Single-Frame Indexing for 3D Hand Pose Estimation

Cassandra Carley Carlo Tomasi
Duke University
Durham, NC USA
{carley, tomasi}@cs.duke.edu

Abstract

Hand pose estimation from 3D sensor data matches a point cloud to a hand model, and has broad applications from gestural interfaces to scene understanding. We propose a novel scheme to index into a database of precomputed hand poses to initialize the match. Our index describes 2D hand silhouettes, which can be computed from either depth maps or standard video, in the form of simple yet expressive signatures*. We compare signatures to each other through a new variant of the Earth Mover's Distance that makes small distances in feature space correlate highly with those in pose space. We present a new technique that uses a depth sensor and a sensor glove to create databases of real images and ground-truth poses for both training and testing. We show state-of-the-art accuracy and speed for both gesture classification and joint-pose regression, even when comparing our 2D single-frame method with those that employ RGB-D features or multi-sensor inputs and report quantitative results.*

1. Introduction

Tracking the detailed motions of a human hand with good accuracy and minimal intrusion would enable applications ranging from gestural interfaces and finger-spelling recognition to medical diagnosis, musical tutoring systems, remote surgery, or animation. Tracking hands with 3D input amounts to matching a point cloud and a hand model by optimizing some measure of fit between them. Finger motions that are fast when compared with typical sensor frame rates suggest viewing each data frame as a separate problem: What is the best estimate of a hand's pose—wrist and finger-joint angles—given just the current point cloud? A popular approach to per-frame hand pose estimation is to use the current frame as an index into a pre-built database of (frame descriptor, hand pose) pairs: Use a descriptor of the input frame to find the database entry with the most similar descriptor, and return the associated hand pose as the estimate. This paper addresses several technical challenges within this approach.

First, our indices describe hand silhouettes, which have the advantage of being easy to compute from point clouds, and even from standard video imagery if hand and background look different enough. Thus, although we use 3D input for tracking, our descriptors are 2D, for added flexibility.

Second, we design both our descriptors and a dissimilarity measure between them so as to capture the main features relevant to matching hands, such as which finger matches which, or how well separated two adjacent fingers are. To this end, we segment a silhouette boundary into its main convexities and concavities using a measure of topological persistence to separate important features from irrelevant ones. This segmentation results into a variable-length descriptor we call a *signature*. We then measure the dissimilarity between two signatures by a variant of the Earth Mover's Distance (EMD), a measure of the amount of work needed to transform one signature into the other. Our variant makes sure that implausible matches between fingers are discarded, and then modulates a measure of dissimilarity between the remaining matches in such a way that similar signatures tend to correspond to similar poses.

Third, instead of sampling the set of all hand poses finely, we use low-dispersion sampling to build a database that populates the space of all natural poses well, given a limited number of pairs one can afford to record and store. We build our database by recording point clouds with an RGB-D sensor

This material is based upon work supported by the NSF under Grants No. IIS-1208245 and CCF-1513816.

while measuring true hand motions with an opto-mechanical hand tracker. The resulting database of real images paired with real hand configurations automatically underrepresents unnatural hand poses and requires no manual annotation—a labor-intensive and quantitatively imprecise alternative.

Fourth, we describe several experiments on both regression and classification tasks that show promising accuracy and speed even when we pit our method based on 2D descriptors against those that employ 3D features or multiple sensors.

Section 2 reviews related work. Sections 3 and 4 describe a hand model and how we construct databases. Sections 5 and 6 describe signatures and the dissimilarity measure between them. Section 7 shows experiments and Section 8 concludes.

2. Related Work

With the introduction of affordable depth sensors, methods that use strictly 2D data [11, 1] are quickly being replaced by those that use depth [12, 14, 8, 9, 19]. Full body tracking methods [14] typically treat the hand and wrist as a single, rigid object [14]. Unlike multi-camera approaches [10, 15, 16] or methods that use colored gloves [21] or data-gloves [2] or bands to identify the wrist [12], we use a single depth camera and no markers. *Erol et al.* provide a general literature review [4].

Ren et al. represent the hand using finger segments from a silhouette [12] while *Sridhar et al.* use a sum of Gaussians [15] and later a sum of anisotropic Gaussians model [16]. Several recent approaches [8, 19, 18] build on the success of full body tracking methods using random forests [14]. *Sridhar et al.* search five separate finger databases for finger articulation to reduce the database size [15]. Others [19, 18] use a training set to learn a map from input to pose and dispose of the need for a runtime database, but are limited by the form of the map.

Tracking based on motion models [4, 10, 9, 15, 16] fails in the presence of fast hand motion [4] and is subject to drift [18]. Gradient descent (*Stoll et al.* [17]), Particle Swarm Optimization [10], interpolation [20], temporal and kinematic constraints [9] or specific hand-assumptions [10, 19] are sometimes used to improve initial estimates. *Wu et al.* use a CyberGlove to learn hand-motion constraints [22].

Our work is most closely related to that of *Ren et al.* [12] in our use of hand silhouettes, discrete descriptors, and EMD to address the problems above. However, we use topological persistence as a robust method to detect and describe segments, and our variant of the EMD accounts for matches between fully and partially extended fingers, and for cases where small differences in hand pose correspond to different categories in classification tasks.

3. Hand Model and Pose Distance

We describe the pose of a hand with a vector χ that collects 6 degrees of freedom (DoF) for wrist rotation/translation plus either 21 angles (χ°, in degrees) or 60 position coordinates (χ^{mm}, in mm) for the 15 joints of a hand (three joints per finger) and 5 fingertips (Figure 1). The thumb has 5 angular DoF in our model: two for flexion and abduction of the trapeziometacarpal (TM) and metacarpophalangeal (MCP) joint and one flexion of the interphalangeal (IP) joint. Each of the other fingers has 4 DoF: one flexion angle for distal interphalangeal (DIP) and proximal interphalangeal (PIP) joint, and two for flexion and abduction of the MCP joint. In our experiments we keep the carpometacarpal (CMC) joints and the fingertips fixed. We label the thumb, index, middle, ring, and little fingers as T, I, M, R, and L. We use forward and inverse kinematics to convert between angles and positions.

We measure *pose distance* between frames λ and λ' as the average Euclidean distance over all joints of interest, using either angles or positions (not both) as needed to compare with existing literature [10]:

$$d_{\chi_{\mathcal{J}}}(\lambda, \lambda') = \frac{1}{|\mathcal{J}|} \sum_{j \in \mathcal{J}} ||\chi_j - \chi'_j||_2 \qquad (1)$$

where $\mathcal{J}$ is the set of parameters of interest.

4. Database

We assume that the space of *natural* hand poses is much smaller than that of all *possible* poses [1], and create a database of 320×240 RGB-D images recorded with an Intel DepthSense 325 sensor. To avoid the need for manual annotation, which is labor intensive and quantitatively imprecise, we simultaneously record poses with a CyberGlove III sensor glove that records 23 joint angles every 11 ms, calibrated and mapped to our model (Figure 1) by standard methods [5, 6]. We synchronize the two sensors counting time from an initialization motion recognizable in both sensors: 'fist', 'open hand', 'fist.'

We asked a single subject to assume 75 predefined poses of their right hand (examples in Figure 3) and then add many examples of random motion. We

keep the elbow fixed and ask the subject to move the wrist through 3 abduction/adduction angles, at each of which the subject undergoes a set of complete flexion/extension motions of the wrist. Our initial set Ω has $14,230$ (image, pose) pairs. In comparison, *Wang and Popović* use $18,000$ samples of finger articulation (no wrist motion) [21].

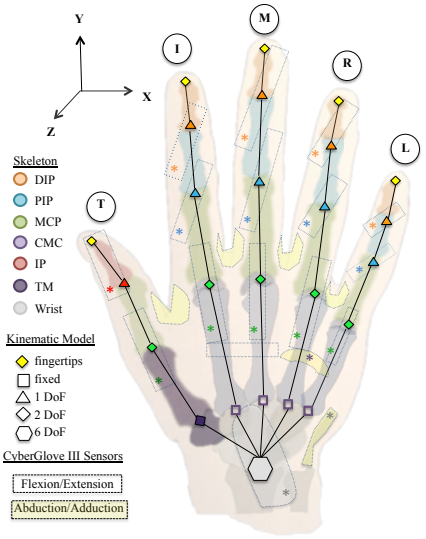

Figure 1: Kinematic model and CyberGlove III sensors.

Low-dispersion sampling [21] is used on Ω to obtain a smaller database Λ such that the distance $\min_{\lambda' \in \Lambda} d\chi^\circ(\lambda, \lambda')$ between any sample $\lambda \in \Lambda$ and its nearest neighbor in Λ is bounded from below. Λ is initialized with the furthest apart pair of points in Ω then iteratively adds the sample from Ω furthest from its nearest point currently in Λ. We iterate until we run out of samples or $\max_{\lambda' \in \Omega} \min_{\lambda \in \Lambda} d\chi^\circ(\lambda, \lambda')$ falls below a predefined ϵ.

5. Silhouette Signatures

The convexities and concavities of outer boundaries of a hand's silhouette capture information about the anatomical features of the hand—fingers, palm, wrist [12]. They are simple closed curves that can be made to be somewhat invariant to wrist motion and the geometry and appearance of individual hands. Silhouettes can be extracted from either color [7] or depth information, or both. We use depth, and assume that the hand is the closest object to the sensor and is well away from the background [12, 18]. If $\mathcal{D}_{\min}$ is the smallest depth in the image and $\tau_{\mathcal{D}}$ is a bit larger than the length of a large hand, the hand is defined as the largest connected component of the pixels whose depth $\mathcal{D} < \mathcal{D}_{\min} + \tau_{\mathcal{D}}$.

Let $\mathbf{c}$ and r be the center pixel and radius of the maximum circle inscribed in the hand region. We trace the silhouette's boundary starting from the leftmost silhouette pixel along the horizontal line through $\mathbf{c}$ to obtain a closed polygon B with n points. We map pixels $\mathbf{p}_s$ on B to their polar coordinates relative to $\mathbf{c}$ and divided by r in norm and 2π in angle:

$$\mathbf{x}_s = (x_s, y_s) = (\mathbf{p}_s - \mathbf{c})/r \quad (2)$$

$$\phi_s = \frac{\texttt{atan2}(y_s, x_s) + \pi}{2\pi} \quad (3)$$

$$\rho_s = \sqrt{x_s^2 + y_s^2}\,. \quad (4)$$

For the index $s \in [0, \ldots, n-1]$ we define $s \oplus z = (s+z) \bmod n$ and $s \ominus z = s \oplus -z$. Centering and normalization account for variations in hand size, distance from the camera, and image position. We found $\mathbf{c}$ to provide a more reliable reference than the hand's centroid, which depends on finger pose. Polar coordinates can be used to yield invariance to 2D rotations in the image by using a standardized starting point—for example, the middle of the wrist.

To reflect changes in hand pose while being insensitive to skin or muscle deformations or noise, we decompose a boundary into a set of segments that separate its main convexities and concavities, as shown next. We then describe the resulting list of segments by a variable-length descriptor, our *signature*, and define a dissimilarity measure between signatures.

5.1. Boundary Decomposition

To decompose a boundary into segments, its pixels are swept in order of decreasing value of ρ_s to determine one segment per local maximum of ρ_s. Boundary segments associated with maxima of low *persistence* [3] are merged with one of their neighbors. Persistence measures the lifetime of an extremum during the sweep [23]. We first describe the sweep, and then explain the role of persistence.

Boundary index s *precedes* s', iff either $\rho_s > \rho_{s'}$ or $(\rho_s = \rho_{s'}$ and $s < s')$, and we then write $s \succ s'$. Index s is a *local maximum* of ρ if $s \succ \mathcal{N}_s$ and a *local minimum* if $\mathcal{N}_s \succ s$ where $\mathcal{N}_s = \{s \ominus 1, s \oplus 1\}$. Let s_l be the index of the l^{th} pixel encountered in the sweep, so that s_0 is the global maximum. Algorithm 1 sweeps the boundary pixels in the order $\succ$ and produces label m_s for index s if this index belongs to the segment associated with a local maximum at m_s.

Initially, all labels are unlabeled (set to -1). The label m_{s_l} of s_l is updated depending on the number of elements in $\mathcal{N}_{s_l}$ that have a valid label:

0 labels: s_l is a *local maximum*, set $\mathrm{m}_{s_l} \leftarrow s_l$.
1 label m**:** s_l is a *regular point*, set $\mathrm{m}_{s_l} \leftarrow \mathrm{m}$.
2 labels: s_l is a *local minimum*.

The two neighboring segments are kept distinct if their persistence is sufficiently high, or are merged otherwise. The minimum s_l is given the label of the older segment. The predicate $merge(\mathrm{m}_1, \mathrm{m}_2)$ in Algorithm 1 is described next.

Algorithm 1 Boundary Segmentation

Inputs: $\mathbf{r} = [\rho_0, \ldots, \rho_{n-1}], \tau_{\text{size}}, \tau_\pi, \tau_{\text{value}}$
Output: $\mathbf{m} = [\mathrm{m}_0, \ldots, \mathrm{m}_{n-1}]$
$(\mathbf{r}, \mathbf{s}) = sort(\mathbf{r}, \succ)$ ▷ Sort $\mathbf{r}$ by $\succ$. Also return sorted indices $\mathbf{s} = [s_0, \ldots, s_{n-1}]$
for $i = 0 : (n-1)$ **do**
  $\mathrm{m}_i = -1$ ▷ Initialize labels
end for
for $l = 0 : (n-1)$ **do**
  switch $|\mathcal{N}_{s_l}| > -1$ **do**
    case 0: $\mathrm{m}_{s_l} \leftarrow s_l$ ▷ New local maximum
    case 1 (m): $\mathrm{m}_{s_l} \leftarrow \mathrm{m}$ ▷ Regular point
    case 2: $\mathcal{N}_{s_l} = \{\mathrm{m}_1, \mathrm{m}_2\}$ with $\mathrm{m}_1 \succ \mathrm{m}_2$ ▷ New local minimum
      $\mathrm{m}_{s_l} \leftarrow \mathrm{m}_1$
      if $merge(\mathrm{m}_1, \mathrm{m}_2, \tau_{\text{size}}, \tau_\pi, \tau_{\text{value}})$ **then**
        relabel all m_2 in $\mathbf{m}$ to m_1
      end if
  end switch
end for
return $\mathbf{m}$

$merge(\mathrm{m}_1, \mathrm{m}_2, \tau_{\textbf{size}}, \tau_\pi, \tau_{\textbf{value}})$

return $|\{s : s = \mathrm{m}_2\}| \leq \tau_{\text{size}} \vee (\rho_{\mathrm{m}_2} - \rho_s \leq \tau_\pi \wedge \rho_{\mathrm{m}_1} - \rho_{\mathrm{m}_2} \leq \tau_{\text{value}})$

5.1.1 Relevant Segments

Persistence [3] helps distinguish between ephemeral local maxima in ρ_s from those that are more likely to correspond to anatomical hand features such as fingers or knuckles. The persistence of a local maximum is the vertical distance between its birth and death. More precisely, a local maximum $s \in B$ is δ-stable if there exist integers a and b with $-n < a < 0 < b < n$ such that (i) for all $z \in [a, b]$ other than s we have $s \succ s \oplus z$ and (ii) $\rho_s \geq \max\{\rho_{s\oplus a}, \rho_{s\oplus b}\} + \delta$. The *persistence* of s is then the maximal δ for which s is $\delta-$stable.

We use persistence as follows. The two immediate neighbors of index s_l at a local minimum belong to segments that are associated with two distinct local maxima, call them $\mathrm{m}_1, \mathrm{m}_2$ during the sweep. Assume that $\mathrm{m}_1 \succ \mathrm{m}_2$, so that m_2 is "younger" than m_1. Segment m_2 is merged into m_1 if m_2 is either insignificant in extent along the boundary, or both of the following conditions are met: the persistence of m_2 is too small [1] and the radial coordinates of the two local maxima $\mathrm{m}_1, \mathrm{m}_2$ are too close to each other. This yields line 18 of Algorithm 1, where τ_{size}, τ_π, and τ_{value} are positive thresholds which we set to 5 pixels, .1 radii, and .1 radii in all our experiments.

The threshold τ_{size} removes very small segments. Of the remaining segments, highly-persistent ones are meant to represent at least partially extended fingertips. Segments that cover enough of the boundary and have a local maximum that is significantly lower than the maximum of an adjacent segment are retained with the intent to capture knuckles, or shorter fingers that touch longer ones.

A boundary *signature* is a concatenation of the descriptors for each segment found by Algorithm 1. Each segment is described by (i) the normalized angular coordinates ϕ_a and ϕ_b of its endpoints; (ii) the persistence π_{m} of its local maximum $(\phi_{\mathrm{m}}, \rho_{\mathrm{m}})$; and (iii) a *weight* $w = (\phi_b \ominus \phi_a)(\rho_{\mathrm{m}} - 1)$ that approximates the area between the segment and the largest inscribed circle. The arm is typically the segment with the greatest weight w, which we remove from the signature. We also remove all segments whose normalized angle ϕ is within 0.2 from the wrist, because they are unlikely to represent fingers.

6. Comparing Signatures

We define a measure of dissimilarity between signatures $S = \{S_1, ..., S_m\}$ and $T = \{T_1, ..., T_n\}$ by (i) defining soft (that is, fractional) matches between segments in S and segments in T and then (ii) measuring the aggregate discrepancies between matched segments, as described next.

6.1. Soft Matches

We perform soft matching through a variant of the Earth Mover's Distance (EMD, [13]), which solves a linear program to determine the smallest amount of work needed to transform the masses w^{S_i} in S into the masses w^{T_j} in T or *vice versa*. Work is the sum

$$\text{WORK}(S, T, \mathbf{F}) = \sum_{i=1}^{m} \sum_{j=1}^{n} d_{ij} f_{ij} \qquad (5)$$

[1] Since the persistence for m_2 is greater than or equal to $\rho_{\mathrm{m}_2} - \rho_s$.

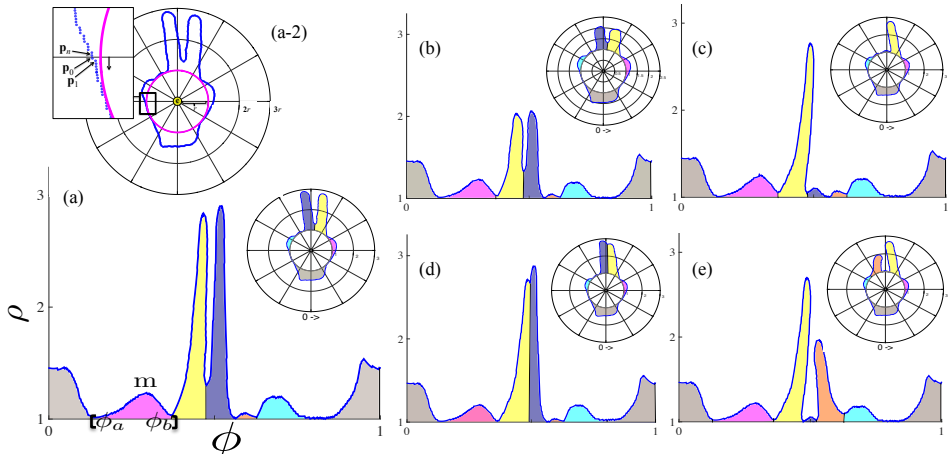

Figure 2: (a-2) shows polar coordinates of normalized boundary. (a, b, c, d, e) each show the signature for a different hand pose, positioned with the wrist (grey) median at 0 and with each segment represented by a different color.

where the matrix $\mathbf{F}$ of unknown *flows* f_{ij} must be nonnegative, have rows and columns that add up to w^{S_i} or w^{T_j}, and have L_1 norm equal to the smaller of the signature masses $w^S = \sum_{i=1}^{m} w^{S_i}$ and $w^T = \sum_{j=1}^{n} w^{T_j}$. We set the *ground distances* d_{ij} in (5) to be equal to the angular distance d_{ij}^{ϕ} used in EMD [12]: zero when the two angular supports overlap fully and $\min(|\phi_a^{S_i} \ominus \phi_a^{T_j}|, |\phi_b^{S_i} \ominus \phi_b^{T_j}|)$ otherwise.

Since it makes little sense for a segment in one signature to match one in the other with a very different angular support, we introduce a distance threshold Δ that is of the order of half the angular extent of a finger width. We set $\Delta = 0.06$ ($\approx 22°$) in all our experiments, finding the value non-critical. We then add extra segments S_0 and T_0 to signatures S and T with weights $w^{S_0} = w^T$ and $w^{T_0} = w^S$, and define

$$d_{ij} = \begin{cases} d_{ij}^{\phi} & i \neq 0, j \neq 0 \\ \Delta & \text{otherwise} \end{cases} . \tag{6}$$

Flows between signatures such that $d_{ij}^{\phi} > \Delta$ are thus shunted into the extra segments, because they incur less work, and matches between excessively distant segments are discarded, *i.e.*, replaced by matches with the extra segments. Setting $d_{00} = \Delta$ prevents S_0 and T_0 from simply matching each other.

6.2. Signature Dissimilarity

The EMD provides an initial measure for the dissimilarities between signatures if modified as follows to account for the extra segments:

$$\texttt{EMD}^{\texttt{x}}(S, T) = \frac{\sum_{i=0}^{m} \sum_{j=0}^{n} d_{ij} \hat{f}_{ij}}{\min(\sum_{i=1}^{m} w^{S_i}, \sum_{j=1}^{n} w^{T_j})} \tag{7}$$

where the flows (soft matches) $\hat{f}_{ij}$ minimize (5). The denominator does not include the extra segments, so finger segments in one signature are either matched to nearby segments in the other signature or discarded if no such match exists. The missing matches are still penalized by Δ units per unit of flow.

Rather than using $\texttt{EMD}^{\texttt{x}}$ directly to measure signature dissimilarity, we introduce an additional cost C for the reasons that follow. Fractional matches between segments account properly for segmentation errors. In addition, a fractional match may represent a match between fingers that are stretched to different extents in the two hands. For instance, Figure 2 shows the signature of a hand with index and middle finger fully extended in (a), and only partially extended in (b). In (c), only the index is visible, fully extended. The EMD computes the correct flows both for the (a, b) pair and for the (a, c) pair. However, the two corresponding signature distances (7) are approximately equal to each other, and this is often undesirable: The pose of the hand in (a) is not too far from that in (b), while that in (c) is more distant from that in (a), at least in the L_2 or L_∞ norm. To address this issue, we introduce optional, additional ground distances k_{i0} and k_{0j} between regular and extra segments, and define these as some convex function of the EMD flows, for instance

$$k_{i0} = \left(\hat{f}_{i0}/w^{S_i}\right)^2 \quad \text{and} \quad k_{0j} = \left(\hat{f}_{0j}/w^{T_j}\right)^2 . \tag{8}$$

Before explaining how these terms are used, we discuss the possible need for another term. In some classification tasks, the pose of a hand whose extended

index and middle finger touch may represent a different category from one where the two fingers are kept slightly separate—for example, the letters 'U' (fingers touching) and 'V' (separate fingers) in finger-spelling. When these distinctions are important, it is useful to add an "abduction" term

$$k_{ij}^{abd} = \alpha^{abd}(\pi_{\mathrm{m}}^{S_i} - \pi_{\mathrm{m}}^{T_j}) \tag{9}$$

which assumes that persistence reflects part separation. For example, in Figure 2 the configurations in (a) and (d) can be differentiated by the persistence of the index finger, which is much greater in (a) than in (d). We set α^{abd} to 0 for regression and to 1 for classification in our experiments.

Our *signature dissimilarity* multiplies k_{i0}, k_{0j}, and k_{ij}^{abd} by the EMD flows $\hat{f}_{ij}$, and levies the resulting penalties *after* the EMD computation:

$$\mathrm{D}_{\mathtt{SIG}}(S, T) = (1-\alpha)\mathtt{EMD}^{\mathtt{x}} + \alpha C \tag{10}$$

where we set $\alpha = 0.5$ in all our experiments,

$$C(S, T, \hat{\mathbf{F}}, \mathbf{K}) = \frac{\sum_{i=0}^{m}\sum_{j=0}^{n} k_{ij}\hat{f}_{ij}}{\min(\sum_{i=1}^{m} w^{S_i}, \sum_{j=1}^{n} w^{T_j})} \tag{11}$$

and the matrix $\mathbf{K}$ has entries

$$k_{ij} = \begin{cases} k_{ij}^{abd} & i \neq 0, j \neq 0 \\ k_{i0} & i \neq 0 \\ k_{0j} & j \neq 0 \\ -\Delta & i = 0, j = 0\ . \end{cases} \tag{12}$$

The term $-\Delta$ for $i = 0$, $j = 0$ subtracts away an irrelevant cost for any flow between extra segments.

Indexing Speedup. The EMD between two signatures is no less than the distance between their centroids if the two signatures have equal mass and the ground distance is induced by a norm [13]. While these assumptions do not hold for our signatures, we have found a threshold on this distance, normalized by total flow, to be an effective heuristic for limiting the number of EMD computations. We also cluster our database by the number of segments with persistence greater than a threshold and match signatures in the same cluster. The combination of these heuristics leads to significant speedups (Section 7). As long as the thresholds used in them are generous (in the sense of over-clustering), their values are not critical.

7. Results

We compare with state-of-the-art methods for gesture classification and joint-pose regression that report quantitative results. The methods compared (including ours, SIG) are summarized in Table 3. Table 4 describes both the test datasets used for evaluation (Q) and the databases (Λ) we used for pose estimation. These include both standard benchmarks and our own databases (CG). We build the latter with an Intel DepthSense 325 depth sensor and a 23 DOF CyberGlove III sensor glove to collect input and ground truth data without manual annotation.

method	% match	mean runtime
$\mathtt{FEMD}_{\mathtt{thresh}}$	90.6	.5004s
$\mathtt{FEMD}_{\mathtt{ncvx}}$	93.9	4.0012s
SIG	97.4	1.021s
SIG (speedup)	97.6	.0417s

Table 1: Classification performance on $Q^{\mathtt{FEMD}}$ for our method SIG with and without indexing speedup shows improvement from thresholded and near-convex FEMD [12].

Classification. Table 1 compares our method (SIG) with *Ren et al.*'s thresholded and near-convex Finger-Earth Mover's Distance (FEMD) [12] methods by the correct-classification rate on the $Q^{\mathtt{FEMD}}$ gesture recognition dataset [12]. Like our method, FEMD does not add local optimization or use temporal information. However, it requires the user to wear a black band to identify the wrist [12]. FEMD does not account for partial correspondences nor does it accommodate an abduction term, and would fail to differentiate, for instance, between 'U' and 'V' in finger-spelling. Our results show significant improvements both in classification rate and speed (when speedup is used). When using the speedup we actually get a slight improvement in classification rate. This is due to eliminating possible false positives and the nature of $Q^{\mathtt{FEMD}}$'s small size, specific gesture classes and variance.

Table 2 shows the classification rate for our method SIG on our own test dataset $Q^{\mathtt{ASL(CG)}}$ using an estimation database $\Lambda^{\mathtt{ASL(CG)}}$ tailored to finger-spelling. $Q^{\mathtt{ASL(CG)}}$ and $\Lambda^{\mathtt{ASL(CG)}}$ are captured at separate times using our CG method and contain data for letters 'A' - 'Z' and numbers '0' - '9' with multiple global orientations and jitter (more details in Supplement). While rates are fairly good, we observe two main types of failures. The first occurs when classes are very similar in pose space and image space. For example, 'Z', 'G' and '1' have slightly different wrist rotations. The second type of failure results from the silhouette not capturing differences in pose. For example, when differentiating between classes like 'A', 'S', and 'T' that have similar 2D silhouettes, depth information could improve performance.

Regression. Figure 4 shows a comparison of our single-camera method with several state-of-the-art

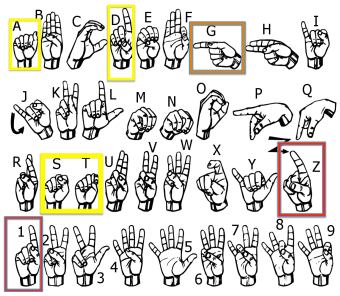

Figure 3: ASL classes used to test $Q^{\mathtt{ASL(CG)}}$ with $\Lambda^{\mathtt{ASL(CG)}}$ (not shown - '0' is fist). 'Red' - slight global abduction/adduction. 'Orange' - more significant rotation. 'Yellow' - cases where silhouettes may not differentiate.

Subset	% match
easy	100
medium	92
hard	87

Table 2: Classification rate for our method SIG on different subsets of our CyberGlove III $Q^{\mathtt{ASL(CG)}}$ dataset. Subset 'easy' is basic finger-spelling, 'med' adds jitter and class variance, and 'hard' increases speed and varies viewpoint.

regression methods using multiple test datasets. For consistency with the literature, we use the mean Euclidean distance between the estimated and ground truth joint positions of the fingertips (in millimeters) averaged over a test dataset.

We compare against *Tang et al.*'s latent regression forests (LRF) [18], *Keskin et al.*'s multi-layer random decision forest classifier (KESKIN) [8] and *Melax et al.*'s model-based approach (MELAX) [9] using the publicly available test datasets $Q^{\mathtt{LRF}}$ [18] and previous reported results [18]. $Q^{\mathtt{LRF}}$ is left-handed, so instead of our right-handed database $\Lambda^{\mathtt{SIG(CG)}}$ we use the provided left-handed training database $\Lambda^{\mathtt{LRF}}$[18].

We also compare with *Sridhar et al.*'s sum of anisotropic Gaussians (SAG, 5 RGB cameras) [16] and earlier sum of Gaussians (SoG, 5 RGBD cameras) [15] approaches using their publicly available Dexter 1 ($Q^{\mathtt{DEX}}$ [16]) with 7 datasets with varying speed, degrees of wrist rotation and occlusions.

We use $Q^{\mathtt{DEX}}$ to compare with SAG, SoG, and LRF, which was evaluated on 3 of the 7 datasets. Performance for our methods on $Q^{\mathtt{DEX}}$ was evaluated using both database $\Lambda^{\mathtt{SIG(CG)}}$ and an approach similar to cross-validation, because $\Lambda^{\mathtt{SIG(CG)}}$ does not contain the range of wrist rotations present in $Q^{\mathtt{DEX}}$. Specifically, we test each of the 7 subsets of $Q^{\mathtt{DEX}}$ using the other 6 subsets as $\Lambda^{\mathtt{DEX}}$. Our performance on $Q^{\mathtt{DEX}}$ has a much greater standard deviation when using database $\Lambda^{\mathtt{DEX}}$ instead of $\Lambda^{\mathtt{SIG(CG)}}$, as some subsets contain poses not found in other subsets. However, the error for our method and baseline is much lower using $\Lambda^{\mathtt{DEX}}$, likely because of similarity in hand shape, environment and motions to $Q^{\mathtt{DEX}}$.

Our own *baseline method* (NN) uses the ground truth to search for the nearest neighbor in pose space to show the relative difficulty of the test datasets, listed increasing to the right: $Q^{\mathtt{ASL(CG)}}, Q^{\mathtt{SIG(CG)}}, Q^{\mathtt{LRF}}, Q^{\mathtt{DEX}}$. This is possibly a result of both the motion complexity and occlusions, as well as database coverage.

We also compare the results for our method using the first nearest neighbor ($\mathtt{SIG}_1$) and the best result from the 5 nearest neighbors ($\mathtt{SIG}_5$).

Our method is the only one that does not take advantage of local optimization or temporal information. We examine the potential impact of adding a local optimization step to our methods by finding the rotation and translation that minimizes the error between ground-truth and estimated pose (Figure 4).

Unlike our methods that only requires single-camera input, SAG uses 5-camera RGB input and SoG requires 5-camera RGB and depth input. Both SAG and SoG also require the user to wear a long black sleeve to help identify the wrist.

8. Conclusion

We estimate per-frame hand pose by encoding image silhouettes with signatures whose elements correlate well with fingers and knuckles, and indexing a database of real (signature, pose) pairs through a novel variant of the EMD that discards implausible matches but treats partial matches appropriately. Our method starts with a single 2D descriptor yet fares well even when compared with multi-camera methods. Our database samples natural hand poses in a balanced way and requires no manual annotation.

The obvious next step is to refine our pose estimates by fitting a hand model to the input point cloud, taking advantage of 3D data to resolve sensitivities to wrist and finger rotation. It will also be interesting to see if our methods can be used to train a pose regressor or a classifier [1], making the database unnecessary at runtime.

References

[1] A. Agarwal and B. Triggs. 3d human pose from silhouettes by relevance vector regression. In *CVPR*, 2004.

method	author	data obs. $\mathcal{Z}$	features	global est.	local opt./constraints
SIG	us	RGB or D	convex components	index Λ	
NN	baseline	RGB or D	$\chi(Q)$	index Λ	
LRF	Tang '14 [18]	RGB-D	vertex info gain	latent reg. forests	error reg. + glob. kin. const.
SAG	Sridhar '14 [16]	5x RGB	2D quadtree (color)	2D aniso. gauss. error	joint and surface const.
SoG	Sridhar '13 [15]	D, 5x RGB	color+depth edges	part based retrieval	voting + sum of gauss. fitting
MELAX	Melax '13 [9]	D	3D point cloud	rigid body simulation	temporal, kin., collision const.
KESKIN	Keskin '12 [8]	RGB-D	pixel color+pos.	global expert network	local expert network
$\text{FEMD}_{\text{ncvx}}$	Ren '11 [12]	RGB-D	near-convex parts	index Λ Finger-EMD	
$\text{FEMD}_{\text{thresh}}$	Ren '11 [12]	RGB-D	height thresh. seg.	index Λ Finger-EMD	

Table 3: Methods Compared: FEMD is a classification only method; all others estimate hand-model parameters. All assume the hand to be the foremost object in the view. Additionally, SAG and SoG require a black long-sleeve shirt and FEMD a black wrist band. SAG and SoG are multi-camera approaches, and all except SIG, NN and FEMD do local optimization.

test Q	$\lvert Q\rvert$	ground-truth pose (χ)	database Λ	$\lvert\Lambda\rvert$	description	compared
$Q^{\text{ASL(CG)}}$	$1,900$	class + $\mathcal{M}^{\circ,mm}$ (CG)	$\Lambda^{\text{ASL(CG)}}$	$4,900$	'A'-'Z', '0'-'9'	
$Q^{\text{SIG(CG)}}$	$4,140$	$\mathcal{M}^{\circ,mm}$ (CG)	$\Lambda^{\text{SIG(CG)}}$	$14,230$	random motion	
Q^{FEMD} [12]	$1,000$	class	$Q^{\text{FEMD}} \cap q$	999	10 gestures	$\text{FEMD}_{\text{thresh}}$, $\text{FEMD}_{\text{ncvx}}$
Q^{LRF} [18]	$1,850$	RMT^{mm} ([9] + manual)	Λ^{LRF} [18]	$301,095$	random motion	LRF, KESKIN, MELAX
Q^{DEX} [16]	$3,155$	$FTIP^{mm}$ (manual)	$\subset Q^{\text{DEX}}/\Lambda^{\text{SIG(CG)}}$	$< 3,155/14,230$	7 subsets	SAG, SoG, LRF

Table 4: Test Datasets: All include RGB-D data and assume the hand to be the foremost object. Our new datasets (CG) are captured using an Intel DepthSense 325 and CyberGlove III (CG) simultaneously to obtain RGB-D and ground truth. For consistency with the literature, ground truth pose is either for all DoF of our model ($\mathcal{M}$), or for positions of fingertips ($FTIP$) [15, 16] or for finger root, middle, and tip (RMT) [18]. Q^{FEMD} is evaluated by the nearest neighbor for each query (q) in the dataset other than itself. We evaluate Q^{DEX} with $\Lambda^{\text{SIG(CG)}}$ (see text).

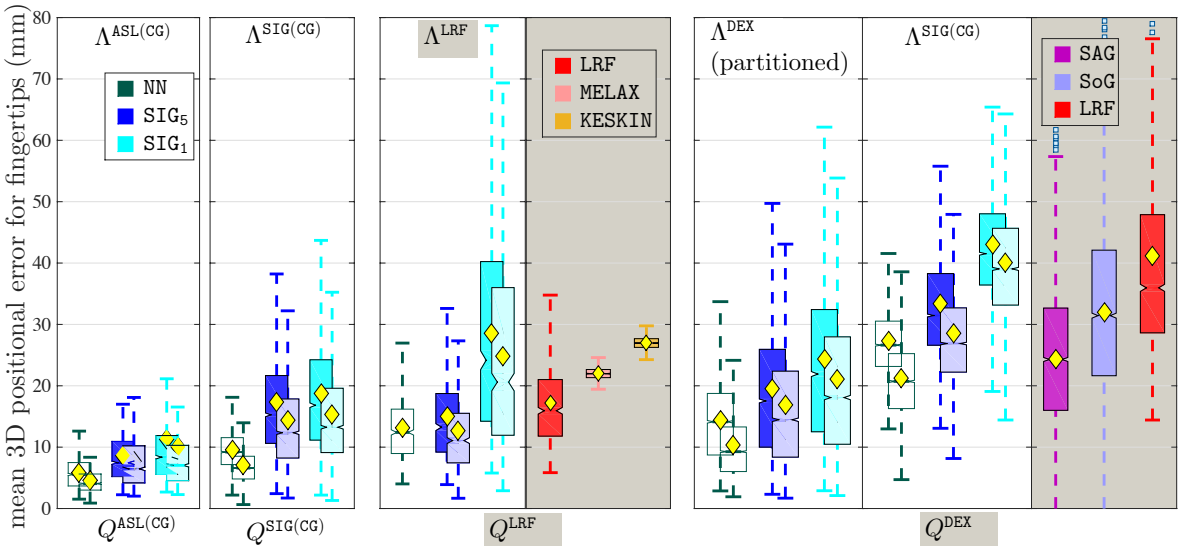

Figure 4: Regression Results (best viewed in color): Grey background is related work; white background is ours. Boxplot median at notch, mean at yellow diamond. When two boxplots overlap, the left one is actual results and right one is error for same method but with local optimization applied (using ground truth to show the potential). NN provides a bound for our performance, SIG_1 shows the results of SIG's nearest neighbor and SIG_5 shows the results for SIG using the best of the 5 nearest neighbors. For KESKIN and MELAX standard deviation is set $\sigma = 1$ and because only mean error for finger root, mean and tip (RMT) was available [18], fingertip ($FTIP$) error is found using an adjustment ratio obtained by results for which we had both RMT and $FTIP$ error such that ${d_{\chi_{RMT}}}^{mm}/{d_{\chi_{FTIP}}}^{mm} \approx .8$. SAG and SoG require a black long-sleeve shirt and FEMD a black wrist band. SAG and SoG are multi-camera, and all except SIG, NN and FEMD do local optimization.

[2] A. Baak, T. Helten, M. Müller, G. Pons-Moll, B. Rosenhahn, and H.-P. Seidel. Analyzing and evaluating markerless motion tracking using inertial sensors. In *Trends and Topics in Computer Vision*. 2012.

[3] H. Edelsbrunner, D. Letscher, and A. Zomorodian. Topological persistence and simplification. In *Proceedings of 41st Annual Symposium on Foundations of Computer Science*, 2000.

[4] A. Erol, G. Bebis, M. Nicolescu, R. D. Boyle, and X. Twombly. A review on vision-based full dof hand motion estimation. In *CVPR Workshops*, 2005.

[5] W. B. Griffin, R. P. Findley, M. L. Turner, and M. R. Cutkosky. Calibration and mapping of a human hand for dexterous telemanipulation. *ASME IMECE Symposium on Haptic Interfaces for Virtual Environments and Teleoperator Systems*, 2000.

[6] M. Huenerfauth and P. Lu. Calibration guide for CyberGlove. *ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, 2013.

[7] S. K. Kang, M. Y. Nam, and P. K. Rhee. Color based hand and finger detection technology for user interaction. *International Conference on Convergence and Hybrid Information Technology (ICHIT)*, 2008.

[8] C. Keskin, F. Kiraç, Y. E. Kara, and L. Akarun. Hand pose estimation and hand shape classification using multi-layered randomized decision forests. In *ECCV*, 2012.

[9] S. Melax, L. Keselman, and S. Orsten. Dynamics based 3d skeletal hand tracking. In *Proceedings of Graphics Interface*, 2013.

[10] I. Oikonomidis, N. Kyriazis, and A. A. Argyros. Efficient model-based 3D tracking of hand articulations using kinect. In *BMVC*, 2011.

[11] J. M. Rehg and T. Kanade. Visual tracking of high dof articulated structures: an application to human hand tracking. In *ECCV*. 1994.

[12] Z. Ren, J. Yuan, and Z. Zhang. Robust hand gesture recognition based on finger-earth movers distance with a commodity depth camera. In *Proc. of ACM Multimeida*, 2011.

[13] Y. Rubner, C. Tomasi, and L. J. Guibas. The earth mover's distance as a metric for image retrieval. *International Journal of Computer Vision*, 40(2):99–121, 2000.

[14] 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*, 2011.

[15] S. Sridhar, A. Oulasvirta, and C. Theobalt. Interactive markerless articulated hand motion tracking using rgb and depth data. In *ICCV*, 2013.

[16] S. Sridhar, H. Rhodin, H.-P. Seidel, A. Oulasvirta, and C. Theobalt. Real-time hand tracking using a sum of anisotropic gaussians model. *International Conference on 3D Vision (3DV)*, 2014.

[17] C. Stoll, N. Hasler, J. Gall, H.-P. Seidel, and C. Theobalt. Fast articulated motion tracking using a sums of gaussians body model. In *ICCV*, 2011.

[18] D. Tang, H. J. Chang, A. Tejani, and T.-K. Kim. Latent regression forest: Structured estimation of 3d articulated hand posture. In *CVPR*, 2014.

[19] D. Tang, T.-H. Yu, and T.-K. Kim. Real-time articulated hand pose estimation using semi-supervised transductive regression forests. In *ICCV*, 2013.

[20] C. Tomasi, S. Petrov, and A. Sastry. 3d tracking = classification + interpolation. In *ICCV*, 2003.

[21] R. Y. Wang and J. Popović. Real-time hand-tracking with a color glove. *ACM Transactions on Graphics (TOG)*, 28:63, 2009.

[22] Y. Wu, J. Lin, and T. S. Huang. Analyzing and capturing articulated hand motion in image sequences. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 27(12):1910–1922, 2005.

[23] Y. Zheng, S. Gu, and C. Tomasi. Topological persistence on a jordan curve. *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3693–3696, 2012.