Chenhao Xu

Personal Information

Status:	**Undergraduate Student**
Program:	**Mathematics and Computer Science**
School:	**New York University Abu Dhabi**
Website:	**https://www.linkedin.com/in/%E7%90%9B%E6%98%8A-%E5%BE%90-067852124/**
RA Period:	**From 2019-09 to 2020-05**

Biography

I'm a Co-founder of Opus foundation. Before that, I was a research assistant in New York University Abu Dhabi, advised by Professor Yi Fang. I am broadly interested in 3D Computer Vision, Pattern Recognition and Deep Learning.

Capstone Project: EPDNet: Encoder-less 3D Shape Descriptor Extraction

1 Description

Existing 3D shape descriptor extraction methods based on neural network encoders have not been able to replicate the same success of its 2D counterpart, due to the increased time-space complexity brought by the extra dimension and the unstructured nature of point cloud data. To tackle this challenge, we propose EPDNet (Encoder-less Point Drift Net), a novel formulation based on auto- decoder networks that allows us to extract joint shape descriptors of 3D point cloud shape pairs without needing an encoder network, by directly optimizing latent shape descriptors for the self- supervised coherent point drift task. We show that by using this formulation, we can bypass the difficulty of designing encoders for 3D data and directly obtain informative shape descriptor vectors that capture the relationship between a pair of 3D shapes. We further demonstrate the effectiveness of our method by applying the extracted shape descriptors in other computer vision tasks such as classification.

2 Method

This project develop an encoder-less 3D shape descriptor extraction method, based on auto-decoder networks, that directly operates on point cloud data. As displayed in Figure.1, instead of producing latent encodings with an encoder like an auto-encoder, an autodecoder directly accepts a latent vector as an input. A randomly initialized latent vector is assigned to each data point, and the optimal latent vector is found using gradient descent. we describe the formulation of EPDNet. Given a pair of 3D shapes A and B, we assume that there exists a latent vector z that describes the relation between the 2 shapes. This latent vector is the desired end product of our model, and can be used to determine whether the two objects are similar or from the same class, which is useful for applications such as retrieval and classification. For our implementation, the transformation function f is realized as a fully-connected neural network, from here on referred to as the decoder network. As the transformation function takes both the latent encoding vector and 3D point positions as input, the input to our

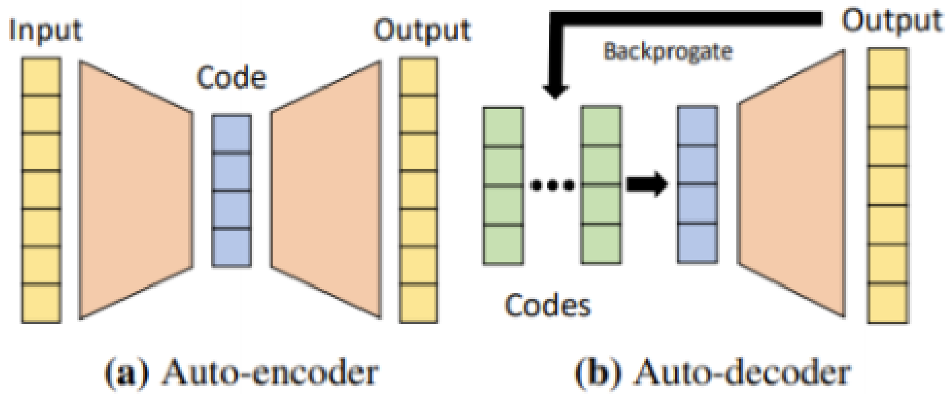

Figure 1: The pipeline of the proposed method.

decoder network is the concatenation of 3D point coordinates and the shape descriptor vector. Figure.2 shows that coherent point drift is the task of predicting a geometrically coherent drift to the source point set to match it with the target point set

3 Results

In this section, we conduct experiments to demonstrate the effectiveness of the

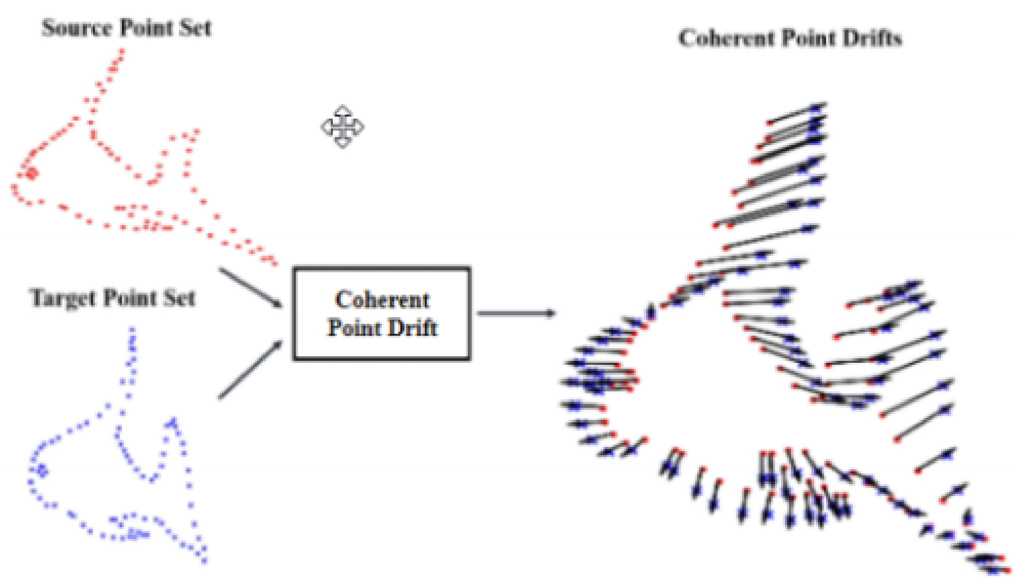

Figure 2: Coherent point drift.

Model	Total Accuracy	Same Class F1-score	Different Class F1-score
Base CompNet	*0.87*	*0.87*	*0.87*
SVM	0.82	0.81	0.83
Decision Trees	0.73	0.73	0.74
Logistic Regression	0.82	0.81	0.82
Random Forest Classifier	0.78	0.78	0.78
Naive Bayes	0.78	0.78	0.78
Ensemble CompNet 1	**0.97**	**0.97**	**0.97**
Ensemble CompNet 2	**0.96**	**0.96**	**0.96**

Table 1: The results with different model configurations.

proposed EPDNet. We used a subset of the ModelNet training set which consists of a total of 10817 3D shapes from 7 different object classes. Each shape consists of 2048 points, represented as a tensor of shape (3, 2048). For fast experiment iterations, we then only sample every third point for each shape, and each resultant shape now consists of 683 points. For each of 10817 shapes, we then randomly sample 2 target shapes from the training set, one from the same class as the original shape and the other from a different class, thus creating a total of 21634 shape pairs. The purpose of sampling pairs from both same and different object classes is to ensure the auto-decoder is capable of producing both kinds of transformation, which may benefit the classification task that we experiment with at a later stage. As shown in Table 1, the model achieves 96% overall accuracy on the test data, with 96% F1 scores on both positive and negative samples. The reason for the model's improved performance is likely due to the use of the ensemble learning technique of bagging: by averaging multiple weakly correlated models, the resultant ensemble model has lower variance and avoids over-fitting, thus performing better on the test data. The impressive performance the model was able to achieve also indicates that the encodings found using EPDNet preserve rich information on the original shape pair.