Dimensionality Reduction Method for 3D-Handwritten Characters Based on Oriented Bounding Boxes

YU ZHANG1, JINGWEI LU1, KUI WANG1, JUFENG ZHAO1, GUANGMANG CUI1, AND XIUMIN GAO2

1School of Electronics and Information, Hangzhou Dianzi University, Hangzhou 310018, China
2School of Optical-Electrical and Computer Engineering, University of Shanghai for Science and Technology, Shanghai 200093, China

Corresponding author: Xiumin Gao (gxm@usst.edu.cn)

This work was supported by the National Natural Science Foundation of China under Grant 61372156.

ABSTRACT Dimensionality reduction of 3D-handwritten characters can be problematic because of random mirror rotations and angle rotations which appear after using the traditional algorithms. In order to overcome the above drawbacks of the traditional methods, this study proposes a new algorithm for dimensionality reduction of 3D-handwritten characters based on oriented bounding boxes. First, we get a 3D discrete point set \( T \) and generate a 3D trajectory. Then, we apply an oriented bounding box model and determine the projection surface. Next, we perform three coordinate transformations, including (1) pre-transforming the 3D discrete point set \( T \) into the projection point set \( T_1 \), (2) converting \( T_1 \) to a two-dimensional point set \( T_2 \) which has solved the problem of mirror rotation, and (3) converting \( T_2 \) to a dimensionally reduced point set \( T_3 \) which has solved the problem of angle rotation. Finally, we obtain a dimensionally reduced image without mirror and angle rotations. The experimental results confirm that the proposed method can not only obtain a better visual dimensionally reduced image, but also has a higher recognition rate than the conventional ones.

INDEX TERMS 3D-handwritten character recognition, dimension reduction, oriented bounding box.

I. INTRODUCTION

3D-Handwritten character recognition is one of important method to realize non-contact human-computer interaction. Leap motion can provide 3D point cloud to track 3D finger movements with 0.01 millimeter accuracy. Based on Leap motion, many algorithms of 3D-Handwritten character recognition have been designed [1]–[3], which proceed the 3D-Handwritten character recognition directly on the 3D track. However, high dimensionality has brought great pressure to the transmission and storage [4]. Moreover, three are still mature algorithms of 2D-Handwritten character recognition. As a result, another way to recognize the 3D-Handwritten character is transforming 3D-Handwritten character into 2D character by dimensionality reduction method firstly, and then 2D character containing the equivalent information of 3D-Handwritten character is recognized by the 2D character recognition method to finish the recognition of 3D-Handwritten character.

Dimensionality reduction is an important process for retaining as much high-dimensional data as possible in a low-dimensional space. The existing dimensionality reduction algorithms can be linear or nonlinear. Traditional linear dimensionality reduction algorithms include Principal Component Analysis [5] (PCA) which is to select the low-dimensional space with the largest variance as the projection space and Linear Discriminant Analysis [6] (LDA) which is to maximize the distance between classes and minimize the distance within the class. The dimensionality reduction algorithms for nonlinear data can be divided into those retaining global features and retaining local features. Among them, the algorithms that preserve local features include Locally Linear Embedding [7] (LLE), which is an unsupervised learning algorithm that computes low-dimensional, neighborhood-preserving embeddings of high-dimensional inputs. Laplacian Eigenmaps [8] (LE), which is that the points that are related are as close as possible in the space after dimension reduction and it can reflect the internal manifold structure of the data. Local Tangent Space Alignment [9] (LTSA), which uses the local tangent space to represent...
the local geometry of the basic manifold structure, Hessian Eigenmaps [10] (HE), which adapts the weights in LLE to minimize the Hessian operator. And Locality Preserving Projections [11] (LPP), which arises by solving a variational problem that optimally preserves the neighborhood structure of the data set.

From the set of nonlinear algorithms that retain the global features, several algorithms use kernel technology such as Kernel Principal Component Analysis [12] and Kernel Fisher Discriminant Analysis [13], which can be seen as nonlinear versions of classic algorithms such as Principal Component Analysis (PCA) and Linear Discriminant Analysis (LDA). Another type of dimensionality reduction algorithms is based on distance retention and includes Multiple Dimensional Scaling [14] (MDS) which keeps the distance between the original spatial samples and the distance in the low-dimensional space as much as possible, Isometric Mapping [15] (Isomap) which is a non-iterative globally optimized dimensionality reduction algorithm that requires the geodesic distance to remain constant before and after dimensional reduction, and Diffusion Maps [16] which is dimensionally reduced by diffusion processes to remove redundant information.

However, applying the existing dimensionality reduction algorithms to process 3D-handwritten characters has the disadvantage that the 2D-characters may be subject to mirror rotation and angle rotation. Moreover, this problem will affect the recognition rate for 3D-handwritten characters. Figure 1 shows the images with mirror rotation (b) and angle rotation (c), compared to the original image (a).

In order to solve the problems of random appearance of mirror rotation and angle rotation, which are caused by dimensionality reduction of 3D handwritten characters, we propose a new dimensionality reduction algorithm based on oriented bounding boxes.

In this study, we acquire the 3D coordinates of handwritten characters by recording the 3D trajectory of the moving finger in the projection region of Leap Motion (projection region is a inverted pyramid region centered on Leap Motion, with an effective range from 26 nm to 600 nm above the device), define the coordinates of 3D-handwritten character as a 3D discrete point set \( T \), and generate a 3D track of handwritten characters according to \( T \). Then, an oriented bounding box that surrounds the 3D-handwritten character’s discrete points is determined. Next, the projection surface is obtained by ascertaining the normal vector of the largest area in the oriented bounding box and the geometric center of the 3D discrete point. Next, we pre-transform the 3D discrete point set \( T \) into a projection point set \( T_1 \) and establish a 2D Cartesian coordinate system \( X'O'Y' \). Following this, by converting the projection point set \( T_1 \) to a two-dimensional point set \( T_2 \) and converting the 2D point set \( T_2 \) to a dimensionally reduced point set \( T_3 \), we can solve the problem of mirror and angular rotations in dimensionally reduced images. Finally, a reduced-dimension image in a fixed direction is the output.

The remainder of the paper is organized as follows. Section 2 presents the proposed algorithm. The experimental results and a performance comparison with other state-of-the-art parameter estimation approaches are reported in Section 3. Section 4 concludes our study.

II. PROPOSED METHOD

The flowchart of the proposed method is shown in Figure 2. It is comprised of the following seven steps.

The proposed method is mainly aimed to solve the problem of random mirror and angle rotation of 2D-handwritten characters obtained by traditional dimensionality reduction methods. In our case, converting the projection point set \( T_1 \) to a 2D point set \( T_2 \) solves the problem of mirror rotation, and converting the 2D point set \( T_2 \) to the dimensionality reduction point set \( T_3 \) solves the problem of angular rotation.

Step ① (Get 3D Discrete Point Set T and Generate 3D Trajectory): The proposed algorithm is based on the 3D coordinate system XYZ built into the Leap Motion somatosensory controller, which is defined as the initial 3D coordinate system of handwritten characters.

The characters are written by moving the index finger within the detection region of the Leap Motion. The 3D point set obtained by moving the index finger is represented by the set \( T, T = \{T_i(x_i, y_i, z_i) \}, i=1,2,\ldots,n \). \( T_i \) is the spatial coordinate of the i-th discrete point. n is the number of discrete points obtained by Leap Motion. Wherein \( x_i, y_i, z_i \)
The average value can be calculated as (1), \( n \) is the number of input discrete points.

\[
\begin{align*}
  x_c &= \frac{\sum_{i=1}^{n} x_i}{n} \\
  y_c &= \frac{\sum_{i=1}^{n} y_i}{n} \\
  z_c &= \frac{\sum_{i=1}^{n} z_i}{n}
\end{align*}
\]

\( (1) \)

respectively represent the coordinate values of the X-axis, the Y-axis, and the Z-axis of the discrete point in the initial 3D coordinate system \( XYZ \). Then, 3D discrete points are sequentially connected to obtain a 3D trajectory of handwritten characters.

**Setp 2** (Establish an Oriented Bounding Box Model): After obtaining the 3D discrete points of the handwritten characters, we can get an oriented bounding box surrounding all 3D discrete point sets. The parameters of the oriented bounding box are given below.

**A. DIRECTION OF THE ORIENTED BOUNDING BOX**

Since 3D discrete point set \( T \) can be obtained, so the mean \( T_c \) \((x_c, y_c, z_c)\) of all discrete points can be calculated, wherein \( x_c \), \( y_c \), \( z_c \) respectively represent the mean coordinate values of the X-axis, the Y-axis, and the Z-axis of the center point of the discrete point set \( T \) in the initial 3D coordinate system \( XYZ \). The average value can be calculated as (1), \( n \) is the number of input discrete points.

\[
\begin{align*}
  x_c &= \frac{\sum_{i=1}^{n} x_i}{n} \\
  y_c &= \frac{\sum_{i=1}^{n} y_i}{n} \\
  z_c &= \frac{\sum_{i=1}^{n} z_i}{n}
\end{align*}
\]

\( (1) \)

The covariance matrix \( C \) can be calculated as (2), as shown at the bottom of the next page. The discrete points need to be normalized of data, that is \((x_i - x_c, y_i - y_c, z_i - z_c)\).

In (2), \( \text{cov}(x_i - x_c, y_i - y_c) = E[(x_i - x_c)(y_i - y_c)] - E[x_i - x_c]E[y_i - y_c] \).

Since the covariance matrix \( C \) is a \( 3 \times 3 \) symmetric matrix, there are \( \lambda \) and non-zero vectors \( \alpha \) to satisfy (3). \( \lambda \) is feature value in \( C \), and \( \alpha \) is feature vector of \( C \) corresponding to the feature value. The eigenvalues \( \lambda = (\lambda_1, \lambda_2, \lambda_3) \) and the corresponding eigenvectors \( \alpha = (\alpha_1, \alpha_2, \alpha_3) \) can be calculated as (3), wherein each feature vector is \( 3 \times 1 \) vector.

\[
C\alpha = \lambda \alpha
\]

\( (3) \)

Because the eigenvectors of the symmetric matrix \( C \) are orthogonal, the three eigenvectors \((\alpha_1, \alpha_2, \alpha_3)\) only need to be normalized to obtain the base vector \((v_1, v_2, v_3)\) as (4).

\[
v_j = \frac{\alpha_j}{\|\alpha_j\|} \quad (j = 1, 2, 3)
\]

\( (4) \)

The three-axis direction of the oriented bounding box is defined by the base vector \((v_1, v_2, v_3)\) and a \( 3 \times 3 \) matrix consisting of \((v_1, v_2, v_3)\) as the rotation matrix \( R \), which is the rotation matrix \( R \) of the oriented bounding box.

**B. VERTICES OF THE ORIENTED BOUNDING BOX**

The maximum and minimum values of the X-axis, Y-axis, and Z-axis can be determined based on the maximum and minimum values of all discrete points \( T \) on each coordinate.
axis of the initial coordinate system, that is, $x_{\text{max}}, x_{\text{min}}, y_{\text{max}}, y_{\text{min}}, z_{\text{max}}, z_{\text{min}}$.

The following shows eight points defined according to the above six coordinate values: $A_1(x_{\text{max}}, y_{\text{max}}, z_{\text{min}}), B_1(x_{\text{min}}, y_{\text{max}}, z_{\text{min}}), C_1(x_{\text{min}}, y_{\text{min}}, z_{\text{max}}), D_1(x_{\text{max}}, y_{\text{min}}, z_{\text{max}}), E_1(x_{\text{max}}, y_{\text{min}}, z_{\text{min}}), F_1(x_{\text{min}}, y_{\text{min}}, z_{\text{min}}), G_1(x_{\text{min}}, y_{\text{min}}, z_{\text{max}}),$ and $H_1(x_{\text{max}}, y_{\text{min}}, z_{\text{max}})$.

A bounding box composed of the above eight points can surround all discrete points of 3D-handwritten characters, but this box cannot represent the direction of the characters. To obtain an oriented bounding box, it is necessary to multiply the eight points $A_1, B_1, C_1, D_1, E_1, F_1, G_1$ and $H_1$ by the rotation matrix $R$ to obtain the direction vectors $A, B, C, D, E, F, G, H$.

Let us take point $A$ as an example, where $A_2$ is a $1 \times 3$ matrix, $R$ is a $3 \times 3$ matrix, and $A_1$ is multiplied by $R$ to obtain a $3 \times 3$ matrix, that is, the 3D coordinates of the vertex $A$ in the oriented bounding box. The 3D coordinates of $B, C, D, E, F, G,$ and $H$ can be obtained similarly in (5).

$$
\begin{align*}
A &= A_1 \ast R = (x_{\text{max}}, y_{\text{max}}, z_{\text{min}}) \ast (v_1, v_2, v_3) \\
B &= B_1 \ast R = (x_{\text{min}}, y_{\text{max}}, z_{\text{min}}) \ast (v_1, v_2, v_3) \\
C &= C_1 \ast R = (x_{\text{min}}, y_{\text{min}}, z_{\text{max}}) \ast (v_1, v_2, v_1) \\
D &= D_1 \ast R = (x_{\text{max}}, y_{\text{min}}, z_{\text{max}}) \ast (v_1, v_2, v_3) \\
E &= E_1 \ast R = (x_{\text{max}}, y_{\text{min}}, z_{\text{min}}) \ast (v_1, v_2, v_3) \\
F &= F_1 \ast R = (x_{\text{min}}, y_{\text{min}}, z_{\text{min}}) \ast (v_1, v_2, v_3) \\
G &= G_1 \ast R = (x_{\text{min}}, y_{\text{min}}, z_{\text{max}}) \ast (v_1, v_2, v_3) \\
H &= H_1 \ast R = (x_{\text{max}}, y_{\text{min}}, z_{\text{max}}) \ast (v_1, v_2, v_3)
\end{align*}
$$

(5)

C. SIDE OF THE ORIENTED BOUNDING BOX

We project the discrete point set $T$ to the rotation matrix $R$ ($v_1, v_2, v_3$), and then we find the projection interval of all the discrete points to the three directions, that is, the length of three projection intervals are the three sides of oriented bounding box. The maximum and minimum values of the $i$-th discrete point $T_i$ projected along the respective directions are calculated as (6) and (7).

$$
\begin{align*}
&u_i = \max(T_i \cdot v_1), \quad i = 1, 2, \ldots, n \\
&u_2 = \max(T_i \cdot v_2) \\
&u_3 = \max(T_i \cdot v_3) \\
&l_1 = \min(T_i \cdot v_1), \quad i = 1, 2, \ldots, n \\
&l_2 = \min(T_i \cdot v_2) \\
&l_3 = \min(T_i \cdot v_3)
\end{align*}
$$

In (6), $u_1, u_2$ and $u_3$ are the maximum values of point $T_i$ along the direction of $v_1$, $v_2$ and $v_3$. In (7), $l_1, l_2$ and $l_3$ are the minimum values of point $T_i$ along the direction of $v_1$, $v_2$ and $v_3$. Then, three side length of the oriented bounding box is $u_1 - l_1, u_2 - l_2$ and $u_3 - l_3$.

The center point of the oriented bounding box is denoted as $c$, which is shown in (8).

$$
c = \frac{1}{2} (l_1 + u_1) v_1 + \frac{1}{2} (l_2 + u_2) v_2 + \frac{1}{2} (l_3 + u_3) v_3
$$

(8)

FIGURE 3. Definition of the points $M, N, P, Q.$

FIGURE 4. $T_i$ is converted to $T_i^\prime$.

Step 3 (Determine the Projection Surface $MNPQ$): The projection surface $MNPQ$ is determined based on the parameters of the oriented bounding box. The normal vector of the projection surface $MNPQ$ is the same as the normal vector of the largest area in the oriented bounding box. The projection surface $MNPQ$ passes through the center point $T_c$ of the discrete point set $T$.

In Figure 3, $MNPQ$ is defined. The connection order of $MNPQ$ is the same as the connection order of $ABCD$.

Step 6 (The 3D Discrete Point Set $T$ is Pre-Transformed Into the Projection Point Set $T_1$): The set of projection points is represented by a set $T_1$, $T_1 = \{T_1^\prime (x_i^\prime, y_i^\prime, z_i^\prime) \}$, $i = 1, 2, \ldots, n$. $T_1^\prime$ is the coordinates of the $i$-th projection point. Wherein $x_i^\prime, y_i^\prime, z_i^\prime$ respectively represent the coordinate values of the X-axis, Y-axis, and Z-axis of a certain projection point $T_i^\prime$ in the initial 3D coordinate system $XYZ$.

The process of this step is to project the set of discrete points $T$ onto the projection surface $MNPQ$ to obtain the projection point set $T_1$, that is, Figure 4 shows the projection point $T_i(x_i, y_i, z_i), i = 1, 2, \ldots, n$ is converted to the i-th projection point $T_i^\prime (x_i^\prime, y_i^\prime, z_i), i = 1, 2, \ldots, n$.

The process of pre-conversion is derived as follows. Assume that the normal vector of the projection plane $MNPQ$
is $\vec{n}(a, b, c)$, and the projection plane passes $T_c(x_c, y_c, z_c)$. Projection point $T'_c$ can be calculated according to projecting the discrete point $T_i(x_i, y_i, z_i)$ onto $MNPQ$, as shown from (9) to (11).

The definition of $MNPQ$ is as (9):

$$a(x - x_c) + b(y - y_c) + c(z - z_c) = 0 \quad (9)$$

Since $T'_c$ is over the $MNPQ$, there is the following (10):

$$a(x'_i - x_c) + b(y'_i - y_c) + c(z'_i - z_c) = 0 \quad (10)$$

Because of $T_i T'_c \parallel \vec{n}$,

$$\frac{x'_i - x_i}{a} = \frac{y'_i - y_i}{b} = \frac{z'_i - z_i}{c} \quad (11)$$

Substituting (11) into (9):

$$\begin{cases}
    x'_i = ax_c + \frac{b^2}{a}x_i - by_i + by_c + \frac{c^2}{a}x_i - cz_i - cz_c \\
    y'_i = \frac{b}{a}(x'_i - x_i) + y_i \\
    z'_i = \frac{c}{a}(x'_i - x_i) + z_i
\end{cases} \quad (12)$$

In (12), the $T'_c$ is obtained after a discrete point $T_i$ is projected onto the projection surface $MNPQ$. According to the above method, we can calculate the projection point sets $T_1$ corresponding to the discrete point set $T$.

As shown in Figure 5(a), the 3D trajectory of the handwritten characters and the oriented bounding box is obtained. In Figure 5(b), the projection point set $T_1$ obtained by projecting 3D discrete point set $T$ to $MNPQ$ is determined.

**Step 5 (Establish 2D Cartesian Coordinate System $X'O'Y'$):** Establishing a 2D Cartesian coordinate system is divided into two steps. First, we project points $M$, $N$, $P$ onto the XOY plane in the initial 3D coordinate system to get the point $M_1$, $N_1$, $P_1$, and it is judged whether the points $M_1$, $N_1$, $P_1$ are clockwise or counterclockwise. Then, we establish a 2D Cartesian coordinate system based on two different situations.

In the initial 3D coordinate system $XYZ$, we define $M$ $(a_1, b_1, c_1)$, $N$ $(a_2, b_2, c_2)$ and $P$ $(a_3, b_3, c_3)$, so that the coordinates of $M_1$, $N_1$, and $P_1$ in $XYZ$ are $M_1(a_1, b_1, 0)$, $N_1(a_2, b_2, 0)$, and $P_1(a_3, b_3, 0)$, respectively. The direction feature $d$ can be calculated as (13).

$$\begin{vmatrix}
    a_1 & a_2 & a_3 \\
    b_1 & b_2 & b_3 \\
    1 & 1 & 1
\end{vmatrix} = (a_1 - a_3) \cdot (b_2 - b_3) - (b_1 - b_3) \cdot (a_2 - a_3) \quad (13)$$

If $d < 0$, as shown in Figure 6(a), $M_1$, $N_1$, and $P_1$ are counterclockwise. Then, we establish a 2D Cartesian coordinate system $X'O'Y'$. The positive direction of the $Y'$ axis is the direction of the point $M$ to the point $Q$, and the positive direction of the $X'$ axis is the direction of the $M$ to $N$.

If $d \geq 0$, as shown in Figure 6(b), $M_1$, $N_1$, and $P_1$ are clockwise. Then, we establish a 2D Cartesian coordinate system $X'O'Y'$. The positive direction of the $Y'$ axis is the direction of the point $M$ to the point $N$, and the positive direction of the $X'$ axis is the direction of $M$ to $Q$.

**Step 6 (The Projection Point Set $T_i$ is Converted to 2D Point Set $T_2$):** The 2D point set $T_2$ is defined as follows: $T_2 = \{T''_i(x''_i, y''_i)\}$, $i = 1, 2, \ldots, n$. Here, $T''_i$ is the i-th 2D point. Wherein $x''_i$, $y''_i$ respectively represent the coordinate values of the $X'$-axis and $Y'$-axis of a certain projection point $T''_i(x'_i, y'_i, z'_i)$ in the coordinate system $X'O'Y'$. In this step, the projection point set $T_1$ is converted to a 2D point set $T_2$, that is, the projection point $T'_i(x'_i, y'_i, z'_i)$ is converted to the i-th 2D point $T''_i(x''_i, y''_i)$, where $i = 1, 2, \ldots, n$.

$T''_i(x''_i, y''_i)$ is defined according to whether $M_1$, $N_1$, and $P_1$ are clockwise or counterclockwise. If $M_1$, $N_1$, and $P_1$ are connected counterclockwise, we establish a 2D coordinate system $X'O'Y'$ as shown in Figure 7, $p_i$ and $q_i$ mean the corresponding lengths of 2D coordinates.
If $M_T$ and $N_T$ are connected clockwise, the i-th 2D point set $T_i''$ is defined as (16).

\[
\begin{align*}
    x_i'' &= MT_i \cdot \mu_{MQ}, \quad i = 1, 2, \ldots, n \\
    y_i'' &= MT_i \cdot \mu_{MN},
\end{align*}
\]  

(16)

The coordinate value of the i-th 2D $T_i''$ is as shown in (14).

\[
\begin{align*}
    x_i'' &= MT_i \cdot \mu_{MQ}, \quad i = 1, 2, \ldots, n \\
    y_i'' &= MT_i \cdot \mu_{MN},
\end{align*}
\]  

(14)

Among them, $MT_i$ refers to the vector where the point $M$ points to the point $T_i'$. The unit vector of $MQ$ and $MN$ is denoted as $\mu_{MQ}$ and $\mu_{MN}$, which is shown as (15).

\[
\begin{align*}
    \mu_{MQ} &= \frac{MQ}{||MQ||} \\
    \mu_{MN} &= \frac{MN}{||MN||}
\end{align*}
\]  

(15)

Step 6 (The 2D Point Set $T_2$ is Converted to Dimensionally Reduced Point Set $T_3$): The 2D point set is defined by set $T_3$, $T_3 = \{(x''_i, y''_i)\}, \ i = 1, 2, \ldots, n$. $T_i''$ is the i-th dimensionality reduction point. Wherein $x''_i$, $y''_i$ respectively represent the coordinate values of the $X''$-axis and the $Y''$-axis in the coordinate system $X''O''Y''$ after a certain 2D point $T_i''(x''_i, y''_i)$ is rotated $\theta$ counterclockwise around the $T_i''$.

The 2D point set $T_2$ is converted to the dimensionality reduction point set $T_3$, that is, the 2D point $T_i''(x''_i, y''_i)$ is converted to the dimensionality reduction point $T_i'''(x'''_i, y'''_i)$. Here, $T_i'''(x'''_i, y'''_i)$ is defined as (17).

\[
\begin{align*}
    x'''_i &= (x''_i - \sum_{i=1}^{n} x''_i) \cos \theta - (y''_i - \sum_{i=1}^{n} y''_i) \sin \theta + \frac{\sum_{i=1}^{n} x''_i}{n} \\
    y'''_i &= (y''_i - \sum_{i=1}^{n} y''_i) \sin \theta - (y''_i - \sum_{i=1}^{n} y''_i) \cos \theta + \frac{\sum_{i=1}^{n} y''_i}{n}
\end{align*}
\]  

(17)

Wherein the coordinates of the rotation center point $T''_c$ of the dimension reduction point set $T_3$ is defined as (18).

\[
T''_c = \left(\frac{\sum_{i=1}^{n} x''_i}{n}, \frac{\sum_{i=1}^{n} y''_i}{n}\right), \quad i = 1, 2, \ldots, n
\]  

(18)

$\theta$ is the angle at which the 2D point set $T_2$ is rotated counterclockwise with a center in $T''_c$ from the positive direction of the $X'$-axis to the positive direction of the $Y'$-axis. A new 2D Cartesian coordinate system $X''O''Y''$ is established by rotating $\theta$ in the coordinate axis $X'O'Y'$, wherein the coordinate value in $X''O''Y''$ corresponds to $T_3$. All dimensionality reduction results are output in the coordinate system $X''O''Y''$.

The angle $\theta$ is determined by comparing the different coordinate values of $M'$ and $P'$ in the 2D coordinate system $X'O'Y'$. In the initial 3D coordinate system $XYZ$, $M(a_1, b_1, c_1)$ and $P(a_3, b_3, c_3)$ are defined. We assume that the values of...
FIGURE 8. 3D space handwritten character trajectory, 3D OBB and five dimensionality reduction images. (a) A 3D trajectory of the 26 Arabic characters and their 3D OBB; (b) Visualization results with PCA; (c) Visualization results with KPCA; (d) Visualization results with MDS; (e) Visualization results with Isomap; (f) Visualization results with the proposed method.
III. EXPERIMENTAL RESULTS

There are two ways to evaluate the dimensionality reduction of 3D-handwritten characters. The first approach is to consider visualizations of the results of dimensionality reduction. The second approach is to calculate the recognition rate of the image after dimensionality reduction.
FIGURE 8. (Continued.) 3D space handwritten character trajectory, 3D OBB and five dimensionality reduction images. (a) A 3D trajectory of the 26 Arabic characters and their 3D OBB; (b) Visualization results with PCA; (c) Visualization results with KPCA; (d) Visualization results with MDS; (e) Visualization results with Isomap; (f) Visualization results with the proposed method.
Therefore, the experimental process in this paper is divided into two parts: one for visual comparison and another for character recognition.

In order to verify the visual processing effect of the proposed method, we compared it with the dimensionality reduction results of four other algorithms: PCA, KPCA, MDS, and Isomap, which are mainstream and effective algorithms.

In order to compare the effectiveness of dimensionality reduction methods fairly and objectively, we select three most
FIGURE 8. (Continued.) 3D space handwritten character trajectory, 3D OBB and five dimensionality reduction images. (a) A 3D trajectory of the 26 Arabic characters and their 3D OBB; (b) Visualization results with PCA; (c) Visualization results with KPCA; (d) Visualization results with MDS; (e) Visualization results with Isomap; (f) Visualization results with the proposed method.
common and simple recognition algorithms: k-nearest neighbor (KNN) [17], Support Vector Machine (SVM) [18], [19], and Naïve Bayes [20], [21]. We compare the recognition rate produced by these algorithms for the reduced-dimensional images obtained by five previously described dimensionality reduction algorithms. In order to highlight and compare the effectiveness of the dimensionality reduction method, the three most common and simplest identification methods are selected.

A. COMPARISON OF VISUALIZATION RESULTS OF DIFFERENT DIMENSION REDUCTION METHODS

The experimental platform is Sublime Text 2, and the experimental environment is Windows 10 operating system.
FIGURE 8. (Continued.) 3D space handwritten character trajectory, 3D OBB and five dimensionality reduction images. (a) A 3D trajectory of the 26 Arabic characters and their 3D OBB; (b) Visualization results with PCA; (c) Visualization results with KPCA; (d) Visualization results with MDS; (e) Visualization results with Isomap; (f) Visualization results with the proposed method.
The experimenter wrote 26 3D lowercase letters by the tip of the index finger with Leap Motion.

As shown in Figure 8, the red curve represents the character trajectory for the letters a-z. The acquired 3D-handwritten character points are adaptively bounded by an oriented bounding box. The dimensionality reduction algorithms based on PCA, KPCA, MDS, Isomap and the proposed method, respectively, obtain the following visualization results, as shown in Figure 8.

Through the experimental results, it can be clearly found that the 2D characters obtained by the PCA, KPCA, MDS and Isomap will randomly produce mirror and angle rotations.
FIGURE 8. (Continued.) 3D space handwritten character trajectory, 3D OBB and five dimensionality reduction images. (a) A 3D trajectory of the 26 Arabic characters and their 3D OBB; (b) Visualization results with PCA; (c) Visualization results with KPCA; (d) Visualization results with MDS; (e) Visualization results with Isomap; (f) Visualization results with the proposed method.
FIGURE 8. (Continued.) 3D space handwritten character trajectory, 3D OBB and five dimensionality reduction images. (a) A 3D trajectory of the 26 Arabic characters and their 3D OBB; (b) Visualization results with PCA; (c) Visualization results with KPCA; (d) Visualization results with MDS; (e) Visualization results with Isomap; (f) Visualization results with the proposed method.

Since the direction of the dimensionality reduction image is not fixed, this will lead to a lower recognition rate. On the contrary, the proposed method overcomes the problem of mirror and angle rotations in the reduced-dimensional image by converting the projection point set \( T_1 \) to a 2D point set \( T_2 \), and converting the 2D point set \( T_2 \) to a dimensionality reduction point set \( T_3 \), therefore the visualization looks better than in other methods.

B. COMPARISON OF THE RECOGNITION RATES OF DIFFERENT DIMENSION REDUCTION METHODS

In this paper, the 3D-handwritten character trajectory is processed in a lowercase letters database. The data from the lowercase database is collected from 10 experimenters. Each experimenter recorded 26 lowercase letters 40 times. There are 10,400 samples in the database, 10-fold cross validation is used for training and testing. Then we calculate the
FIGURE 8. (Continued.) 3D space handwritten character trajectory, 3D OBB and five dimensionality reduction images. (a) A 3D trajectory of the 26 Arabic characters and their 3D OBB; (b) Visualization results with PCA; (c) Visualization results with KPCA; (d) Visualization results with MDS; (e) Visualization results with Isomap; (f) Visualization results with the proposed method.
The recognition rate of every single 3D-handwritten character in the lowercase database, and the average recognition rate of all characters.

The recognition rate of a single character can be calculated as (19).

$$R = \frac{n_1}{N_1}$$

Among them, $R$ is the recognition rate of a single character. $n_1$ is the number of single 3D-handwritten characters to be identified correctly. $N_1$ is the total number of single 3D-handwritten characters.

For all samples in the lowercase database, single 3D-handwritten character recognition rates for five kinds of dimensionality reduction methods under three recognition
methods are shown in Tables 1. In Table 1, under the different recognition methods, the value with the highest recognition rate is shown in bold.

As can be seen from Table 1, under the KNN, SVM and NB identification method, the single 3D-handwritten character recognition rate of the dimensionality reduction image obtained by the proposed method is higher than those obtained by PCA, KPCA, MDS and Isomap. Because the proposed method solves the problem of mirror and angular rotation of 2D characters after dimensionality reduction, the recognition rate is improved. Therefore, we obtained the coincident results to the visualization experiment section.

In addition, as the latest method of recognize the characters from the 3D tracks directly, the recognition rate of the method in literature [1] is also compared with that of the proposed method. From Table 1, you can see that under the SVM and KNN identification method, the proposed dimensionality reduction method can obtain better and comparable result, compared with the method in [1]. It is also can be seen from Table 1 that, beside the method of dimensionality reduction, the recognition rate is also dependent on the identification method of 2D characters.

For all samples in the 3D-handwriting lowercase character database, the average recognition rate of all characters processed by different dimensionality reduction algorithms are shown in Figure 9.

As can be seen from Figure 9, the average recognition rates of the proposed method under the KNN, SVM and NB are 89.9%, 91.5% and 86.9%, respectively. The similar conclusions as in Table 1 and Figure 8 can be obtained, showing that the average recognition rate of the proposed algorithm is better than for other algorithms. Similarly, the highest recognition rate is obtained because the problem of random mirror and angle rotations of the reduced-dimensional image is solved.
IV. CONCLUSION
In this paper, an algorithm for dimensionality reduction of 3D-handwritten characters based on an oriented bounding box was proposed. By converting the projection point set \( T_1 \) to a 2D point set \( T_2 \), and converting the 2D point set \( T_2 \) to a dimensionality reduction point set \( T_3 \), we solve the problem of random mirroring and angular rotation of traditional dimensionally reduced images. The experimental results indicate that the proposed algorithm can obtain 2D dimensionality reduction images in a fixed direction, and the proposed method improves the recognition rate by 32.6%, 31.4% and 32.4% under the three identification methods of SVM, KNN and NB.

REFERENCES
[1] P. Kumar, R. Saini, P. P. Roy, and D. P. Dogra, “3D text segmentation and recognition using leap motion,” Multimedia Tools Appl., vol. 76, no. 15, pp. 16491–16510, Aug. 2017.
[2] D. K. Charles, K. Pedlow, S. Mcondonough, K. Shek, and T. Charles, “An evaluation of the leap motion depth sensing camera for tracking hand and fingers motion in physical therapy,” in Proc. Interact. Technol. Games Conf., 2013.
[3] S. Vikram, L. Li, and S. Russell, “Handwriting and gestures in the air, recognizing on the fly,” in Proc. ACM Conf. Hum. Factors Comput. Syst. (CHI) Extended Abstr., 2013, pp. 1179–1184.
[4] H. Xu, H. Zhang, W. He, and L. Zhang, “Superpixel-based spatial-spectral dimension reduction for hyperspectral imagery classification,” Neurocomputing, vol. 360, pp. 138–150, Sep. 2019.
[5] I. T. Jolliffe and J. Cadima, “Principal component analysis: A review and recent developments,” Philos. Trans. Roy. Soc. A, vol. 374, no. 2065, Apr. 2016, Art. no. 20150202.
[6] Y. Zhu, C. Zhu, and X. Li, “Improved principal component analysis and linear regression classification for face recognition,” Signal Process., vol. 145, pp. 175–182, Apr. 2018.
[7] T. Cai and W. Liu, “A direct estimation approach to sparse linear discriminant analysis,” J. Amer. Stat. Assoc., vol. 106, no. 496, pp. 1566–1577, Dec. 2011.
[8] B. Jiang, C. Ding, and B. Luo, “Robust data representation using locally linear embedding guided PCA,” Neurocomputing, vol. 275, pp. 523–532, Jan. 2018.
[9] Z. Lai, W. K. Wong, Y. Xu, J. Yang, and D. Zhang, “Approximate orthogonal sparse embedding for dimensionality reduction,” IEEE Trans. Neural Netw. Learn. Syst., vol. 27, no. 4, pp. 723–735, Apr. 2016.
[10] J. Wang, X. Sun, and J. Du, “Local tangent space alignment via nuclear norm regularization for incomplete data,” Neurocomputing, vol. 273, pp. 141–151, Jan. 2018.
[11] Q. Ye and W. Zhi, “Discrete Hessian Eigenmaps method for dimensionality reduction,” J. Comput. Appl. Math., vol. 278, pp. 197–212, Apr. 2015.
[12] R. Wang, F. Nie, R. Hong, X. Chang, X. Yang, and W. Yu, “Fast and orthogonal locality preserving projections for dimensionality reduction,” IEEE Trans. Image Process., vol. 26, no. 10, pp. 5019–5030, Oct. 2017.
[13] X. Huang, A. Maier, J. Hornegger, and J. A. Suykens, “Indefinite kernels in least squares support vector machines and principal component analysis,” Appl. Comput. Harmon. Anal., vol. 43, no. 1, pp. 162–174, Jul. 2017.
[14] X.-Z. Liu and C.-G. Zhang, “Fisher discriminant analysis based on kernel cuboid for face recognition,” Soft Comput., vol. 20, no. 3, pp. 831–840, Mar. 2016.
[15] L. Lin and D. Fong, “Bayesian multidimensional scaling procedure with variable selection,” Comput. Statist. Data Anal., vol. 129, pp. 1–13, Jan. 2019.
[16] M. Salhov, A. Bermanis, G. Wolf, and A. Averbuch, “Approximately-isometric diffusion maps,” Appl. Comput. Harmon. Anal., vol. 38, no. 3, pp. 399–419, May 2015.
[17] À. Fernández, A. M. González, J. Díaz, and J. R. Dorrorsoro, “Diffusion maps for dimensionality reduction and visualization of meteorological data,” Neurocomputing, vol. 163, pp. 25–37, Sep. 2015.
[18] S. Zhang, X. Li, M. Zong, X. Zhu, and R. Wang, “Efficient kNN classification with different numbers of nearest neighbors,” IEEE Trans. Neural Netw. Learn. Syst., vol. 29, no. 5, pp. 1774–1785, May 2018.
[19] S. Zeng, C. Gao, X. Wang, L. Jiang, and D. Feng, “Multiple kernel–based discriminant analysis via support vectors for dimension reduction,” IEEE Access, vol. 7, pp. 35418–35430, 2019.
[20] P. Zhao, Q. Miao, J. Song, Y. Qi, R. Liu, and D. Ge, “Architectural style classification based on feature extraction module,” IEEE Access, vol. 6, pp. 52598–52606, 2018.
[21] G. Feng, J. Guo, B.-Y. Jing, and T. Sun, “Feature subset selection using naive Bayes for text classification,” Pattern Recognit. Lett., vol. 65, pp. 109–115, Nov. 2015.
[22] L. Jiang, C. Li, S. Wang, and L. Zhang, “Deep feature weighting for naive Bayes and its application to text classification,” Eng. Appl. Artif. Intell., vol. 52, pp. 26–39, Jun. 2016.

YU ZHANG received the M.S. degree in circuits and systems and the Ph.D. degree in microelectronics and solid state electronics from Tianjin University, Tianjin, China, in 2006 and 2009, respectively. Since July 2009, she has been with the School of Electronics and Information, Hangzhou Dianzi University, where she is currently a Full Professor. Her research interests include designing the on-chip signal processing algorithm of CMOS image sensors and VLSI realization, and 3D-handwriting character recognition.

JINGWEI LU received the M.S. degree from the School of Electronics and Information, Hangzhou Dianzi University, Hangzhou, China, in 2019. Her research interest includes designing the algorithm of 3D handwriting character recognition.

KUI WANG is currently pursuing the M.S. degree with the School of Electronics and Information, Hangzhou Dianzi University, Hangzhou, China. Her research interest includes designing the algorithm of 3D-handwriting character recognition.

JUFENG ZHAO received the B.S. degree from Zhejiang University, Hangzhou, China, in 2008, and the Ph.D. degree from the State Key Laboratory of Modern Optical Instrumentation, Zhejiang University, in 2013. Since 2013, he has been an Associate Professor with the School of Electronics and Information, Hangzhou Dianzi University, Hangzhou. His research interest includes a wide range of topics in the representation and analysis of visual images, including digital image assessment.

GUANGMANG CUI received the B.S. degree from Zhejiang University, Hangzhou, China, in 2011, and the Ph.D. degree in optical engineering from the State Key Laboratory of Modern Optical Instrumentation, Zhejiang University, in 2016. He is currently a Lecturer with the School of Electronics and Information, Hangzhou Dianzi University, Hangzhou, China. His research interests include optical imaging systems, image processing, and computational imaging.

XIUMIN GAO received the Ph.D. degree in optical engineering from the Shanghai Institute of Optical Precision Machinery, Chinese Academy of Sciences, in 2006. He is currently a Full Professor with the School of Optoelectronic Information and Computer Engineering, University of Shanghai for Science and Technology, Shanghai, China. His research interests include optical imaging systems, image processing, and light field regulation.

***