A Clustering Method for Geometric Data based on Approximation using Conformal Geometric Algebra

Minh Tuan Pham
Nagoya University
minhtuan@cmplx.cse.nagoya-u.ac.jp

Kanta Tachibana
Kogakuin University
kanta@cc.kogakuin.ac.jp

Tomohiro Yoshikawa
Nagoya University
yoshikawa@cse.nagoya-u.ac.jp

Takeshi Furuhashi
Nagoya University
furuhashi@cse.nagoya-u.ac.jp

***Abstract*—Clustering is one of the most useful methods for understanding similarity among data. However, most conventional clustering methods do not pay sufficient attention to the geometric properties of data. Geometric algebra (GA) is a generalization of complex numbers and quaternions able to describe spatial objects and the relations between them. This paper uses conformal GA (CGA), which is a part of GA, to transform a vector in a real vector space into a vector in a CGA space and presents a proposed new clustering method using conformal vectors. In particular, this paper shows that the proposed method was able to extract the geometric clusters which could not be detected by conventional methods.**

***Index Terms*—conformal geometric algebra, hyper-sphere, inner product, distance, clustering.**

I. Introduction

Because nowadays enormous amounts of data are available in various fields, data mining, which analyzes data to reveal interesting information, becomes increasingly necessary. Clustering is an important technique in data mining. Methods of clustering [1] iteratively minimize an objective function based on a distance measure to label each individual data point. To take the geometric structure of data into consideration, [2] adopted a distance measure based on straight lines, and [3] adopted a distance measure based on spherical shells. The straight line measure and the spherical shell measure are applicable to datasets which have clusters generated respectively by translational motion and by rotational motion in m-dimensional space. But so far, a distance measure which detects a mixture of clusters generated by either translational or rotational motions has not been proposed. Furthermore, the spherical shell measure cannot detect a cluster around a spherical shell with dimension less than m. For example, it cannot express a data distribution around a circle in three-dimensional space because it optimizes an objective function based on distance from a three-dimensional spherical shell. This paper proposes a clustering method which can detect clusters with either translational or rotational relations. The proposed method can also detect clusters around spheres and planes of any dimension less than m because they are well expressed as elements of conformal geometric algebra.

In this study, we use geometric algebra (GA), which can describe spatial vectors and the higher-order subspace relations between them [4],[5],[6]. There are already many successful examples of its use in image processing, signal processing, color image processing, and multi-dimensional time-series signal processing with complex numbers or quaternions, which are low-dimensional GAs [7],[8],[9],[10],[11],[12],[13]. In addition, GA-valued neural network learning methods for learning input–output relationships [14] and the feature extraction based on GA [15] are well studied. This paper uses the vectors in conformal GA (CGA) [16] space, which is a part of GA to propose a new approach of approximation and a new clustering method for geometric data. A vector in CGA is called a conformal vector, and it can express both m-dimensional spheres and planes (infinite radius spheres). A conformal bivector, called a 2-blade in CGA, expresses the intersection of two m-dimensional elements, and thus it is able to express an $m-1$-dimensional sphere or plane. Likewise, a k-blade in CGA expresses an $m-k$-dimensional element. A conformal vector can also express a point as a sphere with zero radius.

In this paper, data vectors in the original space are first translated into conformal vectors expressing the data points. Then, the conformal vectors expressing the cluster centers are optimized using a distance measure on the cluster center conformal vectors and the data points. This optimization problem is solvable as an eigen-value decomposition using the Lagrange multiplier formulation. Each eigen-value is proportional to the sum of distances from an m-dimensional element expressed by the corresponding eigen conformal vectors to each of the data points. The sum of k eigen-values can be considered to be the sum of the distances from the intersection of k elements to the data points. The k eigen conformal vectors corresponding to the smallest k eigen-values can detect a cluster which is distributed over an $m-k$-dimensional spherical or planar surface.

A characteristic of the proposed method is that it is able to detect a cluster distributed over a sphere, plane, or intersection of some combination of the three, for example, a straight line or arc. Section II describes our proposal to approximate geometric data as conformal vectors and shows an algorithm of the proposed clustering method. Section III demonstrates the effectiveness of the proposed method through two experiments. One experiment estimates the probability density distribution of geometric data, and the other is a clustering experiment. The results of the first experiment demonstrate that the proposed method can approximate 2 express geometric data precisely.

The results of the second experiment demonstrate that the kernel alignment [17] between the true labels and the labels detected by clustering using the proposed method is better than those from using conventional methods.

II. Proposed Method

A. Conformal Geometric Algebra and Estimation of Hyper-spheres (-planes) from Geometric Data Sets

CGA is a special part of GA, which is also called Clifford algebras. GA defines the signature of $p+q$ orthonormal basis vectors $\mathcal{O} = \{\mathbf{e}_1, \ldots, \mathbf{e}_p, \mathbf{e}_{p+1}, \ldots, \mathbf{e}_{p+q}\}$, $\{\mathbf{e}_i\}$, such that $\mathbf{e}_i^2 = +1, \forall i \in \{1, \ldots, p\}$ and $\mathbf{e}_i^2 = -1, \forall i \in \{p+1, \ldots, p+q\}$. This paper denotes the geometric algebra determined from $\mathcal{O}$ by $\mathcal{G}_{p,q}$. For example, the geometric algebra determined from the m-dimensional real vector space $\mathbf{R}^m$ is denoted by $\mathcal{G}_{n,0}$.

A CGA space is extended from the real vector space $\mathbf{R}^m$ by adding 2 orthonormal basis vectors. Thus, a CGA space is determined by $m+2$ basis vectors $\{\mathbf{e}_1, \ldots, \mathbf{e}_m, \mathbf{e}_+, \mathbf{e}_-\}$, where $\mathbf{e}_+$ and $\mathbf{e}_-$ are defined so that

$$\begin{aligned} \mathbf{e}_+^2 = \mathbf{e}_+ \cdot \mathbf{e}_+ &= 1, \\ \mathbf{e}_-^2 = \mathbf{e}_- \cdot \mathbf{e}_- &= -1, \\ \mathbf{e}_+ \cdot \mathbf{e}_- = \mathbf{e}_+ \cdot \mathbf{e}_i = \mathbf{e}_- \cdot \mathbf{e}_i &= 0, \forall i \in \{1, \ldots, m\}. \end{aligned} \quad (1)$$

Therefore, a CGA space can be expressed by $\mathcal{G}_{m+1,1}$. This paper then defines the converted basis vectors $\mathbf{e}_0$ and $\mathbf{e}_\infty$ as

$$\mathbf{e}_0 = \frac{1}{2}(\mathbf{e}_- - \mathbf{e}_+), \mathbf{e}_\infty = (\mathbf{e}_- + \mathbf{e}_+). \quad (2)$$

From Eq. (1) and Eq. (2), we get

$$\begin{aligned} \mathbf{e}_0 \cdot \mathbf{e}_0 = \mathbf{e}_\infty \cdot \mathbf{e}_\infty &= 0, \\ \mathbf{e}_0 \cdot \mathbf{e}_\infty = \mathbf{e}_\infty \cdot \mathbf{e}_0 &= -1, \\ \mathbf{e}_0 \cdot \mathbf{e}_i = \mathbf{e}_\infty \cdot \mathbf{e}_i &= 0, \forall i \in \{1, \ldots, m\}. \end{aligned} \quad (3)$$

We also make use of a transform based upon that proposed by Hestenes[16]. The real vector $\mathbf{x} = \sum_i^m x_i \mathbf{e}_i \in \mathbf{R}^m$ is extended to a point $P \in \mathcal{G}_{m+1,1}$ according to the equation

$$P = \mathbf{x} + \frac{1}{2}\|\mathbf{x}\|^2 \mathbf{e}_\infty + \mathbf{e}_0. \quad (4)$$

Note that the inner product of P with itself, $P \cdot P$, is 0. Then, a sphere is represented as a conformal vector

$$\begin{aligned} S &= P - \frac{1}{2} r^2 \mathbf{e}_\infty \\ &= \mathbf{x} + \frac{1}{2}\{\|\mathbf{x}\|^2 - r^2\}\mathbf{e}_\infty + \mathbf{e}_0, \end{aligned} \quad (5)$$

where the sphere has center $\mathbf{x}$ and radius r in real space $\mathbf{R}^m$. The inner product $S \cdot Q$ is 0 for any point Q on the surface of the sphere. From Eqs. 4 and 5, we can see that a point is naturally expressed as a sphere with radius $r = 0$ in CGA space $\mathcal{G}_{m+1,1}$.

CGA also expresses a plane as a conformal vector:

$$L = \mathbf{n} + d\mathbf{e}_\infty, \quad (6)$$

where $\mathbf{n}$ is the normal vector, and d is a scalar coefficient of the plane $\mathbf{n} \cdot \mathbf{x} + d = 0$ in real space $\mathbf{R}^m$. The inner product $L \cdot Q$ is 0 for any point Q in the plane.

A conformal vector S in $\mathcal{G}_{n+1,1}$ is generally written in the following form:

$$S = \mathbf{s} + s_\infty \mathbf{e}_\infty + s_0 \mathbf{e}_0. \quad (7)$$

$\mathbf{s} = \sum_i^m s_i \mathbf{e}_i$ is a vector in the real vector space $\mathbf{R}^m$. s_∞ and s_0 are the scalar coefficients of the basis vectors $\mathbf{e}_\infty$ and $\mathbf{e}_0$.

In this paper, we use that the inner product between a point P and a conformal vector S in CGA space is proportional to the distance between them. From Eqs. (3), (4), and (7), the inner product of P and S is

$$\begin{aligned} d(P,S) &\propto P \cdot S \\ &= \left(\mathbf{x} + \frac{1}{2}\|\mathbf{x}\|^2 \mathbf{e}_\infty + \mathbf{e}_0\right) \cdot (\mathbf{s} + s_\infty \mathbf{e}_\infty + s_0 \mathbf{e}_0) \\ &= \mathbf{x} \cdot \mathbf{s} - s_\infty - \frac{1}{2}\|\mathbf{x}\|^2 s_0. \end{aligned} \quad (8)$$

When $d(P,S) = 0$, we get

$$\begin{aligned} & \mathbf{x} \cdot \mathbf{s} - s_\infty - \frac{1}{2}\|\mathbf{x}\|^2 s_0 = 0 \\ \Leftrightarrow & \begin{cases} \mathbf{x} \cdot \mathbf{s} - s_\infty = 0 & (s_0 = 0), \\ \|\mathbf{x} - \frac{1}{s_0}\mathbf{s}\|^2 = \frac{\|\mathbf{s}\|^2 - 2 s_\infty s_0}{s_0^2} & (s_0 \neq 0). \end{cases} \end{aligned} \quad (9)$$

Eq. (9) shows that when $d(P,S) = 0$, the point P lies on the hyper-plane or hyper-sphere expressed in CGA space as conformal vector S. Herein, we use this characteristic to approximate geometric data having rotational or the translational relations and propose a new approach for clustering.

Given a set of vectors $\mathcal{X} = \{\mathbf{x}_k = \sum_i^m x_{k;i} \mathbf{e}_i \in \mathbf{R}^m, k = 1, \ldots, n\}$, the proposed method transforms each m-dimensional vector to a point in CGA space using Eq. (4) to yield the set $\mathcal{P} = \{P_k = \mathbf{x}_k + \frac{1}{2}\|\mathbf{x}_k\|^2 \mathbf{e}_\infty + \mathbf{e}_0, k = 1, \ldots, n\}$ in $\mathcal{G}_{m+1,1}$. Then, the proposed method approximates the set of points $\mathcal{P}$ using a vector $S = \mathbf{s} + s_\infty \mathbf{e}_\infty + s_0 \mathbf{e}_0$ in $\mathcal{G}_{m+1,1}$, which is estimated using least squares. The error function E is defined as

$$\begin{aligned} E &= \sum_{k=1}^n d^2(P_k, S) \\ &= \sum_{k=1}^n \left(\mathbf{x}_k \cdot \mathbf{s} - s_\infty - \frac{1}{2}\|\mathbf{x}_k\|^2 s_0\right)^2. \end{aligned} \quad (10)$$

Eq. 9 can be solved by modifying the equation while preserving the equality, such as by multiplying both sides by a scalar value. Therefore, $\mathbf{s}$ can be restricted to $\|\mathbf{s}\|^2 = 1$. In this case, the optimization problem becomes

$$\min \sum_{k=1}^n \left(\mathbf{x}_k \cdot \mathbf{s} - s_\infty - \frac{1}{2}\|\mathbf{x}_k\|^2 s_0\right)^2 \quad (11)$$

subject to

$$\|\mathbf{s}\|^2 = 1. \quad (12)$$

Therefore, we might be tempted to express the previous problem by means of a non-negative Lagrange multiplier λ as the minimization of

$$L = \frac{1}{n}\sum_{k=1}^{n}\left(\mathbf{x}_k \cdot \mathbf{s} - s_\infty - \frac{1}{2}\|\mathbf{x}_k\|^2 s_0\right)^2 - \lambda\left(\|\mathbf{s}\|^2 - 1\right). \quad (13)$$

The critical values of L occur when its gradient is zero. The partial derivatives are

$$\frac{\partial L}{\partial \mathbf{s}} = \frac{2}{n}\sum_{k=1}^{n}\left(\mathbf{x}_k \cdot \mathbf{s} - s_\infty - \frac{1}{2}\|\mathbf{x}_k\|^2 s_0\right)\mathbf{x}_k - 2\lambda\mathbf{s} = 0, \quad (14)$$

$$\frac{\partial L}{\partial s_\infty} = -\frac{2}{n}\sum_{k=1}^{n}\left(\mathbf{x}_k \cdot \mathbf{s} - s_\infty - \frac{1}{2}\|\mathbf{x}_k\|^2 s_0\right) = 0, \quad (15)$$

$$\frac{\partial L}{\partial s_0} = -\frac{1}{n}\sum_{k=1}^{n}\left(\mathbf{x}_k \cdot \mathbf{s} - s_\infty - \frac{1}{2}\|\mathbf{x}_k\|^2 s_0\right)\|\mathbf{x}_k\|^2 = 0. \quad (16)$$

From Eqs. (15) and (16),

$$s_\infty = \mathbf{f}_\infty \cdot \mathbf{s}, \quad (17)$$

$$\frac{1}{2}s_0 = \mathbf{f}_0 \cdot \mathbf{s}, \quad (18)$$

where

$$\mathbf{f}_\infty = \frac{-\sum_{k=1}^{n}\|\mathbf{x}_k\|^4\sum_{k=1}^{n}\mathbf{x}_k + \sum_{k=1}^{n}\|\mathbf{x}_k\|^2\sum_{k=1}^{n}\|\mathbf{x}_k\|^2\mathbf{x}_k}{\sum_{k=1}^{n}\|\mathbf{x}_k\|^2\sum_{k=1}^{n}\|\mathbf{x}_k\|^2 - \sum_{k=1}^{n}\|\mathbf{x}_k\|^4\sum_{k=1}^{n}1}, \quad (19)$$

$$\mathbf{f}_0 = \frac{\sum_{k=1}^{n}\|\mathbf{x}_k\|^2\sum_{k=1}^{n}\mathbf{x}_k - \sum_{k=1}^{n}1\sum_{k=1}^{n}\|\mathbf{x}_k\|^2\mathbf{x}_k}{\sum_{k=1}^{n}\|\mathbf{x}_k\|^2\sum_{k=1}^{n}\|\mathbf{x}_k\|^2 - \sum_{k=1}^{n}\|\mathbf{x}_k\|^4\sum_{k=1}^{n}1}. \quad (20)$$

More simply, we define the equation

$$\mathbf{f}(\mathbf{x}) = \mathbf{x} - \mathbf{f}_\infty - \|\mathbf{x}\|^2\mathbf{f}_0 \in \mathbf{R}^m. \quad (21)$$

Then, Eq. (13) can be rewritten as

$$L = \mathbf{s}^T\mathbf{A}\mathbf{s} - \lambda\left(\|\mathbf{s}\|^2 - 1\right), \quad (22)$$

where

$$\mathbf{A} = \frac{1}{n}\sum_{k=1}^{n}\mathbf{f}(\mathbf{x}_k)\,\mathbf{f}^T(\mathbf{x}_k). \quad (23)$$

Therefore, the optimization problem is solved by the eigen decomposition

$$\mathbf{A}\mathbf{s} = \lambda\mathbf{s}. \quad (24)$$

An eigen vector $\mathbf{s}$ is an eigen conformal vector defining the hyper-sphere (-plane) $S = \mathbf{s} + s_\infty\mathbf{e}_\infty + s_0\mathbf{e}_0$, and the eigen-value λ is the variance, i.e., the sum of the squared distances between P_k and S. The estimated hyper-sphere or -plane will exactly fit the data set if the eigen-value λ is 0. Because there are m eigen-values when the original data space is m-dimensional, we can find m solutions of hyper-spheres (or planes). Thus, we can also detect the $m-k$-dimensional hyper-spheres (or planes) using the k eigen conformal vectors corresponding to the smallest k eigen-values. For example, in the case of three-dimensional data, we can detect a circle which is a intersection between two spheres having the smallest eigen-values. Furthermore, by also using the sphere with the largest eigen-value, we can detect a cluster of data with an arc, i.e., part of a circle. So, the proposed method is expected to improve clustering performance.

B. Geometric Clustering Method

This paper proposes a new clustering method based on the proposed approximation for data with both the mixture of rotational and translational relationships. For each cluster $\mathcal{X}_c = \{\mathbf{x}_k = \sum_i^m x_{k;i}\mathbf{e}_i \in \mathbf{R}^m \mid y_k = c, k \in \{1,\ldots,n\}\}$, the proposed method uses Eqs. (17), (18), and (24) to estimate the m pairs of hyper-spheres (-planes) $S_{c,l} = \mathbf{s}_{c,l} + s_{\infty c,l}\mathbf{e}_\infty + s_{0c,l}\mathbf{e}_0, l \in \{1,\ldots,m\}$ and eigen-values $\lambda_{c,l}$, where the eigen-values are the variances based on the distance $d(P, S_{c,l})$ between a point $P = \mathbf{x} + \frac{1}{2}\|\mathbf{x}\|^2\mathbf{e}_\infty + \mathbf{e}_0, \mathbf{x} \in \mathcal{X}_c$ and the hyper-sphere (-plane) $S_{c,l}$. Herein, we assume that the variance based on $d(P, S_{c,l})$ follows a Gaussian distribution. Then the density function is

$$p(\mathbf{x} \mid \lambda_{c,l}) = \frac{1}{\sqrt{2\pi\lambda_{c,l}}}\exp\left(-\frac{d^2(P, S_{c,l})}{2\lambda_{c,l}}\right). \quad (25)$$

Next, we assume that the variances corresponding to the eigenvalues are mutually statistically independent. Under these assumptions, the posterior probability density function for $\mathbf{x}$ given the cluster $y = c$ can be defined as

$$p(\mathbf{x} \mid c) = \prod_l^m p(\mathbf{x} \mid \lambda_{c,l}). \quad (26)$$

Then the following shows an algorithm of the proposed method using Eq. (26).

- Step1 Choose the number of clusters, C.
- Step2 Randomly assign label $y_k \in \{1,\ldots,C\}$ for each data point k.
- Step3 Estimate the posterior probability density function $p(\mathbf{x} \mid c)$ for all clusters $\mathcal{X}_c = \{\mathbf{x}_k \mid y_k = c\}$.
- Step4 Update the data label by $y_k = \arg\max_c p(\mathbf{x} \mid c)P(c)$, where $P(c) = \frac{|\mathcal{X}_c|}{n}$ is the prior distribution of cluster $\mathcal{X}_c$, and $|\mathcal{X}_c|$ is the number of data in cluster $\mathcal{X}_c$.
- Step5 Repeat the two previous steps until the algorithm has converged.

III. Experiments and Discussion

A. Estimation of Probability Density Function for Geometric Data Set

This subsection focuses on estimation of the probability density function for three kinds of two-dimensional geometric data: a circle, a line, and an arc. Tables I, II, and III show

the generating parameters and the parameters of the estimated conformal vectors. α and ϵ are random variables following the uniform distribution. x and y are the coordinates of the generated data. λ, $\mathbf{s}$, s_∞, and s_0 are the parameters which are estimated by the proposed method. $r = \sqrt{\frac{\|\mathbf{s}\|^2 - 2s_\infty s_0}{s_0^2}}$ is a radius calculated from the estimated parameters.

TABLE I
GENERATING AND ESTIMATED PARAMETERS FOR THE CIRCLE.

Generating Parameters				
α	uniform distribution on $[-\pi; \pi]$			
ϵ	uniform distribution on $[-0.5, -0.5]$			
x	$5 + (3.5 + \epsilon)\sin\alpha$			
y	$5 + (3.5 + \epsilon)\cos\alpha$			
Estimated Parameters				
λ	$\mathbf{s}$	s_∞	s_0	r
0.017	-0.71, -0.70	-2.70	-0.14	3.61
6.626	0.70, -0.71	0.12	-0.02	46.69

TABLE II
GENERATING AND ESTIMATED PARAMETERS FOR THE LINE.

Generating Parameters				
α	uniform distribution on $[1.5, 8.5]$			
ϵ	uniform distribution on $[-0.5, -0.5]$			
x	α			
y	$\alpha + \epsilon$			
Estimated Parameters				
λ	$\mathbf{s}$	s_∞	s_0	r
0.038	-0.75, 0.66	-0.17	-0.01	83.40
0.283	-0.66, -0.75	-3.07	-0.14	2.98

TABLE III
GENERATING AND ESTIMATED PARAMETERS FOR THE ARC.

Generating Parameters				
α	uniform distribution on $[-\frac{\pi}{3}; \frac{\pi}{3}]$			
ϵ	uniform distribution on $[-0.5, -0.5]$			
x	$5 + (3.5 + \epsilon)\sin\alpha$			
y	$5 + (3.5 + \epsilon)\cos\alpha$			
Estimated Parameters				
λ	$\mathbf{s}$	s_∞	s_0	r
0.013	-0.78, -0.63	-3.18	-0.13	2.85
0.399	0.63, -0.78	3.37	-0.04	28.77

Table I shows that the circle data is approximated by a circle with radius $r_{circle} = 3.61$ when the smallest eigen-value equals 0.017. The approximated circle is very similar to the circle of the generating parameters. Table II shows that the line data is approximated by a circle with a large radius ($r_{line} = 83.40$) when the smallest eigen-value equals 0.038. Thus, it is easy to see that the parameters which are estimated using the proposed method shown in Table II are reasonable if one considers a line as a circle with infinite radius. Table III shows the generating and estimated parameters for the arc data. As shown, the radius $r_{arc} = 2.85$ of the estimated circle is smaller than the generating radius $\hat{r}_{arc} = 3.5$. This shows that the radius of the estimated circle with smallest eigen-value in the case of arc data is different from the case of circle or line data. Thus, the proposed method is expected to be able to express arc data using a combination of all the estimated circles.

Next, we will use Eq. (26) to estimate the probability density function for geometric data. Fig. 1 shows the data distribution and the estimation results of the probability density function using the proposed method. Figs. 1(a), (b), and (c) show the data distributions for the circle, line, and arc data generated using the parameter values listed in Tables I, II, and III, respectively. Fig. 1(d), (e), and (f) show the estimated probability densities corresponding respectively to the data shown in Figs. 1(a), (b), and (c) using the proposed method. The results shown in Fig. 1 demonstrate that the proposed method can estimate the probability density for an arc, not just circles and lines. For a general m-dimensional vector space, it is easy to understand that the proposed method can estimate exactly the probability density for the hyper-sphere, -plane, and -arc.

B. Clustering Experiment

This subsection uses 2 kinds of triangular pyramid data for an experiment of clustering. Table IV shows the coordinate of vertexes of triangular pyramids which was used in this experiment. For each triangular pyramid $i \in \{1, 2\}$, we have a set of vectors $\xi_i = \{\mathbf{p}_{ij} = (\mathbf{p}_{ij,x}, \mathbf{p}_{ij,y}, \mathbf{p}_{ij,z}) \in \mathbf{R}^3; j = 1, 2, 3, 4\}$. We generated a data set $D = \{f(\xi_i, \Sigma)\}\}$ that randomly rotated each triangular pyramid 250 times, where Σ is a random rotation matrix. For each rotated triangular pyramids, we have a feature vector $f(\xi_i) = [\mathbf{p}_{i1}\Sigma, \ldots, \mathbf{p}_{i1}\Sigma] \in \mathbf{R}^{12}$ made by arranging 4 groups of 3 coordinates of each vector in a row. This experiment compares k-means[1], hard C Spherical Shells (HCSS) [3] and proposed clustering algorithm. In this paper, we choose true cluster number $C = 2$ for all algorithms.

TABLE IV
COORDINATE OF VERTEXES OF TRIANGULAR PYRAMIDS.

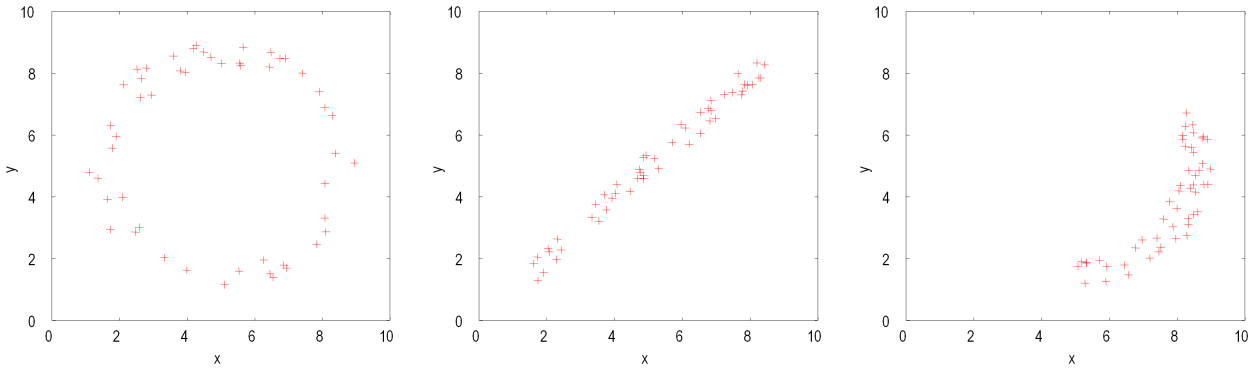
	Triangular pyramid 1			Triangular pyramid 2		
	x	y	z	x	y	z
$\mathbf{p}_1$	0	1	1	1	0	0
$\mathbf{p}_2$	1	0	1	0	1	0
$\mathbf{p}_3$	1	1	0	0	0	1
$\mathbf{p}_4$	0	0	0	-1	-1	-1

Because the results depend on the randomly assigned initial labels in most clustering methods, this experiment is performed using five different random sets of initial labels for each clustering method. The following kernel alignment function is used to evaluate the performance of the clustering results:

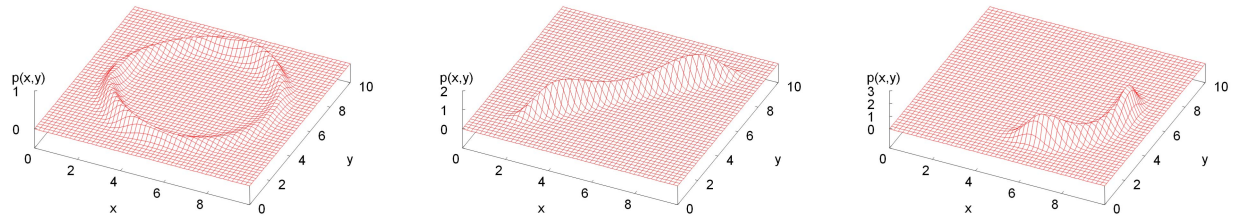
$$A(Y_t, Y_e) = \frac{\sum_{k,l} y_{kl;t} y_{kl;e}}{\sqrt{\sum_{k,l} y_{kl;t}^2} \sqrt{\sum_{k,l} y_{kl;e}^2}} \in [0, 1], \tag{27}$$

where $Y_t = [y_{kl;t}]$ and $Y_e = [y_{kl;e}]$ are symmetric matrices with n rows and n columns corresponding respectively to the generating data labels and the estimated data labels. y_{kl} indicates whether labels are different between instance k and instance l, and is defined by

$$y_{kl} = \begin{cases} 1 & (y_k = y_l), \\ 0 & (y_k \neq y_l). \end{cases} \tag{28}$$



(a) circle (b) line (c) arc



(d) probability density for a circle (e) probability density for a line (f) probability density for an arc

Fig. 1. Estimated probability densities for the (a) circle, (b) line, and (c) arc.

Table V shows the average of the kernel alignment when using k-means, HCSS, and the proposed method. As shown in Table V, the kernel alignment when using the proposed method was better than when using k-means or HCSS.

TABLE V
AVERAGE ALIGNMENTS OF K-MEANS, HCSS, AND THE PROPOSED METHOD.

Method	Average alignment
k-mean	0.50053
HSSC	0.50082
CGA	1.00000

The clustering based on k-means uses the center of each cluster as representative, and so it is not able to cluster geometric data based on hyper-spheres. HCSS can cluster the data based on hyper-spheres. But in this experiment, the vectors made by rotating from two triangular pyramids have a norm $\|f(\xi_i)\|_2 = \sum_j \|\mathbf{p}_{ij}\|^2 = 6, \forall i \in \{1,2\}$. So, these vectors lie on the same hyper-sphere in 12-dimensional space. Therefore, the clustering result of HCSS was not good. Unlike k-means and HCSS, the proposed method can detect clusters around spheres of any dimension less than 12 because they are well expressed as elements of conformal geometric algebra. Together, the results of this experiment verifies that the clustering results using the proposed method are better than those using these conventional methods.

IV. CONCLUSION

This paper proposed a new approach of approximating geometric data, including that based on hyper-spheres or hyper-plane using CGA. The proposed method used a Lagrange function and eigen decomposition to find eigen conformal vectors for constructing the approximation. In m-dimensional space, the proposed method could find m solutions of hyper-spheres (-planes) and m eigen-values. The method expresses various data distributions, such as those based on hyper-spheres (-planes), circles (lines), or arcs, by using probability density functions based on the Gaussian function for the distance defined by CGA. Then, this paper proposed a new clustering method using the proposed approximation. The results of the first experiment showed that the proposed method was able to express exactly geometric data based on circles, lines, or arcs. The results of the second experiment showed that the kernel alignment based on the clustering results using the proposed method was better than those of clustering using conventional methods.

REFERENCES

[1] J. B. MacQueen, Some Methods for classification and Analysis of Multivariate Observations, Proceedings of 5th Berkeley Symposium on Mathematical Statistics and Probability, vol. 1, pp. 281–297, 1967.

[2] J. C. Bezdek, C. Coray, R. Gunderson and J. Watson, Detection and characterization of cluster substructure I. Linear structure fuzzy c-lines, SIAM Jour. of Appl. Math., vol.40, no.2, pp. 339-357, 1981.

[3] R. Krishnapuram, O. Nasraoui and H. Frigui, The fuzzy c spherical shells algorithms: a new approach, IEEE Trans. on Neural Networks, vol.3, no.5, pp. 663–671, 1992.
[4] C. Doran and A. Lasenby, Geometric algebra for physicists, Cambridge University Press, 2003.
[5] D. Hestenes, New foundations for classical mechanics, Dordrecht, 1986.
[6] L. Dorst, D. Fontijne, and S. Mann, Geometric Algebra for Computer Science: An Object-oriented Approach to Geometry (Morgan Kaufmann Series in Computer Graphics), 2007.
[7] I. Sekita, T. Kurita, and N. Otsu, Complex Autoregressive Model for Shape Recognition, IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. 14, No. 4, 1992.
[8] A. Hirose, Complex-Valued Neural Networks: Theories and Applications, Series on Innovative Intelligence, Vol. 5, 2006.
[9] N. Matsui, T. Isokawa, H. Kusamichi, F. Peper, and H. Nishimura, Quaternion neural network with geometrical operators, Journal of Intelligent and Fuzzy Systems, Volume 15, Numbers 3–4, pp. 149–164, 2004.
[10] S. Buchholz and N. Le Bihan, Optimal separation of polarized signals by quaternionic neural networks, 14th European Signal Processing Conference, EUSIPCO 2006, September 4–8, Florence, Italy, 2006.
[11] T. Nitta, An Extension of the Back-Propagation Algorithm to Complex Numbers, Neural Networks, Volume 10, Number 8, pp. 1391–1415(25), November 1997.
[12] D. Hildenbrand and E. Hitzer, Analysis of point clouds using conformal geometric algebra, 3rd International conference on computer graphics theory and applications, Funchal, Madeira, Portugal, 2008.
[13] E. Hitzer, Quaternion Fourier Transform on Quaternion Fields and Generalizations, Advances in Applied Clifford Algebras, 17(3), pp. 497–517 (2007).
[14] G. Sommer, Geometric Computing with Clifford Algebras, Springer, 2001.
[15] M. T. Pham, K. Tachibana, E. M. S. Hitzer, T. Yoshikawa, and T. Furuhashi, Classification and Clustering of Spatial Patterns with Geometric Algebra, AGACSE 2008 Leipzig, 2008.
[16] D. Hestenes and G. Sobczyk. Clifford Algebra to Geometric Calculus: A unified language for mathematics and physics, Reidel, 1984.
[17] N. Cristianini, J. Kandola, A. Elisseeff, and J. Shawe-Taylor, On kernel target alignment. Journal of Machine Learning Research, 2002.