Feature Point Extraction using 3D Separability Filter for Finger Shape Recognition

Ryoma Yataka, Lincon Sales de Souza, and Kazuhiro Fukui
Graduate School of Systems and Information Engineering, University of Tsukuba
1-1-1 Tennodai, Tsukuba, Ibaraki 305-8573, Japan
Email: {yataka, lincons}@cvlab.cs.tsukuba.ac.jp, kfukui@cs.tsukuba.ac.jp

Abstract—In this paper, we propose a method for recognizing finger shapes based on the geometrical structure of 3D feature points such as fingertips and finger joints, which are extracted from a depth image. In the proposed method, the two sets of feature points to be compared are represented by different shape subspaces in a high-dimensional vector space. Then, classification is performed by measuring the canonical angles between the two corresponding shape subspaces as the similarity. To perform this method effectively, it is essential to extract the feature points stably and accurately from an input depth image. Extracting such feature points is a difficult problem because the finger shape and size varies largely across individuals. To address this issue, we introduce a 3D separability filter that can extract the 3D positions of the finger feature points stably and accurately from a depth image. This 3D separability filter is an extension of the 2D separability filter, which has been widely used to extract feature points such as pupils and nostrils in face image analysis. The proposed filter extracts points with high separability in a 3D depth image space, where the Fisher criterion of the voxel information of the inner and outer regions of a 3D filter mask at each point is used as the separability. We demonstrate the effectiveness of the proposed method by an evaluation experiment using finger shapes pointing at targets on a screen.

Keywords—3D separability filter, Shape subspace, Finger shape recognition.

I. INTRODUCTION

In this paper, we propose a method for recognizing a finger shape using the 3D geometrical structure of feature points as fingertips and finger joints. Finger shape recognition has been actively studied with the aim of improving applications such as human–computer interfaces and human–robot interaction [1]–[3]. Recently, many methods have been proposed for recognizing finger shapes from a depth image [4]–[6], because capturing 3D finger shape information has become easier with the emergence of various types of depth sensors.

The proposed method is based on the 3D geometrical structure of feature points of a finger, which are extracted from an input depth image. Two sets of feature points are represented by different shape subspaces in a high-dimensional vector space. An input set of feature points is then classified by measuring the canonical angles between the two corresponding shape subspaces as their similarity [7], [8]. As the concept of the shape subspace is simple and applicable, it has been used in various applications, for example, identification of persons based on the geometrical structure of micro facial feature points [7].

To perform classification based on the shape subspace representation effectively, it is essential to extract such feature points stably and accurately from an input depth image. However, extraction can be difficult because the finger shape and size varies largely across individuals and may contain extreme occlusions.

To address this issue, we introduce a 3D separability filter that can extract the 3D positions of the finger feature points from an input depth image. Fig. 1 shows the concept of this filter. This 3D separability filter is an extension of the 2D separability filter [9]–[11], which has been widely used to extract various feature points. In particular, circular feature points, such as pupils and nostrils for face image analysis, have been extracted. The proposed filter extracts multiple points with high separability in the 3D voxel space of a finger depth image, where the separability is defined as the Fisher criterion, namely, the class separability between the information of the inner and outer regions of a 3D filter mask in a 3D voxel space, as shown in Fig. 1. We demonstrate the effectiveness of the proposed method by an evaluation experiment using finger shapes pointing at targets on a screen.

The rest of the paper is organized as follows. In Sec. II, we explain the proposed 3D separability filter in detail. In Sec. III, we explain the proposed framework for recognizing finger shape. In Sec. IV, we conduct experiments using finger shapes pointing at targets on a screen and evaluate their results. Sec. V concludes the paper.
II. PROPOSED 3D SEPARABILITY FILTER

First, we describe feature point extraction using a 2D separability filter. Then, we extend the 2D separability filter to the 3D separability filter to deal with 3D voxel data.

A. 2D separability filter

The separability value indicates the class separability between information, such as the image intensities, colors, and textures, of two local regions. A rectangle or circular mask is used to calculate the separability [9]–[11]. Consider the case of using a circular mask that consists of two regions, \( R_1 \) and \( R_2 \), as shown in Fig. 2. In this case, the separability \( \eta \) can be calculated as follows:

\[
\eta = \frac{\sigma_b^2}{\sigma_T^2}, \quad (1)
\]

where \( \sigma_b^2 \) denotes the between-class variance between \( R_1 \) and \( R_2 \), and \( \sigma_T^2 \) denotes the total variance.

\[
\begin{align*}
\sigma_b^2 &= \frac{N_1}{N} (\overline{P_1} - \overline{P})^2 + \frac{N_2}{N} (\overline{P_2} - \overline{P})^2 \quad (2) \\
\sigma_T^2 &= \sum_{P_i \in (R1 \cup R2)} (P_i - \overline{P})^2 \\
&= \overline{P^2} - (\overline{P})^2 \quad (3)
\end{align*}
\]

Here, \( N \) denotes the total number of elements in \( R_1 \) and \( R_2 \), and \( N_1 \) and \( N_2 \) denote the numbers of elements in \( R_1 \) and \( R_2 \), respectively. Further, \( \overline{P_1} \) denotes the value of the \( i \)th element, and \( \overline{P_1} \) and \( \overline{P_2} \) denote the mean values of \( R_1 \) and \( R_2 \), respectively. \( \overline{P} \) denotes the mean value of all elements in \( R_1 \) and \( R_2 \), and \( \overline{P^2} \) denotes the mean square value of all elements in \( R_1 \) and \( R_2 \).

B. Extension to 3D Separability Filter

We extend the 2D separability filter to the 3D separability filter by using the depth information of a depth image. For this purpose, we consider the volume data obtained from a depth image as described below. Below, we describe the 3D separability filter and explain how it is applied to a depth image.

First, we convert a 2D depth image to a 3D array by converting each pixel into a vector with the same number of elements as the maximum depth value of the image, as shown in Fig. 3(a). We refer to the 3D array as the volume data. Assuming that the maximum value in an \((H \times W)\) depth image is \( D_{\text{max}} \), the generated volume data is an \( (H \times W \times D_{\text{max}}) \) array. We refer to the \( D_{\text{max}} \) direction as the depth axis. In

(a) We convert a 2D depth image into volume data by transforming each pixel in the image into a vector with the dimension set to that of the largest element of the depth value. In this figure, \( D_{\text{max}} \) is the maximum depth value in the image. Each vector’s element has a value of 0 or 1.

(b) Voxel filter. The separability between concentric regions \( R_1 \) and \( R_2 \) is calculated.

Next, we apply a 3D voxel filter with the 3D mask configuration shown in Fig. 3(b) to each element of the \((H \times W \times D_{\text{max}})\) volume data (Fig. 4). The 3D voxel filter consists of two concentric cubic regions \( R_1 \) and \( R_2 \). The half-side lengths of the inner cube and outer cube are \( r \) and \( 2r \), respectively, and the center of the filter corresponds to the target element in the volume data. This filtering operation is repeated for all elements of the volume data, so that the separability \( \eta \) between the neighborhoods \( R_1 \) and \( R_2 \) surrounding each element can be calculated by using Eq.(1). The output of this filter is voxel data of the same size as the original volume data, where each output element corresponds to the separability of each volume data element. In other words, we obtain new 3D volume data, separability volume data, in which each element consists of a separability value.

Finally, we sum the separability volume data along the depth axis (Fig. 4). With the above operation, we can generate a separability map as shown in Fig. 5. Note that this separability map is a 2D image showing the center of the feature points as the local peak of separability. To find feature points, we
extract local maximum points in the 2D separability map. The \((x, y, z)\) components corresponding to these local maximum points are used as the coordinates of the 3D feature points.

Filtering the volume data incurs high computational cost, because the separability needs to be calculated one element at a time. Therefore, we reduce the computational time by introducing a speed-up technique with integral images [11].

III. PROPOSED FINGER SHAPE RECOGNITION

The core concept in our finger shape recognition framework is to measure the similarity between two feature point sets. Below, we first describe the method for generating a shape subspace. Next, we describe how to calculate canonical angles from two shape subspaces and classify an input finger shape by using the canonical angles as the similarity.

A. Generation of Shape Subspace

A shape subspace is defined as a 3-dimensional subspace in a high-dimensional vector space. We generate a shape subspace from 3D feature points that are obtained from a depth image. Suppose that \(N\) number of 3D feature points are extracted from a given depth image. In this case, a shape subspace \(S\) is spanned by the three column vectors of an \(N \times 3\) matrix \(S\) that is defined as

\[
S = (s_1, s_2, \ldots, s_N)^\top = \begin{pmatrix} x_1 & y_1 & z_1 \\ x_2 & y_2 & z_2 \\ \vdots & \vdots & \vdots \\ x_N & y_N & z_N \end{pmatrix}
\]

where \(s_p = (x_p, y_p, z_p)^\top (1 \leq p \leq N)\) denotes the positional vector of the \(n\)-th 3D feature point. The column vectors of \(S\) are orthogonalized to span the orthonormal basis vectors of a three-dimensional shape subspace in \(\mathbb{R}^N\).

B. Definition of Similarity with Canonical Angles

The similarity of two shape subspaces is quantitatively evaluated using canonical angles [7]. Consider \(n_1\) and \(n_2\)-dimensional shape subspaces, where \(n_1 \leq n_2\). The canonical angles \(\theta_i (1, \ldots, n_1)\) are defined as

\[
\cos \theta_i = \max_{u \in S_1, v \in S_2} u^\top v = u_i^\top v_i,
\]

s.t. \(\|u\| = \|v\| = 1\), \(u_i^\top u_j = v_i^\top v_j = 0\), \(j = 1, \ldots, i - 1\), where \(u_i\) and \(v_i\) are the canonical vectors that form the \(i\)-th smallest canonical angle, \(\theta_i\).

A shape subspace is a 3-dimensional subspace; therefore, we can calculate three canonical angles. The similarity between two shape subspaces \(S_1\) and \(S_2\) is defined as

\[
sim (S_1, S_2) = \frac{1}{3} \sum_{i=1}^{3} \cos^2 \theta_i.
\]

If the two shape subspaces correspond to each other completely, \(\sim = 1\) because all canonical angles are 0. In contrast, if the two shape subspaces are orthogonal to each other, \(\sim = 0\).

The mutual subspace method (MSM [12], [13]) is the simplest classification method using canonical angles. In MSM, we calculate the similarities between an input subspace \(S\) and reference subspaces \(D_i (i = 1, 2, \ldots)\). The input subspace is classified into the class with the highest similarity.

However, its classification ability for structural similarity is not effective when discriminating fingers that have large inner class variation, such as individual hand characteristics and slight changes of hand position. This is because MSM is sensitive to slight changes in the basis representation. To improve the classification ability of MSM, we use the framework of Grassmann discriminant analysis (GDA [14]); this is a generalization of linear discriminant analysis (LDA) on the Grassmann manifold, in which each subspace is represented by a point on the manifold. GDA can find the discriminant space that maximizes the class separability. The details of the procedure when using the relative similarities between subspaces can be referred to in the paper by [14].

C. Process Flow of Proposed Method

Fig. 6 shows the flow of the classification by the proposed framework. Here, we consider the problem of classifying an input finger shape into one of \(C\) classes (finger shapes), given a set of \(N_c\) depth images for each class.
Training phase

1) We extract the feature point sets from all reference depth images using the 3D separability filter.
2) We set reference shape matrices $D_i^c (c = 1, \ldots, C; i = 1, \ldots, N_c)$ as given in Eq.(4). In our framework, we need to distinguish the finger shapes by the difference in the pointing directions. To do so, we generate the shape matrices by using the camera position coordinates as the origin of the space, instead of centering it around each finger shape (Fig. 7).
3) Although a shape subspace can be easily generated as the column space of the shape matrix, as mentioned in Sec.III-A, the shape subspace changes if the order of the feature points changes. Thus, points between two shape matrices need to correspond before the similarity of the shape subspaces can be calculated. In our study, we use a method based on the autocorrelation matrix [8]. These autocorrelation matrices $A_i^c$ are used as the references in the classification phase.

Classification phase

1) We extract the feature point set from an input depth image using the 3D separability filter.
2) We set the input shape matrix $S_{in}$ in the same way as in the training phase, by setting the camera position as the origin of space.
3) We generate the autocorrelation matrix $A_{in}$ to correspond to feature points of reference shape matrices $D_i^c$.
4) After completing the corresponding process, we calculate the similarity among reference shape subspaces $D_i^c$ using Eq.(6).
5) With the similarities obtained above, we first calculate a discriminant space on the Grassmann manifold by applying the GDA. Next, we project each shape subspace onto the discriminant space and then calculate the distance between the projected input $S_{in}$ and reference $D_i^c$.
6) The input $S_{in}$ is classified into the class of the nearest reference shape subspace.

IV. EXPERIMENTS

We evaluated the validity of the proposed method by the following two experiments. In the first experiment, we qualitatively evaluated the effectiveness of the proposed 3D separability filter with synthetic data. In the second experiment, we evaluated the performance of our framework with a real finger shape data set by using both qualitative and quantitative results.

A. Experiment with Artificial Data

In this experiment, we compared the 3D separability filter with the conventional 2D separability filter in terms of robustness to shape noise and non-descriptive concave and convex local shapes, such as muscles and wrinkles of the finger. For the purpose of the comparison, we used the synthetic data shown in Fig. 8(a). The data consists of a depth image comprising five different shapes imitating the shapes of finger and hand joints, and their surroundings contain small bumps that represent shape noise.

These shapes for the evaluation were generated as follows. From the left, the four shapes have a bottom radius of 10 pixels and top radius of 5 pixels, and each is a truncated cone with a height of 50, 40, 30, and 20 pixels, respectively. The rightmost shape is a quadrangular pyramid with a height of 40 pixels and rectangular shapes of $15 \times 30$ pixels and $10 \times 20$ pixels as the bottom and top surfaces, respectively. The size of the whole image is $50 \times 260$ pixels.

Fig. 8(b) shows the resulting heat map of the conventional 2D separability filter, and Fig. 8(c) shows the result of the proposed 3D separability filter. In these figures, bright red and dark blue represent high and low separability values, respectively. We can confirm that by using the conventional 2D separability filter, the difference in separability between the target shapes and the other noises is small; this may cause the algorithm to select misleading points as shape descriptors. On
The other hand, when using the proposed filter, the separability of the target shapes was well emphasized and noise was limited effectively. These results show that the feature points in the depth image can be extracted more effectively by using the proposed 3D separability filter.

B. Experiment with Real Finger Shapes

In this experiment, we evaluated the validity of the proposed framework with real finger shape data. First, we conducted an experiment to extract feature points that are useful for shape description, such as fingertips and vertices of bent finger joints, by using the 3D separability filter. Next, we performed a recognition experiment using these feature point data. Below, we first describe the data used in this experiment.

1) Experimental Settings: Depth images were captured from 6 subjects by using a depth sensor (DS325/SoftKinetic) that was attached to the top of a display. The distance from the sensor to the subjects sitting on a chair is approximately 0.5 m. Fig. 9 shows examples of the finger shapes collected from 9 classes. We captured 50 images for each class (50 × 9 = 450 depth images).

The captured depth images include objects other than the finger shapes, such as the face or body of the subjects. As the information of these regions is not used for recognition, we cropped only the hand region according to the distance between it and the depth sensor. Furthermore, we performed noise reduction by applying the median filter and the bilateral filter to the cropped images.

2) Feature Extraction from Finger Shape Data: We evaluated the robustness of the proposed 3D separability filter to inner class shape changes between subjects by extracting feature points from depth images captured from 6 subjects. In this experiment, 6 points were manually set as correct feature points (fingertips and vertices of bent finger joints) in advance. We expected that these feature points can be automatically extracted by using the proposed 3D separability filter.

We conducted the experiment by the following procedure. First, we varied the filter size from 4 to 6 to obtain multiple separability maps for each size. The separability values from these maps were ranked from highest to lowest, and then, 6 points with the highest separability were extracted.

Fig. 10 shows the experimental results. Each column represents a subject. Row A indicates the correct positions of feature points that were determined manually. These include the thumb, index finger, and second joint. Row B indicates the feature points extracted using the 3D separability filter; these are indicated by (+).

The results in row B confirm that the fingertip is accurately extracted from each index finger. Some joints are extracted redundantly or not at all. If the joint of the thumb is occluded, its extraction fails. However, overall, we observe that the proposed filter can extract joints with high accuracy.

3) Validity of Proposed Framework: In the second part of this experiment, we compared the recognition accuracy with a small number of feature points extracted by the 3D separability filter and with 100 feature points that were extracted randomly.

We conducted the evaluation experiment by the following
TABLE I. ERROR RATES, EQUAL ERROR RATES, AND PROCESSING
TIME OF THE EXPERIMENT WITH REAL FINGER DATA. THE 3D
SEPARABILITY FILTER IN THE TABLE INDICATES THE FEATURE POINT
SELECTION METHOD THAT CHOOSES 10 FEATURE POINTS IN DESCENDING
ORDER OF HIGHER SEPARABILITY. RANDOM SAMPLING INDICATES THE
METHOD THAT CHOSES 100 FEATURE POINTS RANDOMLY.

<table>
<thead>
<tr>
<th></th>
<th>ER(%)</th>
<th>EER(%)</th>
<th>Average Time</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>MSM</td>
<td>GDA</td>
<td></td>
</tr>
<tr>
<td>3D Sep.</td>
<td>6.2</td>
<td>5.6</td>
<td>4.6</td>
</tr>
<tr>
<td>Random</td>
<td>8.0</td>
<td>2.7</td>
<td>14.6</td>
</tr>
</tbody>
</table>

procedure. First, two data sets were generated: (1) 10 feature points extracted using a 3D separability filter and (2) 100 feature points that were extracted randomly for 450 images. Then, the leave one out cross-validation was applied, in which nine images from 9 different shapes were used for the test and the remaining 49 × 9 images were used for training to construct shape subspaces. MSM and GDA were used to classify the shape subspaces.

Table I shows the results of this experiment. The achieved performance depends on which classifier, MSM or GDA, is used. However, overall, we can observe the effectiveness of the 3D separability filter from the result. When using the GDA, the performance of the 3D separability filter was not so high compared to that of random sampling. This is probably because the number of extracted feature points from the 3D separability filter was too few to use the GDA. With regard to the computation time, discrimination was faster with the 3D separability filter than with random sampling.

These results suggest an alternative method to further improve the accuracy and processing time of our framework: apply the proposed 3D separability filter to extract a small number of effective feature points and then select several points randomly from these feature points for recognition.

V. CONCLUSION

In this paper, we proposed a method for classifying finger shapes based on the 3D structure of feature points such as fingertips and finger joints. In the proposed method, the 3D structure is represented by a shape subspace in a high-dimensional vector space. To extract such feature points stably, we proposed a 3D separability filter that can extract finger feature points effectively even from an input depth image. This filter outputs the separability between the voxel data of the inner and outer regions of a 3D filter mask. We experimentally confirmed the validity of the proposed framework based on a 3D separability filter by using the depth image data of subjects pointing to targets on a screen.

In a future study, we will develop a method for combining the 3D separability filter and random sampling. In addition, we will further validate the proposed framework through experiments with multiple subjects. Furthermore, in feature point extraction using a 3D separability filter, we will need to solve the problem of instability in feature point extraction when the finger joints are occluded.

ACKNOWLEDGMENT

Part of this work was supported by JSPS KAKENHI Grant Number JP16H02842.

REFERENCES


