COMPUTER VISION PROJECT

Learning Implicit Fields for Generative Shape Modeling

https://github.com/Computer-Vision-IIITH-2021/project-no-flux-given.git

The Team

Namratha Gopalabhatla - 20171017

Sagrika Nagar - 20171204

Nishant Sachdeva - 2018111040

Sumanth Balaji - 2018114002

Presentation Highlights

1. Introduction
2. Method Overview
3. Goals/Deliverables
4. Expected Timeline

Introduction

Our task is shape generation. For that purpose, we look to establish the use of implicit fields for learning generative models of shapes and introduce an implicit field decoder for shape generation, aimed at improving the final visual quality of these generated shapes.

Context

- An implicit field assigns a value to each point in 3D space so that a shape can be extracted as an iso-surface. Our encoder is trained to perform this assignment by means of a binary classifier. Specifically, it takes a point coordinate, along with a feature vector encoding a shape, and outputs a value that indicates whether the point is outside the shape or not.
- By replacing conventional decoders with our implicit de-coder for representation learning and shape generation, we demonstrate superior results for tasks such as generative shape modeling, interpolation, and single-view 3D reconstruction, particularly in terms of visual quality.

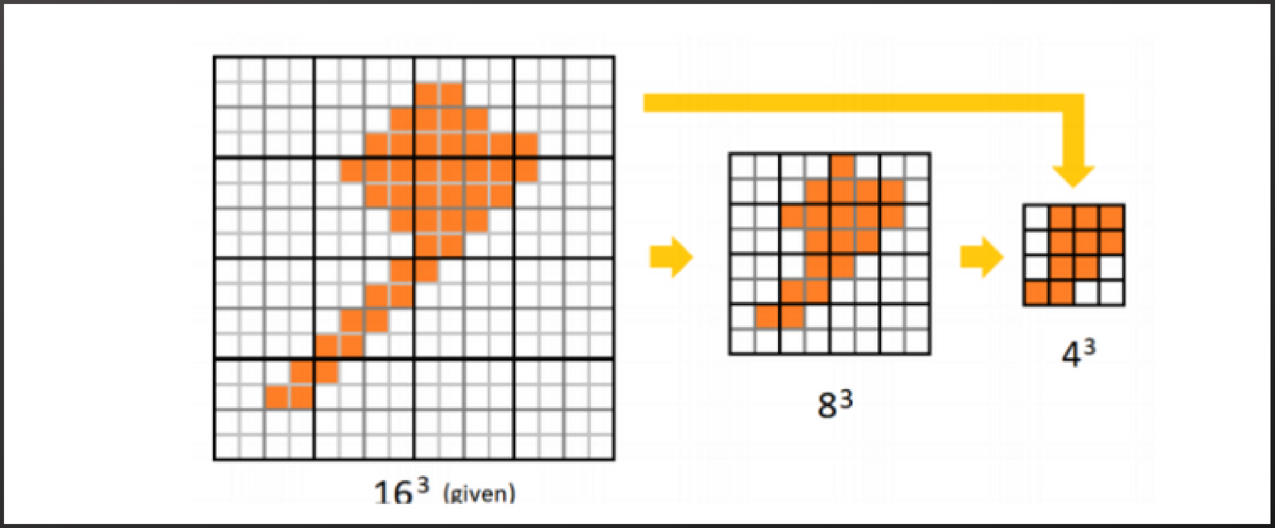
Method Overview

The major steps that are involved are:

1. Data preparation
2. Structure of the Model
    a. Encoder
    b. Decoder
3. Application of Implicit Decoders

Data Preparation

- Input data is in the form of point-value pairs.
- The original voxel models and rendered views are from Hierarchical Surface



Data Preparation

- Prediction (HSP). Since our network takes point-value pairs, the voxel models require further sampling.
- Sampling is done by taking the center of each voxel and producing n^3 points. Lesser points can be generated with a more efficient approach involving more sampling near shape surfaces.

Encoder

- The encoder's basic task is to take the input data and generates the corresponding feature vectors that are then fed to the decoder. The type of encoder can be changed based on the application of the model.
- The encoder that we are going to use to generate the feature vectors of the shape will either be a CNN or a PointNET depending on the application and structure and dimension of the input data.

Decoder

- The structure of our implicit decoder will be that of a cascade of fully connected layers with leaky ReLU as our non-linearity.
- To increase the speed of learning, skip connections will be added in the form of copy and concatenate connections.
- If the feature vector becomes too large, increasing the size of the model, these skip connections can also be removed.

Decoder

- The implicit decoder that we are implementing here will have a total of 7 layers including the input layer and the final Sigmoid layer.
- The loss function used to train the complete model is a weighted mean square error between the ground truth labels and the labels predicted by the model for each sampled point of the target shape.

Expected Timeline

We expect to be able to stick to the following production deadlines

- Data Preparation : Early March
- Encoder : Early March - Mid Evaluation
- Decoder : Mid Evaluation - April

Thank
You