IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received March 17, 2022, accepted April 9, 2022, date of publication April 18, 2022, date of current version April 27, 2022.

*Digital Object Identifier 10.1109/ACCESS.2022.3168674*

# Methods for Measurement and Analysis of Full Hand Angular Kinematics Using Electromagnetic Tracking Sensors

**PRAJWAL SHENOY[1,2], VIGNESH SOMPUR[1,3], AND VARADHAN SKM[1]**

[1]Department of Applied Mechanics, Indian Institute of Technology Madras, Chennai, Tamil Nadu 600036, India
[2]Department of Mechatronics Engineering, Manipal Institute of Technology, Manipal Academy of Higher Education, Manipal, Karnataka 576104, India
[3]Department of Engineering Design, Indian Institute of Technology Madras, Chennai, Tamil Nadu 600036, India

Corresponding author: Varadhan Skm (skm@iitm.ac.in)

The work of Varadhan SKM was supported by the Department of Science and Technology, Government of India, through Cognitive Science Research Initiative (CSRI) under Grant SR/CSRI/97/2017/87.

This work involved human subjects or animals in its research. Approval of all ethical and experimental procedures and protocols was granted by the institutional ethics committee at IIT Madras under Approval No. IEC/2020-03/SKM/02/10.

**ABSTRACT** The human hand needs a large number of sensors to measure kinematics owing to its large number of degrees of freedom. Existing devices like data gloves and optical trackers are associated with calibration, line of sight, and accuracy problems. In this paper, we attempt to measure the full hand kinematics using Electromagnetic Tracking Sensors (EMTS) which are accurate, free of line-of-sight problems, and require no calibration. However, EMTS provides output in the form of rotation groups which are defined on a nonlinear manifold. Hence, linear operations required for experimental analysis such as linear dimensionality reduction are not valid. Also, these sensors are expensive, utilize space in terms of cabling, and require a reduced sensor layout. In this paper, we present measurement methods, test the utility of linear and non-linear dimensionality reduction techniques on quaternions and exponential maps. We also performed sensor reduction using a Gini feature selection based on random forest algorithm. The kinematic measurement results show that EMTS yield superior posture reproduction with an error of less than 1 degree. Autoencoder, a nonlinear dimensionality reduction technique, was successfully applied on quaternions which was tuned to perform better than Principal Component Analysis (PCA) in reducing dimensions. The reduced sensor layout with 8 sensors was able to predict full hand kinematics with a Root Mean Square Error (RMSE) of 5.1 degrees.

**INDEX TERMS** Hand kinematics, synergies, quaternions, Euler angles, Polhemus, electromagnetic tracking sensor, data glove, PCA, autoencoders, neural networks, random forest algorithm.

## I. INTRODUCTION

The hand allows us to manipulate objects with complex contours with ease, owing to its unparalleled dexterity. The strategy employed by the Central Nervous System (CNS) for the control of such a large number of joints has always intrigued researchers working in the fields of robotics and neuroscience alike. This has led to a surge in the development of advanced prosthetic hands, rehabilitation methods, and assistive devices [1]–[4]. To understand how the CNS controls the hand, human hand kinematics are recorded while the participants perform different postures, activities of daily living, or grasp different objects [5]–[11]. Data-driven modeling using dimensionality reduction is usually performed to extract neural correlates of control such as postural synergies. Also, human hand movement data is crucial in many other avenues like the sports and animation industry. The success of these studies, however, relies on the quality of data that is measured. Most of the existing studies are performed using a data glove (Ex: CyberGlove) [5], [6], [11]–[16], or camera-based trackers [10], [17], [18] the latter considered to be the gold standard. Data gloves are embedded with flex sensors, which provide an analog output corresponding to the amount of flexion. The analog output from these sensors is then converted to joint angles using appropriate calibration routines [19]. But the absence of an

The associate editor coordinating the review of this manuscript and approving it for publication was Vivek Kumar Sehgal.

appropriate ground truth forces the use of approximations and computational techniques such as linear regression to obtain the joint angles [11]. These methods could induce error in measurement resulting in a lower quality of data being collected. Also, the presence of a glove can induce a sense of equipment awareness in the participants during data recording, thereby reducing dexterity. A recent study has demonstrated that fine motor skills reduce by 29% when wearing gloves [20]. The output of the data glove is dependent on the hand size of the participant and hence calibration is necessary for each participant. Moreover, flex sensors of these data gloves sometimes produce nonlinear outputs for the actual joint angles [21]. Camera-based optical trackers make use of one or multiple cameras and a set of markers on the body to track its movements. These systems are highly accurate, but it requires a line of sight between the camera and markers during the entire recording session. Since the hand movements are fine as well as complex, it is rather challenging to establish a constant line of sight between the markers and cameras. Some studies have also developed customized IMU-based gloves to measure hand kinematics [22]–[25]. These devices are extremely inexpensive and serve well for measurement in clinical practice and applications involving non-laboratory conditions but do not possess the accuracy and resolution like the optical trackers or EMTS.

The drawbacks of data gloves, optical trackers and IMU's can be addressed by electromagnetic tracking sensors. These systems are highly accurate and miniaturized which leads to lower sensor awareness during data capture for the participant. EMTS provide a significant advantage over data gloves and optical trackers in the sense that they do not require any calibration procedures, nor do they have a line-of-sight problem. However, electromagnetic (EM) interference caused due to presence of metallic components in the experimental space could produce inaccurate measurements. Such errors can be reduced by making sure that the experimental space has minimum metallic objects that could cause EM interference. Some of the commercially available EMTS are Polhemus, NDI Aurora, and Ascension MiniBird. While only a few pieces of literature are available that discuss the utility of EMTS in measuring hand kinematics [26], [27], a thorough analysis and methods for recording, performing dimensionality reduction, and selection of an optimal number of sensors is scarce. The challenges associated with measurement and analysis will be discussed in the subsequent sections. The methods presented in this paper can also be applied to IMU's in which sensor fusion algorithms are used to convert the accelerometer, magnetometer, and gyroscope data into orientation data in the form of quaternions or Euler angles.

### A. PROBLEM OF NONLINEAR REPRESENTATION OF ORIENTATION

The first problem associated with EMTS is the representation of orientation. Data gloves provide analog output in response to the deflection of bend sensors which are later calibrated to obtain joint angles. Optical trackers provide the spatial positions (3 d.o.f) of the markers in real-time and the joint angles are calculated using forward kinematic techniques. EMTS provide both spatial positions and orientations (6 d.o.f) of the sensors. But computing joint angles using forward kinematics involving only spatial positions requires a large number of sensors. This is not feasible as the EMTS's provide options for up to only 16 sensors. Hence one has to resort to orientations for computing kinematics. EMTS and IMU's provide orientation output using the following 3 representations viz. rotation matrices, Euler angles, and quaternions. These orientation outputs belong to a special class called rotation groups. The angular measurements obtained from data glove and optical trackers parametrize motion in a linear Euclidean domain unlike rotation matrices, quaternions, and Euler angles which are defined on the surface of a hypersphere [28]. These representations are susceptible to problems like singularities, antipodal symmetry, and gimbal lock and must be carefully handled. Also, since these representations are defined on a nonlinear domain, Euclidean operations like averaging, linear filtering might not be valid [28]. Apart from this, it is not straightforward to perform important operations like dimensionality reduction using linear methods like PCA, which is the basis of most of the experimental research on hand kinematics [29]. In this paper, we attempt to measure full-hand kinematics involving a full thumb reconstruction of 4 participants performing postures and object grasps. Quaternions are used as an initial joint angle parametrization method to avoid singularities and gimbal lock. Relative orientations between joints are computed using quaternion operations and joint angles are computed by converting relative quaternions to Euler angles as the last step for visualization and animation. Since the relative joint orientations involve small angular changes, the computed joint angles from relative quaternions are completely free of orientation discontinuities and gimbal lock by selecting a suitable coordinate system. We also present methods to perform sensor to body segment alignment and methods to benchmark sensor values. For Euclidean operations, like filtering and computing RMSE, the quaternions are converted to exponential maps, a linear representation of orientation.

### B. PROBLEM OF PERFORMING DIMENSIONALITY REDUCTION WHILE REPRESENTING ORIENTATION USING ROTATION GROUPS

Neuroscience research has hypothesized that a simplified set of motor primitives are responsible for controlling the complex biomechanical structure of the fingers. As a result, natural flexural patterns of its joints depict a coordinated activation. Previously, PCA has been successfully applied on hand kinematics obtained using data glove and optical tracking systems [5], [6], [8], [11]–[18]. As mentioned earlier, these measurement systems parametrize motion in a linear Euclidean domain, unlike rotation groups that

are used while using EMTS. Dimensionality reduction using PCA performed on data obtained from data glove and optical trackers have shown that the first 3 principal components (PCs) account for around 80% of the variance indicating that fewer joint parameters can control the entire range of movement (ROM) of fingers without reducing the degrees of freedom (d.o.f) [16]. The results showed that the first and second principal components involve the movements of the proximal interphalangeal joint (PIP) and metacarpophalangeal joints (MCP) of the fingers, while the third PC involves the thumb movements. Previously, it has been hypothesized that applying PCA directly on quaternions is not desirable since quaternions are defined along a nonlinear manifold whereas PCA is strictly a linear algorithm [29]. Because of this, animators have explored the possibility of performing a linear dimensionality reduction on exponential maps [29], [30]. As described earlier, exponential maps are a linear representation of quaternions in the tangent space. A comparison of dimensionality reduction on different joint angle representations using functional principal component analysis (fPCA), which is a linear method, has shown that exponential maps and quaternions fare better than Euler angles to reconstruct postures using reduced dimensions [31]. But the efficiency of the fPCA algorithm depends on the continuity in data. In the case of discontinuities (due to quaternion antipodal symmetry, singularities of exponential maps), fPCA appeared to perform poorly. Other than fPCA, we haven't found any literature that addresses the application of dimensionality reduction techniques directly on quaternions when analyzing movements. Recently, a nonlinear dimensionality reduction using autoencoders has been shown to provide superior reconstruction using latent dimensions when compared to that performed using PCA when measured using a data glove [5]. In this paper, we first test if PCA on hand kinematics using exponential map data representation produces the same result as previously reported for data gloves and optical trackers. We then test the performance of autoencoders in reducing dimensions using quaternions. The performance of the quaternion autoencoder is compared with exponential map PCA by examining its ability to reconstruct postures from reduced latent space.

### C. PROBLEM OF SENSOR COST AND REQUIREMENT OF A LARGE NUMBER OF SENSORS/CABLES

Thirdly, EMTS are expensive compared to other devices and require a cable per sensor resulting in system bulkiness. Hence a reduced sensor setup is necessary that will record the full hand kinematics with minimum sensors and still measure kinematics as accurately as possible. Considering a large number of joints and the requirement of a large number of sensors, the problem of the cost of the sensors and time for setup needs to be addressed. Previously, dimensionality reduction using PCA has been used to identify the first few PCs (also known as Synergies) that explain the maximum variance and identify the joints that contribute to the PCs [32]. Hence, these joints represent maximum variance in the data and will be able to reconstruct data with reasonable accuracy. While such a method would help in identifying joints that portray coordination while explaining maximum variance, the method would be participant-specific and cannot be generalized since the synergies are different for different participants. However, synergies acquired from data recorded during ADL can be clustered to obtain a set of synergies that are representative of the global population [33]. While this approach provides promising results, it is not always possible to obtain a cluster of synergies that would contain synergies from all participants [34]. This could especially be true while performing unconstrained hand postures like American Sign Language (ASL) letters or numbers. Prediction of hand kinematics based on contact parameters of objects to be grasped has also been performed previously using Neural networks [35]. Apart from PCA, an inverse kinematics-based approach has also been used previously to obtain a reduced set of sensors [36], [37]. In this study, we utilize a data-driven technique using random forest algorithm to rank the sensors using the kinematic data of all participants. Gini importance score (a default random forest feature selection algorithm) is utilized to select the optimal set of sensors. Using the selected set of sensors, the orientation of other joints is predicted by training a simple neural network with a single hidden layer. The RMSE between the predicted and original kinematics using the full sensor setup is presented.

The overall methodology followed in the paper is depicted in the flowchart in Figure 1:

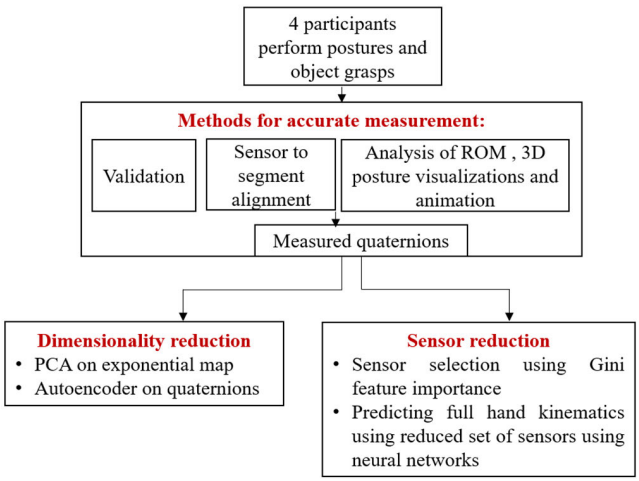


**FIGURE 1.** Methodology employed in the paper for the measurement of finger kinematics and analysis for dimensionality reduction and sensor reduction.

## II. JOINT ANGLE REPRESENTATIONS

Rotation matrix and Euler angles are the most commonly used joint angle parametrization methods in biomechanics, aerospace, and mechanical engineering applications. Both are very intuitive for visualizing or animating postures in real-time. But both have their limitations in terms of ambiguities, computational requirements, and singularities. Quaternions on the other hand are free of all these problems and provide

an efficient and compact way of representing orientation data. In this study, rotation matrices are not utilized since they require 9 values per sensor to represent orientation in 3D space. The 2 modes of joint angle representation viz. Euler angles and quaternions available for motion capture using EMTS are discussed in the next section. Also, a linear form of quaternions called the exponential map is also discussed, which is been extensively used in this study.

### A. EULER ANGLES

Euler angles represent the orientation of any rigid body in 3D space by 3 consecutive rotations about 3 orthogonal axes. If x, y, and z are 3 orthogonal axes, as the object rotates, the angles made by the new x, y, and z axes with respect to the x,y,z axes are the Euler angles. Since Euler angles are defined as consecutive rotations, the output depends on the choice of order or sequence of rotations. Euler angles are suitable only for measurements involving small angular changes. This is due to the singularity that occurs when the measured angle exceeds 180 degrees. The resulting measurements jump to $-179$ degrees instead of 181 degrees and so on, resulting in a discontinuity in data. The second problem associated with Euler angles is the gimbal lock where 2 axes lock up when the second angle approaches 90 degrees. This reduces rotational d.o.f because of which information is lost. Hence it becomes difficult to compute hand kinematics using Euler angles alone. Apart from this, it is difficult to perform some calculations using only Euler angles. It is not possible to add, subtract or multiply Euler angles as such operations will not consider the effect of interaction between the axes. Hence differentiation, averaging, etc. which are operations that are crucial for experimental research are not straightforward. Hence, in this study, Euler angles are used only to animate hand postures and visualize temporal angular variation which is obtained by converting quaternions to Euler angles in the last step.

### B. QUATERNIONS

Problems associated with Euler angles and rotation matrices can be resolved by using quaternions as the first joint angle parametrization method and Euler angles can be obtained from quaternions whenever it is necessary to visualize or animate postures. Quaternions were first described by Hamilton in 1860 and takes the following form (1):

$$q = q_w + q_x i + q_y j + q_z k \tag{1}$$

where $(q_w, q_x, q_y, q_z \in R)$ and $i^2 = j^2 = k^2 = ijk = -1$. The first quaternion component $q_w$ forms the scalar part of the quaternion and $q_x$, $q_y$, $q_z$ form the vector part. For a quaternion to represent pure rotation (no translation), the quaternion should be a unit quaternion. A unit quaternion of the form shown in (1) can be constructed from a rotation of an angle $\theta$ about an axis v using (2) as

$$q_w = \cos\left(\frac{\theta}{2}\right); q_v = q_{[x,y,z]} = v_{[x,y,z]}\sin\left(\frac{\theta}{2}\right) \tag{2}$$

where $[v_x, v_y, v_z]$ form a unit vector that represents the rotation axis and $\theta$ is the angle of rotation.

Thus, a unit quaternion can be used to represent the rigid body rotations and the kinematics is defined along the surface of a unit hypersphere (S3 $\subset$ R4). Hence, all operations on quaternions are performed considering this shape of the surface and are not based on Euclidean geometry. Unlike Euler angles, they are free of orientation ambiguities and gimbal lock and provide smooth interpolation between any 2 orientations.

Multiplying 2-unit quaternions ($q_1 \otimes q_2$) causes $q_2$ to be rotated further by an angle defined by $q_1$. To rotate back $q_2$ to $q_1$, one can use expression (3) by applying the quaternion conjugate operation.

$$q_{relative1=}q_1^{conj} \otimes q_2 \tag{3}$$

where $\otimes$ indicates quaternion multiplication.

Quaternions are easy to interpolate and average and are hence suitable for experimental analysis [28].

### C. EXPONENTIAL MAPS

Exponential maps are a linear reparameterization of quaternions on a tangent space. Exponential map representation can be obtained from a quaternion using a logarithmic map and is not directly obtained as an output format by the EMTS. Logarithmic mapping translates the quaternions defined on a hypersphere to a 3-dimensional tangent plane defined at the identity. The vectors in the tangent plane are linear. These vectors can then be mapped back to the corresponding quaternion using the exponential map, hence the name [39].

The logarithmic map for a general arbitrary quaternion $q = (q_w, q_v)$ is defined using the expression (4) as

$$lnq = \left(ln\,|\mathrm{q}|\,, \left(\frac{1}{\|q_v\|} arccos\frac{q_w}{|q|}\right) q_v\right) \tag{4}$$

Since we use unit quaternions to represent rotations, in expression (4), $ln\,|q| = 0$. Thus, we get a vector in $R^3$ (called a pure vector or quaternion with zero scalar part) which is linear and encodes the information of the axis (direction of $lnq$) as well as an angle (magnitude of $lnq$) using a 3-element vector.

Also, because $lnq$ is a pure vector of the form $v = [0, v_x, v_y, v_z]$, we can define the quaternion as an exponential of the pure vector using the expression (5)

$$q = e^v = \left[cos\left(\frac{\theta}{2}\right), sin\left(\frac{\theta}{2}\right)\frac{v}{\|v\|}\right] \tag{5}$$

where $\theta = \|v\|$

The transformation between the logarithmic map and the exponential map is shown in Figure 2.

The expression (5) maps a pure vector v in tangent space to a unit quaternion on the hypersphere that represents a 3D rotation by an angle $\boldsymbol{\theta} = \|\boldsymbol{v}\|$ about an axis v. In this paper, we will call the linear vector (R3) in tangent space as the exponential map format. Since the exponential map is parametrized in R3, it is susceptible to singularities similar

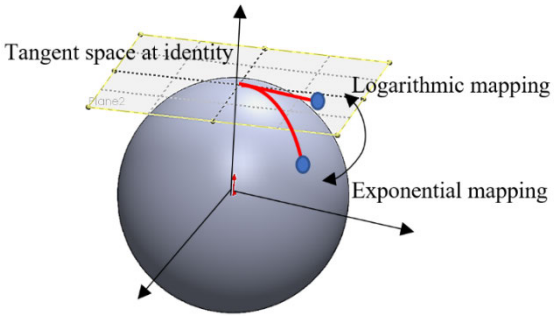


**FIGURE 2.** Representation of mapping between a quaternion and an exponential map.

to Euler angles. The singularity occurs because rotation by $2n\pi (where\ n = 1, 2 \ldots)$ about any axis represents a rotation of 0 degrees, and any point in R3, distant at $2\pi$ from the origin maps back to identity in S3. Nevertheless, the mapping can be made singularity-free if the vector in R3 is restricted to operate in a sphere of radius $2\pi$ [39]. In the case of measuring hand kinematics, we convert relative quaternions between any two phalanges to an exponential map representation. Since the relative angles are small, the singularities are very far from the actual operating region. Hence exponential maps also provide smooth and continuous angular variations. In this paper, exponential maps are used wherever linear operations are to be performed which is not possible using quaternions. The quaternions are converted to exponential maps, linear operations are applied, and the exponential map is then converted to quaternions. One could prefer to completely rely on exponential maps. But the disadvantage with the exponential map is that it is not possible to compose rotations, unlike rotation matrices and quaternions which can be used to rotate a vector to a new orientation [39].

## III. MATERIALS AND METHODS

### A. TERMINOLOGIES AND HAND MODEL

To compute the full hand kinematics (joint angles), the hand is modeled as a 21 d.o.f manipulator with 15 joints. The d.o.f's mentioned are used only for plotting and animating hand posture data for visualization purposes. For dimensionality reduction and prediction, all d.o.f at a joint are considered. Each finger and the thumb are modeled to have 3 joints. The joints of the fingers are the MCP, PIP, and distal interphalangeal joint (DIP). The joints of the thumb are the Carpometacarpal joint (CMC), MCP, and interphalangeal (IP) joint. The CMC of the thumb is modeled to have 3 d.o.f, its MCP and IP joints with 1 d.o.f each. For the fingers, all MCP joints are modeled to have 2 d.o.f and PIP/DIP joints have 1 d.o.f each. The movements of the CMC joint comprise flexion-extension (pitch), abduction/adduction (yaw), and thumb rotation (roll). The MCP and IP joint of the thumb have only flexion-extension movements. In the case of the fingers, all MCP joints comprise flexion-extension and abduction-adduction movements. The PIP and DIP joints have only flexion-extension movements. To compute the full hand kinematics with the joint d.o.f mentioned earlier, we make use of 16 EMTS placed on the dorsal side of the hand. The sensor positioning, sensor numbering, and joint d.o.f's are depicted in Figure 3a. The finger movement types and sign convention are depicted in Figure 3b.

### B. DETAILS OF ACQUISITION INSTRUMENT

Sixteen EMTS (Model: Liberty Microsensors, Polhemus Inc., Colchester, VT, USA) were used for data acquisition. The system can transmit data at an update rate of up to 240 Hz. The microsensors were placed on the dorsal side of the hand at approximately the center of all the phalanges of the fingers. For the thumb, sensors were placed on the metacarpal, proximal and distal segments each. The microsensors that were used are 1.8 mm in diameter and have a resolution of 1.27 microns. They have a static position accuracy and static angular orientation accuracy of 0.76 mm and 0.15 degree respectively. All the sensors are capable of providing position as well as orientation data as output. Each sensor was configured only to provide 4 valued quaternion orientations as the output. Hence the recorded data had 64 dimensions (16 sensors x 4 channels). The microsensors were attached to the hand using surgical tapes. Before performing the actual experiment on hand, a sensor validation procedure was performed. To validate the static angles, we used Polhemus liberty standard sensors. Standard sensors are EMTS similar to microsensors in terms of working, except they are packed in a rugged casing with mounting holes. Since they come with screw holes in their casing that can be used for mounting on 3D printed surfaces they can be used for accurate validation. Microsensors and standard sensors are shown in Figure 4.

### C. COMPUTATION OF JOINT ANGLES FOR VISUALIZATION

To compute the angles for all joints, the quaternions $q_1$ and $q_2$ measured across any joint were converted to relative quaternions using (3). The relative quaternion was then converted to Euler angles using the ''XZY'' rotation sequence indicating a roll, yaw, and pitch respectively. The corresponding components of Euler angles were used to represent rotation, abduction-adduction, and flexion-extension angles wherever applicable.

During the process of converting quaternions to Euler angles, gimbal lock could still occur if the second angle approaches 90 degrees. However, this is addressed in the current setup by aligning yaw movement to abduction/adduction by the appropriate orientation of the source box. Since none of the abduction adduction angles of the fingers or the thumb approaches 90 degrees, gimbal lock is completely avoided.

### D. FILTERING

To filter the data, the relative joint quaternions were converted to exponential map format, and a linear 2$^{nd}$ order zero-lag Butterworth filter was applied. A cut-off frequency of 5 Hz was used to remove any effect of physiological tremors [38].

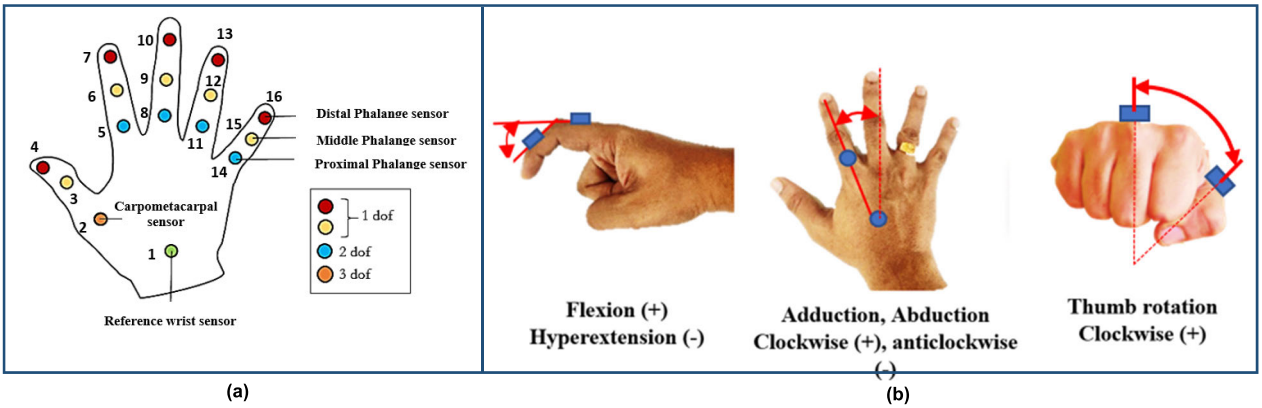


**FIGURE 3.** Hand model utilized in this study (a) Sensor location, numbering scheme, and degrees of freedom (b) Type of joint movements and sign convention.

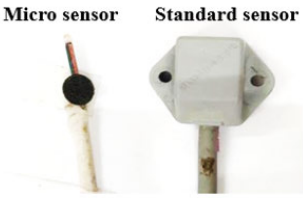


**FIGURE 4.** Micro and standard sensors.

The filtered exponential map was then converted to relative quaternions. Since the exponential map does not incur a singularity, the conversion to and from quaternions is one to one [39].

### E. VALIDATION OF STATIC JOINT ANGLES

To validate the joint angles computed from the sensors, standard sensors were mounted on two different 3D printed models of fingers and one model of the thumb. Standard sensors were selected for validation purposes as they can be accurately mounted on the 3D printed finger replicas. The microsensors on the other hand are difficult to mount accurately on 3D printed surfaces. The 3D printed finger models with attached sensors were placed in the lower left quadrant of the source box as shown in Figure 5a. The angles for the joints were selected to mimic actual finger flexion (design A), thumb movements (design B), and finger abduction-adduction (design C) as illustrated in Figure 5b. The relative quaternion was computed using (2) and then converted to Euler angles using the XZY rotation sequence indicating rotation, yaw, and pitch respectively. The corresponding components of Euler angles were used to represent roll, abduction-adduction, and flexion-extension angles wherever applicable.

### F. PARTICIPANTS

Four right-handed male volunteers of age 27 $\pm$ 3.1 years (mean $\pm$ S.D) were recruited for the study. Written informed consent was obtained from them. There were no previous upper extremity disorders or injuries reported by any of the individuals. The institutional ethics committee at IIT Madras approved the experimental procedures (IEC/2020-03/SKM/02/10).

### G. EXPERIMENTAL DESIGN

The experiment involved performing simple hand postures and object manipulations. The participants were comfortably seated on a wooden chair. The source box reference frame was positioned with X-axis pointing away from the participant and Y-axis pointing towards the left. To avoid magnetic interference, the experimental space had minimum metallic components. Also, care was taken that no metallic components were placed in a sphere of 3x radius where x is the distance from the source box to the sensors mounted on the hand. The participants had to perform 2 types of tasks. In the first task, the participants had to perform 26 postures shown on a monitor screen placed in front of them. These postures included eight Bharatanatyam postures, ten ASL letters, and eight ASL numbers. In the second experiment, the participants had to grasp ten different objects placed in front of them. In both the experiments, the participants had to start from the predefined start position, perform the posture/grasp and hold the position static till the trial was completed. Each trial lasted for 5 seconds (from the predefined start position to end) and 3 trials per posture/object were performed. The trial was repeated if the posture could not be performed within the stipulated time. The quaternion data was recorded at 100 Hz using a custom LabVIEW program. The data analysis was performed in MATLAB. The data collected was utilized to compute joint angles, visualize postures, perform dimensionality reduction and sensor reduction.

### H. SENSOR TO SEGMENT ALIGNMENT

The data obtained from the sensors is the orientation of the sensor with respect to the global frame (source box frame) and not the orientation of finger segments with respect to the global frame. To align the sensors with respect to the segments and hence the segments with respect to the global frame, a sensor-to-segment alignment has to be performed. This process is called bore-sighting. Upon bore-sighting, the

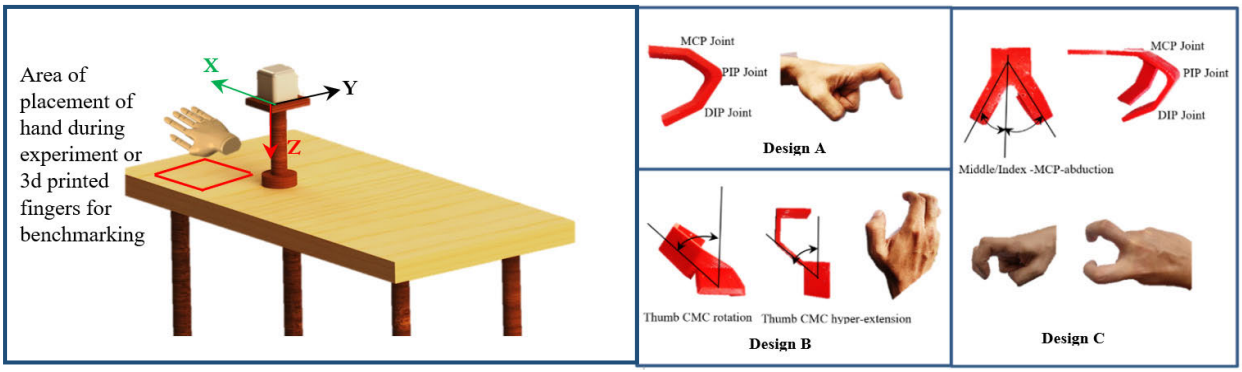


**FIGURE 5.** (a) Placement of hand and 3D printed replicas along with the source box (b) Finger replicas used for validation.

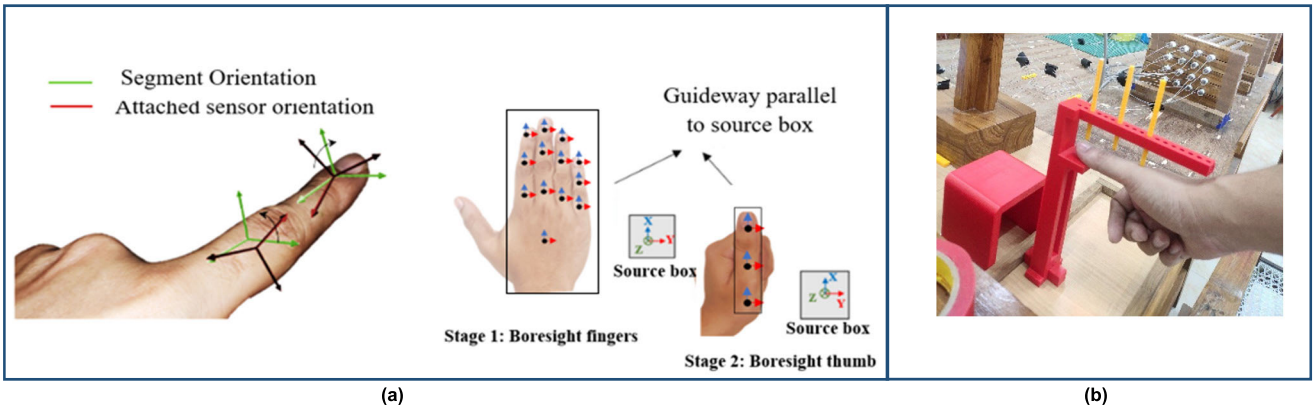


**FIGURE 6.** (a) Positioning of hand for sensor to segment alignment (b) 3D printed setup to reduce parallax error during bore-sighting.

reference frames of all sensors are set to zero in the bore-sighted position. However, since the hand model includes the 3 d.o.f of the CMC joint, upon bore-sighting the hand in the conventional reference position (all fingers and thumb in adducted position), thumb CMC rotation becomes zero and it becomes difficult to reconstruct the thumb posture correctly because the initial value of thumb rotation is lost. Hence, we propose a 2 stage bore-sighting process for the fingers and the thumb separately before the commencement of the experiments. In the first step, the sensors on the 4 fingers and the wrist are bore-sighted by placing them in a position such that all the sensor's frame of reference aligns with the source box reference frame. This is achieved by placing the hand on the reference box placed on the experiment table. The reference box is aligned with the source box by placing it in a guideway mounted on the experiment table. Markings are made on the reference box for placing the hand to ensure alignment with the source box. Step 2 involves the bore-sighting of only the thumb. For this, the thumb is placed such that all the 3 sensors on the thumb are parallel to the source box frame. Also, the thumb is rotated about the CMC joint so that the thumb orientation is identical to the other fingers during the bore-sighting procedure (Figure 6a). To align the thumb with the global frame without any parallax errors, a 3D-printed thumb guide was used. The thumb guide is placed in the same guideway which was previously used to align it with the source box (Figure 6b). Using this 2-stage approach, all the movements of the 4 fingers & the thumb can be accurately captured

### *I. HEMISPHERIZATION*

Before performing any analysis on the kinematic data, it is essential to hemispherize quaternions due to their antipodal symmetric representations. A given orientation can be represented by 2 quaternions: q and -q. Upon performing certain operations like averaging or PCA, such discontinuity can cause accuracy issues hence this must be addressed in the first step [29]. For a given array of quaternion Q of size Nx4, the mean is first computed (described in the next section). A dot product between each sample in the array and the mean of the array is performed. If the dot product is negative, the quaternion is flipped by negating all the components of the quaternions. All the quaternions are then made to lie on the side of the hemisphere where the mean lies. Upon running the hemispherization process, we kept a check on the number of quaternion samples that need flipping. We found that all the samples lay on the same side of the hemisphere and none of the quaternions required flipping. This indicates that all quaternions were constrained to operate within a single quaternion hemisphere resulting in smooth and

continuous data. The pseudo-code for the same is presented in Algorithm 1.

**Algorithm 1** Quaternion Hemispherization.

Input: A quaternion column array Q of dimension Nx4. Each quaternion $q_i$ in the array is a unit quaternion
Output: Q, a hemispherized quaternion array
Compute column mean $\mu$
for $i = 1, 2..N$
  $d = q_i \cdot \mu$
  if $d < 0$
    $q_i = -q_i$
*end if*
*end for*

### J. COMPUTING AVERAGE

Averaging of measurements is very important in experimental analysis. Unlike Euler angle representations, quaternions can be easily averaged. The averaging algorithm is based on the eigenvector decomposition of a quaternion array [40]. The 4 x 1 eigenvector corresponding to the largest eigenvalue obtained from the covariance matrix of a quaternion array (of size N x 4) is considered as the average quaternion. However, it has been shown previously that direct averaging of quaternions also yields superior results when the noise is low [28]. However, upon direct averaging, the quaternions no longer represent rotation as they do not lie on the unit hypersphere. Hence a renormalization is necessary. In this work, we use Markley's algorithm to compute the average wherever necessary. The pseudo-code for computing average is presented in Algorithm 2.

**Algorithm 2** Quaternion Averaging (Markley's Algorithm).

Input: A quaternion column array Q of dimension Nx4. Each quaternion $q_i$ in the array is a unit quaternion
Output: Average quaternion $q_\mu$
$C = (Q^T Q) \rightarrow size(4x4)$
$[e, v] = eigen\ decompsoition\ (C)$ ($e = eigen\ vectors\ of\ C$, $v = eigen\ values\ of\ C$)
$e_{max} = eigen\ vector\ corresponding\ to\ the\ maximum\ eigen\ value$
$q_\mu = e_{max}$

### K. DATA VISUALIZATION

The results of the experiment are demonstrated through 3D hand model visualizations, temporal joint angle variations, and box plot analysis.

### L. DIMENSIONALITY REDUCTION

To perform the dimensionality reduction using PCA for each participant separately, orientation data in exponential map format was utilized as suggested in [29]. The postural data of all 3 trials for all postures were utilized. Thus, the dataset consists of 54000 rows (36 movements x 3 trials x 500 sample points) and 60 columns (15 joints x 4 valued quaternions). The mean for all quaternion block columns (each of size 54000 x 4) was computed using Markley's algorithm. Every sample point from a block column (column containing 4 valued quaternions) was rotated about the block column mean using quaternion conjugate multiplication (similar to centering the data about zero in regular PCA). The centered data were mapped to the tangent space using logarithmic mapping resulting in a linearized data matrix (exponential maps) of size (54000 x 45). Standard PCA was performed on the data matrix and analyzed for Eigen postures, explained variance and reconstruction error. To compute the reconstruction error, exponential maps were converted to quaternions. RMSE error between original quaternion data matrix Q and a quaternion matrix $\boldsymbol{Q_r}$ reconstructed using k principal components was computed using expression (6).

$$(RMSE) = \sqrt{\sum_{i=1}^{n} \frac{1}{n} \left\| ln(Q_i^{conj} \otimes Q_{r_i}) \right\|^2} \tag{6}$$

The complete details of the algorithm can be found in [29]. A pseudo-code for the same is provided in Algorithm 3.

To perform nonlinear dimensionality reduction, autoencoders were utilized. For this purpose, the data of each participant was split into 80% train and 20% test data. Two different autoencoder configurations were tested. The first configuration with 1 hidden layer and linear activation function was used for both the encoding side and decoding sections (Figure 7a). This would replicate the linear PCA process on quaternions using autoencoders and make the results comparable with the results of PCA performed on exponential maps. For the second configuration, a neural network with 5 hidden layers including a bottleneck layer in between was used (Figure 7b). The purpose of this configuration was to demonstrate the advantage of autoencoder to better reconstruct postures from reduced dimensions using nonlinear functions when using rotation groups.

The second configuration was used on both quaternions and exponential maps. As demonstrated in [5], a nonlinear tanh activation function was used for the hidden layers and a linear function was used for the bottleneck layer.

For quaternions, the input and output layers had 60 neurons (15 joints x 4 valued quaternions). The second layer had 40 ($2/3^{rd}$ of input neurons) neurons and the third layer had 20 neurons ($1/3^{rd}$ of input neurons). For the bottleneck layer, the performance was tested by varying the neurons from 1 to 6 (similar to 6 PCs). For exponential maps, the input and output layers had 45 neurons (15 joints x 3 valued exponential maps). The second layer had 30 neurons ($2/3^{rd}$ of input neurons) neurons and the third layer had 15 neurons ($1/3^{rd}$ of input neurons). Reconstruction error for both cases was calculated

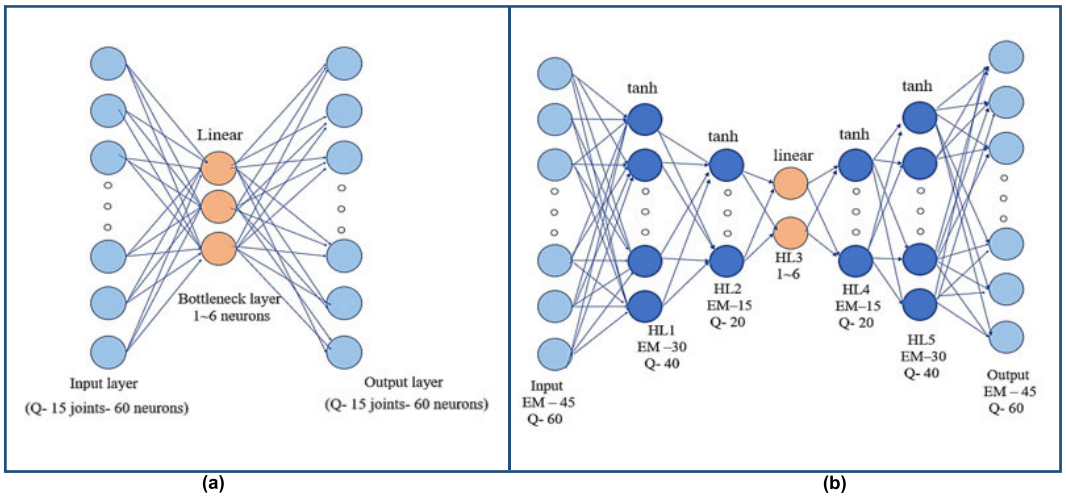


**FIGURE 7.** Neural network parameters (EM- Exponential map, Q-quaternions) (a) 1 Hidden layer (HL) with all linear activation functions (b) 5 hidden layers with linear activation function for the bottleneck layer and tanh activation function for all other hidden layers.

**Algorithm 3** PCA Using Exponential Maps.

**Input:** Data matrix Q consisting of 15, 4-valued quaternion block columns. $Q_i$ represents a single block column vector, $q_{ij}$ is each quaternion sample in $Q_i$

**Output:** Eigenvectors (Linear basis), percentage explained variance, and reconstructed quaternion matrix using reduced dimension

***for*** $i = 1 : 15$ quaternion block columns

Compute mean$Q_{i\mu}$using Merkleys algorithm

***for*** $j = 1 : 108$*movements*

Hemispherize all samples $q_{ij}$ such that it lies on the side of the mean$Q_{i\mu}$

$x_{ij} = \ln\left(Q_{i\mu}^{conj} \otimes q_{ij}\right)$ // Rotate each quaternion $q_i$ about mean $Q_{i\mu}$ to center data around zero and perform alogarithmic map to get a linear vector $x_{ij}$

**end** for

**end** for

Arrange the linearized vectors $x_{ij}$ as blocked columns for each joint to form a single linear matrix X.

$C = \left(X^T X\right) \rightarrow size(45 \times 45)$ // Compute covariance matrix of X

$[E, V] = eigendecompsoition\,(C)\,(E = eigen\ vectors\ of\ C, V = eigen\ values\ ofC)$

**Return** E and percentage explained variance computed using V

P=X $E_k$ // Project linearized data into Eigenspace using first k principal components

R= $E_k P$ // Reconstruct linearized vectors using first k principal components. Each block column in R is $R_i$ with each sample in a block column being $r_{ij}$

***for*** $i = 1 : 15$ block columns

***for*** $j = 1 : 108$ movements

$q_{ij} = Q_{i\mu} \otimes \exp(r_{ij})$ //Convert each linear reconstructed vector to quaternion using exponential mapping and rotate back from the mean position

**end** for

**end** for

Arrange the reconstructed quaternions $q_{ij}$ as blocked columns for each joint to form a single reconstructed quaternion matrix $Q_r$.

**Return** $Q_r$

using (1). Hyper tuning of parameters was not performed. Although, the ratio of convergence of the autoencoder was kept the same for both exponential map and quaternion formulation for the sake of comparison.

### *M. SENSOR REDUCTION*

In this section, an attempt is made to predict the full hand kinematics using a limited set of sensors. The data matrix used for this study consisted of posture data of all participants

in exponential map format. While the data used in this study is of only 4 participants, the entire dataset of all participants consisting of 36 movements repeated 3 times results in a total of 2,16,000 data points (54000 rows per participant x 4 participants) which should be enough training data for the algorithms used.

There are 2 ways to perform sensor reduction. One way is to predict the output joint orientations using the raw data of a limited set of sensors. The second way is to predict the output joint orientations (relative orientations) using a limited set of joint orientations whose values have been obtained previously, computed using 2 sensors. The first method is not used in this study because the raw data of the sensors is dependent on the orientation of the sensor with respect to the source box. It is possible to orient the hand in such a way that, when the raw quaternion values are converted to an exponential map (for filtering or predictions), the exponential map could lie in the singularity region and hence would result in inaccurate readings.

To obtain an optimum number of sensors or joints, a random forest classification algorithm was used to rank features (joint variables in this case). A random forest algorithm is a supervised learning algorithm that generates a group of decision trees. A decision tree is a recursive algorithm based on yes or no questions that split the data set using the decision nodes until pure nodes or leaf nodes (with variables belonging to only one class) are obtained. In other words, the impurity is reduced. In the case of a random forest algorithm, the data is sampled randomly both in columns and in rows and divided into n number of data sets (bootstrapping) based on which, the decision trees are built. Each tree then splits the randomly selected data until a leaf node is obtained. Once leaf nodes are obtained for all trees, the algorithm is considered to have converged. When a new data point is entered, a majority voting is performed based on the output of all the decision trees to make a decision. A Gini impurity index is computed based on the ability of a feature to reduce the impurity at any decision node. If a condition based on a particular feature decreases the impurity by a greater extent, the feature is considered to be important. The Gini impurity index at any node is calculated using (7).

$$\text{Gini index} = \mathbf{1} - \sum_{i=1}^{n} (\boldsymbol{P_i})^{\mathbf{2}} \tag{7}$$

where Pi is the probability of classifying any element to a distinct class (in this case postures and object grasps).

Such a method has been previously used for the analysis of spectral data [41], text data [42], [43], and image analysis [44]. In this study, the features were ranked based on their ability to classify the postures and objects grasped. Since the trees are built on randomly selected rows and features, the algorithm is less sensitive to the changes in training data and hence has a lesser chance of overfitting. The inbuilt random forest algorithm of the python package Sklearn was utilized.

The input data matrix consisted of 216000 rows and 45 columns (15 joints with 3 valued exponential maps). The labels include 36 postures. The model was trained using 80% training and 20% test data with 10-fold cross-validation. Feature importance was then computed using the default Gini algorithm based on the mean decrease in impurity. The disadvantage of ranking based on a mean decrease in impurity is that the algorithm favors columns with high cardinality (like columns that contain extremely unique categorical entries) [45]. Since in our data, none of the columns do contain categorical entries with unique values, the problem of cardinality is not present. The second disadvantage of the algorithm is its inability to rank appropriately, the features that are correlated. In the case of correlated features, the importance could be spread across similar features. In such a case, where the ranks are similar, we test all such possible sensor arrangements with similar importance scores. Since the data contains 45 features, the algorithm outputs 45 scores (3 for each joint). The scores of each joint are summed up and the joints are ranked. The top 5 joints are selected ($1/3^{rd}$ of the total joints) for the prediction of the remaining joints.

Once the joints are selected, a neural network is trained to predict the remaining joints. The neural network was built with an input layer with 15 neurons (5 joints x 3 value per joint), one hidden layer with 20 neurons, and an output layer with 30 neurons (10 predicted joints x 3 values per joint). A batch size of 256 was selected. The hyperparameters are first tuned using the data of a participant not involved in the analysis. 5 different activation functions – sigmoid, tanh, reLU, linear, and exponential were tested for 500 epochs with a learning rate of 0.01. To test the performance of the activation functions, a ''variance accounted for'' (VAF) measure as demonstrated in [5] was utilized which is a measure of reconstruction error (8). If the VAF is 1, then the predicted and actual values are the same.

$$\boldsymbol{VAF}\,(\%) = \left(\mathbf{1} - \frac{\boldsymbol{var}(\boldsymbol{y} - \hat{\boldsymbol{y}})}{\boldsymbol{var}\,(\boldsymbol{y})}\right) \times \mathbf{100} \tag{8}$$

where y = original variable, $\hat{y}$ = predicted variable

Once the best activation function was selected, the effect of the learning rate was evaluated for the selected activation function. Four different learning rates 0.001, 0.01, 0.025, 0.05 were selected based on which the number of epochs and learning rate were decided. The predicted joint orientations with reduced sensors are then compared with the actual orientations measured using all sensors.

## IV. RESULTS

### A. JOINT ANGLE VALIDATION

The static joint angles of 3D printed replicas measured using standard sensors are shown in Table 1.

It can be observed that the sensors produce accurate measurements and the maximum error observed is 1 degree. This error could also be due to slight variations in the 3D printed surface especially near the region of mounting holes. Hence, it can be observed that the joint angles computed using quaternion could yield superior quality posture reproduction which is not susceptible to gimbal lock and ambiguities while measuring hand kinematics.

**TABLE 1.** Validation results of static joint angles.

| Joint | Design A | | Design B | | Design C | |
|---|---|---|---|---|---|---|
| | Set | Measured | Set | Measured | Set | Measured |
| CMC/ F | - | - | -50 | -49.921 | - | - |
| CMC/R | - | - | 60 | 60.397 | - | - |
| MCP/F | 30 | 30.57 | 50 | 50.088 | 0 | 0.5987 |
| MCP/A | - | - | - | - | 30 | 29.4843 |
| PIP/F | 90 | 89.101 | 90 | 89.75 | 30 | 29.64 |
| DIP/F | 30 | 29.703 | - | - | 90 | 89.01 |

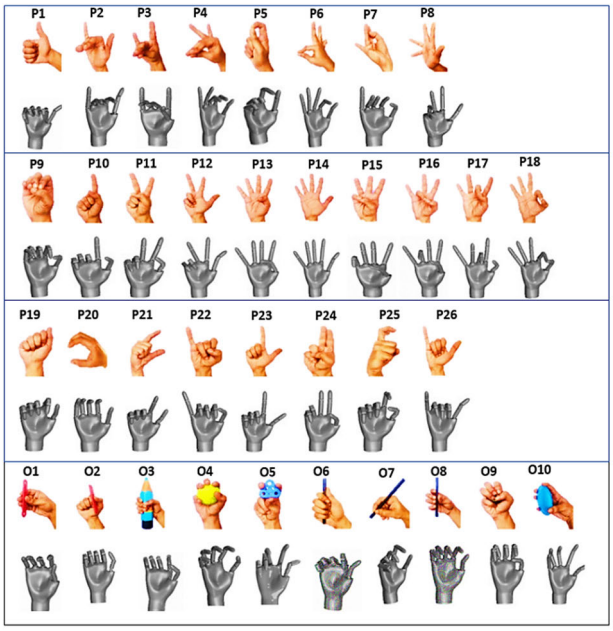


**FIGURE 8.** Reproduction of static postures. Quaternions converted to Euler angles and reproduced using a hand model in SOLIDWORKS. Row 1- Bharatanatyam Postures (P1 to P8), Row 2- ASL numbers (P9-P18), Row 3- ASL letters (P19-P26), Row 4- Objects to be grasped (O1-O10).

### B. POSTURE VISUALIZATION IN 3D

The relative quaternions were converted to Euler angles in the XZY sequence as described earlier. To convert quaternions to Euler angles, the 'eulerd' function from the MATLAB sensor fusion and tracking toolbox was used. To plot the hand postures, a hand model was built using SOLIDWORKS with 21 d.o.f. As can be seen from Figure 8, quaternions help in producing superior quality posture from EMTS's avoiding all other problems like singularities and gimbal lock. The reproduced static postures of randomly selected participants from a randomly selected trial are shown. The thumb CMC being a complex joint was modeled as a simple combination of three, 1 d.o.f joints in SOLIDWORKS to facilitate the thumb assembly. An animation (done using Matlab- Simulink) of all the postures performed by a participant is presented in supplementary material 1.

### C. VISUALIZATION OF JOINT ANGLE VARIATION AND RANGE OF MOVEMENTS (ROM)

The variation in the joint angle while performing a particular posture P19 is shown in Figure 9. The abduction and adduction joint angle variations are not plotted as these angle variations are minimum for the selected posture. The particular posture was selected to show distinct variation in joint angles between PIP, DIP, and MCP joints. For the selected posture, PIP joints show maximum variations (dash lines) while MCP joints show minimum variation (dotted lines). The thumb CMC flexion-extension joint angles are negative because of how the sensor-to-segment alignment has been performed.

It was demonstrated in section III (E) that the thumb was aligned in a position similar to the fingers and then bore-sighted. As a result, when the thumb is rotated to the normal position after bore-sighting, the CMC hyperextends and hence the angle is negative. Only in a few postures, where there is excessive flexion of CMC joint (opposition task) like while performing ASL-number 4, the thumb moves significantly inside towards the little finger resulting in a positive value. This is demonstrated as an inset image in Figure 9.

As expected, the thumb rotation varies from 50 degrees to 80 degrees as the thumb moves from adducted position to the final posture. A box plot analysis to visualize the ROM of each joint across all participants is presented in Figure 10. Outliers, if any, for the joints are not plotted. Data from all participants and all three trials are shown. As can be seen from the graphs, the PIP joints (J5, J8, J12, J16, J20) exhibit a higher range of movement followed by MCP joints of the fingers (J6, J10, J14, J18). The thumb CMC flexion-extension joint angles are negative because of how the sensor-to-segment alignment has been performed as explained in section 4.3. The thumb IP joint shows a greater variation in the negative direction because of the requirement of producing significant hyperextension while grasping some of the objects (O1, O5, O6, and O8).

### D. DIMENSIONALITY REDUCTION USING EXPONENTIAL MAPS AND QUATERNIONS

The results of dimensionality reduction using exponential maps and quaternions are discussed in this section. Scree plot for PCA performed on exponential maps was plotted separately for postures, objects, and combined data (Figure 11). The first 3 PCs account for around 80% variance in data which is in accordance with previous results [11], [15], [16]. Hence PCA using exponential maps provide a simple and computationally inexpensive way of performing a dimensionality reduction when parametrizing data using rotation groups. To interpret the information in the latent dimension, the first 4 Eigen postures (derived from eigenvectors) were plotted for all participants (Figure 12). The maximum and minimum Eigen postures are computed using (9) and (10). The Eigen postures are transformed from

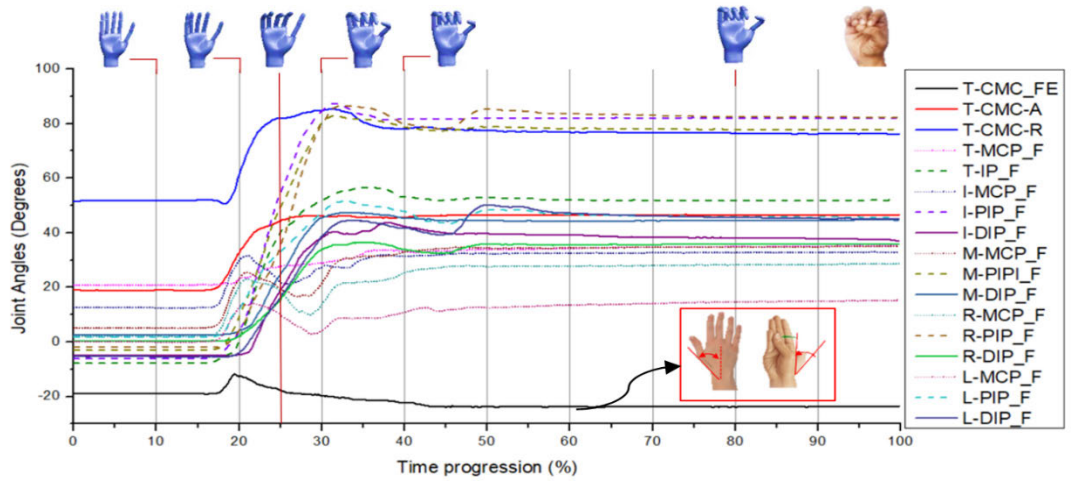


**FIGURE 9.** Variation in joint angles for P19. Abduction adduction angles of fingers are not plotted as the variation in those angles are minimum.

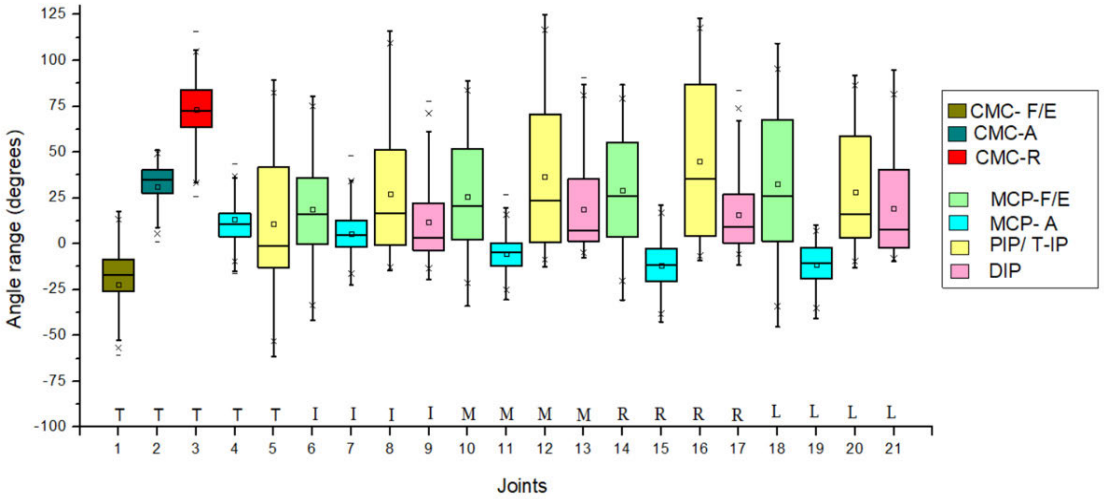


**FIGURE 10.** Box plot analysis to visualize the active range of movements (T-Thumb, I-Index, M-Middle, R-Ring, L-Little).

exponential maps to quaternions and then scaled using (9) and (10) to obtain the movements within the normal active range of movements.

$$Eigenposture_{max} = Qs_i \otimes Q\mu_i \tag{9}$$

$$Eigenposture_{min} = Qs_i^{conj} \otimes Q\mu_i \tag{10}$$

where $Qs_i$ is the i$^{th}$ synergy or eigenvector and $Q\mu_i$ is the mean posture.

Since all participants were trained with the same set of postures, PC1-PC3 was similar for all participants. PC1 and PC2 represent the PIP and MCP joint movements resulting in opening and closing action through the flexion-extension of the joints. PC3 involves the coordinated movement of the index finger DIP joint and the thumb IP joint. PC4 differs across participants. These results are also in accordance with those obtained using data gloves and optical trackers [10], [11], [15], [16]. Thus, exponential maps provide a good representation of orientation with a larger tolerance band for singularities and provide ease of performing dimensionality reduction using PCA.

The reconstruction error for PCA and autoencoder is shown in Figure 13. As can be seen from the graph, an autoencoder with 1 hidden layer and linear activation function on quaternions produces the same result as PCA performed on exponential maps. The advantage of using autoencoders is that a better dimensionality reduction can be achieved by increasing the hidden layers and the use of nonlinear activation functions. The 5 hidden layer autoencoder produces a reduction in RMSE by more than 5 degrees. This was tested for both exponential maps and quaternions. While quaternions produced slightly better performance, both representations provide good dimensionality reduction. The advantage of PCA is the ability to interpret the latent

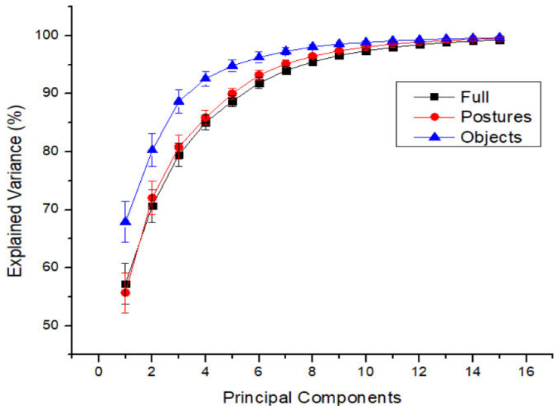


**FIGURE 11.** Explained variance computed using PCA on exponential maps, separately for postures, object grasps, and combined dataset.

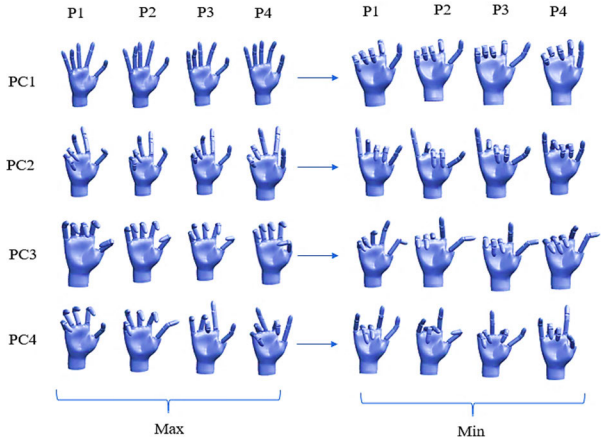


**FIGURE 12.** Eigen postures (PC1-PC4) plotted for all 4 participants (P1-P4).

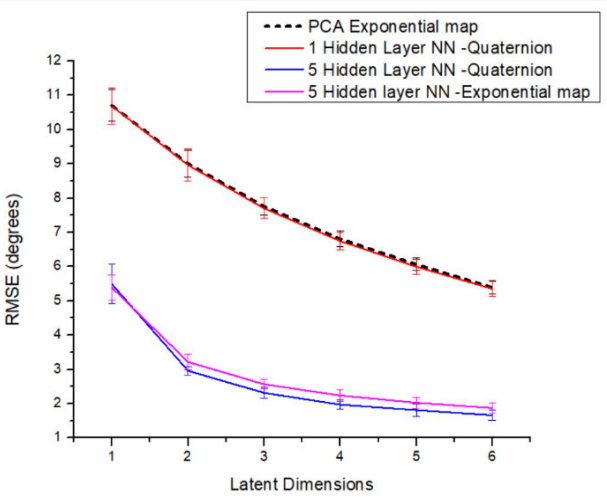


**FIGURE 13.** Comparison of reconstruction error for PCA (using the exponential map) and autoencoder (using both exponential map and quaternion representation).

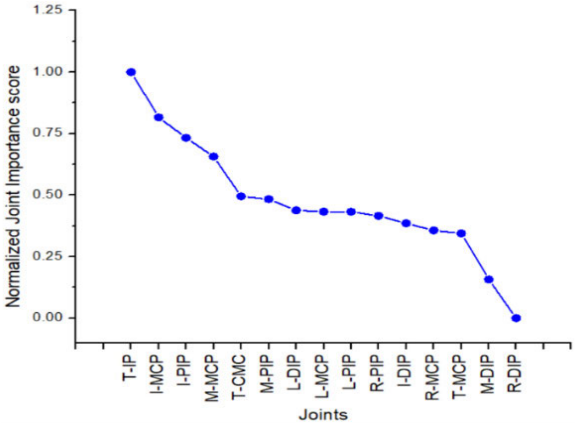


**FIGURE 14.** Normalized joint importance scores computed using random forest algorithm (Gini feature selection).

dimensions. In the case of PCA, the weights of the PCs can be analyzed to obtain the Eigen postures which can be used for prosthetic design [46], [47] as well as training to improve hand function performance [48]. This is not possible with autoencoders, particularly when using nonlinear activation functions.

### *E. SENSOR REDUCTION FOR MEASUREMENT OF FULL HAND KINEMATICS*

The random forest algorithm trained on all participants produced a classification accuracy of 100 %. Such a high classification accuracy has been previously reported [5] while classifying hand postures. No hyperparameter tuning was performed since the accuracy was 100%. The normalized joint importance score was then calculated using the Gini algorithm (Figure 14). Based on the results, the top joints (T-IP, I-MCP, I-PIP, M-MCP) were selected. The next 9 joints were ranked similarly as they contribute equally in classifying postures. To decide the 5th joint, combinations of selected 4 joints and one of next 9 joints (T-CMC, M-PIP, L-DIP, L-MCP, L-PIP, R-PIP, I-DIP, R-MCP, and T-MCP) were selected and RMSE was computed. The combination with minimum RMSE was to be selected. The last 2 joints were not selected due to a very low importance score.

The results of hyper-parameter tuning for a neural network for prediction are shown in Figure 15a. For this purpose, the first 5 joints were taken as input to the neural network. All 5 activation functions were run for 500 epochs with a default learning rate of 0.001. Linear activation function yielded lower prediction accuracy while exponential activation function overfitted the data with less test accuracy. We were free to select any of the other three activation functions. We selected a sigmoid function for the prediction. The plot of different learning rates and their effect on performance is shown in Figure 15b. While all learning rates converged well within 500 epochs, a learning rate of 0.01, 0.025, and 0.05 produced similar and faster convergence compared to a learning rate of 0.001. A learning rate of 0.01 was selected and the number of epochs was selected to be 250.

The results of different combination in terms of the RMSE of test data of the first 4 selected joints and one of the additional 9 joints is shown in Figure 16. Training data

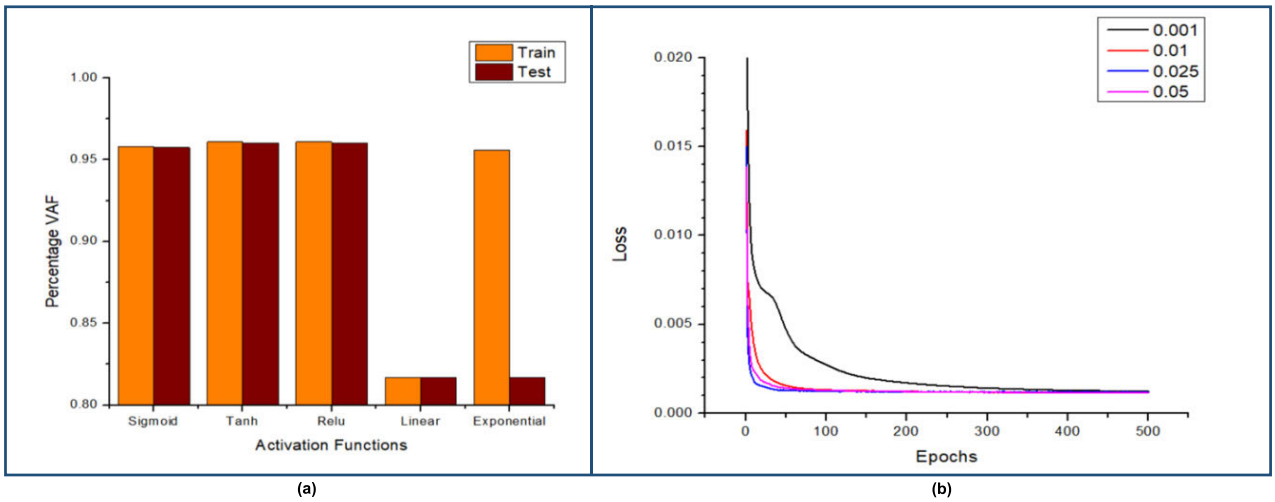


**FIGURE 15.** Hyper-parameter tuning of neural network (a) Percentage VAF for different activation functions (b) Effect of learning rate on network performance.

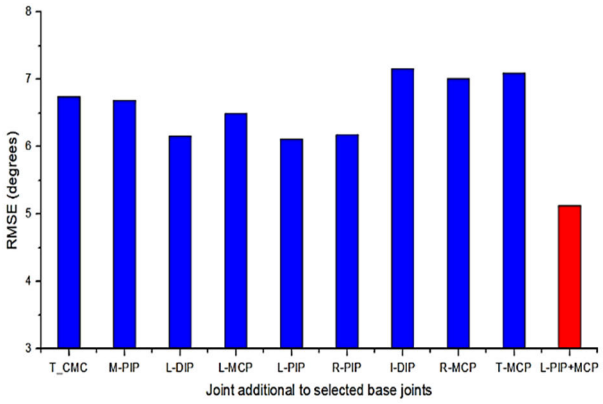


**FIGURE 16.** RMSE for joints selected additional to selected top 4 joints.

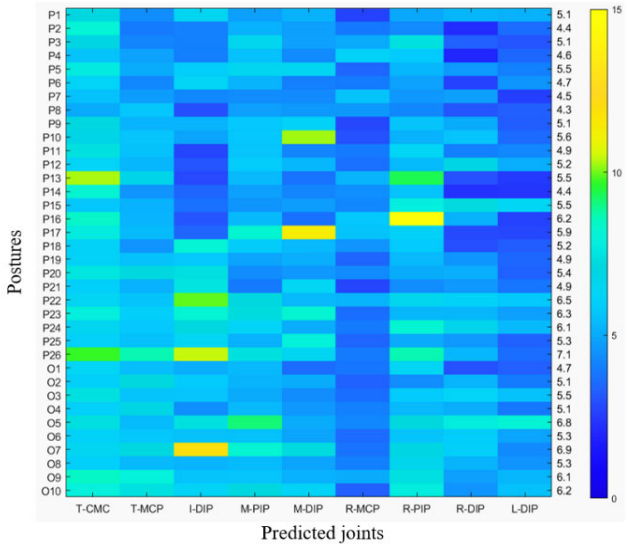


**FIGURE 17.** RMSE for each joint (X-axis) for different postures (Y-axis left). Average RMSE for each posture is displayed on the right Y-axis.

accuracy was also the same as test data and no overfitting was observed (also seen in Figure 15a). While all combinations produced similar RMSE, the lowest RMSE was associated with the PIP joint of the little finger. However, to measure the orientation of the PIP joints, 2 sensors are required: one placed on the proximal phalange and the other placed on the middle phalange. Since a reference sensor is utilized already due to the selection of MCP joints (Index and MCP), the sensor on the proximal phalange of the little finger could be used to measure the MCP orientation as well. Hence, 6 joints were selected (T_IP, I_MCP, I_PIP, M_MCP, L_PIP, L_MCP) and the RMSE in predicting the remaining joints is presented as a red bar in Figure 16. The RMSE was 5.1 degrees.

A detailed analysis of predicted values was performed to identify the postures and joints that performed poorly using the decided sensor combinations. The RMSE for full dataset data (test + train) was segregated based on postures/ object grasp and RMSE is reported for each type of posture and each joint as well. The results are depicted in Figure 17. The values on the right-hand side of the plot depict the average value of RMSE for each posture. As can be seen from Figure 17, the maximum RMSE was observed for Posture 26 with maximum error at the index DIP.

Hence, the real-time visualization of that posture was selected and is demonstrated in Figure 18. From the figure, it is evident that the thumb CMC rotation has a larger error initially while the error reduces as the final posture is obtained. Similarly, the index DIP shows a larger error, but the error reduces and settles down towards the end of the posture. The proposed sensor selection method using a random forest algorithm provides an RMSE of about 5 degrees for the postures and object grasps selected. Postures reconstructed using such an RMSE could be used for applications involving virtual reality applications in which such subtle changes in postures are not evident. For other applications requiring higher accuracy, an increase in sensor number would be necessary for one or many of the following joints - thumb

**TABLE 2.** Comparison of performance.

| | Mora et. Al. [35] | Wheatland et. al. [32] | Jarque et.al. [33] | Current work |
|---|---|---|---|---|
| **Participants** | 6 | 1 | 22 | 4 |
| **Degrees of freedom** | 15 | 18 | 23 | 21 |
| **Sensor selection method** | N.A (Based on dimensions of objects to be grasped | Ranking based on metrics derived from PCA of a single participant | Ranking based on clustered PCA of all many participants | Gini feature selection based on Random Forest algorithm |
| **Tasks** | Grasping | ASL | ADL | Bharatnatyam postures, ASL, and object grasp |
| **Prediction algorithm** | Neural Networks | Locally weighted regression | Nonlinear least square curve fitting | Single hidden layer Neural Network |
| **Measurement domain** | Joint angles (Linear) | Joint positions (Linear) | Joint angles (Linear) | Quaternions (Non-Linear) |
| **Measurement Device** | Data glove | Optical Tracking | Data Glove | Electromagnetic Tracking sensors |
| **Prediction error (RMSE)** | Up to 15 degrees | 22.9 degrees (6 markers) | 15 degrees (8 sensors) | 5.1 degrees (8 sensors) |

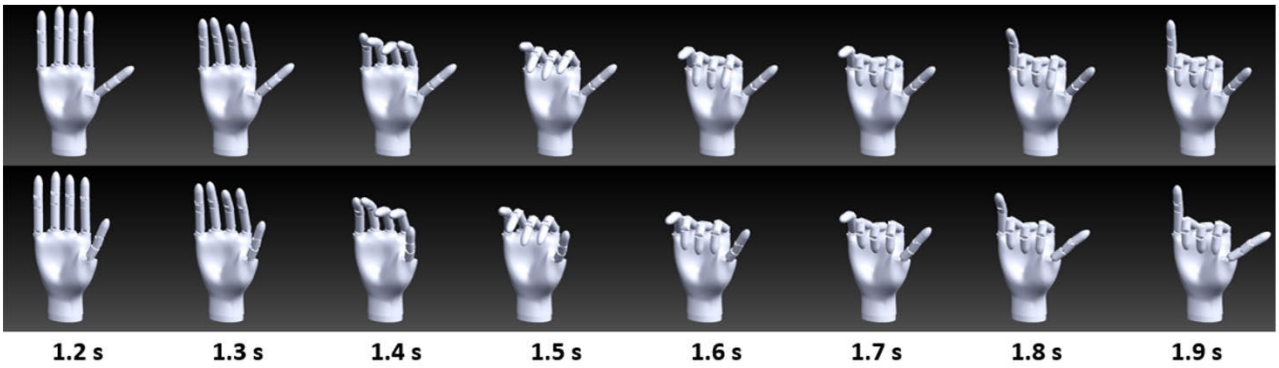


**FIGURE 18.** Visualization of real-time movement for original full set sensors (top row) and movement predicted using only 8 sensors measuring the orientation of 6 joints (bottom row).

CMC, index DIP, Middle DIP, and ring PIP, which have produced higher RMSE occasionally for some postures (Figure 17). A comparison of the results with previous studies is demonstrated in Table 2. While the method adopted in this study produces better accuracy in predicting kinematics from a reduced number of sensors, the study was performed on only 4 participants and a thorough evaluation with a larger number of participants is needed which we view as the future scope for this study. This method can also be applied to IMU's that measure orientation using rotation groups like quaternions.

## V. CONCLUSION

In this paper, we presented methods for the analysis of finger kinematics using EMTS that measure orientation in the form of rotation groups. The sensors were observed to provide good accuracy in terms of measured joint angles without the need for calibration procedures or the presence of a line-of-sight problem. Of the available joint angle representation methods, we chose quaternions over Euler angles and rotation matrices as an initial measurement method considering their advantages. Based on the benchmarking results performed on 3D printed finger designs, we saw that the sensors are

accurate up to at least 1 degree. This is the instrument accuracy and actual accuracy varies during the actual experiment due to several factors. Some of the most common factors are skin artifacts, artifacts due to movement of bones and tendons called soft tissue artifacts, errors induced due to incorrect sensor placement, and errors induced during sensor to segment alignment (bore-sighting). Skin artifacts arise as a result of the sliding of skin and hence the sliding of the sensor over the bone during movement. Soft tissue artifact is prominent near the wrist reference sensor due to the rising and falling of tendons while performing different postures. The area of location of the reference sensor must be therefore carefully inspected before attaching the sensor. Tendon and palm arch-induced errors are mostly induced in the rotation values of the thumb CMC joint due to the error caused by the rotation of the sensor. Upon ensuring minimum artifacts, EMTS along with quaternions provide a reliable approach to measure hand kinematics.

For dimensionality reduction on rotation groups, we presented the results of traditional PCA on exponential maps and autoencoders directly on quaternions. While traditional PCA has the advantages of ease and speed of computation as well as the ability to interpret the latent dimensions to make decisions while designing prosthetics or training for rehabilitation, autoencoders can be configured to provide better compression in the latent dimensions and hence produce better reconstruction compared to PCA. Our results are in agreement with those presented in [5]. While using a single hidden layer with linear activation functions, autoencoders on quaternions produce the same results as that obtained using PCA on exponential maps. However, upon using 5 hidden layers with nonlinear activation functions, autoencoders with both quaternions and exponential maps produced RMSE's up to 5 degrees lesser than that obtained using regular PCA or autoencoders with single hidden layers. For sensor reduction, we presented the preliminary results of a random forest algorithm to select optimal sensors and predict orientations of other sensors using a simple 1 hidden layer neural network. Based on the ranking algorithm, 6 joint orientations measured using 8 sensors were selected (Thumb-IP, Index- MCP, Index- PIP, Middle – MCP, Ring-MCP, and Ring-PIP). The selected sensors produced an RMSE of 5.1 degrees. This could be further reduced by using more neurons in the hidden layers or by adding more hidden layers. While such accuracy can be acceptable for Virtual reality studies, its applicability for synergy analysis must be studied. Also, the method needs to be tested with a larger number of participants. This will be taken up as a future study.

As an application, the methods used for measurement and analysis in this study could also be applied to IMU based hand kinematics measurement systems. Recently, these devices have proven their ability to measure kinematics in a non-laboratory environment with comparable accuracy while being inexpensive. These devices are also programmed to provide output data in the form of quaternions and Euler angles. This way, the EMTS could also be used as a gold standard to compare the performance of the IMU's and hence check their suitability for applications such as dimensionality reduction and measurement of ROM that are necessary for research in neuroscience as well as for clinical diagnosis.

## SUPPLEMENTARY MATERIAL

The animation video of one participant performing postures is attached with the article.

## REFERENCES

[1] S. Li, X. Sheng, H. Liu, and X. Zhu, "Design of a myoelectric prosthetic hand implementing postural synergy mechanically," *Ind. Robot: Int. J.*, vol. 41, no. 5, pp. 447–455, Aug. 2014.

[2] S. Fani, M. Bianchi, S. Jain, J. S. Pimenta Neto, S. Boege, G. Grioli, A. Bicchi, and M. Santello, "Assessment of myoelectric controller performance and kinematic behavior of a novel soft synergy-inspired robotic hand for prosthetic applications," *Frontiers Neurorobotics*, vol. 10, p. 11, Oct. 2016.

[3] T. Geng, M. Lee, and M. Hülse, "Transferring human grasping synergies to a robot," *Mechatronics*, vol. 21, no. 1, pp. 272–284, Feb. 2011.

[4] M. Santello, M. Bianchi, M. Gabiccini, E. Ricciardi, G. Salvietti, D. Prattichizzo, M. Ernst, A. Moscatelli, H. Jörntell, A. M. L. Kappers, K. Kyriakopoulos, A. Albu-Schäffer, C. Castellini, and A. Bicchi, "Hand synergies: Integration of robotics and neuroscience for understanding the control of biological and artificial hands," *Phys. Life Rev.*, vol. 17, pp. 1–23, Jul. 2016.

[5] A. A. Portnova-Fahreeva, F. Rizzoglio, I. Nisky, M. Casadio, F. A. Mussa-Ivaldi, and E. Rombokas, "Linear and non-linear dimensionality-reduction techniques on full hand kinematics," *Frontiers Bioeng. Biotechnol.*, vol. 8, p. 429, May 2020.

[6] I. De Feudis, D. Buongiorno, G. D. Cascarano, A. Brunetti, D. Micele, and V. Bevilacqua, "A nonlinear autoencoder for kinematic synergy extraction from movement data acquired with HTC vive trackers," in *Progresses in Artificial Intelligence and Neural Systems*. Singapore: Springer, 2021, pp. 231–241.

[7] S. Tang, L. Chen, M. Barsotti, L. Hu, Y. Li, X. Wu, L. Bai, A. Frisoli, and W. Hou, "Kinematic synergy of multi-DoF movement in upper limb and its application for rehabilitation exoskeleton motion planning," *Frontiers Neurorobot.*, vol. 13, p. 99, Nov. 2019.

[8] N. J. Jarque-Bou, A. Scano, M. Atzori, and H. Müller, "Kinematic synergies of hand grasps: A comprehensive study on a large publicly available dataset," *J. Neuroeng. Rehabil.*, vol. 16, no. 1, pp. 1–4, Dec. 2019.

[9] M. K. Burns, V. Patel, I. Florescu, K. V. Pochiraju, and R. Vinjamuri, "Low-dimensional synergistic representation of bilateral reaching movements," *Frontiers Bioeng. Biotechnol.*, vol. 5, p. 2, Feb. 2017.

[10] C. Della Santina, M. Bianchi, G. Averta, S. Ciotti, V. Arapi, S. Fani, E. Battaglia, M. G. Catalano, M. Santello, and A. Bicchi, "Postural hand synergies during environmental constraint exploitation," *Frontiers Neurorobot.*, vol. 11, p. 41, Aug. 2017.

[11] N. Jarrassé, A. Ribeiro, A. Sahbani, W. Bachta, and A. Roby-Brami, "Analysis of hand synergies in healthy subjects during bimanual manipulation of various objects," *J. NeuroEngineering Rehabil.*, vol. 11, no. 1, p. 113, 2014.

[12] R. Vinjamuri, M. Sun, C.-C. Chang, H.-N. Lee, R. J. Sclabassi, and Z.-H. Mao, "Temporal postural synergies of the hand in rapid grasping tasks," *IEEE Trans. Inf. Technol. Biomed.*, vol. 14, no. 4, pp. 986–994, Jul. 2010.

[13] R. Vinjamuri, M. Sun, D. Crammond, R. Sclabassi, and Z.-H. Mao, "Inherent bimanual postural synergies in hands," in *Proc. 30th Annu. Int. Conf. IEEE Eng. Med. Biol. Soc.*, Aug. 2008, pp. 5093–5096.

[14] E. Todorov and Z. Ghahramani, "Analysis of the synergies underlying complex hand manipulation," in *Proc. 26th Annu. Int. Conf. IEEE Eng. Med. Biol. Soc.*, vol. 2, Sep. 2004, pp. 4637–4640.

[15] M. Santello, M. Flanders, and J. F. Soechting, "Patterns of hand motion during grasping and the influence of sensory guidance," *J. Neurosci.*, vol. 22, no. 4, pp. 1426–1435, Feb. 2002.

[16] M. Santello, M. Flanders, and J. F. Soechting, "Postural hand synergies for tool use," *J. Neurosci.*, vol. 18, no. 23, pp. 10105–10115, Dec. 1998.

[17] P. H. Thakur, A. J. Bastian, and S. S. Hsiao, "Multidigit movement synergies of the human hand in an unconstrained haptic exploration task," *J. Neurosci.*, vol. 28, no. 6, pp. 1271–1281, Feb. 2008.

[18] C. R. Mason, J. E. Gomez, and T. J. Ebner, "Hand synergies during reach-to-grasp," *J. Neurophysiol.*, vol. 86, no. 6, pp. 2896–2910, Dec. 2001.

[19] J. Zhou, F. Malric, and S. Shirmohammadi, "A new hand-measurement method to simplify calibration in CyberGlove-based virtual rehabilitation," *IEEE Trans. Instrum. Meas.*, vol. 59, no. 10, pp. 2496–2504, Oct. 2010.

[20] A. Roda-Sales, J. L. Sancho-Bru, M. Vergara, V. Gracia-Ibáñez, and N. J. Jarque-Bou, "Effect on manual skills of wearing instrumented gloves during manipulation," *J. Biomech.*, vol. 98, Jan. 2020, Art. no. 109512.

[21] N. J. Jarque-Bou, M. Atzori, and H. Müller, "A large calibrated database of hand movements and grasps kinematics," *Sci. Data*, vol. 7, no. 1, p. 12, Dec. 2020.

[22] A. Mohan, G. Tharion, R. K. Kumar, and S. R. Devasahayam, "An instrumented glove for monitoring hand function," *Rev. Scientific Instrum.*, vol. 89, no. 10, Oct. 2018, Art. no. 105001.

[23] J. Connolly, J. Condell, B. O'Flynn, J. T. Sanchez, and P. Gardiner, "IMU sensor-based electronic goniometric glove for clinical finger movement analysis," *IEEE Sensors J.*, vol. 18, no. 3, pp. 1273–1281, Feb. 2018.

[24] P.-C. Hsiao, S.-Y. Yang, B.-S. Lin, I.-J. Lee, and W. Chou, "Data glove embedded with 9-axis IMU and force sensing sensors for evaluation of hand function," in *Proc. 37th Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. (EMBC)*, Aug. 2015, pp. 4631–4634.

[25] C. K. Mummadi, F. P. P. Leo, K. D. Verma, S. Kasireddy, P. M. Scholl, and K. Van Laerhoven, "Real-time embedded recognition of sign language alphabet fingerspelling in an IMU-based glove," in *Proc. 4th Int. Workshop Sensor-Based Activity Recognit. Interact.*, Sep. 2017, pp. 1–6.

[26] M. M. Rahman, K. Mitobe, M. Suzuki, C. Takano, and N. Yoshimura, "Analysis of dexterous finger movement for piano education using motion capture system," *Int. J. Sci. Technol. Educ. Res.*, vol. 2, no. 2, pp. 22–31, 2011.

[27] M. M. Rahman, A. B. M. A. Hossain, M. M. Rana, and K. Mitobe, "Hand motion capture system in piano playing," in *Proc. Int. Conf. Informat., Electron. Vis. (ICIEV)*, May 2013, pp. 1–5.

[28] J. H. Challis, "Quaternions as a solution to determining the angular kinematics of human movement," *BMC Biomed. Eng.*, vol. 2, no. 1, p. 5, Dec. 2020.

[29] M. P. Johnson, "Exploiting quaternions to support expressive interactive character motion," Ph.D. dissertation, Dept. Media, Arts Sci., Massachusetts Inst. Technol., Cambridge, MA, USA, 2003.

[30] J. Tilmanne and T. Dutoit, "Expressive gait synthesis using PCA and Gaussian modeling," in *Proc. Int. Conf. Motion Games*. Berlin, Germany: Springer, Nov. 2010, pp. 363–374.

[31] H. Du, M. Manns, E. Herrmann, and K. Fischer, "Joint angle data representation for data driven human motion synthesis," *Proc. CIRP*, vol. 41, pp. 746–751, Jan. 2016.

[32] N. Wheatland, S. Jörg, and V. Zordan, "Automatic hand-over animation using principle component analysis," in *Proc. Motion Games*, Nov. 2013, pp. 197–202.

[33] N. J. Jarque-Bou, J. L. Sancho-Bru, and M. Vergara, "Synergy-based sensor reduction for recording the whole hand kinematics," *Sensors*, vol. 21, no. 4, p. 1049, Feb. 2021.

[34] V. Gracia-Ibáñez, J. L. Sancho-Bru, M. Vergara, N. J. Jarque-Bou, and A. Roda-Sales, "Sharing of hand kinematic synergies across subjects in daily living activities," *Sci. Rep.*, vol. 10, no. 1, p. 6116, Dec. 2020.

[35] M. C. Mora, J. L. Sancho-Bru, and A. Pérez-González, "Hand posture prediction using neural networks within a biomechanical model," *Int. J. Adv. Robotic Syst.*, vol. 9, no. 4, p. 139, Oct. 2012.

[36] M. Schröder, J. Maycock, and M. Botsch, "Reduced marker layouts for optical motion capture of hands," in *Proc. 8th ACM SIGGRAPH Conf. Motion Games*, Nov. 2015, pp. 7–16.

[37] M. Schröder, T. Waltemate, J. Maycock, T. Röhlig, H. Ritter, and M. Botsch, "Design and evaluation of reduced marker layouts for hand motion capture," *Comput. Animation Virtual Worlds*, vol. 29, no. 6, p. e1751, Nov. 2018.

[38] S. Schreven, P. J. Beek, and J. B. J. Smeets, "Optimising filtering parameters for a 3D motion analysis system," *J. Electromyogr. Kinesiol.*, vol. 25, no. 5, pp. 808–814, Oct. 2015.

[39] F. S. Grassia, "Practical parameterization of rotations using the exponential map," *J. Graph. Tools*, vol. 3, no. 3, pp. 29–48, Jan. 1998.

[40] F. L. Markley, Y. Cheng, J. L. Crassidis, and Y. Oshman, "Averaging quaternions," *J. Guid., Control, Dyn.*, vol. 30, no. 4, pp. 1193–1197, Jul. 2007.

[41] B. H. Menze, B. M. Kelm, R. Masuch, U. Himmelreich, P. Bachert, W. Petrich, and F. A. Hamprecht, "A comparison of random forest and its Gini importance with standard chemometric methods for the feature selection and classification of spectral data," *BMC Bioinf.*, vol. 10, no. 1, p. 213, Dec. 2009.

[42] H. Park, S. Kwon, and H.-C. Kwon, "Complete Gini-index text (GIT) feature-selection algorithm for text classification," in *Proc. 2nd Int. Conf. Softw. Eng. Data Mining*, Jun. 2010, pp. 366–371.

[43] W. Shang, "Research on the algorithm of feature selection based on Gini index for text categorization," *J. Comput. Res. Develop.*, vol. 43, no. 10, p. 1688, 2006.

[44] S. Deepa and R. Umarani, "Steganalysis on images using SVM with selected hybrid features of Gini index feature selection algorithm," *Int. J. Adv. Res. Comput. Sci.*, vol. 8, no. 5, pp. 1503–1509, May 2017.

[45] S. Nembrini, I. R. König, and M. N. Wright, "The revival of the Gini importance?" *Bioinformatics*, vol. 34, no. 21, pp. 3711–3718, Nov. 2018.

[46] W. Chen, C. Xiong, and S. Yue, "Mechanical implementation of kinematic synergy for continual grasping generation of anthropomorphic hand," *IEEE/ASME Trans. Mechatronics*, vol. 20, no. 3, pp. 1249–1263, Jun. 2015.

[47] M. K. Burns, K. Van Orden, V. Patel, and R. Vinjamuri, "Towards a wearable hand exoskeleton with embedded synergies," in *Proc. 39th Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. (EMBC)*, Jul. 2017, pp. 213–216.

[48] V. Patel, J. Craig, M. Schumacher, M. K. Burns, I. Florescu, and R. Vinjamuri, "Synergy repetition training versus task repetition training in acquiring new skill," *Frontiers Bioeng. Biotechnol.*, vol. 5, p. 9, Feb. 2017.

**PRAJWAL SHENOY** received the B.Tech. and M.Tech. degrees in mechanical engineering. He is currently pursuing the Ph.D. degree in biomedical engineering with the Department of Applied Mechanics, Indian Institute of Technology Madras, India. He is also an Assistant Professor with the Department of Mechatronics Engineering, Manipal Institute of Technology, MAHE, Manipal. His research interests include the analysis of human hand kinematics and the application of computational techniques in analysis of human hand movements.

**VIGNESH SOMPUR** received the B.Tech. degree in mechanical engineering from the Sardar Vallabhbhai National Institute of Technology, Surat, India, in 2017. He is currently pursuing the dual M.S. and Ph.D. degree in neuro-robotics with the Indian Institute of Technology Madras. His research interests include robotics, neuroscience, and design of mechanisms particularly addressing the design of prosthetic hands.

**VARADHAN SKM** received the Ph.D. degree in the area of kinesiology from Pennsylvania State University, State College, PA, USA. He is currently an Associate Professor with the Biomedical Engineering Group at the Department of Applied Mechanics, Indian Institute of Technology Madras, India. His research interests include motor control, use of experimental and computational techniques in understanding neural control of movement, application of engineering in rehabilitation, and applications of noninvasive techniques in biomedical movement analysis.