Multi-view Rank Pooling for 3D Object Recognition**

Chaoda Zheng*, Yong Xu*†‡, Ruotao Xu*, Hongyu Chi† and Yuhui Quan*§
* School of Computer Science and Engineering, South China University of Technology, Guangzhou, China
†Peng Cheng Laboratory, Shenzhen, China
‡Communication and Computer Network Laboratory of Guangdong, China
§Guangdong Provincial Key Laboratory of Computational Intelligence and Cyberspace Information, China

***Abstract*—3D shape recognition via deep learning is drawing more and more attention due to huge industry interests. As 3D deep learning methods emerged, the view-based approaches have gained considerable success in object classification. Most of these methods focus on designing a pooling scheme to aggregate CNN features of multi-view images into a single compact one. However, these view-wise pooling techniques suffer from loss of visual information. To deal with this issue, an adaptive rank pooling layer is introduced in this paper. Unlike max-pooling which only considers the maximum or mean-pooling that treats each element indiscriminately, the proposed pooling layer takes all the elements into account and dynamically adjusts their importances during the training. Experiments conducted on ModelNet40 and ModelNet10 shows both efficiency and accuracy gain when inserting such a layer into a baseline CNN architecture.**

I. Introduction

3D shape recognition task is a fundamental challenge in computer vision domain. Compared with 2D images, 3D data contain more geometric information and can encode more informative real world structures. However, 3D object analyzing is a still challenging task due to the irregularity of data representation. Unlike 2D images which are in the form of matrices, 3D data can be stored and represented in various format (polygon meshes, point clouds, voxels, multi-view images, etc). With the recent progress in large scale 3D shape datasets, 3D deep learning has shown its great potential in the 3D domain [1]–[14]. And those methods can be divided into two trends in terms of their input formats: shape-based methods and view-based methods. Shape-based methods directly consume the native formats of 3D models such as voxels and point clouds. Voxel-based methods partition the 3D space into regular grids and thus they can use a 3D tensor to represent a 3D shape. Due to the regularity of voxels, it's intuitive to directly apply 3D convolutional neural networks on voxelized 3D shapes [1]–[3], [5], [15]. However, the computational and memory cost of these methods grow cubically as the data resolution increases. Differently, a point cloud represents a 3D shape with a set of surface points, which are permutation invariant. The pioneer of point-based methods is PointNet [16], which utilizes a per-point MLP followed by a max-pooling to extract the global feature of a given point cloud. And PointNet++ [17] further improved the PointNet architecture using a multi-scale grouping scheme, so that the geometric distribution of local regions can be considered. Besides processing point clouds using PointNets as basic blocks, graph convolutional networks can also be used for point clouds analysis, since a point cloud can be easily converted to a graph [6]. Though shape-based methods directly operate on native 3D data, they are sensitive to geometric representation artifacts (e.g. non-manifold geometry, polygon soups, no interior), which are usually present in most of the datasets. View-based methods, however, can effectively avoid such artifacts. And this is one of the main reason why view-based methods are generally better than shape-based methods in object recognition tasks.

View-based methods represents a 3D object via multiple 2D images, which are captured from different viewpoints. MVCNN [8] is one of the very first view-based methods utilizing 2D convolutional neural networks. The network use a shared CNN to extract features from multiple 2D images of a 3D object and use a view-pooling layer to aggregate the view-level features into a global shape-level descriptor. This pipeline is then widely adopted in other view-based approaches [9]–[12]. In object recognition task, the main goal is to design discriminative shape descriptors which can be used to distinguish between objects from different categories. And a well designed shape descriptor must consider the intrinsic properties of the input data. Thus for the view-based deep learning methods, the fundamental challenge is to design a effective view aggregation module which fully considers the intrinsic properties of the multi-view representations. MVCNN [8] simply use a view-wise max-pooling to combine the view-level features. Nevertheless, only keeping the maximums leads to loss of visual information. In order to prevent the loss of visual details, it's intuitive to replace the max-pooling layer with the average-pooling layer. However, the average-pooling was proven to be less effective than the max-pooling in the experiments of [8]. We think this is due to the useful features being contaminated by a large amount of redundant features. Recently, a number of methods were presented to cope with such an issue. For example, [10], [11] used grouping techniques to exploit the similarity among views. [9] presented a harmonized bilinear pooling to consider patch-to-patch similarities. In this paper, we mainly focus on improving the effectiveness of the view aggregation module in the view-based deep learning pipeline. We propose a rank-pooling layer to deal with the shortcomings of the max-pooling layer and the average-pooling layer. The idea of the rank-pooling layer, though is quite intuitive, can be divided into two parts:

**This work was supported in part by the National Natural Science Foundation of China (61672241, 61602184, 61872151, and U1611461), in part by the Natural Science Foundation of Guangdong Province (2016A030308013 and 2017A030313376), in part by the Science and Technology Program of Guangzhou(201707010147 and 20180201005).

978-1-7281-3723-0/19/$31.00 ©2019 IEEE