

Geometric Compression of Orientation Signals for Fast Gesture Analysis

Aswin Sivakumar[†], Rushil Anirudh^{*†}, and Pavan Turaga^{*†}

^{*}School of Arts, Media Engineering

[†]School of Electrical, Computer and Energy Engineering

Arizona State University

Tempe , AZ - 85287

Abstract

This paper concerns itself with compression strategies for orientation signals, seen as signals evolving on the space of quaternions. The compression techniques extend classical signal approximation strategies used in data mining, by explicitly taking into account the quotient-space properties of the quaternion space. The approximation techniques are applied to the case of human gesture recognition from cellphone-based orientation sensors. Results indicate that the proposed approach results in high recognition accuracies, with low storage requirements, with the geometric computations providing added robustness than classical vector-space computations.

1 Introduction

Wearable sensors today are used in a wide variety of applications like health monitoring, fitness tracking, and gaming on smart phones. Many of these applications involve various forms of activity or gesture analysis i.e., making inferences about the type of motion based on the recorded sensor data [1]. For reliable performance, such applications require continuous sensing and recording sensor data, combined with computationally intensive algorithms for inference, both of which can impose a heavy load on resources. This issue manifests, for instance, as reduced battery life on a cellphone while continuously executing activity analysis algorithms. The underlying inference computations can add further load when the data representation has a non-Euclidean interpretation. In the case of activity and gesture analysis, rotation or orientation information is usually represented in the form of quaternions or as rotation matrices, both of which have associated non-Euclidean geometric properties [2]. In such a case, one cannot use classical vector-valued signal approximation strategies, but one needs to take recourse to differential geometry, and extend classical signal approximation techniques with geometric computations. However, non-Euclidean computations are intensive and often iterative, thereby demanding increased resources.

To alleviate this problem, in this paper we propose to use a symbolic approximation method that is an extension of classical vector-quantization approaches used

This research was supported by National Science Foundation grant number CCF-CIF 1320267. Corresponding author - ranirudh@asu.edu

in data mining [3]. Statistical computations on geometric spaces has seen increasing interest in recent years - the geometry of spaces such as the Grassmann and Stiefel manifolds [4], and the hyper-spherical manifolds [5], have been exploited in a wide variety of problems. In this paper, our focus is on quaternion data obtained from wearable and embedded sensors in smart phones which can be used for activity recognition - with the goal to significantly decrease CPU usage for computations without compromising heavily on performance.

Related Work: Analyzing quaternion signals is of significant interest to a wide variety of scientific communities. In particular, the use of differential geometry to analyze orientation signals has received some interest in the past few years addressing problems such as multi-scale representations for manifold valued data [6], multi-view rotation averaging [7, 8] etc. More generally, the issue of reducing the dimensionality of non-linear data is of interest - for example, in the space of Symmetric Positive Definite (SPD) matrices, the geometric properties can be exploited to map the data to a lower dimensional SPD manifold [10]. For the case of numerical manifolds that do not have a defined geometry, manifold learning techniques such as Isomap [9] attempt to visualize the low-dimensional manifold, typically by preserving some property of the original manifold such as neighborhood distances etc. Since searching on non-Euclidean spaces is challenging, hashing techniques have been extended to Riemannian manifolds [11, 12]. However, there is little or no work that addresses the problem of indexing sequences on manifolds which is the main motivation of this work. A recent work addresses efficiently extracting features that are much faster than traditional features [13], this is orthogonal to our contribution and can be combined with the proposed framework for significant improvement in feature extraction and recognition.

2 Signal representations for orientation data

Orientation signals recorded from either wearables or hand-held devices contain significant information about the activity being performed. However, past work in activity analysis has not explicitly exploited orientation data, rather consider it be on par with raw accelerometer data [1]. Representations for orientations are inherently non-Euclidean, which can make further processing difficult. However, we show that it is possible to incorporate non-Euclidean computation within a classical signal matching and approximation framework. We begin with a brief review of the quaternion representation for orientation data, and its associated geometric properties.

2.1 Mathematical framework

Unit quaternions are an efficient means of representing 3D rotations, and have found utility in various application including robotics [14], graphics [15], and animation [16]. A quaternion is a tuple given by $q = [w, x, y, z]^T$, such that $\|q\| = 1$, with the additional constraint that antipodal points $+q$ and $-q$ represent the same rotation. Here, w denotes the magnitude of rotation and x, y, z denote the axis of rotation. One of the benefits of quaternions over Euler angles is that they don't suffer from

gimbal lock, which is the loss of one degree of freedom that occurs when two out of the three axes are driven into a parallel configuration. It is also easier to re-normalize a quaternion in the presence of finite-precision round-off errors.

Base quaternion-space: We denote the base space of quaternions as $Q = \{q \in \mathbb{R}^4 \mid \|q\| = 1\}$ which is also the 3-dimensional hyper-sphere denoted by \mathbb{S}^3 . We call this the base-space, because this representation does not yet account for the fact that antipodal quaternions represent the same physical rotation. The base metric, denoted by d_{base} , can be any of the standard metrics on hyper-spherical manifolds and we will use $d_{base}(q_1, q_2) = \cos^{-1}(\langle q_1, q_2 \rangle)$, which is also the arc-length between two points, or the geodesic distance on the hypersphere.

Quotient-space of quaternions: Since antipodal quaternions represent the same physical orientation, we further define equivalence classes on the base-space to make the representation meaningful. i.e. we define an equivalence relation \sim over Q , such that $q_1 \sim q_2$ if and only if $q_1 = \alpha q_2$, where $\alpha \in \{-1, 1\}$. The resultant space is referred to as a quotient space Q/\sim , or $Q/\{-1, 1\}$ in this case.

Metrics on the quotient-space: The distance between two quaternions is the minimum distance between the corresponding equivalence classes. Thus,

$$d_q([q_1], [q_2]) = \min_{\alpha, \beta \in \{-1, 1\}} d_{base}(\alpha q_1, \beta q_2), \quad (1)$$

where $[q]$ refers to the equivalence class of q . Geometrically, we interpret $Q/\{-1, 1\}$ as a special case of the Grassmann manifold [17] – in this case, each quaternion represents not a point on the hypersphere, but an *axis* passing through the origin and the specified quaternion. This is illustrated in figure 1. Thus, we need to compute the smallest arc-length between two axes passing through the origin and the specified quaternions. This can be written in closed form as $d_q([q_1], [q_2]) = \cos^{-1}(|\langle q_1, q_2 \rangle|)$. (This differs from d_{base} in the $|\cdot|$ operation inside the arccos operation.)

Statistics on the quotient-space: In order to perform signal compression and approximation, we also need to be able to compute sample statistics – such as the average value of the signal in a short time-window, or to obtain a set of quantization prototypes using a clustering procedure for instance. To enable such statistical computations, we take recourse to an extrinsic statistical approach. We *lift* the quaternions to a new space \mathbb{P} - the space of square idempotent rank-1 projection matrices, via the transformation $f : Q \rightarrow \mathbb{P}$, given by $f(q) = qq^T$. Since $f(q) = f(-q)$ the projection matrix is a meaningful representation for the quotient-space $Q/\{-1, 1\}$. The mapping $\Pi : \mathbb{P} \rightarrow Q$ is given by the rank-1 singular

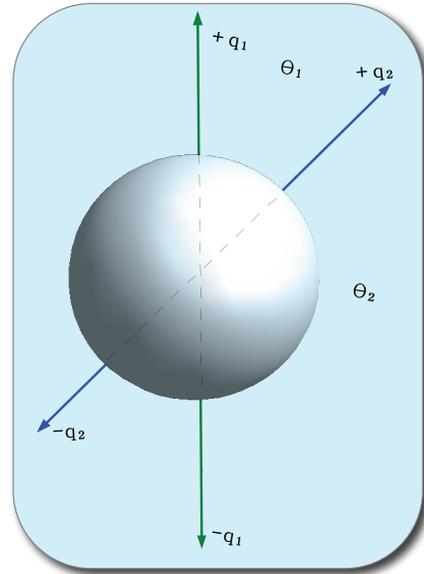


Figure 1: Quaternions represented as axes, and an illustration of the distance computation between two quaternions.

value decomposition (SVD), i.e. $\Pi(P) = U$, where U is the first left singular vector of P . To obtain the sample mean of a set of unit quaternions $q_1, q_2, q_3, \dots, q_n$, we first compute the sample mean of their corresponding projection matrices: $\bar{P} = \frac{1}{n} \sum_{i=1}^n P_i$, where $P_i = q_i q_i^T$. The mean quaternion is given by $\bar{q} = \Pi(\bar{P})$, where Π is as defined above [18].

3 Symbolic approximation of orientation signals

In the applications of interest in this paper, a gesture or activity gives rise to a time-series of quaternions $q(t)$, which we seek to approximate for fast matching with low-storage requirements. We adopt and extend a framework motivated by windowed time-series aggregation used popularly for scalar time-series [3]. In this framework, a time-series $S_j(t)$ of length N is subjected to Piecewise Aggregate Approximation (PAA) where the series is divided into sub-sequences using a window of size w . Each windowed series is replaced with the sample mean of the signal values inside the sub-sequence, which results in a discretized version of the original time-series. The next step called symbolic approximation [3] involves quantization of the aggregated sequence by assigning each aggregated value in the sequence to the nearest ‘symbol’ (or prototype) from a set of pre-defined symbols. In the classical scalar-valued case, this pre-defined set is based on the assumption that the data follows a Gaussian distribution, and therefore quantization levels can be pre-specified [3].

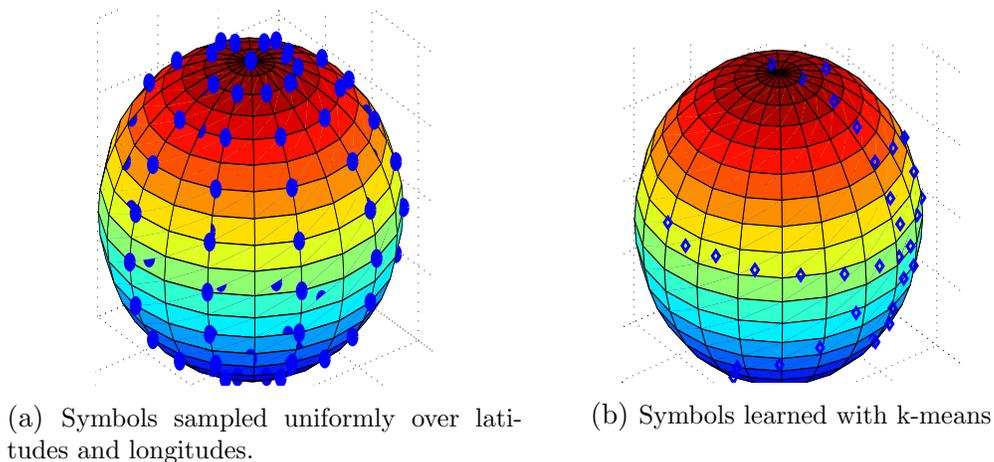


Figure 2: Determining the symbols for quaternion data on \mathbb{S}^2 : (Left) shows symbols distributed across \mathbb{S}^2 , and (Right) shows a set of symbols learned using K-means clustering on \mathbb{P} . This also shows that human activity data results in clusters along sub-manifolds, which necessitates the estimation of quantization symbols from real data.

In the case of non-Euclidean feature spaces, one needs to significantly generalize the notions of aggregation and quantization to take into account the underlying geometry of the space. Using the tools described in section 2, we can estimate a set of

Algorithm 1: K-means on the space of quaternions.

Data: $q_1, q_2 \dots q_N$, Number of clusters K

Result: $\mu_1, \mu_2, \dots, \mu_K$, Cluster memberships $c^{(i)}$

Initialize cluster centroids $\mu_1, \mu_2, \dots, \mu_K$ randomly.

```

while  $t \leftarrow [1 \dots ITER]$  do
  for  $i \leftarrow [1 \dots N]$  do
     $c^{(i)} := \operatorname{argmin}_k d_q(q_i, \mu_k)$ 
    for  $k \leftarrow [1 \dots K]$  do
       $S_k := c^{(i=k)}$ 
       $\mu_k := \Pi \left( \frac{1}{|S_k|} \sum_{j \in S_k} q_j q_j^T \right)$ 
    end
  end
end

```

quantization symbols using K-means clustering on \mathbb{P} – the space of projection matrices, and then map the results back to Q via the operator Π . A simpler approach could be to simply pick symbols at uniformly sampled points along each longitude and latitude on the sphere. In figure 2(a), we show an example of picking quantization symbols at uniformly sampled points, and in figure 2(b) we show symbols that were ‘learned’ using K-means clustering on \mathbb{P} and projected back to Q . The algorithm to perform clustering is outlined in algorithm 1, where $d_q(\cdot)$ is the metric as defined in (1). Note that real movement from humans is not uniformly distributed over the entire space, but clustered along sub-manifolds. This means that with the use of statistical modeling tools one can find fewer symbols that are more effective for signal approximation. One of the advantages of using a pre-learned symbol set is that the distances between symbols can be computed offline – which implies that one can approximately compare two non-Euclidean time-series simply by looking up distances between the corresponding symbolic time-series from a lookup table in constant time. In experiments, we show that this approximate time-series matching is both fast, as well as highly accurate for gesture recognition applications, while also resulting in increased battery life compared to computations on uncompressed data.

Once the quaternion signal has been transformed to a symbolic sequence, we measure the ‘distance’ between two symbolic sequences S_1 and S_2 as

$$D_{min}(S_1, S_2) = \frac{1}{2} \left(\sum_{e \in S_1} \Delta_m(e, S_2) + \sum_{e \in S_2} \Delta_m(e, S_1) \right) \quad (2)$$

where $\Delta_m(x, S)$ returns the distance of the closest point in S from x .

4 Experiments

We perform two sets of experiments to demonstrate the strength of the proposed approach for quaternion signal compression and matching. In the first set of experiments, we use two publicly available datasets to study the effect of quantization and signal approximation on recognition accuracies. In the second experiment, we conduct tests on a dataset we collected using an Android smartphone to illustrate the reduction in smartphone CPU usage while compressing and processing the sensor data vs processing uncompressed data. Before we perform recognition experiments we show that using K-means symbols learned from training data significantly outperforms symbols that are uniformly distributed on Q , the space of quaternions. We collect five gestures typically observed on a smart phone such as tilt left, tilt right, draw a circle, and draw an ‘8’ etc. The rotational information obtained using the gyroscope on the mobile phone is transmitted to a local server. Since these are easier gestures to classify, we are able to approximate the quaternion signals using a large window size of $w = 150$ samples, which is effectively a compression rate of 98.67%, when an overlap of $w/2$ is used. Next, we quantize the aggregated signal using a small symbol set of size $D = 10$ obtained by K-means. As a baseline, we also use a symbol set of the same size distributed uniformly on Q . Table 1 shows the recognition accuracies for the five gestures, and it is evident that the trained symbols do a better job of approximating the sub-manifold with fewer symbols/prototypes.

	Activity 1	Activity 2	Activity 3	Activity 4	Activity 5	Overall
K-means	97.67	97.67	64.78	97.6744	97.6744	91.09
Uniformly Distributed	85.71	76.7	67.7	59.8	97.67	77.54

Table 1: Symbols learned from training data outperform uniformly distributed symbols because they are able to model the sub-manifold more effectively.

4.1 Action recognition on approximated sequences

For this experiment we use the USC-Human Activity Dataset [19] which contains actions recorded on a 3-axis accelerometer and a 3-axis gyroscope sampled at 100Hz. There are 12 action classes such as walk forward, walk left, walk right, go upstairs, go downstairs, run forward, jump up and down, sit and fidget, stand, sleep, elevator up, and elevator down. The dataset contains actions that vary widely in their durations (from 6s to 140s), we choose actions within the dataset that are $\sim 12s - 35s$ long from all actions, which results in a total of 418 actions. Among these actions we consider the first 12s to enforce all the actions to have the same length, this is not a restriction and can be easily alleviated using a warping algorithm such as Dynamic Time Warping (DTW). The accelerometer features lie in \mathbb{R}^3 and the projection matrix interpretation for quaternions on \mathbb{P} . Therefore both the features together lie in a product space of $\mathbb{P} \times \mathbb{R}^3$. The extension of symbolic approximation to a product space is trivial in that

we perform approximation in each space separately and concatenate the final features before classification.

Post processing and classification: To avoid misalignment issues and noise within the data, we represent each discretized and aggregated action sequence in the Fourier Temporal Pyramid (FTP) representation [20]. Here each approximated action sequence is recursively divided into multiple segments in a temporal pyramid, and the low frequency Fourier coefficients from each temporal segment are concatenated to obtain a single feature vector. Next, we perform classification using the k-NN classifier, with $k = 3$, the results are shown in table 2. The parameters that effect the approximation of the orientation signal are the window length w , and the size of the entire symbol set, D or the number of symbols. The trade-off is the following - a larger window size results in a significantly smaller time series, therefore requires lesser storage and computational resources, but also results in larger approximation error and therefore reduced recognition accuracy. Similarly, the approximation error reduces with a large number of symbols, but requires more storage and search time during quantization. The advantage as shown in table 2, is that even by compressing data by 80%, the recognition rate remains within 7.3% of the original uncompressed data. We measure compression ratios as a function of the window size for approximating the sequence, for example a non-overlapping window of size $w = 10$ aggregates a sequence of length 100 to 10, compressing it by 90%.

Compression	Accuracy
0% (original)	78.47
50% + 40 symbols	67.46
50% + 50 symbols	69.38
50% + 60 symbols	69.46
80% + 40 symbols	65.07
80% + 50 symbols	65.55
80% + 60 symbols	67.94
80% + 70 symbols	71.77
80% + 90 symbols	70.33

Table 2: Action recognition at different compression ratios, with non overlapping windows, on accelerometer and quaternion data for the USC-HAD dataset[19]

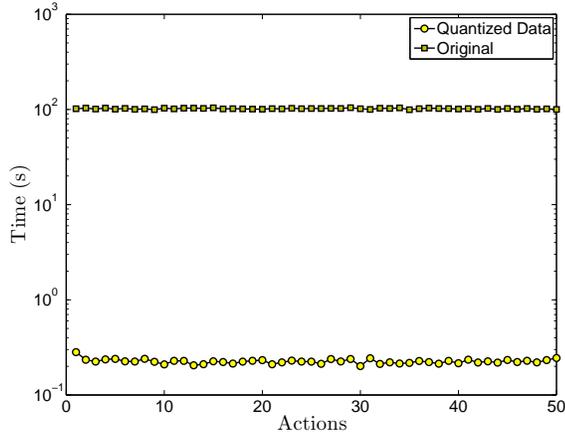


Figure 3: Time taken for a Nearest Neighbor search among 100 action sequences with quantized data is nearly 100× faster than original data of the same length.

4.1.1 Faster searching on the space of quaternions

While a larger window size leads to high compression ratios with negligible loss of recognition accuracy, quantizing the sequence leads to highly efficient searching. This is because the $K \times K$ pairwise distance matrix of the cluster centroids is easily obtained

offline. Therefore comparing two sequences essentially becomes a look-up operation which can be performed in constant time. To demonstrate this, we perform a nearest neighbor search operation on 100 quaternion sequences split into training and testing sets from the the USC-HAD dataset. The time for an action in the test set to find the nearest neighbor in the train set is recorded, and shown in figure 3. It is seen that searching in the symbolic space is nearly 100× faster than searching in quaternion space, all other conditions remaining the same.

Compression	Walking	Jogging	Stairs	Sitting	Standing	Overall
0 %(original)	95.7	98.7	90.7	98.9	98.3	92.98
50% + 30 symbols	92.5	95.2	84.4	98.7	98.4	88
90% + 30 symbols	87.8	93.3	78.7	97.8	89.6	81.37
90% + 60 symbols	88.6	93.7	80.3	98.1	90.4	82.6

Table 3: Variation of recognition accuracy for different compression ratios, and symbols set sizes on WISDM Acti-tracker database [21]

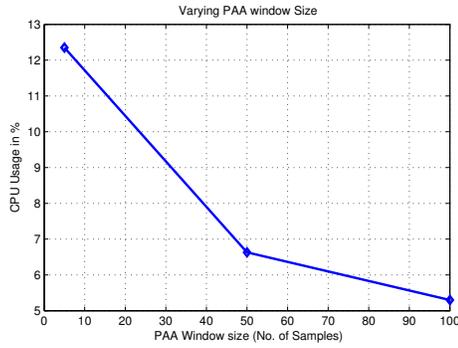
Action recognition on approximated sequences in \mathbb{R}^N : The proposed framework is also general enough to be applied to typical features that lie in Euclidean space. In this experiment, we extract standard statistical features used for activity recognition from accelerometer data such as mean, variance, cross-variance (xy, yz, zx), max value and min value, standard deviation for each dimension. The metric between two sequences in symbolic space is obtained using (2). We perform this experiment on the WISDM Acti tracker dataset dataset [21] which consists of six regularly performed activities: walking, jogging, ascending stairs, descending stairs, sitting, and standing, by 29 users using tri-axial accelerometer data obtained from a smartphone sampled at about 20 Hz. As seen in table 3, we observe that the symbolic sequences are able to preserve much of the important information for reliable recognition, even at high compression rates.

4.2 Computational advantages for approximation

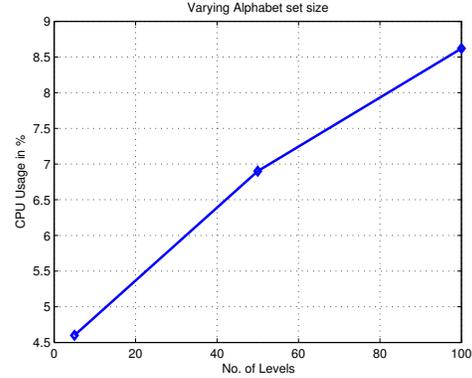
Finally, we demonstrate the advantages of the proposed quaternion signal compression for reduction in smartphone CPU usage and battery consumption while processing, transmission, and recognition of data. We quantify the computational advantage of geometric compression as a function of the two parameters 1) window length and 2) size of the symbol set. Figure 4(a) shows that a larger window size results in reduced number of computations and transmission resources, figure 4(c) shows the same idea applied to battery life decay over time. Similarly, figure 4(b) shows that a larger symbol set requires more processing for the increased search space and storage, which is also reflected in the battery life decay as seen in figure 4(d).

5 Conclusion

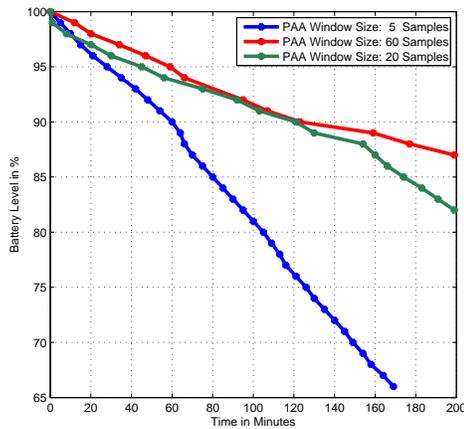
We presented a compression scheme for quaternion time-series, which represent orientation information in 3D space and are often encountered in applications involving



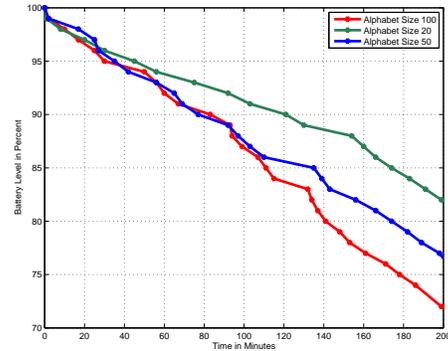
(a) CPU Usage and window size



(b) CPU usage and symbol-set size



(c) Battery usage and window size



(d) Battery usage and symbol-set size

Figure 4: Effects of varying approximation parameters on smartphone CPU usage. Note that there is a significant reduction in CPU usage when working with symbolic compressed versions of data as opposed to uncompressed data.

wearable sensors in medical, fitness, and gaming applications. A major problem for such wearable devices is the limitations on battery life and processing capabilities, which are further complicated because of the non-Euclidean nature of quaternions. We propose to quantize the quaternion time-series using a quantization prototype set that is learned offline using K-means on quaternions. We demonstrate on two publicly available datasets that the approximation of sequences leads to significant reduction in storage and improvement in computational efficiency, but also negligible loss of recognition accuracy as compared to the original features.

References

- [1] O. Lara and M. Labrador, “A survey on human activity recognition using wearable sensors,” *Communications Surveys Tutorials, IEEE*, vol. 15, no. 3, pp. 1192–1209, Third 2013.
- [2] L. Vicci, “Quaternions and rotations in 3-space: The algebra and its geometric interpretation,” Chapel Hill, NC, USA, Tech. Rep., 2001.

- [3] J. Lin, E. J. Keogh, S. Lonardi, and B. Y. chi Chiu, "A symbolic representation of time series, with implications for streaming algorithms," in *DMKD*, 2003, pp. 2–11.
- [4] P. K. Turaga, A. Veeraraghavan, A. Srivastava, and R. Chellappa, "Statistical computations on grassmann and stiefel manifolds for image and video-based recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 11, pp. 2273–2286, 2011.
- [5] X. Pennec, "Intrinsic statistics on Riemannian manifolds: Basic tools for geometric measurements," *Journal of Mathematical Imaging and Vision*, vol. 25, no. 1, pp. 127–154, 2006.
- [6] I. Rahman, I. Drori, V. Stodden, D. Donoho, and P. Schrder, "Multiscale representations for manifold-valued data," *Multiscale Modeling & Simulation*, vol. 4, no. 4, pp. 1201–1232, 2005.
- [7] R. I. Hartley, J. Trumpf, Y. Dai, and H. Li, "Rotation averaging," *International Journal of Computer Vision*, vol. 103, no. 3, pp. 267–305, 2013.
- [8] V. M. Govindu and A. Pooja, "On averaging multiview relations for 3d scan registration," *IEEE Transactions on Image Processing*, vol. 23, no. 3, pp. 1289–1302, 2014.
- [9] J. B. Tenenbaum, V. d. Silva, and J. C. Langford, "A global geometric framework for nonlinear dimensionality reduction," *Science*, vol. 290, no. 5500, pp. 2319–2323, 2000.
- [10] M. T. Harandi, M. Salzmann, and R. Hartley, "From manifold to manifold: Geometry-aware dimensionality reduction for SPD matrices," in *ECCV 2014*, 2014, pp. 17–32.
- [11] P. Turaga and R. Chellappa, "Nearest-neighbor algorithms on non-Euclidean manifolds for computer vision applications," in *ICVGIP*, December 2010.
- [12] R. Chaudhry and Y. Ivanov, "Fast approximate nearest neighbor methods for non-Euclidean manifolds with applications to human activity analysis in videos," in *ECCV*, 2010, pp. 735–748.
- [13] J. Cui and B. Xu, "Cost-effective activity recognition on mobile devices," in *Proceedings of the 8th International Conference on Body Area Networks*. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2013, pp. 90–96.
- [14] L. Sciavicco and L. Villani, *Robotics: modelling, planning and control*. Springer, 2009.
- [15] A. Van Dam and S. K. Feiner, *Computer graphics: principles and practice*. Pearson Education, 2014.
- [16] K. Shoemake, "Animating rotation with quaternion curves," in *ACM SIGGRAPH computer graphics*, vol. 19, no. 3. ACM, 1985, pp. 245–254.
- [17] P.-A. Absil, R. Mahony, and R. Sepulchre, "Riemannian geometry of Grassmann manifolds with a view on algorithmic computation," *Acta Applicandae Mathematicae*, vol. 80, no. 2, pp. 199–220, 2004.
- [18] P. K. Turaga, A. Veeraraghavan, A. Srivastava, and R. Chellappa, "Statistical analysis on manifolds and its applications to video analysis," in *Video Search and Mining*, 2010, pp. 115–144.
- [19] M. Zhang and A. A. Sawchuk, "USC-HAD: A daily activity dataset for ubiquitous activity recognition using wearable sensors," in *ACM International Conference on Ubiquitous Computing (Ubicomp) Workshop on Situation, Activity and Goal Awareness (SAGAware)*, Pittsburgh, Pennsylvania, USA, September 2012.
- [20] J. Wang, Z. Liu, Y. Wu, and J. Yuan, "Mining actionlet ensemble for action recognition with depth cameras," in *CVPR*, June 2012, pp. 1290–1297.
- [21] J. R. Kwapisz, G. M. Weiss, and S. A. Moore, "Activity recognition using cell phone accelerometers," in *Proceedings of the Fourth International Workshop on Knowledge Discovery from Sensor Data*, 2010, pp. 10–18.