3D Model Retrieval Using 3D-DFT Spectral and Cepstral Descriptors of Local Octants

Chang-Hsing Lee*, Jau-Ling Shih, and Yu-Hau Liu

Department of Computer Science and Information Engineering
Chung Hua University, Hsinchu, Taiwan

*Corresponding Author: chlee@chu.edu.tw

Abstract

Conventional 3D-DFT features proposed for 3D model retrieval tried to extract the spectral/cepstral features from the spectrum/cepstrum of the 3D model, without considering the local properties of a 3D model. In this paper, local 3D-DFT spectral descriptor as well as local 3D-DFT cepstral descriptor will be proposed for 3D model retrieval. First, the set of voxels covering the surface of a 3D model is divided into 8 octants. 3D-DFT spectral descriptors and 3D-DFT cepstral descriptors are extracted from each octant and combined to form the local 3D-DFT spectral descriptor and local 3D-DFT cepstral descriptor. Experiments conducted on the Princeton Shape Benchmark (PSB) database have shown that the proposed local 3D-DFT descriptors outperform other 3D-DFT descriptors.

Keywords: 3D model retrieval, 3D discrete Fourier transform (3D-DFT), 3D-DFT spectrum, 3D-DFT cepstrum

1. Introduction

3D models have been widely used in computer aided design, computer animations, movie production, industrial product design, digital library, and so on. Since the number of 3D models grows rapidly, there exist increasing demands for 3D model retrieval systems. Typically, keyword-based search systems were provided to find similar images, videos, or 3D models. To facilitate 3D model retrieval using keyword search mechanism, each 3D model must be labeled with some appropriate keywords. However, it is time-consuming to manually annotate each 3D model with appropriate keywords. Further, it is sometimes hard to find appropriate keywords for some specific models. To overcome this problem, content-based 3D model retrieval mechanism has become widely used to find similar 3D models.

The main challenge to a content-based 3D model retrieval system is how to extract representative features for effectively discriminating different types of 3D models [1, 2]. In general, 3D model retrieval methods can be classified into four categories: histogram-based descriptors [3-19], transform-based descriptors [20-28], 2D view-based descriptors [29-52], and graph-based descriptors [53-56].

Histogram-based descriptors consider the statistical distributions or histograms of local features evaluated at the vertices or meshes of a 3D model. These histogram-based descriptors include geodesic features [3, 4], local statistical features [5], Poisson histogram [6], sliced image histogram [7], beta/distance histograms (BD) [8], grid D2 (GD2) [9], modified D2 descriptor [10, 11], shape distributions [13], shape spectrum descriptor (SSD) [14, 15], 3D Hough transform descriptors [16], shape histograms [19], etc. The main drawback of these spatial shape descriptors is that they do not take into account how the local features are spatially distributed over the model surface.

Transform-based descriptors are extracted by first mapping the 3D data into frequency domain. These frequency domain representations include probability density-based shape descriptors [21], concrete radicalized spherical [22], spherical wavelet transform [23], 3D discrete curvelet transform [24], rotation invariant spherical harmonics [1, 25], spherical harmonics [26], 3D Fourier transform [27], angular radial transform (ART) [28], etc. The effectiveness of these frequency descriptors relies heavily on the resolution of the voxelization (voxel decomposition) operation of a 3D model and how to extract discriminative features from the set of transformed coefficients.
2D view-based descriptors are generally obtained by projecting a 3D model on a number of 2D planes from different views. These projected results can be either binary images representing the silhouettes from different views, or gray-level images representing the curvature, depth, or some geometrical information. One merit of the view-based descriptors is that it is easy to design a query interface which supports a 2D sketch for 3D model retrieval. The view-based descriptors include camera constraint free view-based descriptor [29], Hypergraph [30], Bayesian network lightfield descriptor (BLD) [32], weighted bipartite graph matching (WBG) [32], PCA plane projection [33], spatial shape information [34], Bag-of-Features [35], spatial structure circular descriptor (SSCD) [36], Panoramic views [37], shape impact descriptor (SID) [38], ART elevation descriptor (AED) [39], Multi-Fourier spectra descriptor [40], principal plane analysis [41, 42], spherical trace transform [43], depth image [44], elevation descriptor (ED) [45], curvature map [46-48], grid sphere descriptor (GSD) [49], depth buffer descriptor [50], light field Descriptor (LFD) [51], etc. The major problem with view-based descriptors is that rotation invariance has to be solved by either pose normalization prior to 2D projections, by extracting rotation-invariant features, or by matching 2D feature descriptors over many different alignments.

Graph-based descriptors have the potential of describing the geometrical and topological properties of a 3D model in a more faithful way, particularly for deformable 3D models. The methods include extended cone-curvature [53], medial scaffold [54], Reeb graph [55], skeleton-based descriptors [56], etc.

In our prior work [57], we have designed a 3D model retrieval system using 3D discrete Fourier transform (3D-DFT) spectral features and cepstral features. Experimental results have shown that these descriptors can obtain promising performances. In this paper, local 3D-DFT spectral and cepstral features will be extracted from different local regions of the 3D model and combined for 3D model retrieval.

2. Review of 3D-DFT Descriptors

In this section, we will first describe some 3D-DFT feature extraction methods for 3D model retrieval, including 3D-DFT spectrum [27, 57], 3D-DFT cepstrum [57], 3D-DFT subband spectrum [57], and 3D-DFT subband cepstrum [57].

2.1 3D-DFT Spectrum

The 3D-DFT spectrum was obtained by applying 3D-DFT on the voxels of a 3D model V(x, y, z) defined as follows [27]:

\[ F(u,v,w) = \frac{1}{\sqrt{N}} \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} \sum_{z=0}^{N-1} V(x,y,z) \exp \left( -j2\pi(ux+vy+wz) \right) \]  

where \( u, v \) and \( w \) are 3D-DFT frequency indices \((0 \leq u, v, w < N)\). The magnitudes of those low-frequency coefficients (except \( F(0, 0, 0) \)) were selected to form the 3D-DFT spectral features.

2.2 3D-DFT Cepstrum

By applying 3D-IDFT on the magnitude spectrum \( |F(u, v, w)| \), we can obtain the 3D-DFT cepstrum \( C(a, b, c) \):

\[ C(a,b,c) = \frac{1}{\sqrt{N}} \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} \sum_{z=0}^{N-1} |F(u,v,w)| \exp \left( \frac{j2\pi(au+bv+cw)}{N} \right) \]  

where \( a, b \) and \( c \) are 3D-DFT quefrency (cepstral frequency) indices. The magnitudes of those low-quefrency coefficients were selected to form the 3D-DFT cepstral features.

2.3 3D-DFT Subband Spectrum

To obtain the 3D-DFT subband spectral features, we first decompose the 3D-DFT spectrum into \( K \) \((K = WxHxL)\) subbands, notated by \( SB_{p,q,r} \) \((0 \leq p < W, 0 \leq q < H, 0 \leq r < L)\), by using uniform subband decomposition or logarithmic subband decomposition along the \( x \)-axis, \( y \)-axis, and \( z \)-axis, respectively [57]. Then, the sum of magnitude spectrum within each subband, notated by \( M_{p,q,r} \), is calculated to yield the 3D-DFT subband spectral features:

\[ M_{p,q,r} = \sum_{(u,v,w) \in SB_{p,q,r}} |F(u,v,w)| \]  

2.4 3D-DFT Subband Cepstrum

By applying 3D-IDFT on the subband magnitude spectrum, we can obtain the 3D-DFT subband cepstrum:

\[ C_{id}(a,b,c) = \frac{1}{\sqrt{N}} \sum_{(u,v,w) \in SB_{p,q,r}} M_{p,q,r} \exp \left( \frac{j2\pi(ap+bq+cr)}{N} \right) \]  

The magnitudes of those low-quefrency coefficients (except \( C_{id}(0, 0, 0) \)) were selected as the 3D-DFT subband cepstral features.

3. Proposed 3D Model Retrieval Approach

Prior 3D-DFT feature extraction methods tried to extract the spectral/cepstral features from the spectrum/cepstrum of the 3D model, without considering...
the local properties of a 3D model. Thus, in this paper, we will try to extract local spectral/cepstral features to capture the local characteristics of different parts of a 3D model. First, the principal planes method [42] is used for pose alignment of a 3D model. Then, the 3D model is divided into eight octants. 3D-DFT spectral/cepstral features [27, 57] will then be extracted from each local octant. Finally, the spectral and cepstral features extracted from all octants will be combined to search similar 3D models.

3.1 3D Model Alignment

In this paper, we use the principal planes method [42] to align the pose of each 3D model. First, the smallest bounding box that circumscribes the 3D model is divided into a voxel grid of size $N \times N \times N$ (please see Fig. 1). In this paper, $N = 64$. If there is a polygonal surface passes through the voxel with coordinates $(x, y, z)$, this voxel is defined as an opaque voxel, notated by $V(x, y, z) = 1$; otherwise, this voxel is defined as a transparent voxel, notated by $V(x, y, z) = 0$. As a result, the voxelization operation converts the 3D mesh model into a discrete 3D function with regularly sampled voxels in 3D coordinates. Second, the model’s mass center is moved to $(N/2, N/2, N/2)$. The 3D model is then scaled such that the average distance from all opaque voxels to the mass center becomes $N/4$. Third, the pose of this 3D model is aligned by the grid-based principal component analysis (GPCA) [33]. Thus, the features extracted from this aligned 3D model will be invariant to rotation, scaling, and translation.

3.2 Inverse Distance Transform

The voxelization operation will encode each voxel in the 3D Cartesian coordinate system a value 1 (denoting an opaque voxel) or 0 (denoting a transparent voxel). Such a binary encoding method will produce severe high-frequency artifacts due to the blocky structure of the binary voxel representation. To suppress these high-frequency artifacts, we will employ the inverse distance transform (IDT) proposed by Dutagaci et al. [20] to get a smoothed voxel representation. The IDT value associated with a voxel $V$ is defined as follows:

$$\text{IDT}(V) = \frac{1}{1 + \text{DT}(V)}$$  \hspace{1cm} (5)

where $\text{DT}(V)$ is the distance between $V$ and its nearest opaque voxel in the 3D space given by

$$\text{DT}(V) = \min_{V_o \in N^3} d(V, V_o)$$  \hspace{1cm} (6)

where $d(V, V_o)$ is the distance between voxel $V$ and an opaque voxel $V_o$ measured by the Euclidean distance, $V^o$ is the set of all opaque voxels:

$$V^o = \{V | V(x,y,z) = 1\}$$  \hspace{1cm} (7)

Therefore, the voxels on the surface of the 3D model take the largest $\text{IDT}$ values and the $\text{IDT}$ value decreases smoothly as the voxels move away from the model surface. Fig. 2 gives an example showing how the IDT operation was used to get a smoothed voxel representation.

Fig. 1. 3D model alignment. (a) The 3D monster truck model circumscribed by a bounding box. (b) The bounding box is decomposed into a $N \times N \times N$ voxel grid. (c) The normalized 3D monster truck model.

Fig. 2. An example illustrating IDT. (a) A slice of the motorcycle model (b) A slice from binary voxel representation (c) A slice resulted from IDT operation.
3.3 3D Model Decomposition

To extract the local features, we divide the set of voxels of a 3D model in the 3D space into 8 octants represented as follows:

\[ Q_i = \{ (x, y, z) | x \geq 0, y \geq 0, z \geq 0 \} \]  
\[ (8) \]

\[ Q_2 = \{ (x, y, z) | x < 0, y \geq 0, z \geq 0 \} \]  
\[ (9) \]

\[ Q_3 = \{ (x, y, z) | x < 0, y < 0, z \geq 0 \} \]  
\[ (10) \]

\[ Q_4 = \{ (x, y, z) | x \geq 0, y < 0, z \geq 0 \} \]  
\[ (11) \]

\[ Q_5 = \{ (x, y, z) | x \geq 0, y \geq 0, z < 0 \} \]  
\[ (12) \]

\[ Q_6 = \{ (x, y, z) | x < 0, y \geq 0, z < 0 \} \]  
\[ (13) \]

\[ Q_7 = \{ (x, y, z) | x < 0, y < 0, z < 0 \} \]  
\[ (14) \]

\[ Q_8 = \{ (x, y, z) | x \geq 0, y < 0, z < 0 \} \]  
\[ (15) \]

where \( Q_i (1 \leq i \leq 8) \) denote the \( i \)-th octant (see Fig. 3). Thus, each octant contains some specific voxels (revealing local properties) of a 3D model. Since the size of the bounding box is \( N \times N \times N \), the size of each local octant is \( N/2 \times N/2 \times N/2 \).

For each local octant, the 3D-DFT spectral and 3D-DFT cepstral features will be extracted and concatenated to form the local 3D-DFT spectral descriptor and local 3D-DFT cepstral descriptor.

![Fig. 3. Decomposition of a 3D model into 8 octants.](Image)

3.4 Local 3D-DFT Spectrum

The local 3D-DFT spectrum is obtained by applying 3D-DFT on the voxels within each octant \( Q_i (1 \leq i \leq 8) \) to get the corresponding local spectrum:

\[ F(u,v,w) = \frac{1}{\sqrt{N^3}} \sum_{i,j,k=0}^{N-1} V(x,y,z) \exp \left( -\frac{2\pi i (ux + vy + wz)}{N^2} \right) \]  
\[ (16) \]

where \( u, v \) and \( w \) are 3D-DFT frequency indices \( (0 \leq u, v, w < N) \), \( F_i \) is the spectrum of the \( i \)-th octant. The magnitudes of those \( M \times M \times M \) low-frequency coefficients (except \( F_i(0,0,0) \)) of each local spectrum \( F_i \) will be concatenated to form the local 3D-DFT spectral descriptor. Thus, the length of the local 3D-DFT spectral descriptor is \( 8(M^3-1) \).

3.5 Local 3D-DFT Cepstrum

By applying 3D-IDFT on each local magnitude spectrum \( |F_i(u,v,w)| \), we can obtain the local 3D-DFT cepstrum \( C_i(a, b, c) \), where \( a, b \) and \( c \) are 3D-DFT frequency indices. The magnitudes of those \( D \times D \times D \) low-frequency coefficients (except \( C_i(0,0,0) \)) of each local cepstrum \( C_i \) will be concatenated to form the local 3D-DFT cepstral descriptor. Thus, the length of the local 3D-DFT cepstral descriptor is \( 8(D^3-1) \).

3.6 3D Model Retrieval

Let \( f^S \) denote the feature vector corresponding to the local 3D-DFT spectral descriptor of the query model, and \( f^{L,S} \) denote the feature vector corresponding to the local 3D-DFT spectral descriptor of the \( t \)-th matching model in the database. The distance between the query model and the \( t \)-th matching model evaluated in terms of local 3D-DFT spectral descriptor is defined as follows:

\[ d^{LS} = \sum_{m=1}^{N^3-1} \left| f^{LS}(m) - f^{LS}(m) \right| \]  
\[ (17) \]

where \( f^{LS}(m) \) and \( f^{LS}(m) \) denote respectively the \( m \)-th feature value of \( f^S \) and \( f^{LS} \).

Similarly, let \( f^{LC} \) and \( f^{LC} \) denote the local 3D-DFT cepstral descriptors of the query model and the \( t \)-th matching model, respectively. The distance between the query model and the \( t \)-th matching model evaluated in terms of local 3D-DFT cepstral descriptor is given as follows:

\[ d^{LC} = \sum_{d=1}^{N^3-1} \left| f^{LC}(d) - f^{LC}(d) \right| \]  
\[ (18) \]
where \( f_{L}^{i}(d) \) and \( f_{S}^{L}(d) \) denote respectively the \( d \)-th feature value of \( f_{L}^{i} \) and \( f_{S}^{L} \). Therefore, the models that are most similar to the query one can be found by finding those having smaller distance values.

4. Experimental Results

To demonstrate the effectiveness of the proposed local descriptor, some experiments have been conducted on the Princeton Shape Benchmark (PSB) database [58]. The PSB database contains 1814 models (161 classes) which are divided into 907 training models (90 classes) and 907 testing models (92 classes). The experiments were performed on the 907 testing models (see Fig. 3 and Fig. 4) to keep the results comparable with other methods. The discounted cumulative gain (DCG) [59] will be employed to compare the performance of different descriptors. DCG at the \( r \)-th rank is defined as follows:

\[
DCG_r = \begin{cases} 
DCG_{1-r} + \frac{L_r}{\log_3(r)} & r \geq 2 \\
L_r & r = 1
\end{cases}
\]

where \( L_r \) = 1 if the query model and the \( r \)-th model in the ranked retrieval list have the same class label; otherwise, \( L_r = 0 \). The overall DCG score for a query model is defined as \( DCG_T \), where \( T \) is the number of models in the database. It is clear that if the models appearing in the front of the retrieval list have the same class label as the query one, a larger DCG score can be obtained. On the other hand, if the models having identical class label to the query one appear in the rear of the retrieval list, a small DCG score will be obtained. In our experiments, each model in the database will be presented as a query one to evaluate the average DCG score among all query models:

\[
DCG = \frac{1}{T} \sum_{t=1}^{T} DCG_r(t)
\]

where \( DCG_r(t) \) denotes the DCG score associated with the \( t \)-th matching model.

Table 1 compared the performance of different 3D-DFT descriptors, including (1) 3D-DFT spectrum (notated by 3D-DFT-S), (2) 3D-DFT cepstrum (notated by 3D-DFT-C), (3) 3D-DFT subband spectrum with uniform subband decomposition (notated by 3D-DFT-S-USD), (4) 3D-DFT subband spectrum with logarithmic subband decomposition (notated by 3D-DFT-S-LSD), (5) 3D-DFT subband cepstrum with uniform subband decomposition (notated by 3D-DFT-C-USD), (6) 3D-DFT subband cepstrum with logarithmic subband decomposition (notated by 3D-DFT-C-LSD), and the proposed local 3D-DFT spectrum (notated by 3D-DFT-LS) as well as local 3D-DFT cepstrum (notated by 3D-DFT-LC). We can see that, the proposed local 3D-DFT descriptor outperforms their corresponding 3D-DFT descriptors.

Table 2 compares the proposed local 3D-DFT descriptors with/without performing IDT operation before feature extraction. It can be seen that by performing IDT operation to get a smoothed voxel representation of each 3D model, we can remove those high-frequency artifacts and thus will achieve higher retrieval accuracy.

Table 1. Comparison of different 3D-DFT descriptors on the PSB database in terms of the DCG score. The parentheses show the setting of the parameters.

<table>
<thead>
<tr>
<th>Descriptor</th>
<th>DCG</th>
</tr>
</thead>
<tbody>
<tr>
<td>3D-DFT-S (M = 16)</td>
<td>64.63</td>
</tr>
<tr>
<td>3D-DFT-S-USD (W = H = L = 32)</td>
<td>65.54</td>
</tr>
<tr>
<td>3D-DFT-S-LSD (W = L = 12)</td>
<td>65.96</td>
</tr>
<tr>
<td>3D-DFT-C (D = 16)</td>
<td>63.14</td>
</tr>
<tr>
<td>3D-DFT-C-USD (W = H = L = 32, D = 16)</td>
<td>64.67</td>
</tr>
<tr>
<td>3D-DFT-C-LSD (W = H = L = 12, D = 8)</td>
<td>62.80</td>
</tr>
<tr>
<td>3D-DFT-LS (M = 16)</td>
<td>67.47</td>
</tr>
<tr>
<td>3D-DFT-LC (D = 8)</td>
<td>67.26</td>
</tr>
</tbody>
</table>

Table 2. Comparison of the proposed local 3D-DFT descriptors with/without performing IDT operation. The parentheses show the setting of the parameters.

<table>
<thead>
<tr>
<th>Descriptor</th>
<th>DCG</th>
</tr>
</thead>
<tbody>
<tr>
<td>3D-DFT-LS (M = 16)</td>
<td>67.47</td>
</tr>
<tr>
<td>3D-DFT-LC (D = 8)</td>
<td>67.26</td>
</tr>
<tr>
<td>3D-DFT-LS + IDT (M = 16)</td>
<td>69.24</td>
</tr>
<tr>
<td>3D-DFT-LC + IDT (D = 16)</td>
<td>68.29</td>
</tr>
</tbody>
</table>
5. Conclusions

In this paper, local 3D-DFT spectral descriptor and local 3D-DFT cepstral descriptor are proposed for 3D model retrieval. First, the set of voxels of a 3D model in the 3D space is divided into 8 octants. 3D-DFT spectral features and 3D-DFT cepstral features are extracted from each octant and combined to form the local 3D-DFT spectral descriptor and local 3D-DFT cepstral descriptor. Experiments conducted on the Princeton Shape Benchmark (PSB) database have shown that our proposed local 3D-DFT descriptors outperform other 3D-DFT descriptors.

Acknowledgment

This research was supported in part by the Ministry of Science and Technology of R.O.C. under contract MOST 103-2221-E-216-014.

References


(49) J. L. Shih and H. Y. Chen, “3D model retrieval based on grid sphere and dodecahedral silhouette descriptors”, in Proc. 9th Joint Conf. on Information Science, Taiwan, 2006.


