Research Project Proposal:
3D object reconstruction by shape priors

Cristian Sbrolli
cristian.sbrolli@mail.polimi.it
CSE

POLITECNICO MILANO 1863

HONOURS PROGRAMME HP-SR in Information Technology

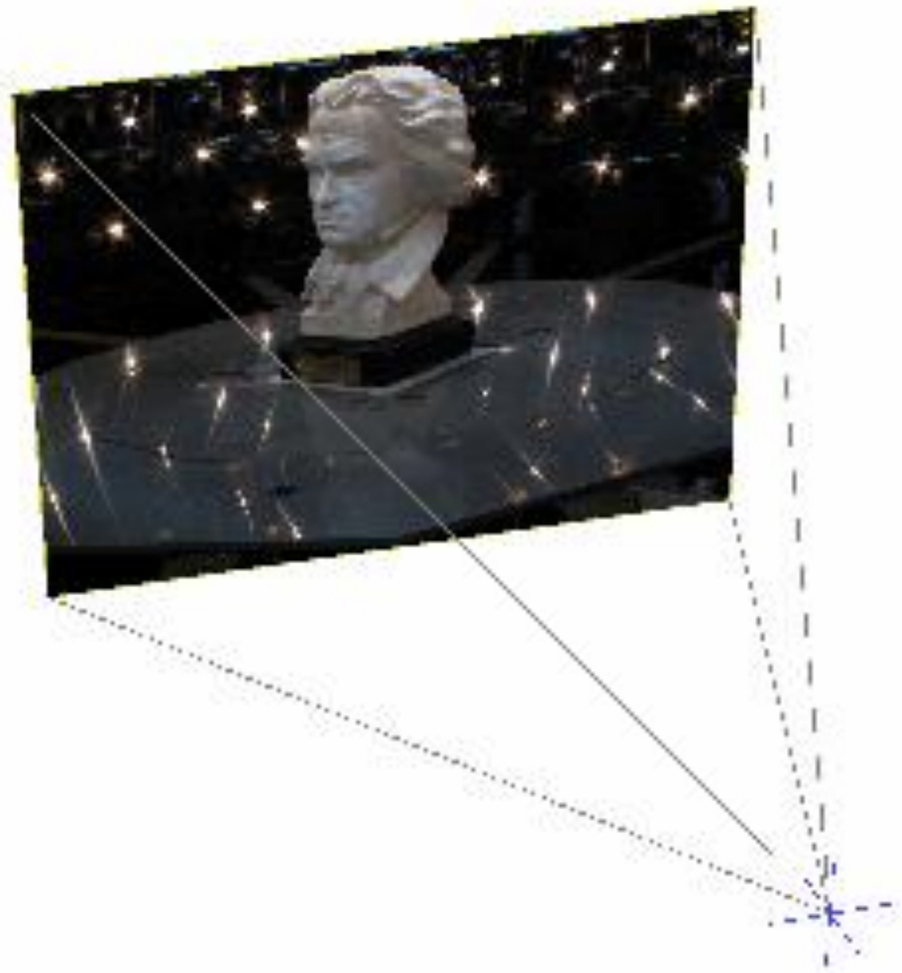
Introduction

- What is 3D object reconstruction?

Input*
Images of an object
$\mathbf{I} = \{I_k, k = 1..n\}$ $n >= 1$

Predictor $f(\vartheta)$

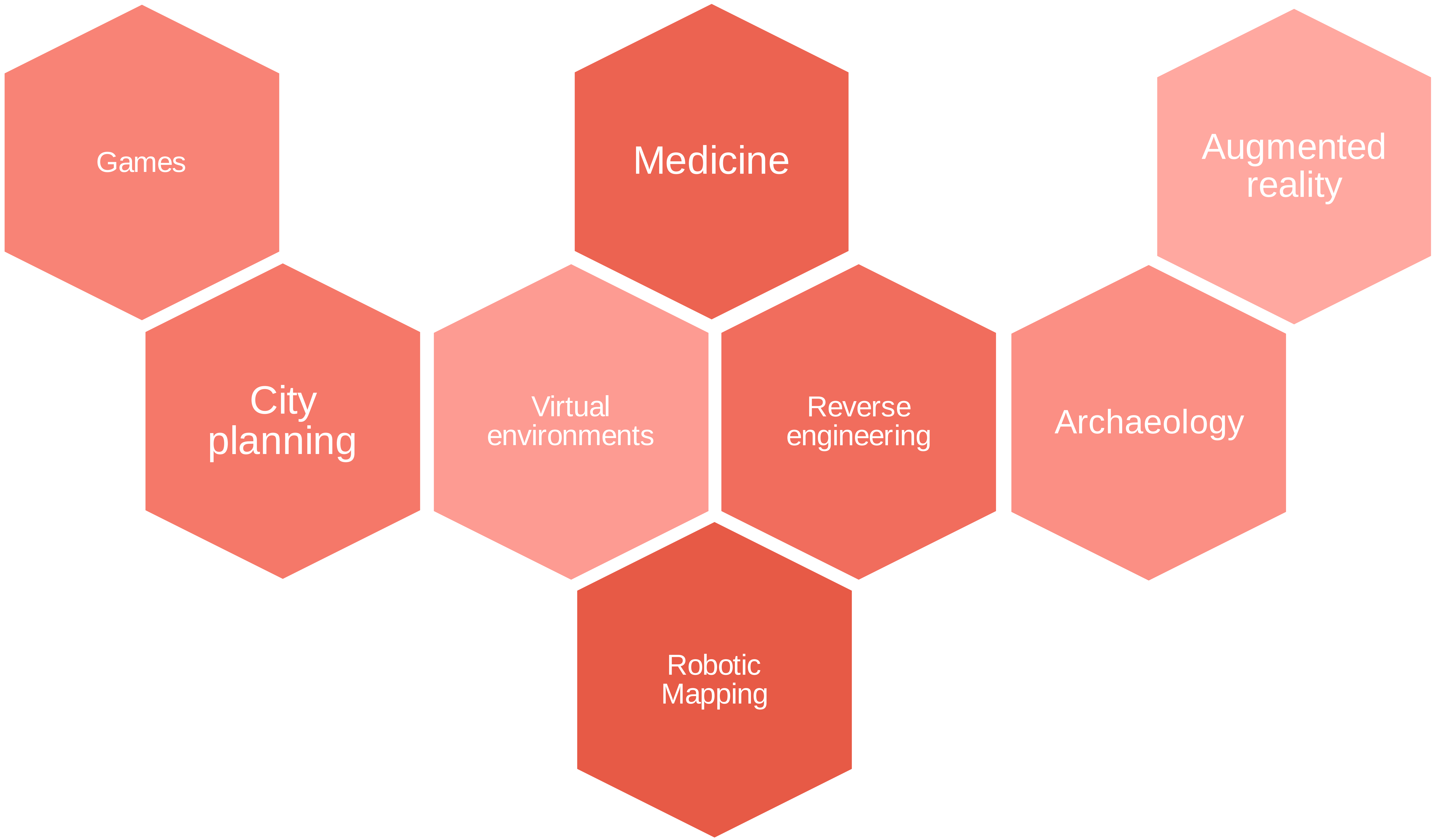
Output
Predicted 3D Shape of
the represented object $\overline{S}$



*Inputs can also be 3D representations as point clouds

Introduction

- Why is it important?



Introduction

Classical approaches
geometric perspective, model 3D to 2D process to solve the inverse problem

× Require multiple images
× Calibrated cameras
× Feature engineering

- How are humans good at this task, even with only one image?

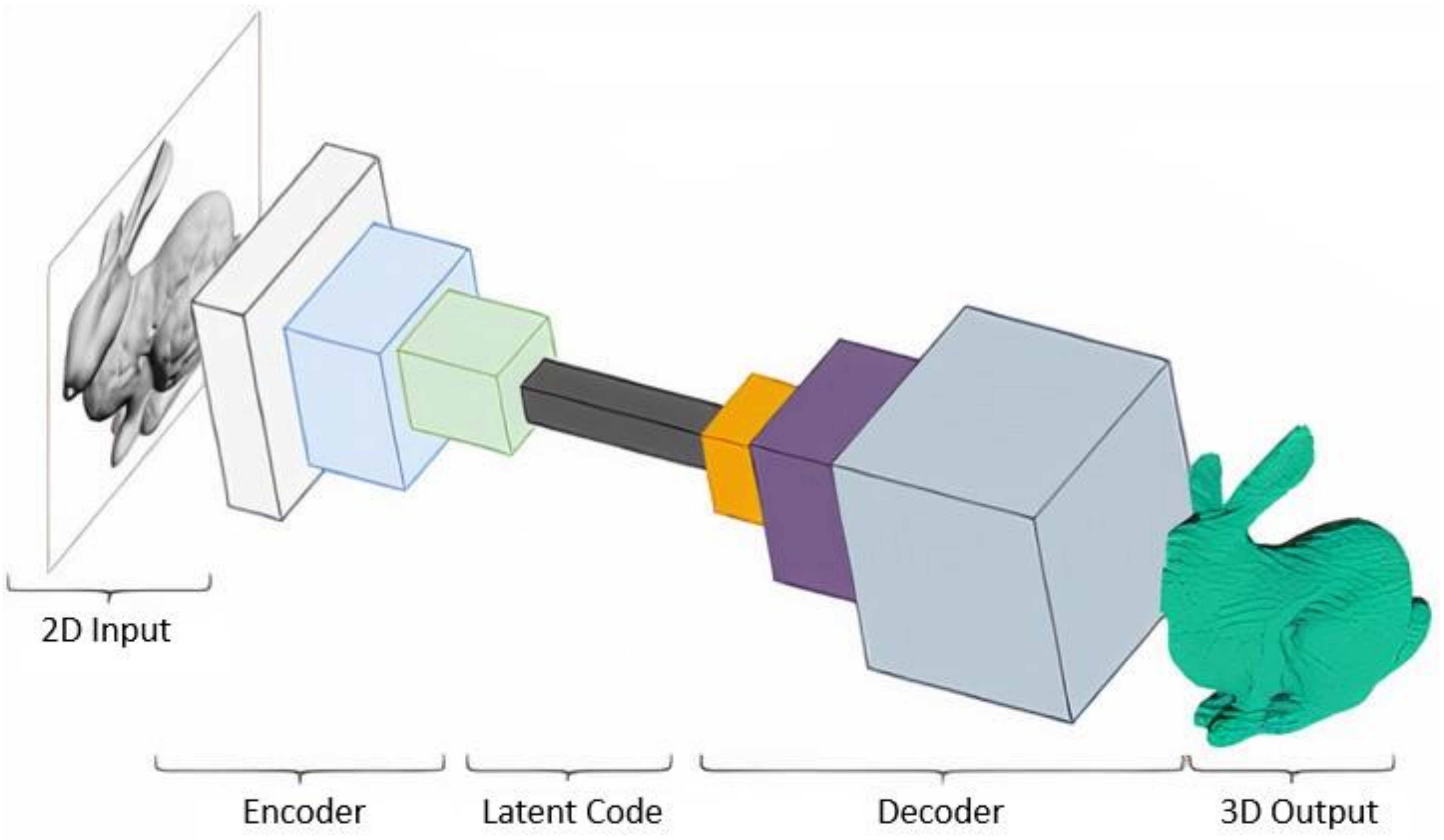
Exploit learnt knowledge about shapes

Deep Learning approaches

Introduction

Deep Learning Approaches
Feature learning and knowledge building

- ✓ Impressive performance even with single view
- ✓ No need of calibrated cameras
- ✓ Feature learning
- × Require large amounts of data
- × Generalization issues to address



Introduction

- What if we use only one image?

➔ 3D information loss
➔ Problems aggravated by single view reconstruction:

× **Unobserved views (Occlusion)**

× **Noisy backgrounds**

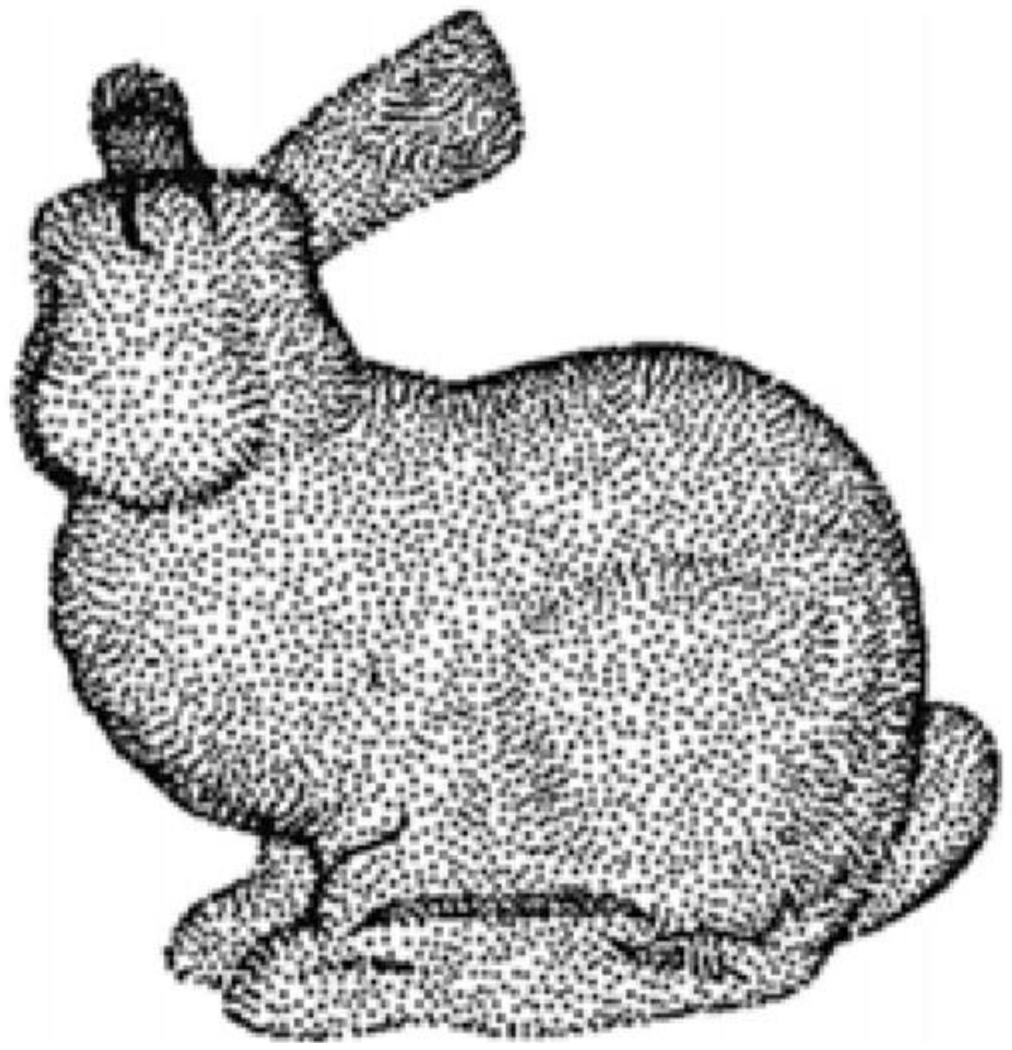
Preliminaries

- How do we represent 3D shapes?

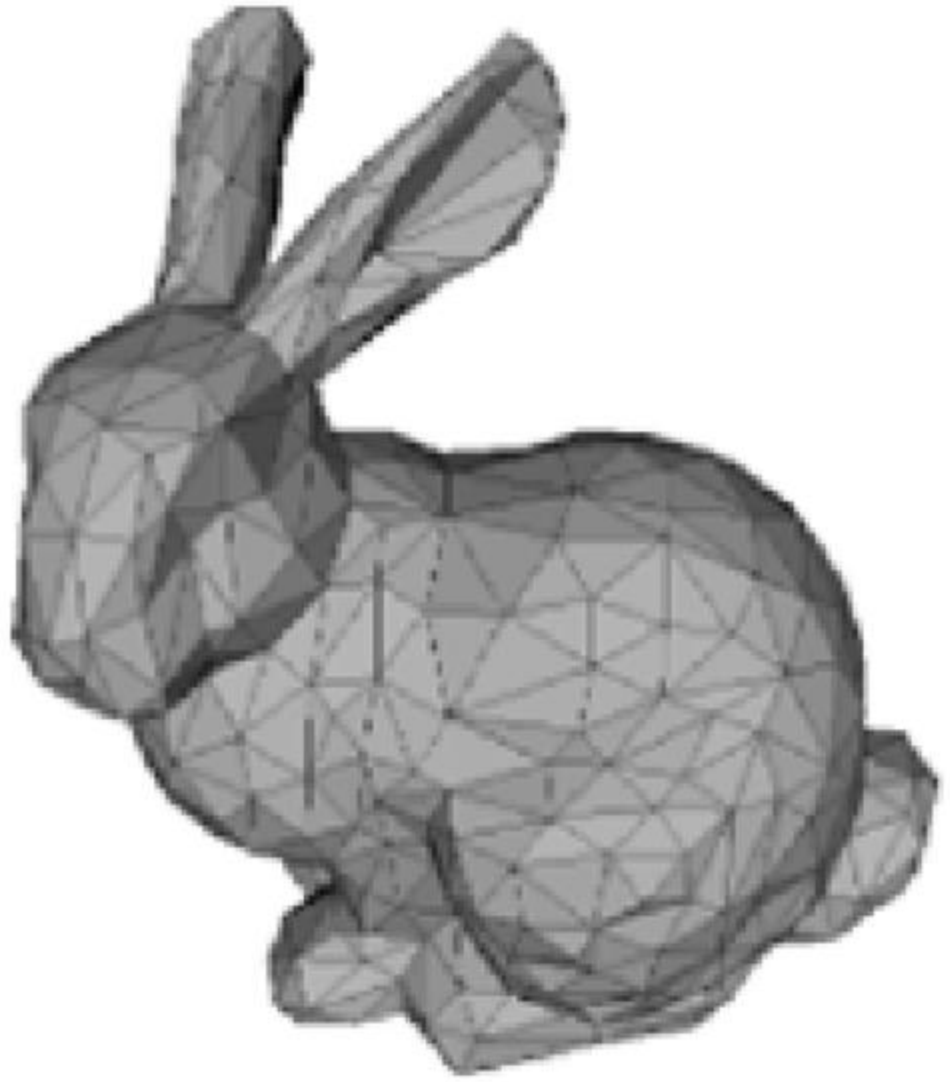
✓ Relatively easy to collect
✓ Exact representation
× Often not directly used
× Do not model connectivity

✓ Easy to render and transform
✓ Computers optimized for it
× Curved objects approximated
× Don't hold up in all resolutions

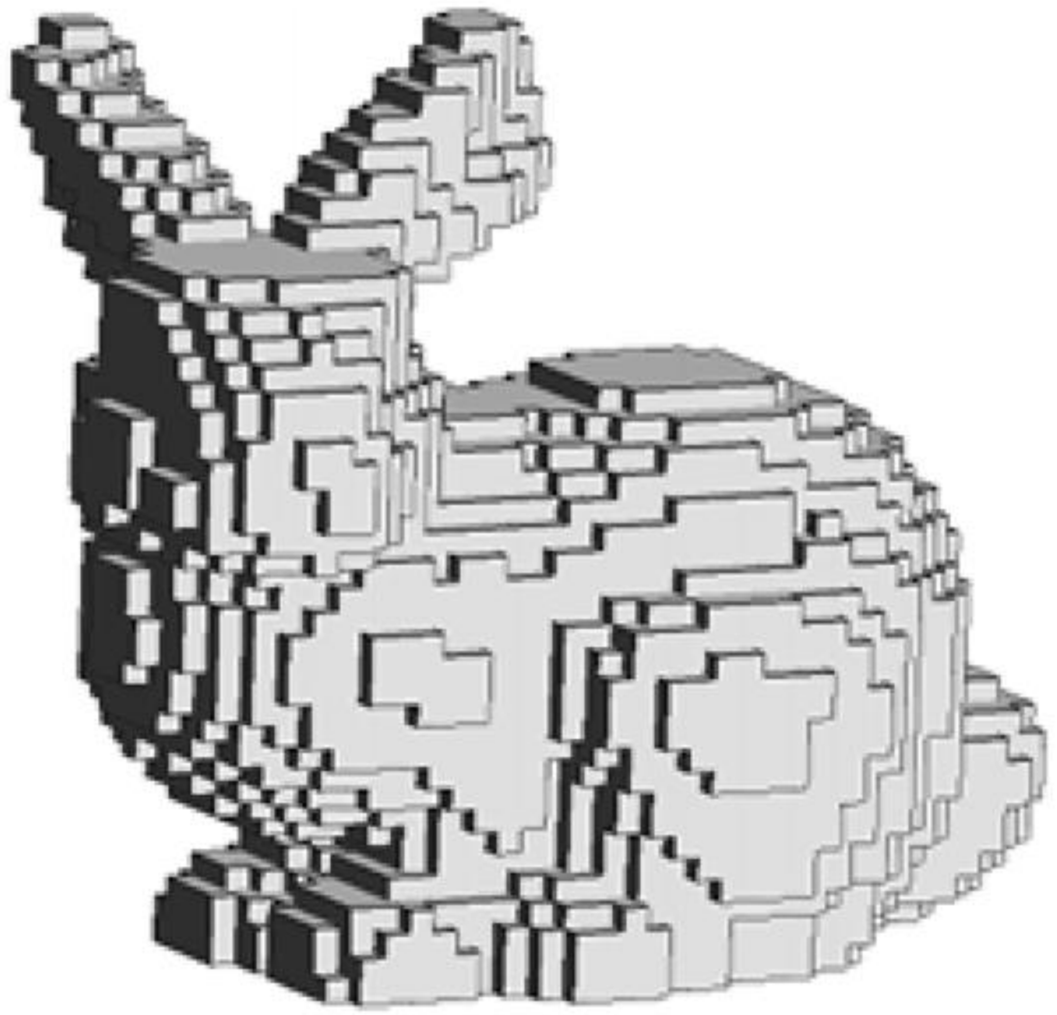
✓ Reflect real world composition
✓ Can have high resolutions
× Memory consumption
× Manhattan world bias



Point cloud



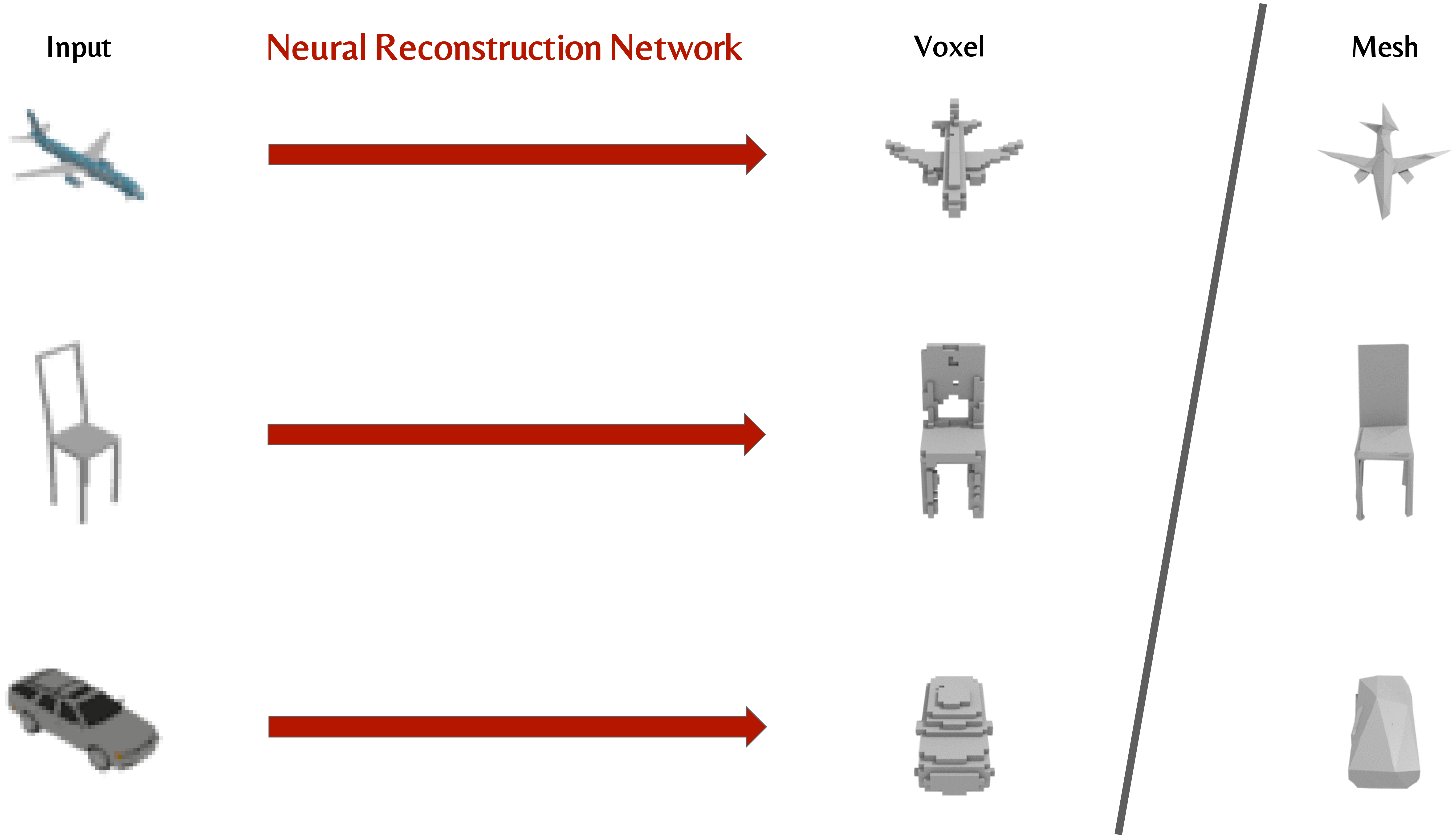
Polygon Mesh



Voxel

Preliminaries

- Different representations examples



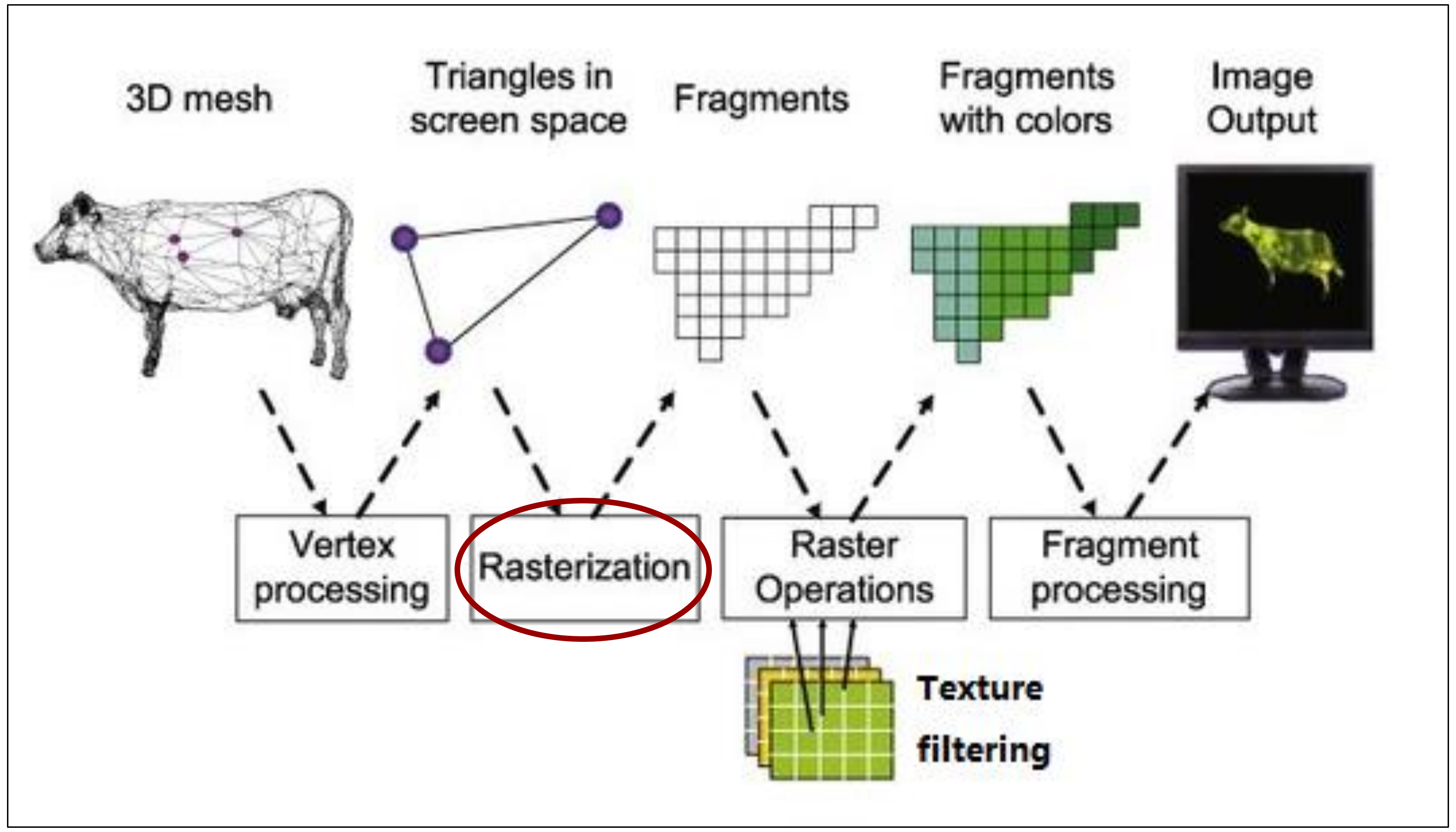
Preliminaries

- What is differentiable rendering?

"Rendering is the process of generating an image from a 2D or 3D model by means of a computer program"

Is it possible to perform automatic differentiation through it?

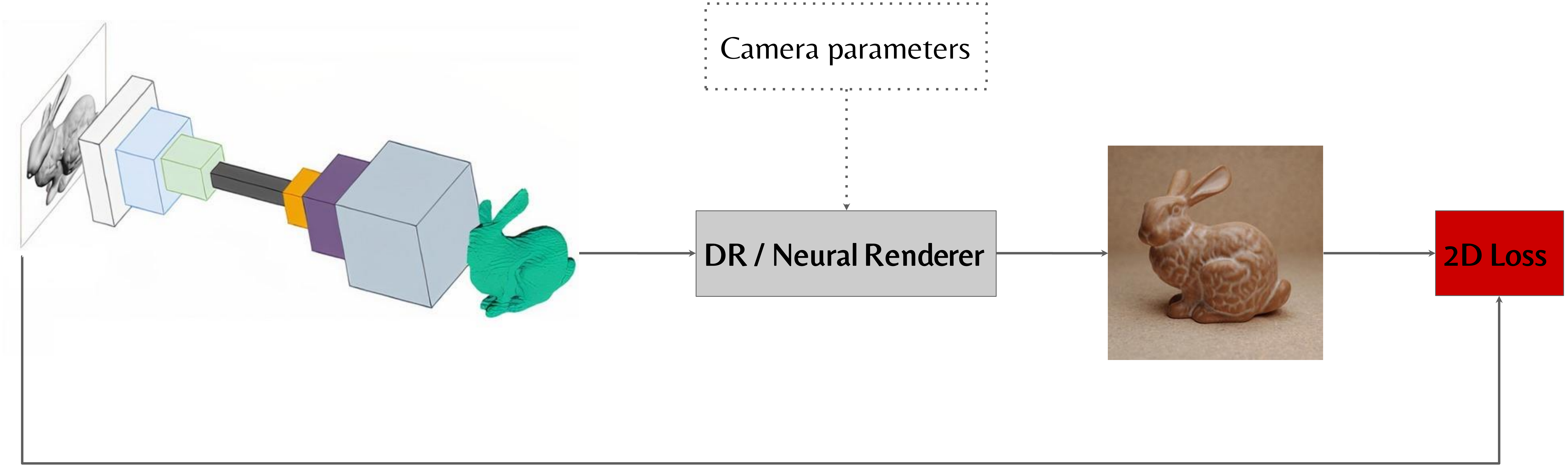
➔ Approximate forward pass (Soft Rasterizer)
➔ Approximate backward pass (OpenDR, NMR)



Alternative approach → Neural Rendering: Learn the rendering process from data

Preliminaries

- How to exploit rendering in reconstruction?

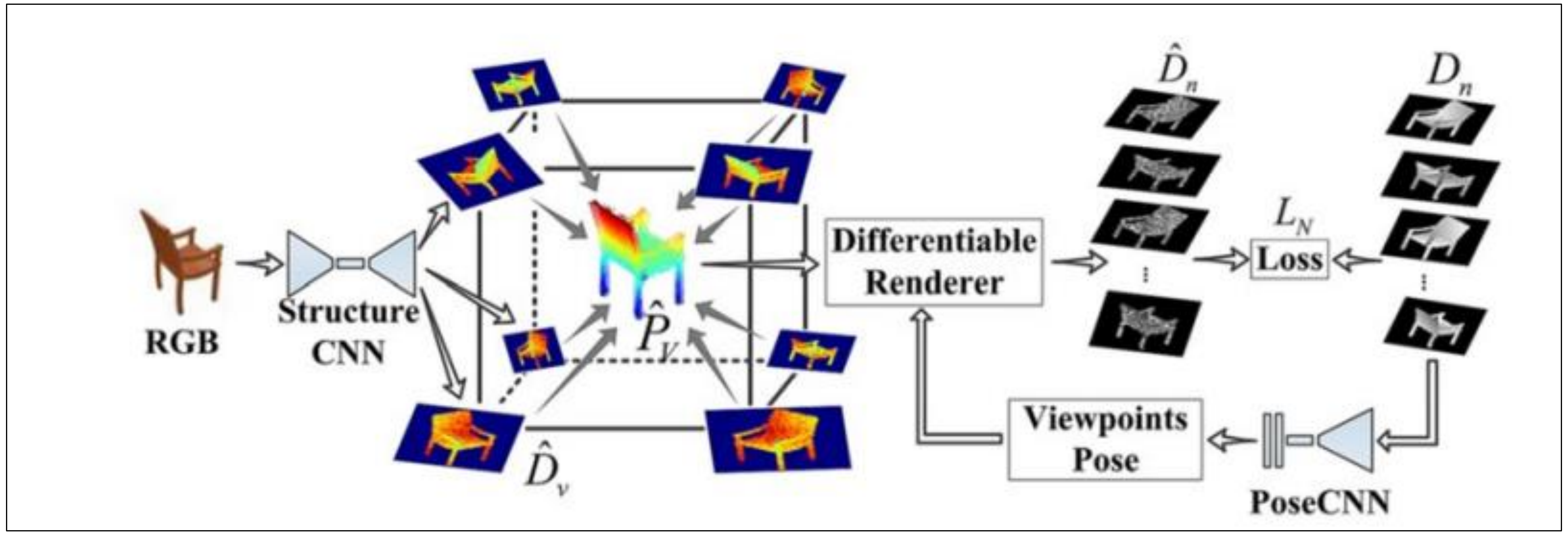


Advantages of 2D annotations w.r.t. 3D annotations:

✓ Collecting 2D data is easier and less costly
✓ Labelling accurately 2D data is easier
✓ Allows self-supervision

Related Works

- A first model exploiting what we just discussed



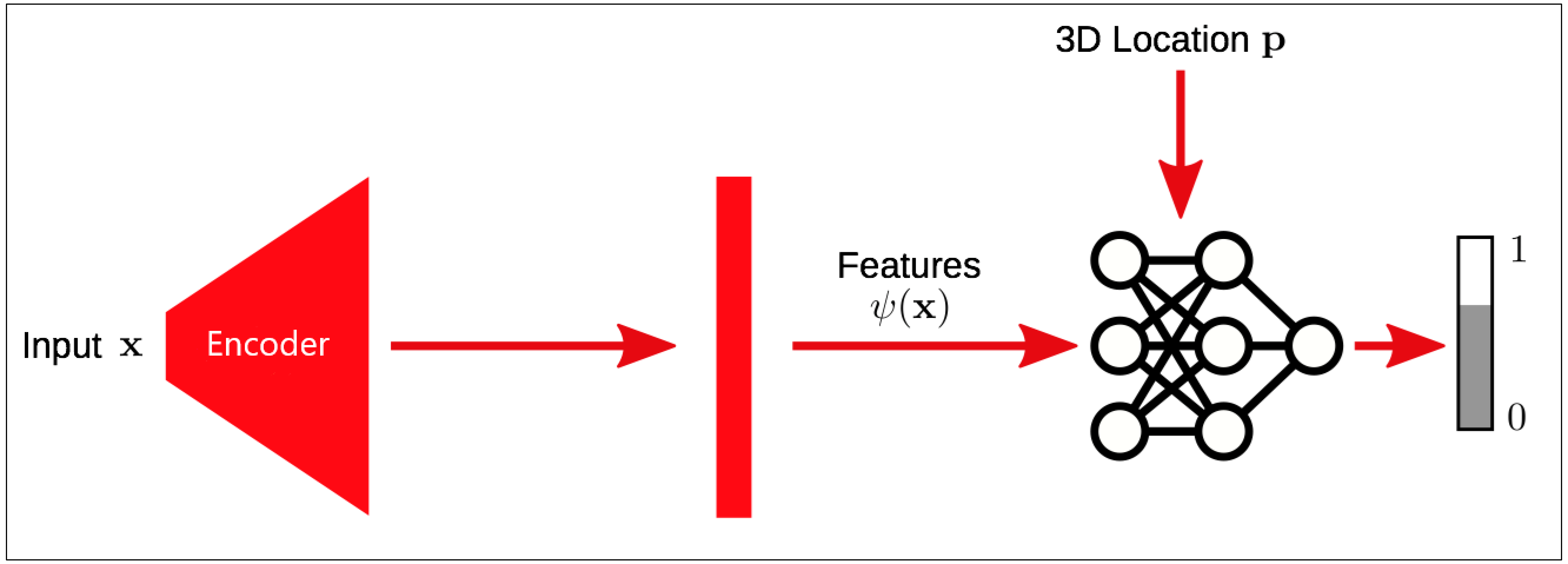
➔ Predicts multiple views to estimate 3D point cloud representation
➔ Differentiable renderer allows depth maps as targets
➔ Training first on syntethic dataset, then on wild images

Jin et al. "Weakly-Supervised Single-view Dense 3D Point Cloud Reconstruction via Differentiable Renderer"

Related Works

- A model exploiting an implicit 3D representation: Occnet

➔ Learn an occupancy function assigning occupancy probability to an input 3D point
➔ Training: sample the GT volume + cross entropy loss
➔ Inference uses an algorithm to extract 3D model



✓ Allows different input representations by changing the encoder
✓ Potentially allow infinite resolution

Ansari et al. "Occupancy Networks: Learning 3D Reconstruction in Function Space"

Related Works

- Occnet successor: D-Occnet

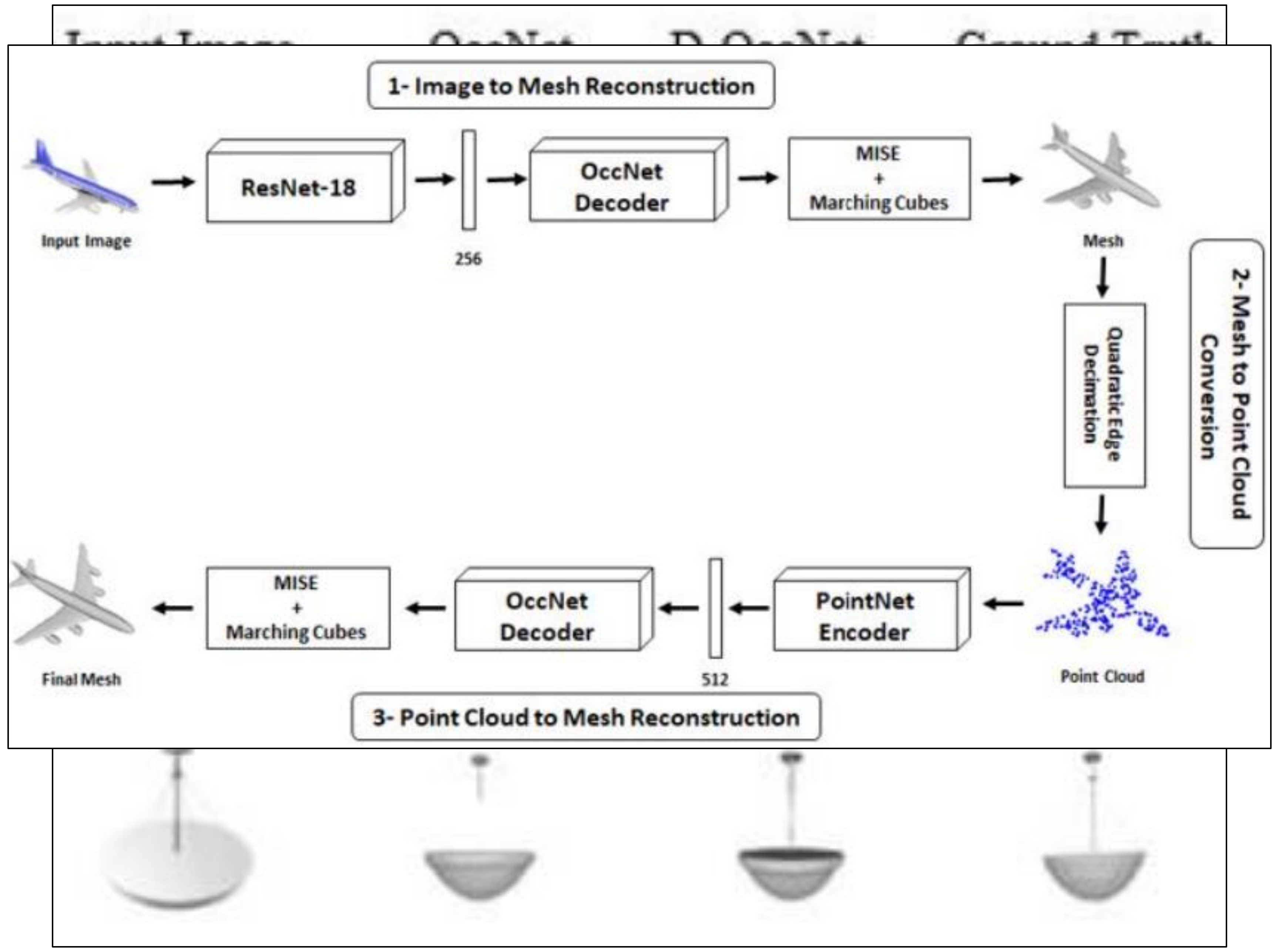
➔ OccNet lacks 3D info
➔ Extend by connecting 2 OccNet together
➔ Effectively add 3D information

OccNet

Image ⟶ Mesh

D-OccNet

Image ⟶ Mesh ⟶ Point Cloud ⟶ Mesh

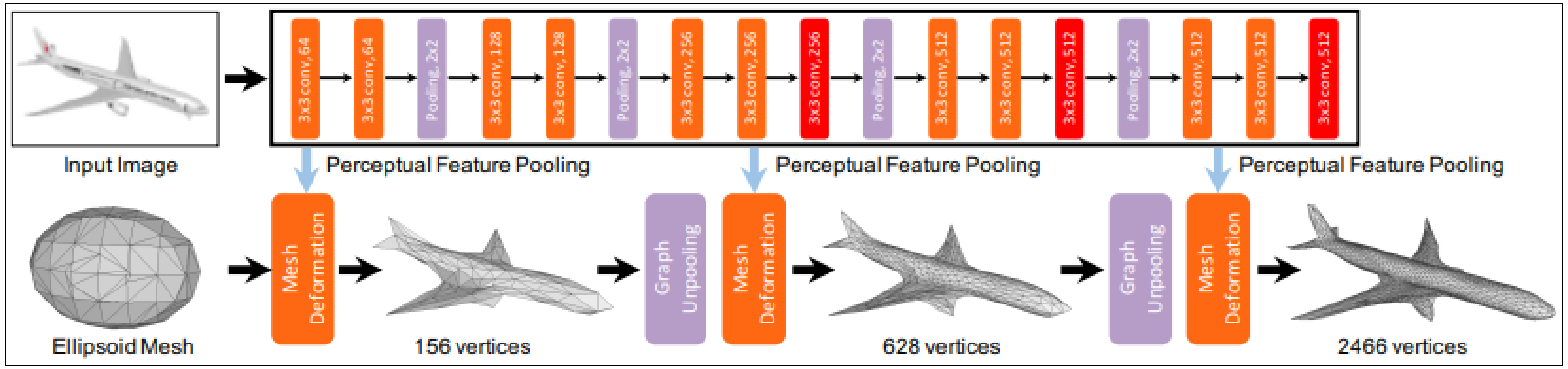


Ansari et al. “D-OccNet: Detailed 3D Reconstruction Using Cross-Domain Learning”

Related Works

- Progressively deforming a predefined shape: Pixel2Mesh

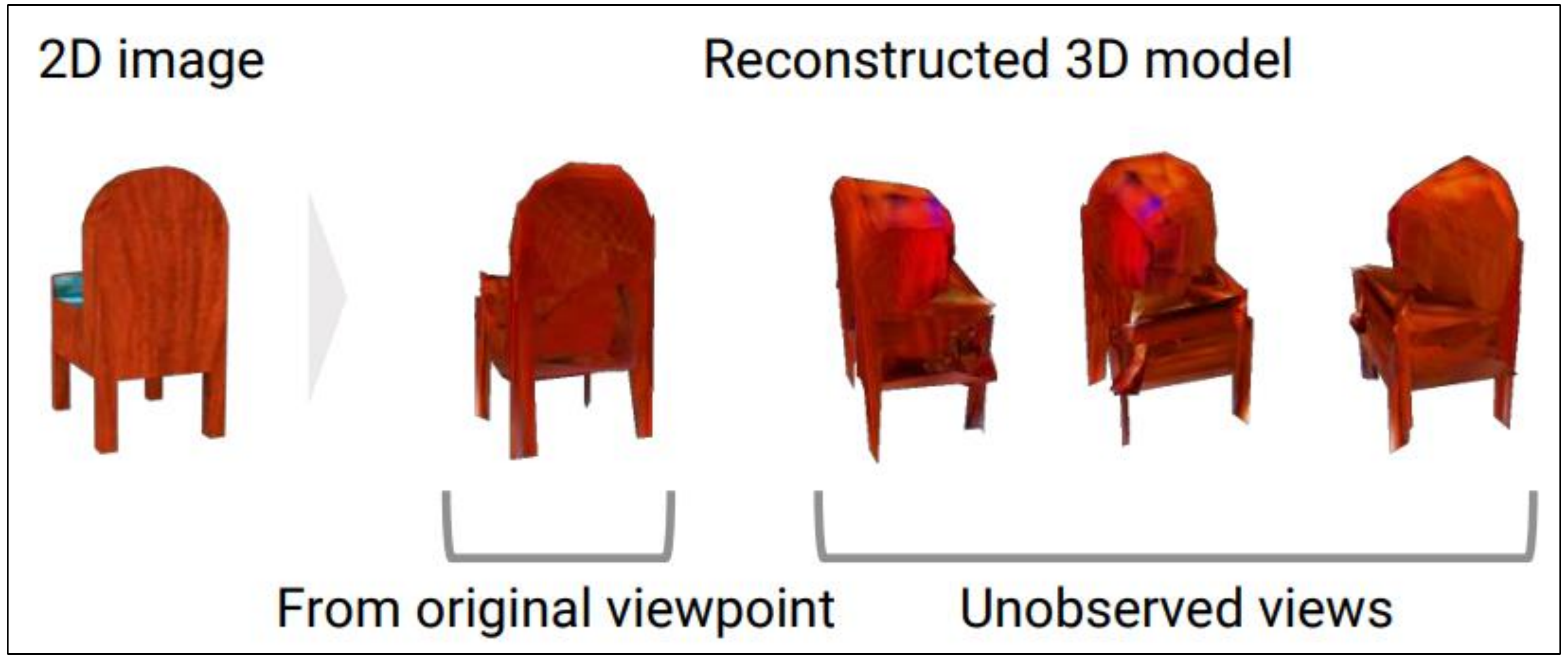
➔ Pre-defined ellipsoid mesh
➔ Image feature network + Mesh deformation network
➔ Mesh deformation through Graph-based Convolutional Neural Network
➔ Progressively add vertices to increase the capacity of handling details



Wang et al. “Pixel2Mesh: Generating 3D Mesh Models from Single RGB Images”

Related Works

- Still, lots of models struggle with the reconstruction of unobserved views:

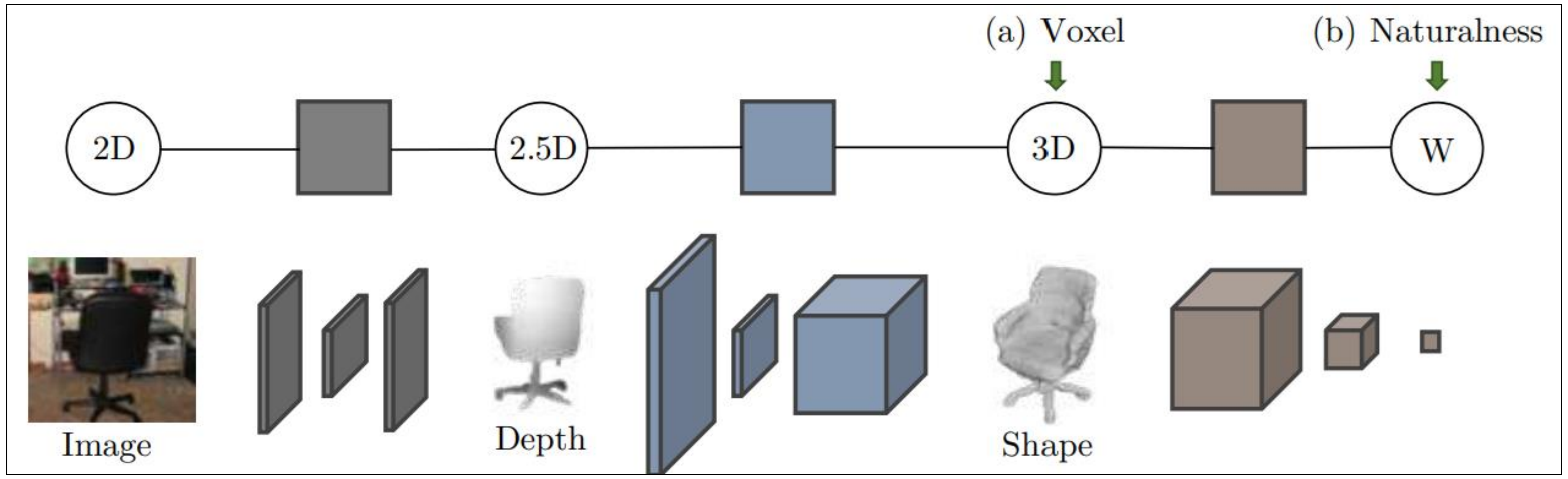


Can we leverage or learn prior shapes?

Related Works

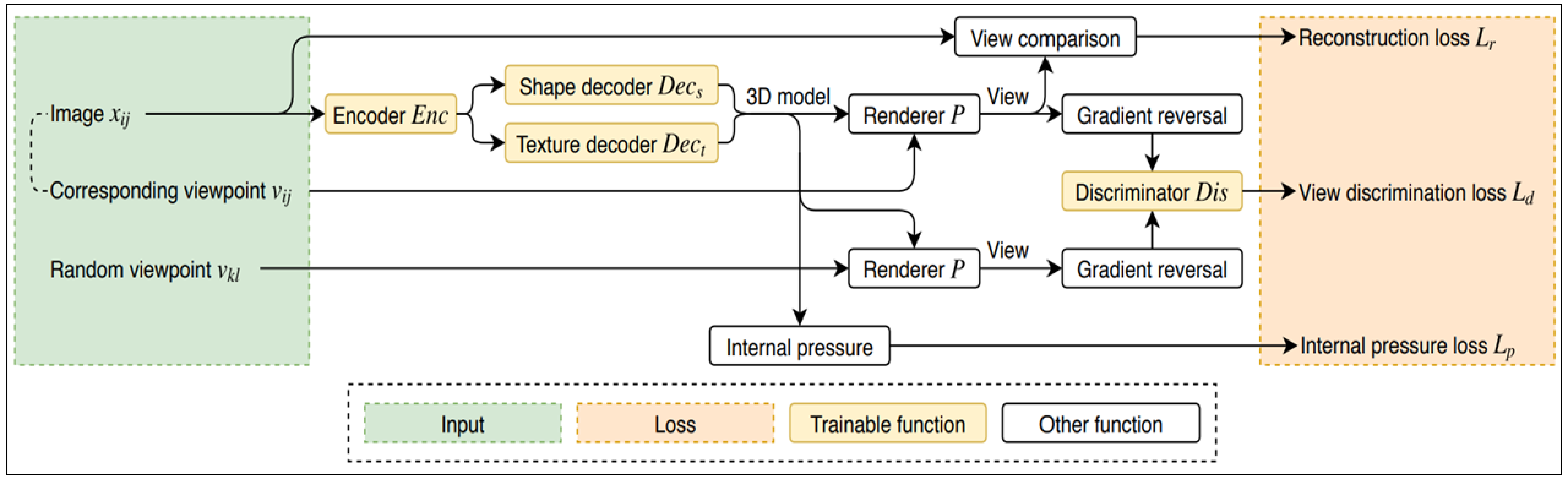
- Adversarial models implicitly learning shape priors:

➔ Penalize the model for unrealistic shapes
➔ Intermediate 2.5D sketches before 3D shape
➔ Pre-trained GAN, only discriminator is kept
➔ Adversarial task: discriminate natural shapes from unnatural ones



Wu et al. "Learning Shape Priors for Single-View 3D Completion and Reconstruction"

➔ Learn priors on 2D views
➔ Generate 3D mesh by moving the vertices of a pre-defined mesh
➔ DR to generate views of the reconstructed shape
➔ Adversarial task: recognize original vs novel views

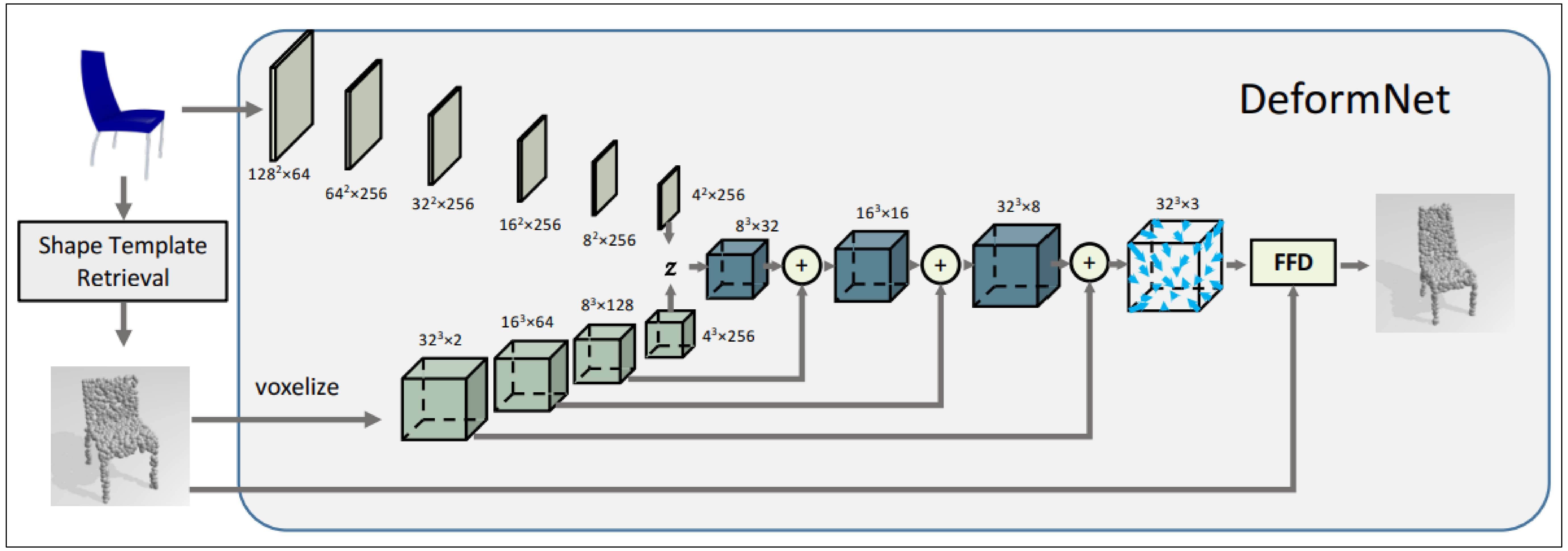


Kato et al. "Learning View Priors for Single-view 3D Reconstruction"

Related Works

- Exploiting a database of high-quality CAD models: DeformNet

➔ Search closest template in database leveraging metric learning
➔ Deform template by moving control points defined by a deformation layer
➔ Decoder output is the offset of the control points



Kuryenkov et al. “DeformNet: Free-Form Deformation Network for 3D Shape Reconstruction from a Single Image”

Related Works

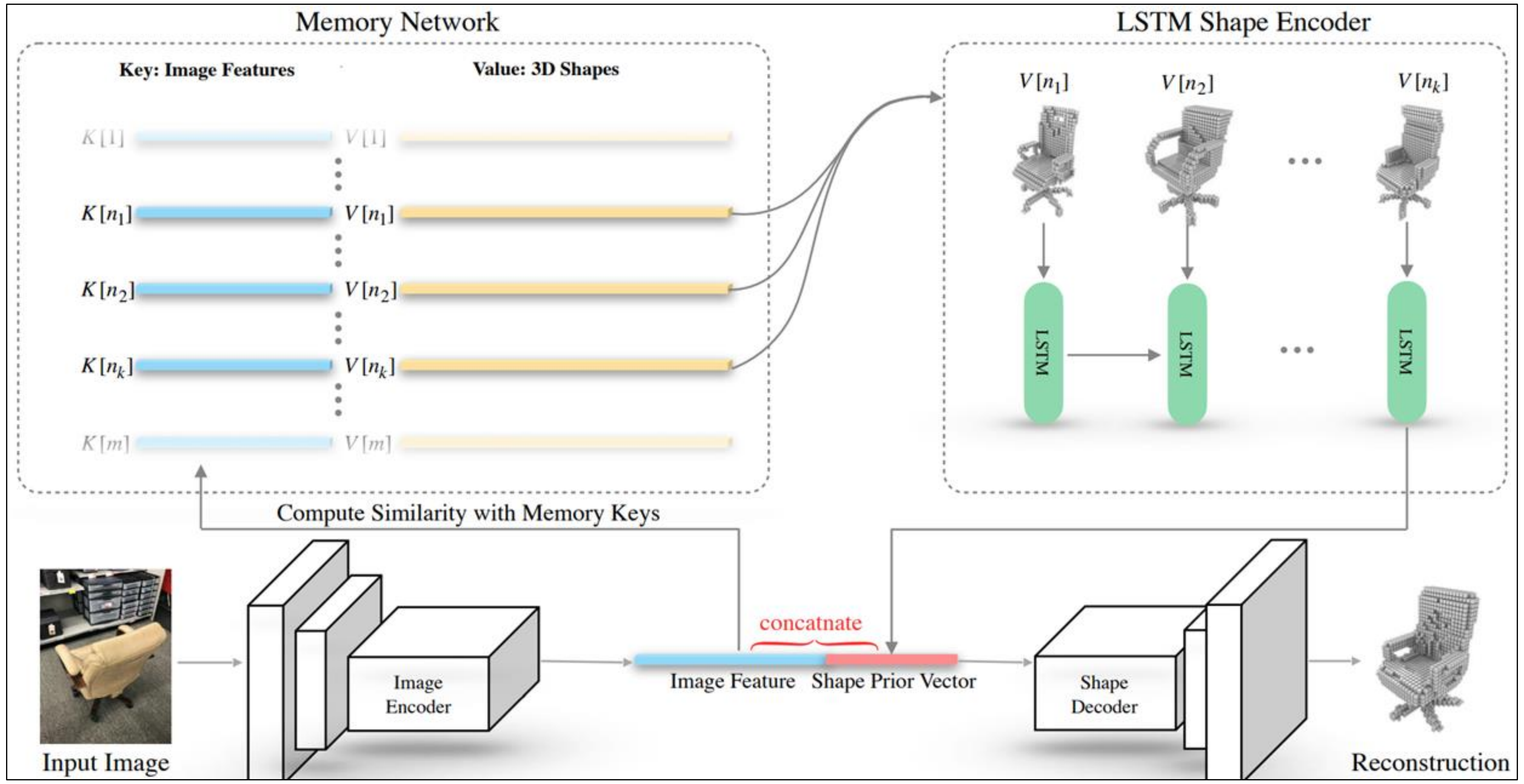
- Memory network storing prior shapes

➔ Memory triplets «K,V,Age»:
  - ◆ K: Image features
  - ◆ V: Voxel shape (GT volumes)
  - ◆ Age: alive time since last successful match

➔ Matching through key similarity

➔ Writing through value similarity

➔ Recurrent network takes all matching priors to produce a shape prior vector



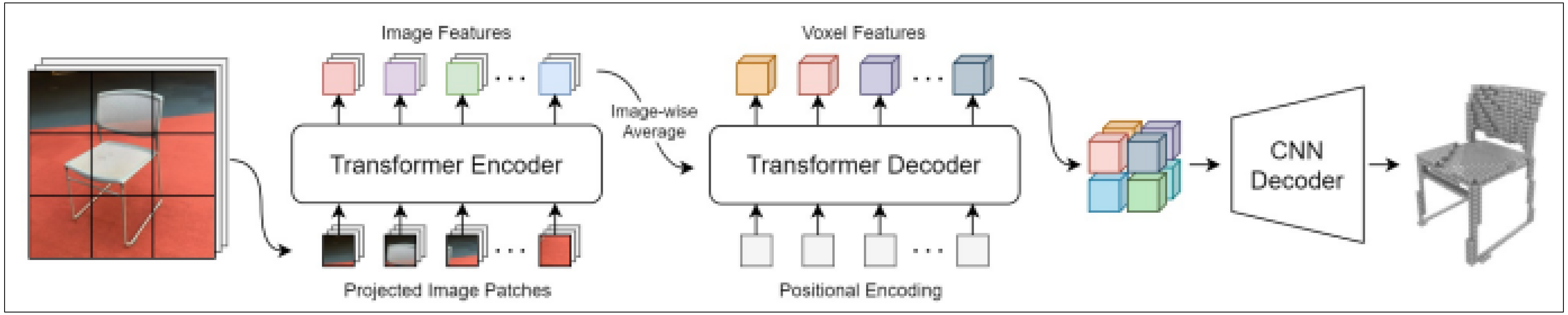
This architecture reminds attention mechanism...

Yang et al. “Single-View 3D Object Reconstruction from Shape Priors in Memory”

Related Works

- Transformer model for 3D reconstruction

➔ Adopts a ViT-like architecture encoding image patches
➔ Decoder processes all M^3 learnable positional encodings in parallel
➔ CNN decoder upsamples with 3D convs the voxel features



Shi et al. "End-to-End Single and Multi-View 3D Reconstruction with Transformers"

Summing up

- DR/NR can be used to train a 3D model on 2D annotations
- Multi-step processing can be useful to progressively add information
- Prior shape knowledge can be exploited in different ways:
    - Knowledge of natural 3D shapes (Implicit)
    - Knowledge of natural 2D views (Implicit)
    - Single prior shape deformation (Explicit)
    - Multiple prior shapes combination (Explicit)
- Graph networks are effective with mesh deformation
- Transformer models can be used effectively and still quite unexplored

How can we combine and enhance this approaches and ideas?

Research Directions

Goal#1
Investigate novel approaches by leveraging shape priors and the new architectures

Challenges
- General architecture and the specific design of its parts
- How to represent priors
- How to use the priors
- Training paradigm
- Dataset(s) to use

Motivation
The previously mentioned works and ideas suggest new possibilities which are worth to explore

Research Directions

Goal#2
Analyze impact of different scale object reconstruction and the possibility of performing scene parsing by parametrized shape priors

Challenges
- Dataset to use
- Metrics have to be changed
- Model architecture

Motivation
Sometimes we are not interested in reconstructing exactly the scene, but to "reproduce" it objectwise by some existing models

Evaluation Metrics

Intersection over Union (IoU):

$$IoU(X', X) = \frac{\sum_i I(X_i > \epsilon) * I(V_i)}{\sum_i I(I(X_i > \epsilon) + * I(V_i))}$$

Chamfer Distance (CD):

$$d_{CD}(S_1, S_2) = \sum_{x \in S_1} \min_{y \in S_2} ||x - y||_2^2 + \sum_{x \in S_2} \min_{y \in S_1} ||x - y||_2^2$$

Earth Mover's Distance (EMD):

$$d_{EMD}(S_1, S_2) = \min_{\phi : S_1 \rightarrow S_2} \sum_{x \in S_1} ||x - \phi(x)||_2$$

Goal#1

We can use these metrics and directly compare to state-of-the-art-models

Goal#2

Using these metrics would not make much sense → We expect to modify them or use different ones

Research Plan

Two phases approach:

I. Build a baseline model to fully experience and understand the problem and its practical challenges

II. progressively refine the model by following the goal(s)