3D Object Classification via Spherical Projections

Zhangjie Cao¹, Qixing Huang², and Ramani Karthik³
¹School of Software, Tsinghua University, China
²Department of Computer Science, University of Texas at Austin, USA
³School of Mechanical Engineering, Purdue University, USA

Figure 1: (left) contour-based projection and network; (right) depth-based projection and network

Summary

1. An novel approach for 3D object classification which unifies two key components:
   - A spherical representation exploiting both depth variation and contour information which can capture geometric details and data dependencies across the entire object.
   - Deep neural networks incorporating large-scale labeled images for training to classify spherical representations of 3D objects.

Challenges

1. Limitations of previous works:
   - For 3D convolutional methods, the resolution of a 3D convolutional neural network is usually very coarse.
   - For all 3D-based methods, the number of available training data is limited.
   - For image-based methods, most do not capture the dependencies between views.

Our spherical projections combine key advantages of these two mainstream 3D classification methods.

Depth-based Projection

The left Figure indicates that each point is recorded as the distance to the first hitting point. Otherwise, the distance is set to be zero.

The right Figure indicates the cylindrical convolution.

We first compute depth values for vertices of a semi-regular quad-mesh whose axes aligns with the longitude and latitude (cos(θ)cos(φ), cos(θ)sin(φ), sin(θ))

Then, we generate the depth value of other points on the sphere by linear interpolation. Given a point with spherical coordinate (θ, φ), where θ ≤ θ_{j-1}, φ ≤ φ_{j-1}, its depth value is given by:

Finally, as in the Figure(right) of depth-projection, we proceed to generate cylindrical strips from the depth projection described above. We first use the strip covering the following area:

Since regions of high latitude suffer from severe distortion, we eliminate them by setting θ_j to 30° to 150°. To utilize information from high latitude regions for classification, we then use the following strip:

### Contour-based Projection

As shown in Figure(left) of contour-based network, contour-based projection shotes a 3x12 grid of images of the input object from 36 view points. The locations are φ_j = 0°, 30°, · · · , 330°, θ_i = −60°, 0°, 60°. The up-right orientation of the camera always points to the north pole. The viewing angle of each image is 45°. The resolution of each image is 256x256. In our experiments, we have varied the value of m and found that m = 12 provides a good trade-off between minimizing the number of views and ensuring that the resulting projections are appropriately invariant to rotating the input object.

### Experiment Setup

- **Datasets:** ModelNet40, ShapeNetCore
- **Parameter selection:** Cross-validation by jointly assessing
- **Methods to compare with:** Image-based methods: MVCNN, MVCNN-MultiRes; 3D-based methods: 3D ShapeNets, VoxelNet, Volumetric CNN, OctNet, combined methods: FusionNet. All of these methods use the upright orientation but do not use the front orientation.

### Results

- **Accuracy of our approaches and the various baseline methods on ModelNet40 and ShapeNetCore**
- **Elevation Degree Variations**

### Running Time Analysis

- **Running Time of All the methods on ModelNet40 and ShapeNetCore**