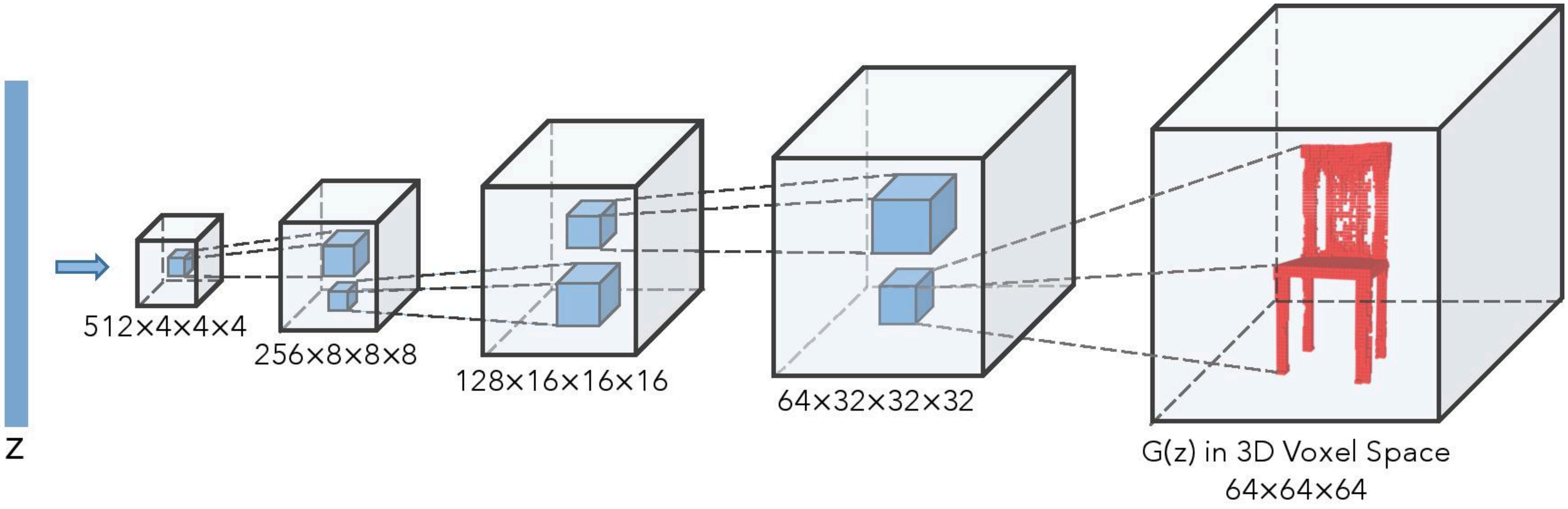
Pixel2Mesh: Generating 3D Mesh Models from Single RGB Images

Yuan Yao

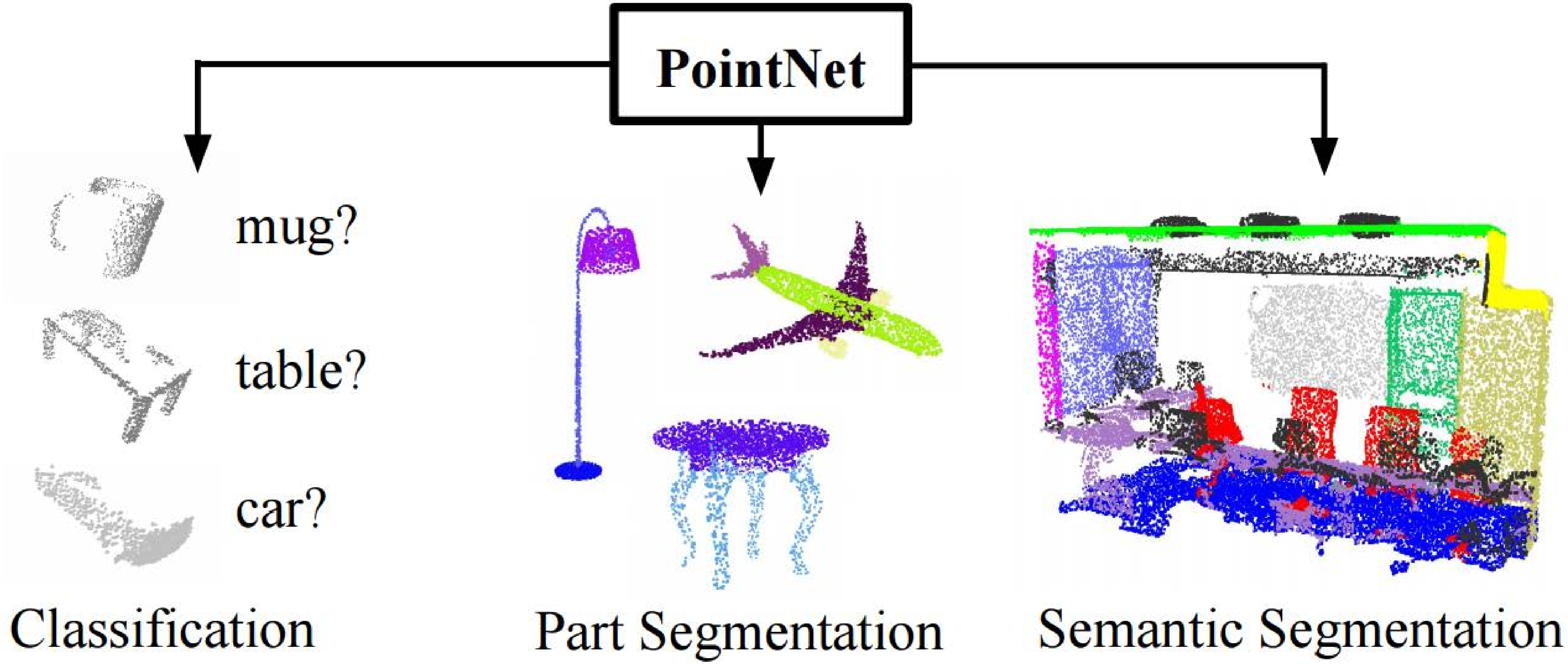
Jan 28, 2019

Representation for 3D Learning



Voxel

Representation for 3D Learning



Point Cloud

Problem

- Lose important surface details
- Non-trivial to reconstruct a surface model

Main Idea

- Deform from mean shape to target geometry
  - Deep learning is better at predicting residual.
  - A series of deformation can be added up together.
  - Enable trade-of between the complexity of deep learning model and quality of the result
- Two modalities: 2D regular grid and 3D geometry
  - Graph based fully convolutional network(GCN)
    - Vertex->node, edge->connection
  - VGG-16 like architecture to extract features

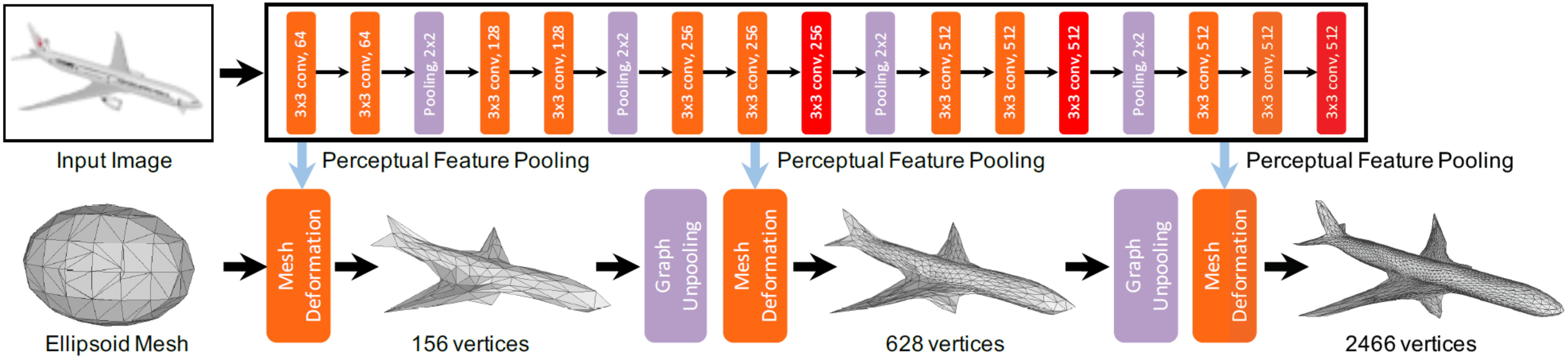
Graph Convolutional Network

- Graph $\mathcal{M} = (\mathcal{V}, \mathcal{E}, \mathbf{F})$, consists of vertices $\mathcal{V} = \{v_i\}_{i=1}^N$, edges $\mathcal{E} = \{e_i\}_{i=1}^E$ and feature vectors $\mathbf{F} = \{f_i\}_{i=1}^N$ attached on vertices.

- Graph based convolutional layer is defined as

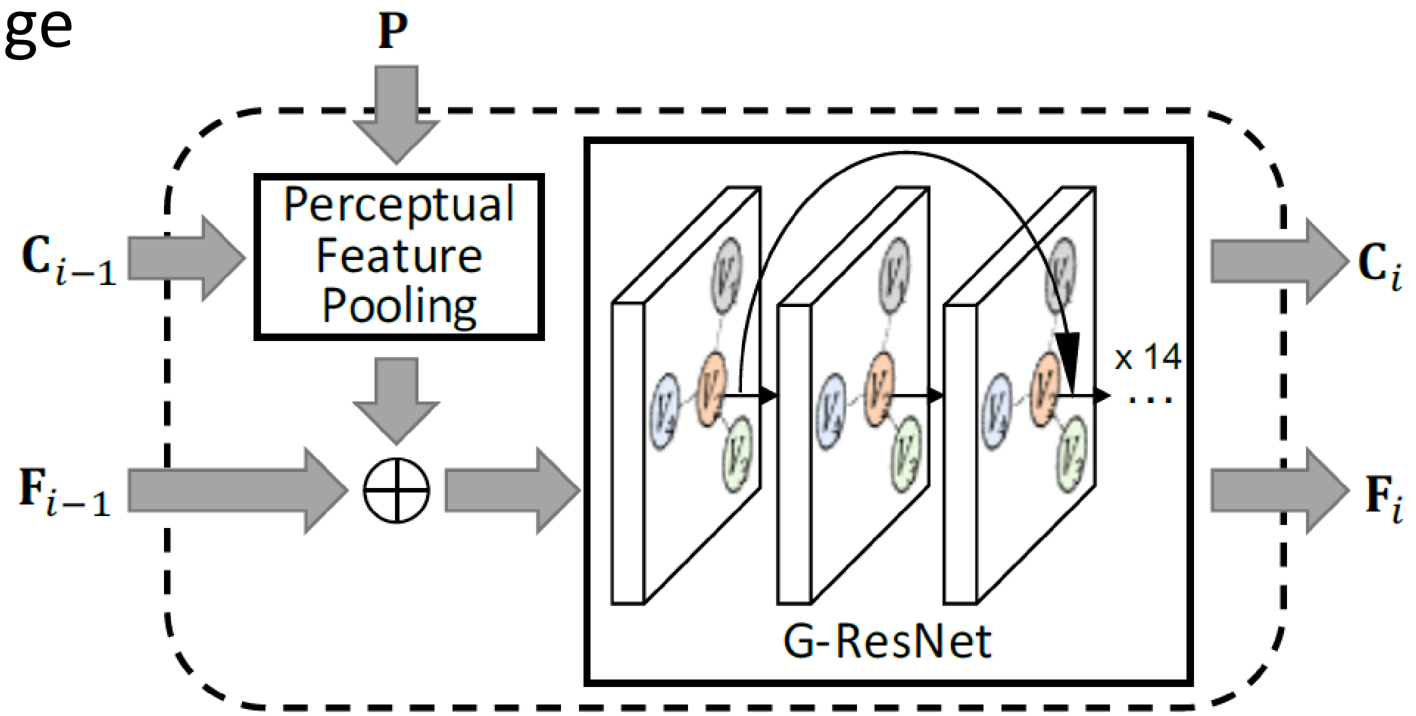
$$f_p^{l+1} = w_0 f_p^l + \sum_{q \in \mathcal{N}(p)} w_1 f_q^l$$

System Overview



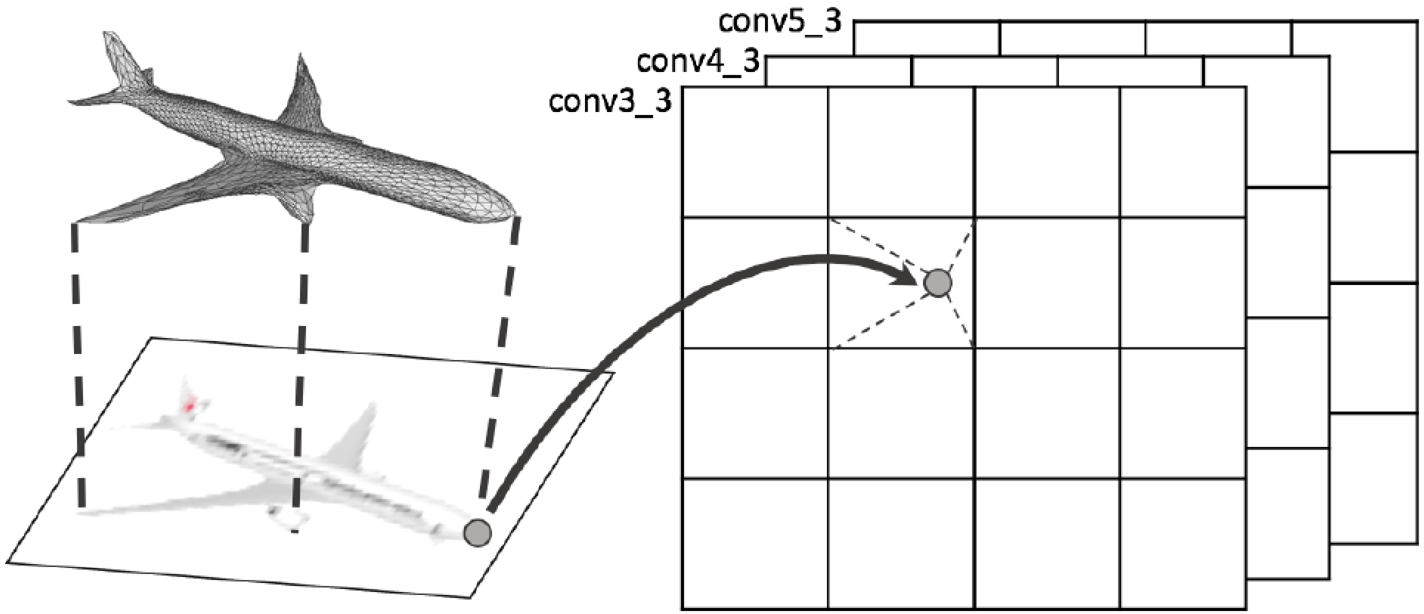
Mesh Deformation Block

- Pool feature from input image
  - Image feature network
  - Location of vertex C_{i-1}
- Feed into G-ResNet to get
  - New coordinates C_{i}
  - New 3d shape feature F_{i}



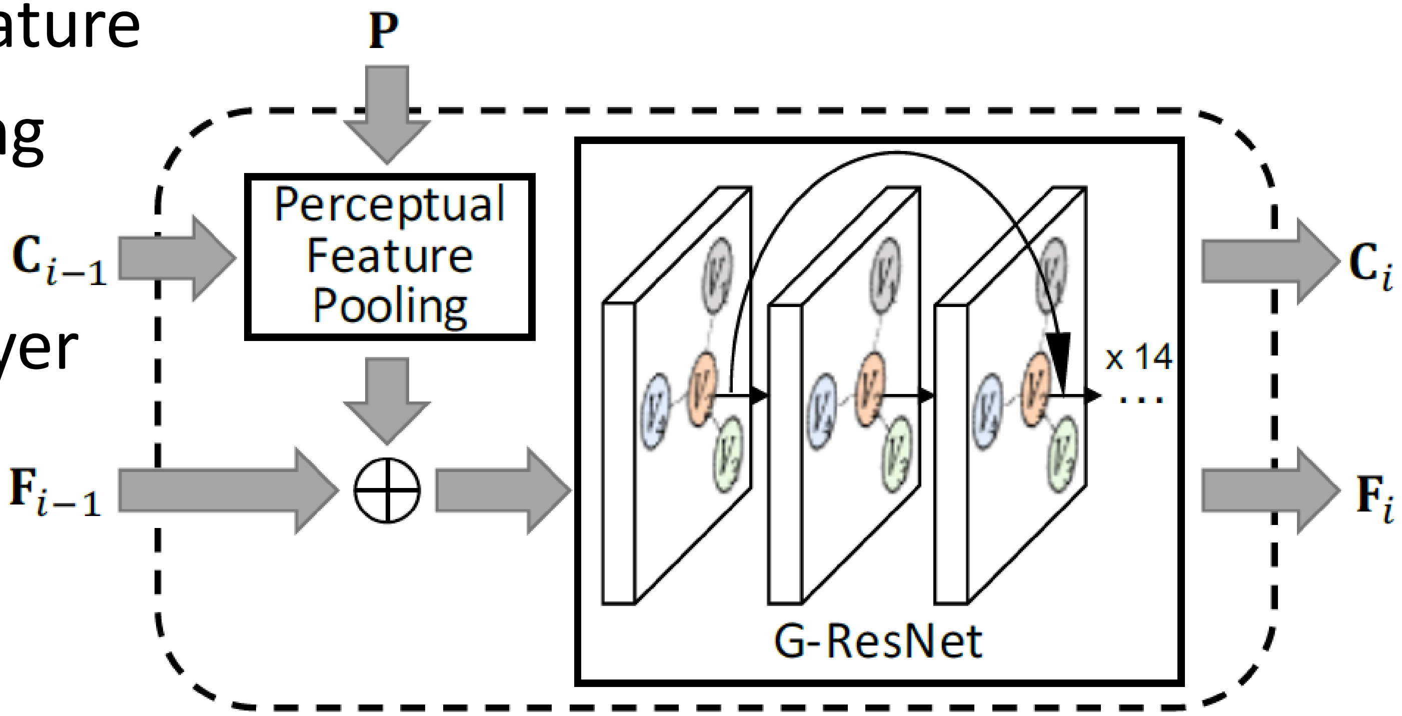
Perceptual feature pooling

- VGG-16 as image feature network
- Project 3D coordinate
- Pool the feature
- conv3_3,conv4_3,conv5_3



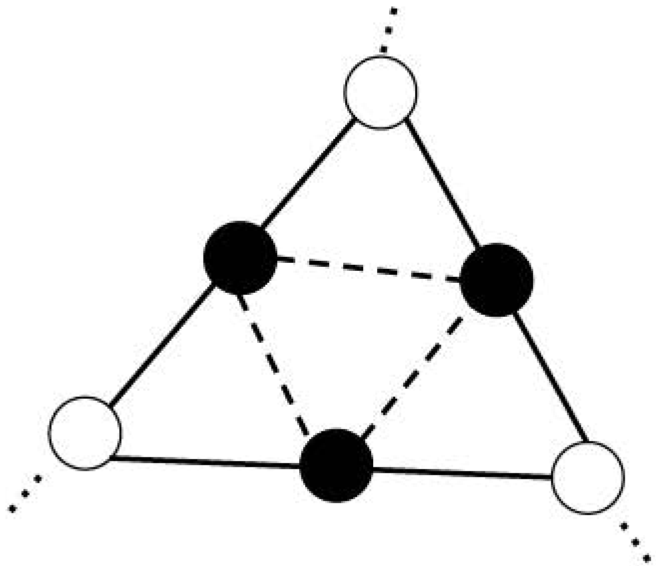
G-ResNet

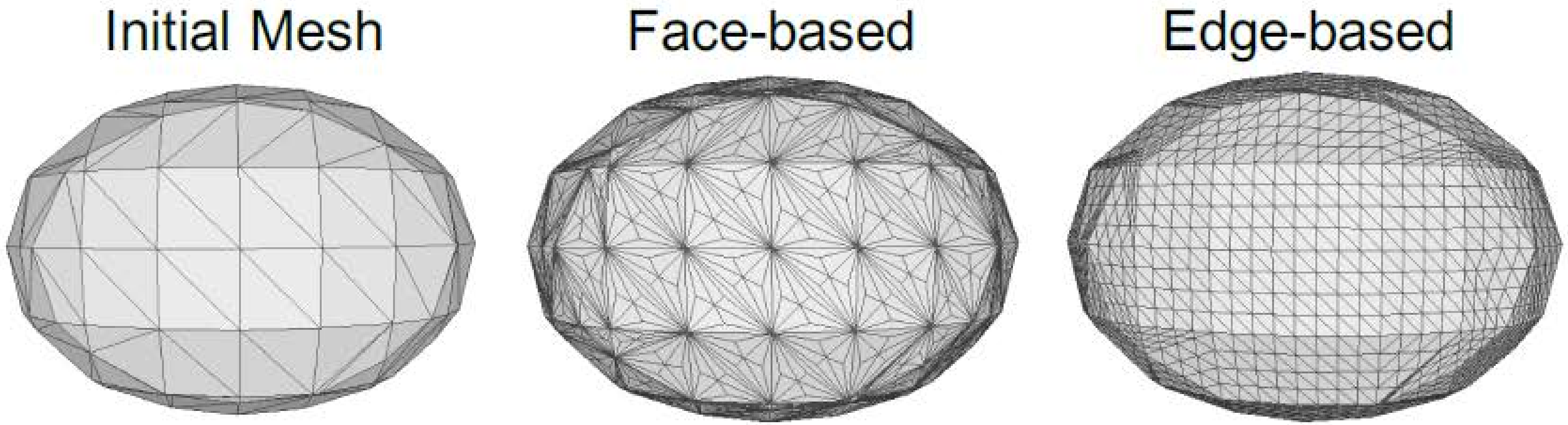
- Predict new location and feature
- Solve information exchanging
- Shortcut connection
- 14 residual convolutional layer
- 128 channel each layer



Graph Unpooling Layer

- Increase the number of vertex
- Edge-based, since face-based causes imbalanced vertex degrees
- Add vertex at center of each edge and average the features
- Connect vertices if they are added on the same triangle





Losses

- Chamfer loss
  - Constrain the location of mesh vertices
  - Measure the distance of each point to the other set as:

$$l_c = \sum_p \min_q \|p - q\|_2^2 + \sum_q \min_p \|p - q\|_2^2$$

- Normal loss
  - Enforce the consistency of surface normal
  - q is the closest vertex for p that is found when calculating the chamfer loss

$$l_n = \sum_p \sum_{q=\arg\min_q(\|p-q\|_2^2)} \|\langle p - k, \mathbf{n}_q \rangle\|_2^2, \text{ s.t. } k \in \mathcal{N}(p)$$

Regularization

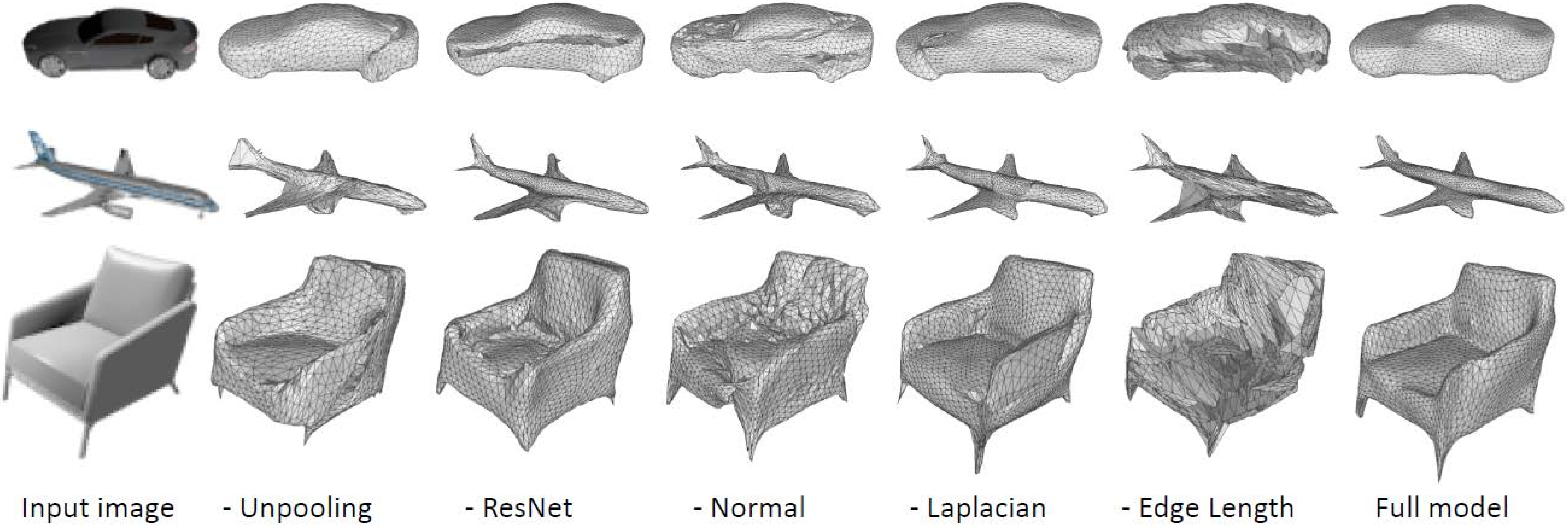
- Laplacian regularization
  - Maintain relative location between neighboring vertices
  - Prevent from moving too freely

- Edge length regularization
  - Penalize flying vertices

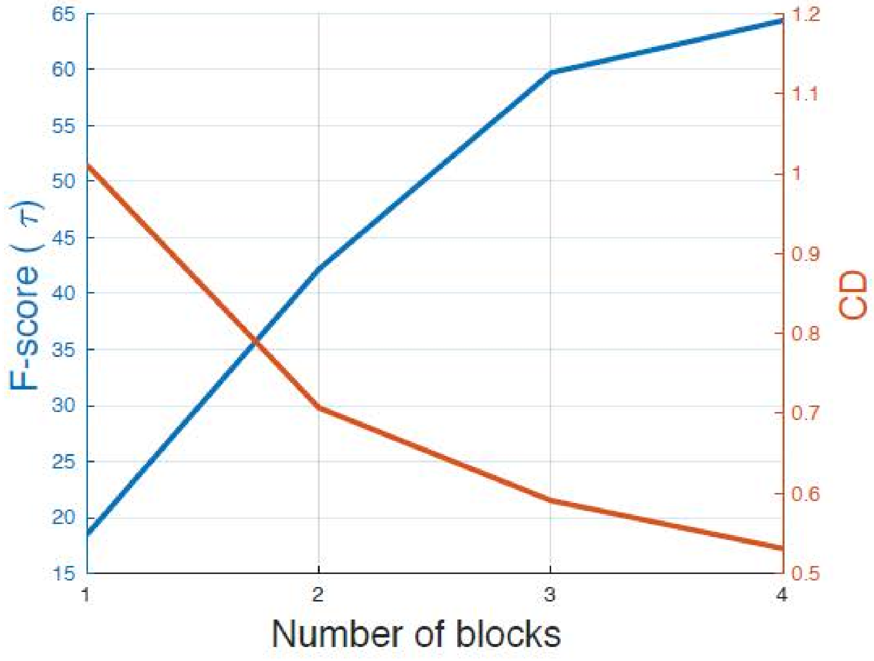
Experiment

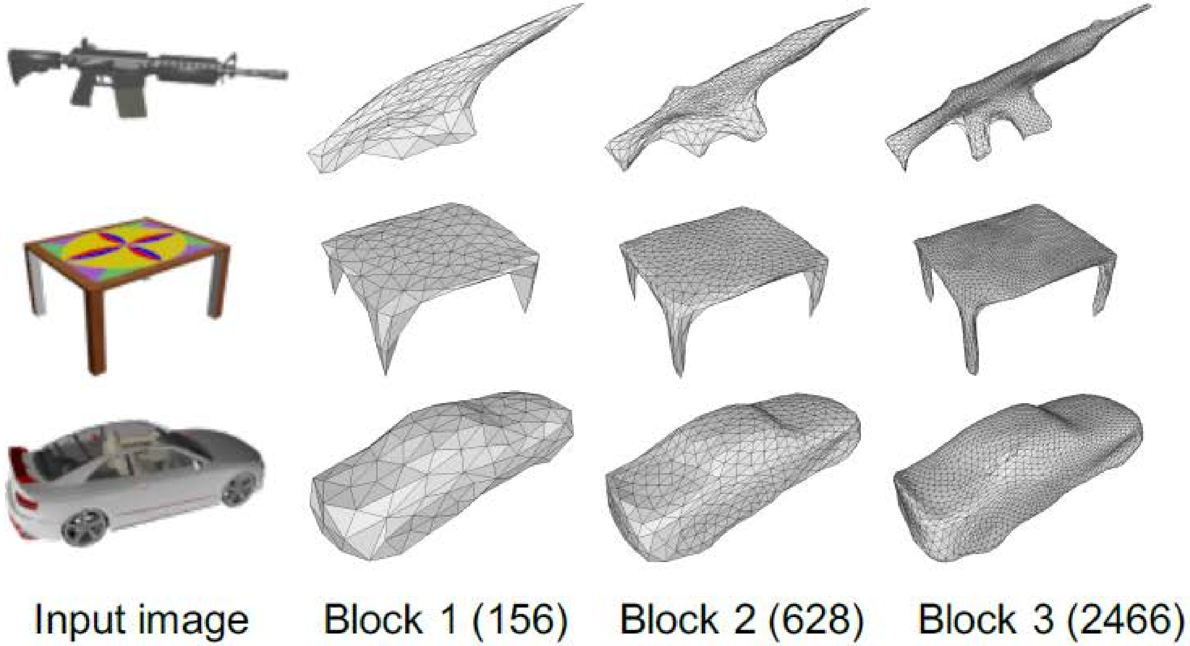
- Data: Synthetic image created from ShapeNet
- Input size 224*224
- Initial ellipsoid, 156 vertices and 462 edges
- Training: 72 hours on one Nvidia Titan X
- Testing: 15.58ms generate mesh with 2466 vertices

Ablation Study



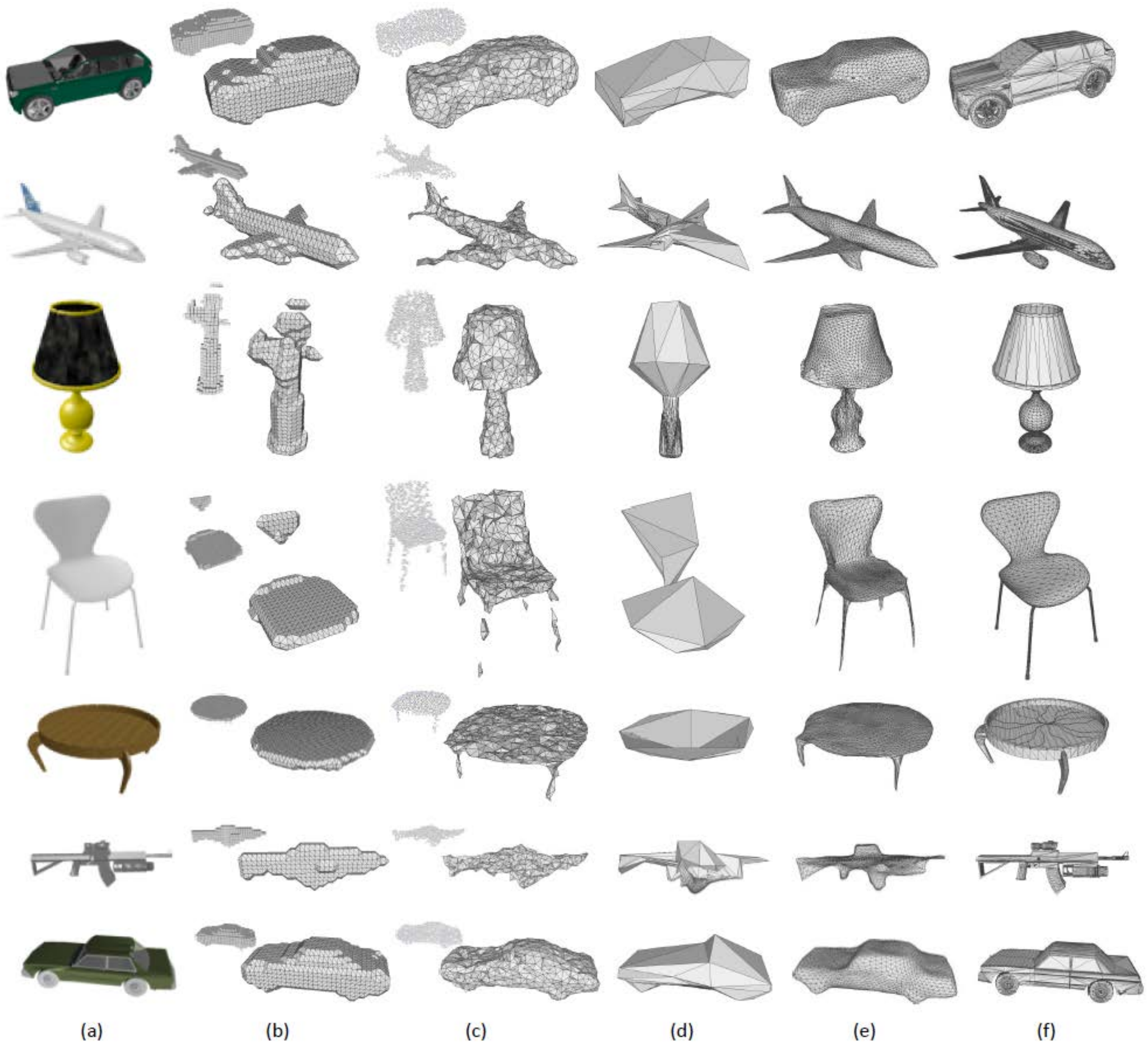
Ablation Study





Reconstruct Real-World images

Comparison



Discussion

- Camera parameters
- Multi-view inputs
- More operations on graph