的可视化也被编码显示出来。

Subject	1	2	3	4	5	6
Scale	1.1	1.0	0.9	0.95	1.1	1.0
FORTH	35.4	19.8	27.3	26.3	16.6	46.2
FORTH*	19.8	15.8	19.8	15.4	16.0	21.0
PSO	29.3	14.8	40.2	17.3	16.2	24.3
PSO*	20.3	12.1	19.2	13.7	14.7	22.4
ICP	29.9	20.7	30.8	23.9	18.5	32.8
ICP*	19.3	15.9	17.5	14.9	11.5	25.9
ICP-PSO	10.1	24.1	13.0	12.8	11.9	20.0
ICP-PSO*	8.6	7.4	9.8	10.4	7.8	11.7

表二。模型比例尺和六个试验者的平均关节跟踪误差（在毫米上）。有*的方法使用了初始化。

E_{init} Metric	15-25 mm				25-35 mm				35-45 mm			
	E_{real}	E_{syn}	S_{real}	S_{syn}	E_{real}	E_{syn}	S_{real}	S_{syn}	E_{real}	E_{syn}	S_{real}	S_{syn}
PSO	8.96	6.97	67.4%	82.8%	13.29	11.17	41.3%	56.2%	18.81	16.89	25.0%	37.6%
ICP	10.48	7.50	52.0%	79.4%	14.46	11.66	28.0%	48.1%	20.00	17.41	13.0%	28.1%
ICP-PSO, $k = 1$	6.68	4.35	85.8%	95.6%	9.42	6.62	70.0%	84.8%	14.40	11.35	52.1%	68.2%
ICP-PSO, $k = 2$	6.42	4.12	87.6%	97.0%	8.81	5.77	71.6%	88.4%	13.39	10.11	55.0%	72.3%
ICP-PSO, $k = 4$	6.25	3.97	88.4%	97.8%	8.65	5.51	74.1%	89.6%	12.78	9.15	56.8%	76.0%

表一，在三个不同难度层次上的三个方法的平均性能指标，使用了真实的和合成的深度图。注意 k 是在 ICP-PSO 上的 k -means 算法的参数。