Topology-Aware 3D Reconstruction for Cable-Stayed Bridges

Reporter：Mr. Fangqiao Hu

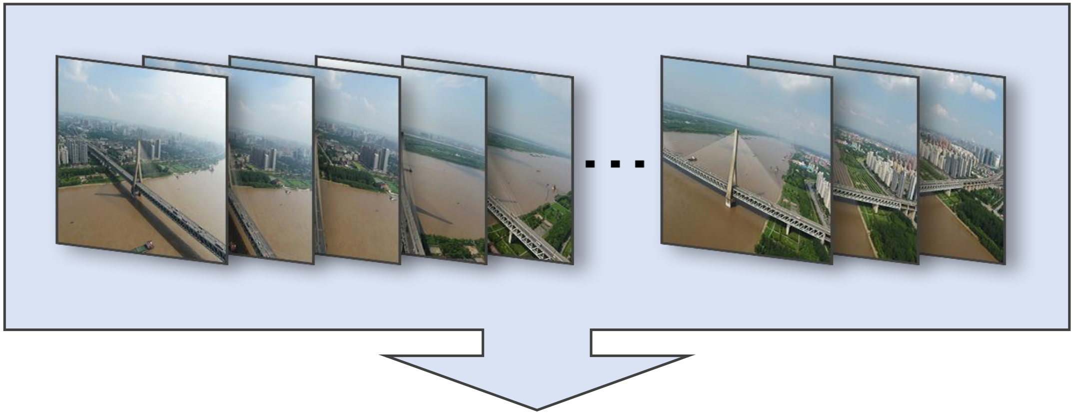
Advisor：Prof. Hui Li

School of Civil Engineering

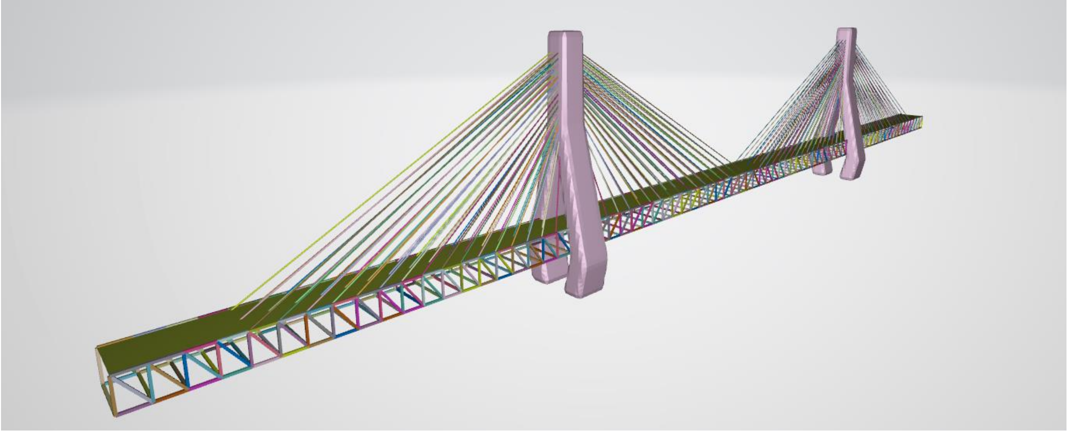
Harbin Institute of Technology

Objective

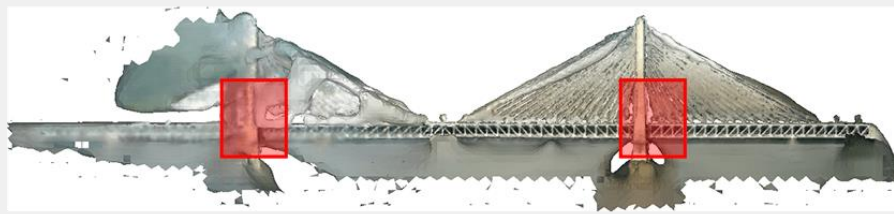
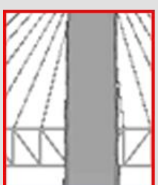
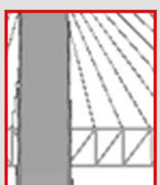
Multi-view images



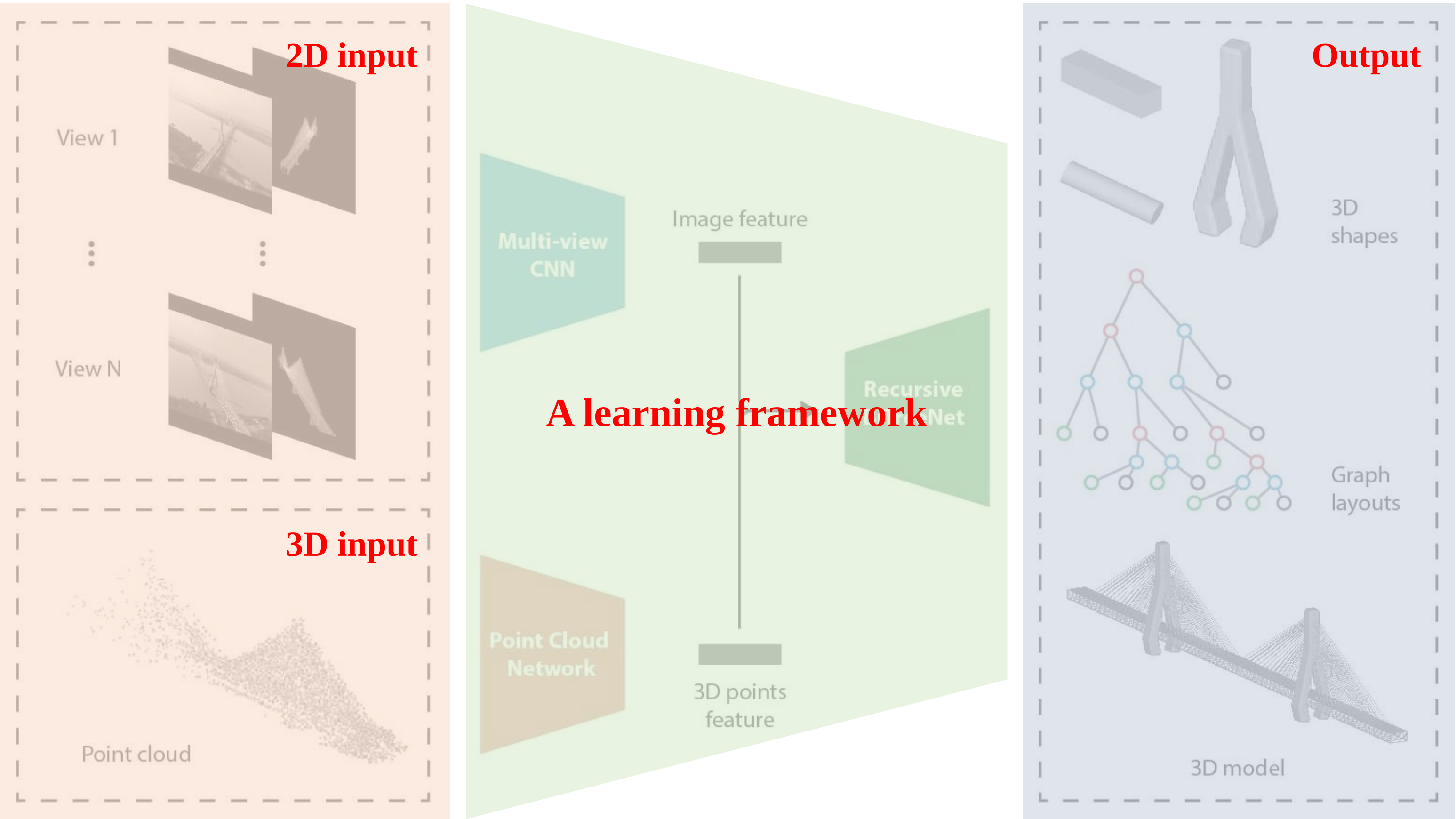
Topology aware 3D model



How did the traditional methods fail?

	Local part 1	Local part 2	Results
Point cloud			
Surface reconstruction			
The proposed method			

Three main problems: (I) the input, (II) the output, (III) from input to output

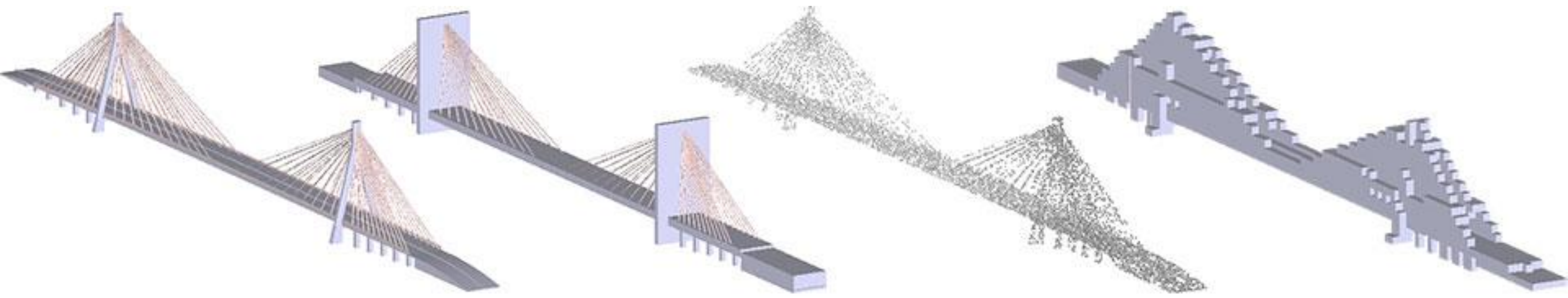


Part I

The output: how to represent a 3D bridge model?

How to represent a 3D bridge model?

How to represent a 3D bridge model?



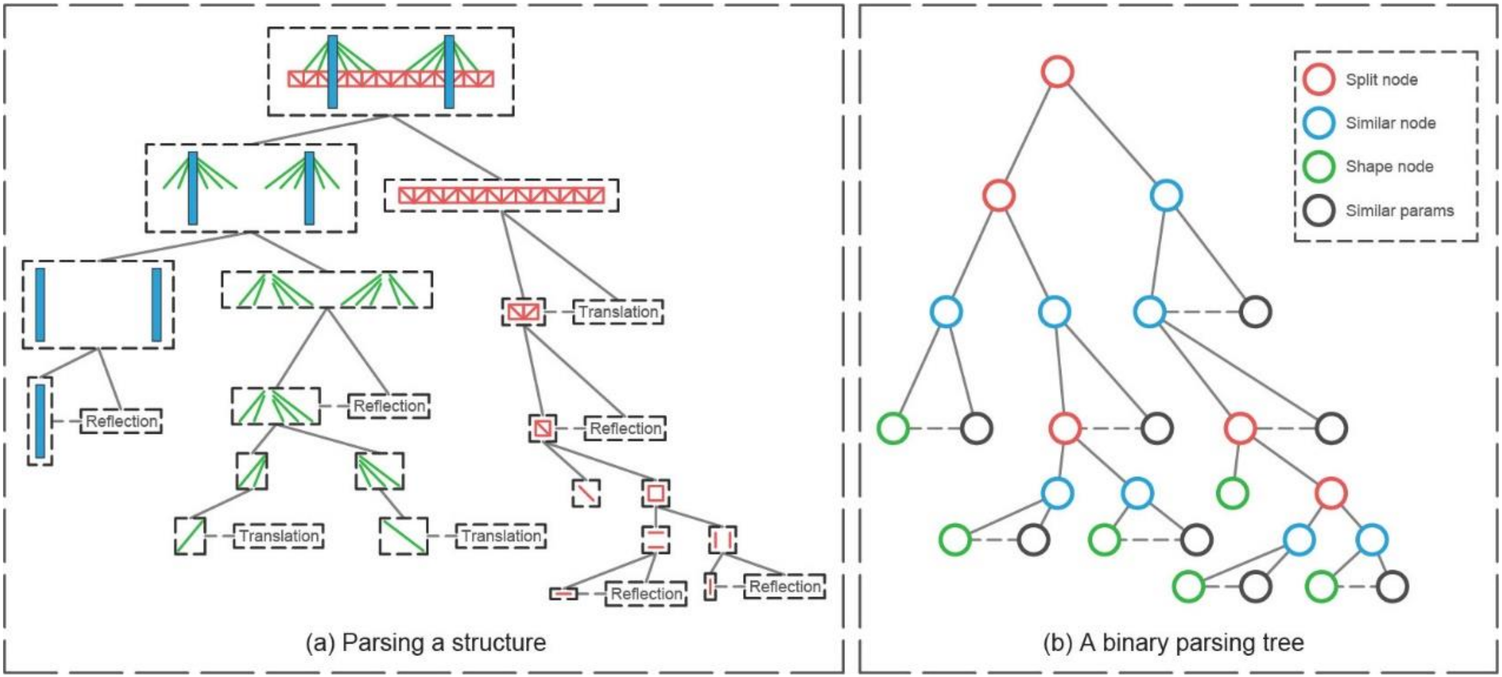
(a) Polygonal Mesh (b) Geometric Primitives (c) Point Cloud (~10K points) (d) Volumetric (128^3)

TABLE: A comparison among different 3D representation forms.

	(a) Polygonal mesh	**(b) Geometric primitives**	**(c) Point cloud**	**(d) Volumetric model**
Expression	$\{V \in \mathbb{R}^{n\times 3}, F \in \mathbb{N}^{m\times 3}\}$	$\left\{\{B_i \in \mathbb{R}^8\}_{i=1}^{m}, \{C_j \in \mathbb{R}^7\}_{j=1}^{n}\right\}$	$\{P_i \in \mathbb{R}^3\}_{i=1}^{n}$	$V \in \mathbb{R}^{n\times n\times n}$
Pros.	Details	Compact	Easy to learn	Easy to learn
Cons.	**Very hard to learn**	Lose details	Low expression ability	Restricted resolution

Human cognition

Parse a 3D bridge model

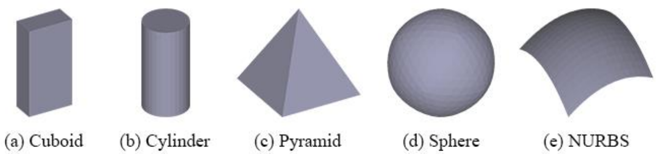


Parsing a 3D model

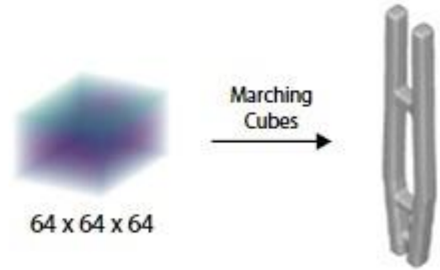
Graph layouts (a binary tree)

Hybrid representation

Geometric primitives

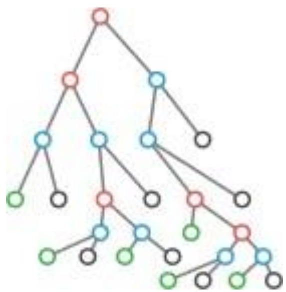


Volumetric model



Hybrid representation

+



=

3D shapes

Graph layouts

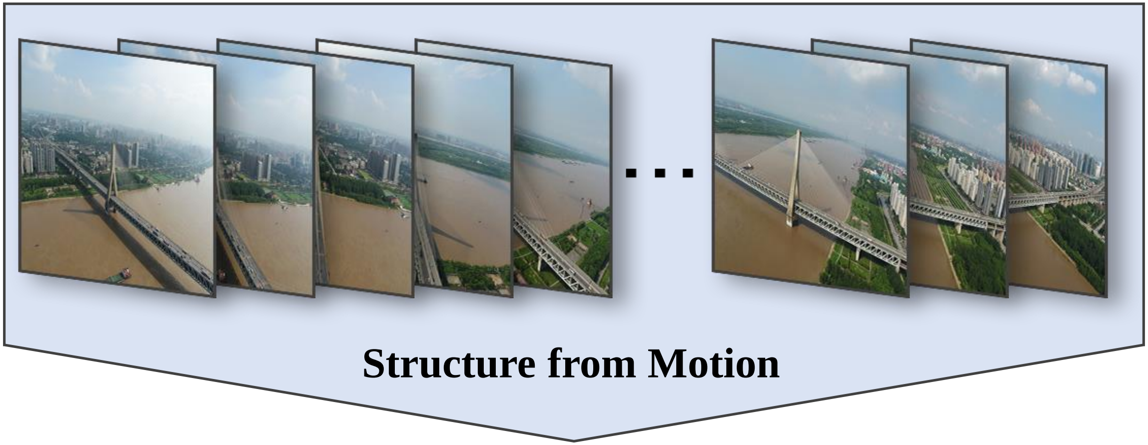
3D model

Part II

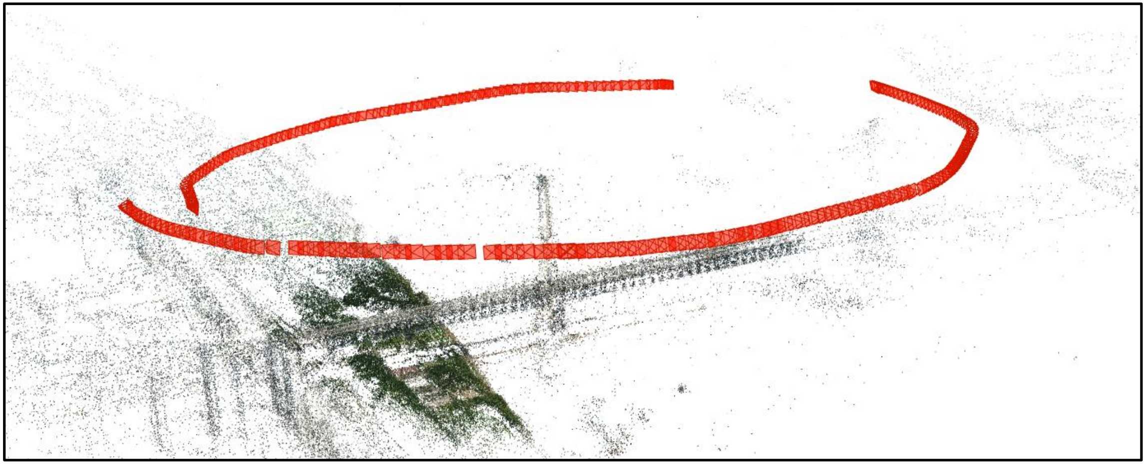
The input: how to mine the input data?

Obtaining rough 3D information

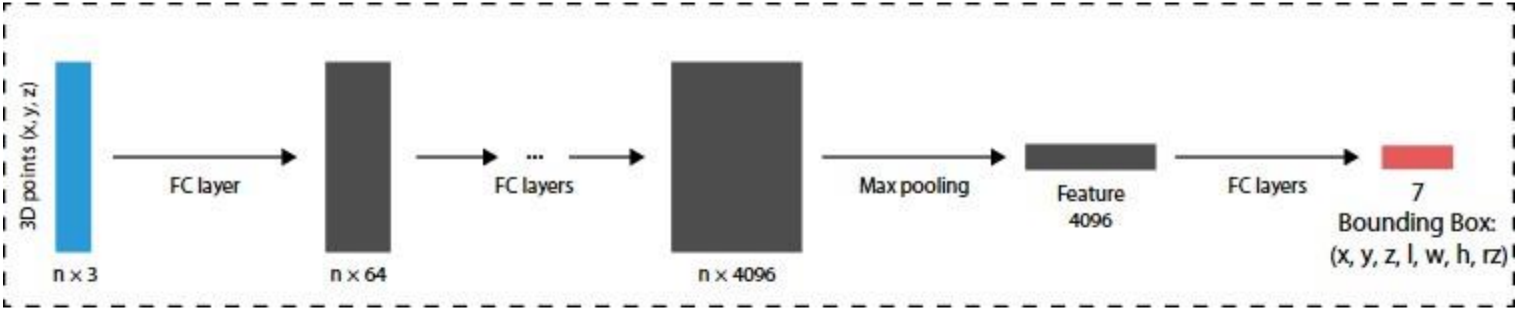
Multi-view images:
Images preserve 2D original information and details.

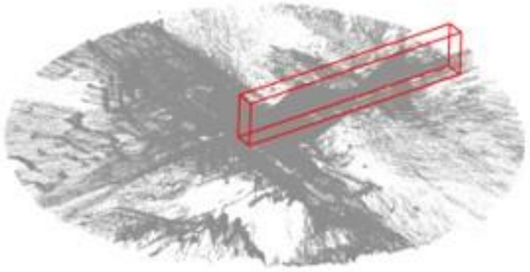


Point cloud:
Point cloud extracts rough 3D information.



Finding RoI in 3D





RoI in 3D:
A 3D orientated bounding box
(3D OBB)

Loss function

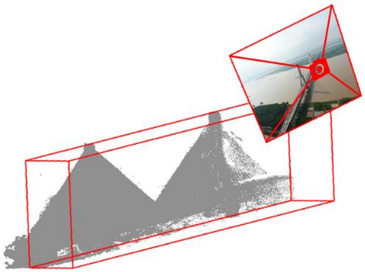
$$E = \left\| \frac{1}{B_{gt}} \otimes (B_{pred} - B_{gt}) \right\|_2$$

B_{gt}: ground truth bounding box

B_{pred}: predicted bounding box

$\otimes$: element-wise product

Finding RoI in 2D



$$p^T = S_\lambda K[R|t]P^T$$

p: points in 2D image

P: points in 3D point cloud

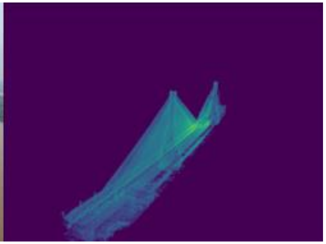
S_λ: radial distortion parameters

K: camera intrinsic parameters

$[R|t]$: camera extrinsic parameters

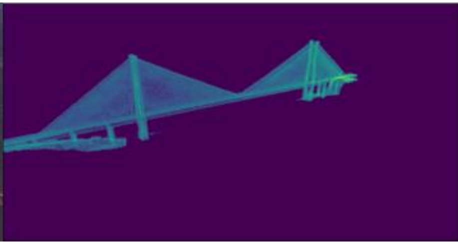
RoI in 2D: foreground-background segmentation mask

Original image



Segmented foreground

Original image

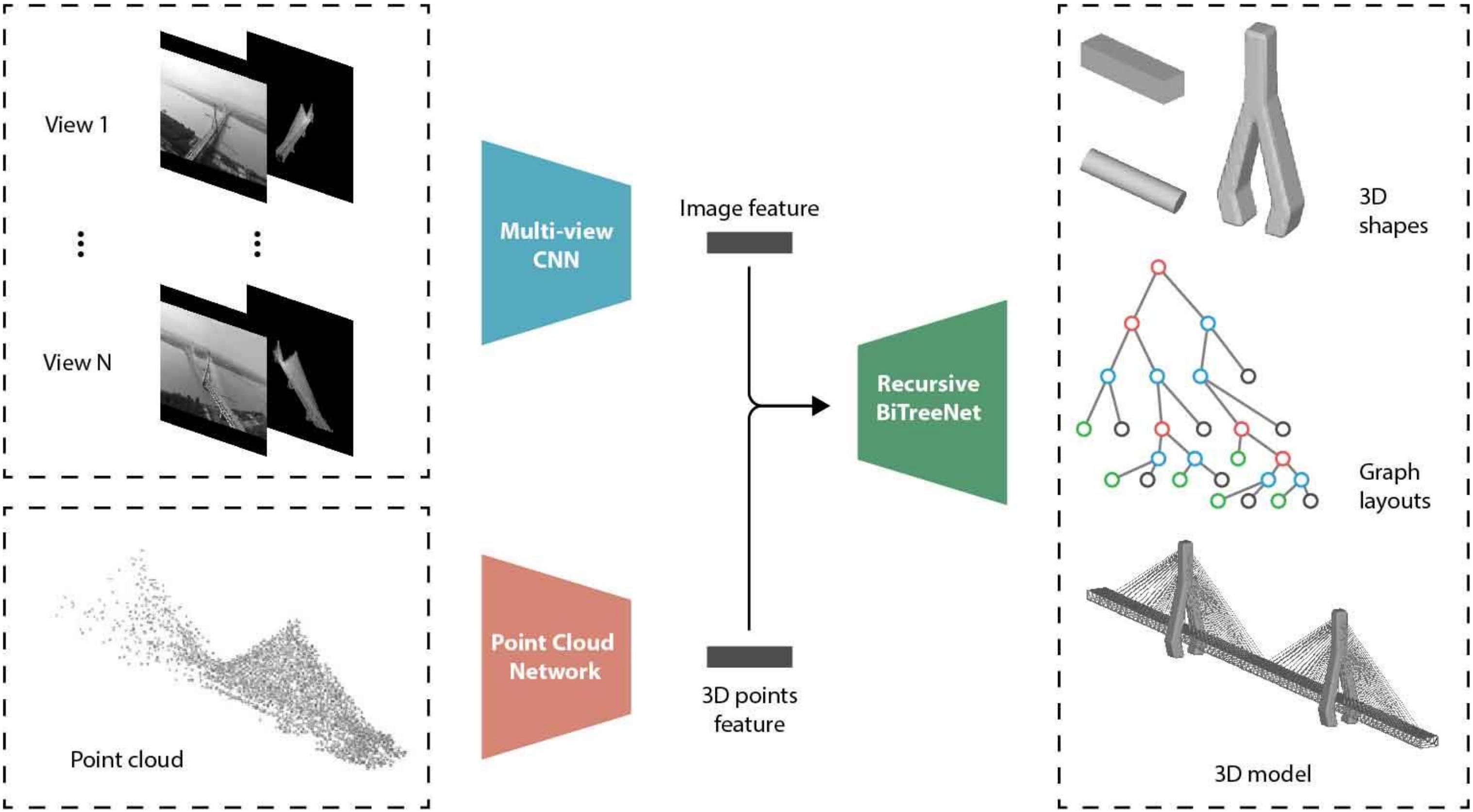


Segmented foreground

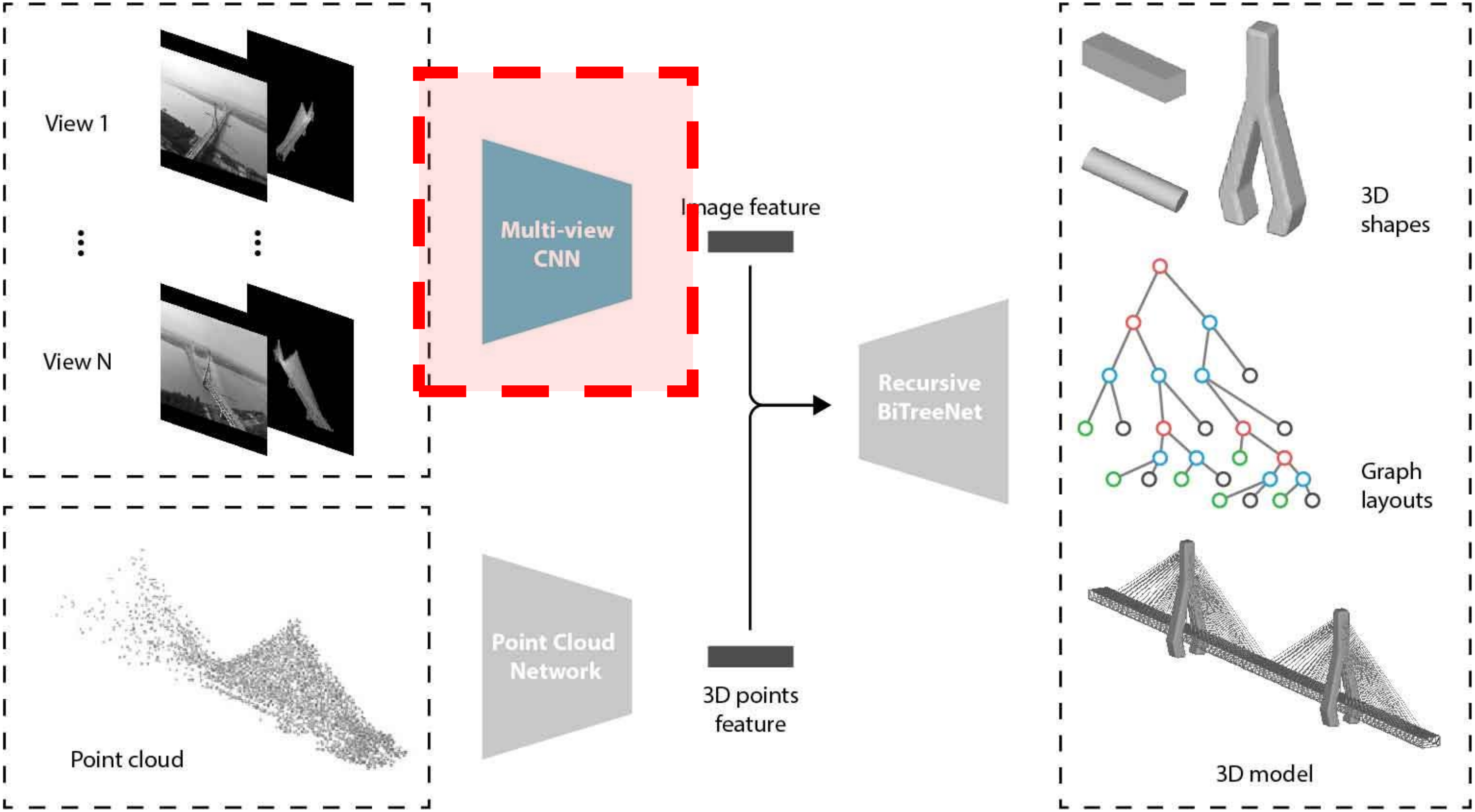
Part III

From input to output: a learning framework

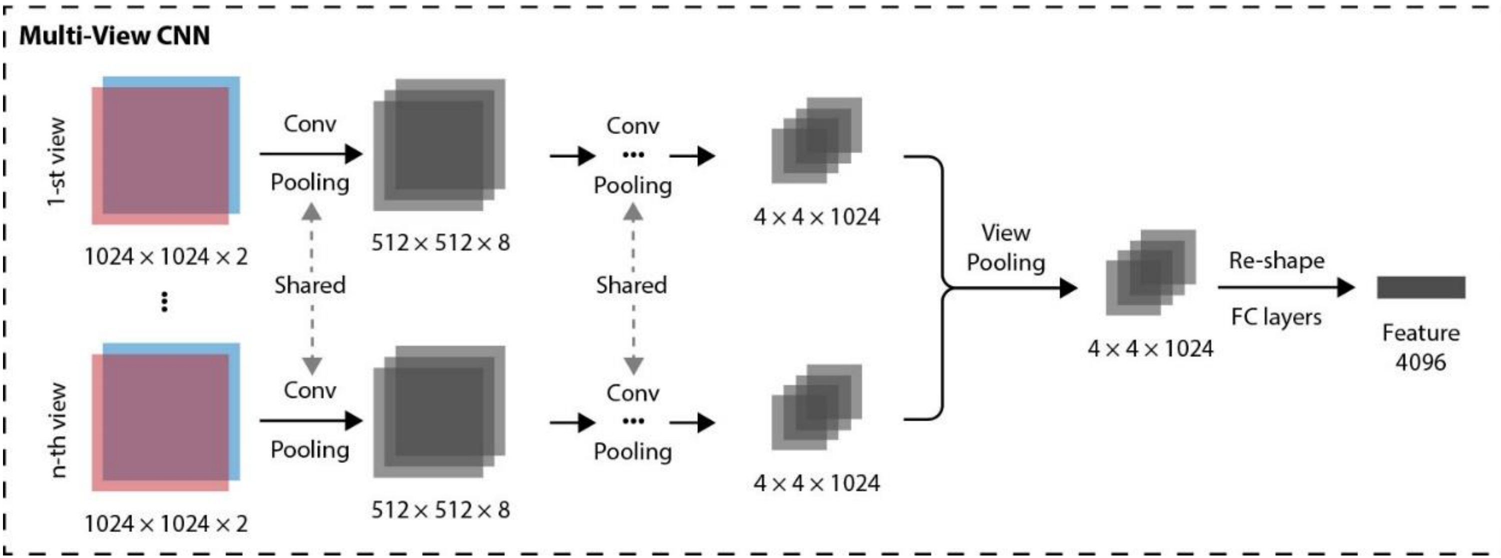
A learning framework



Multi-view CNN



Multi-view CNN



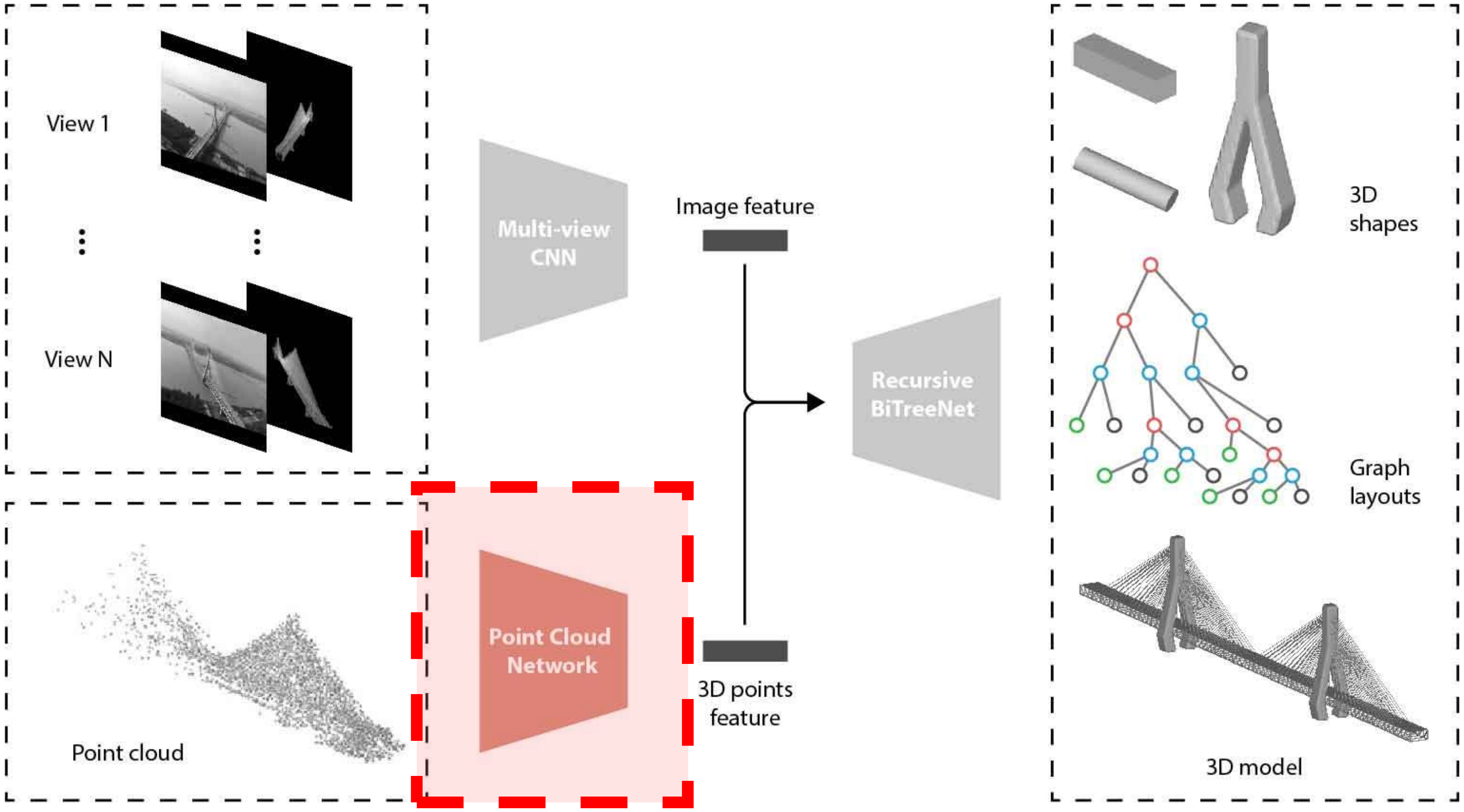
$X = \{I_i\}_{i=1}^{n}$: multi-view images

$I_i \in \mathbb{R}^{1024 \times 1024 \times 2}$: a single-view two-channel image

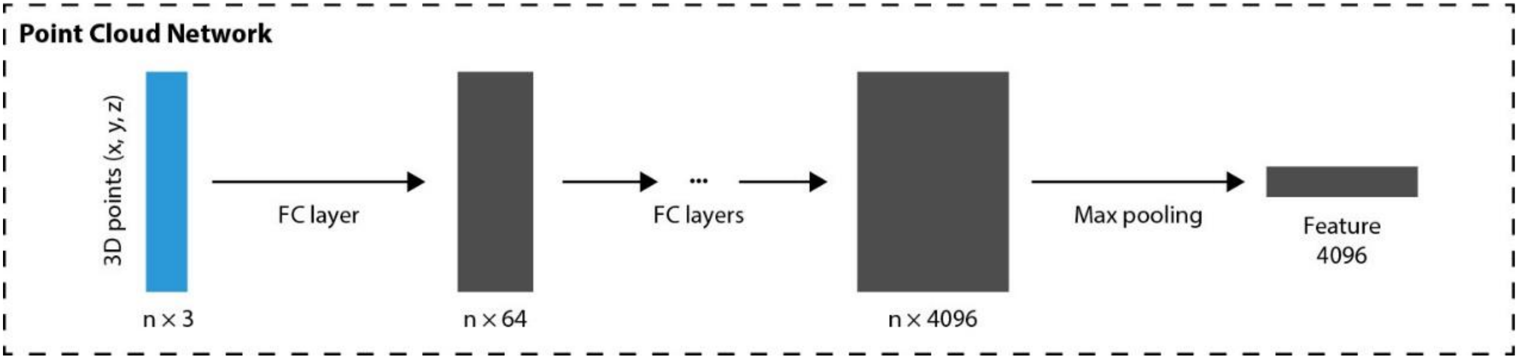
$h \in \mathbb{R}^{4096}$: the learned image feature

$f : X \rightarrow h$: the multi-view CNN

Point cloud network



Point cloud network



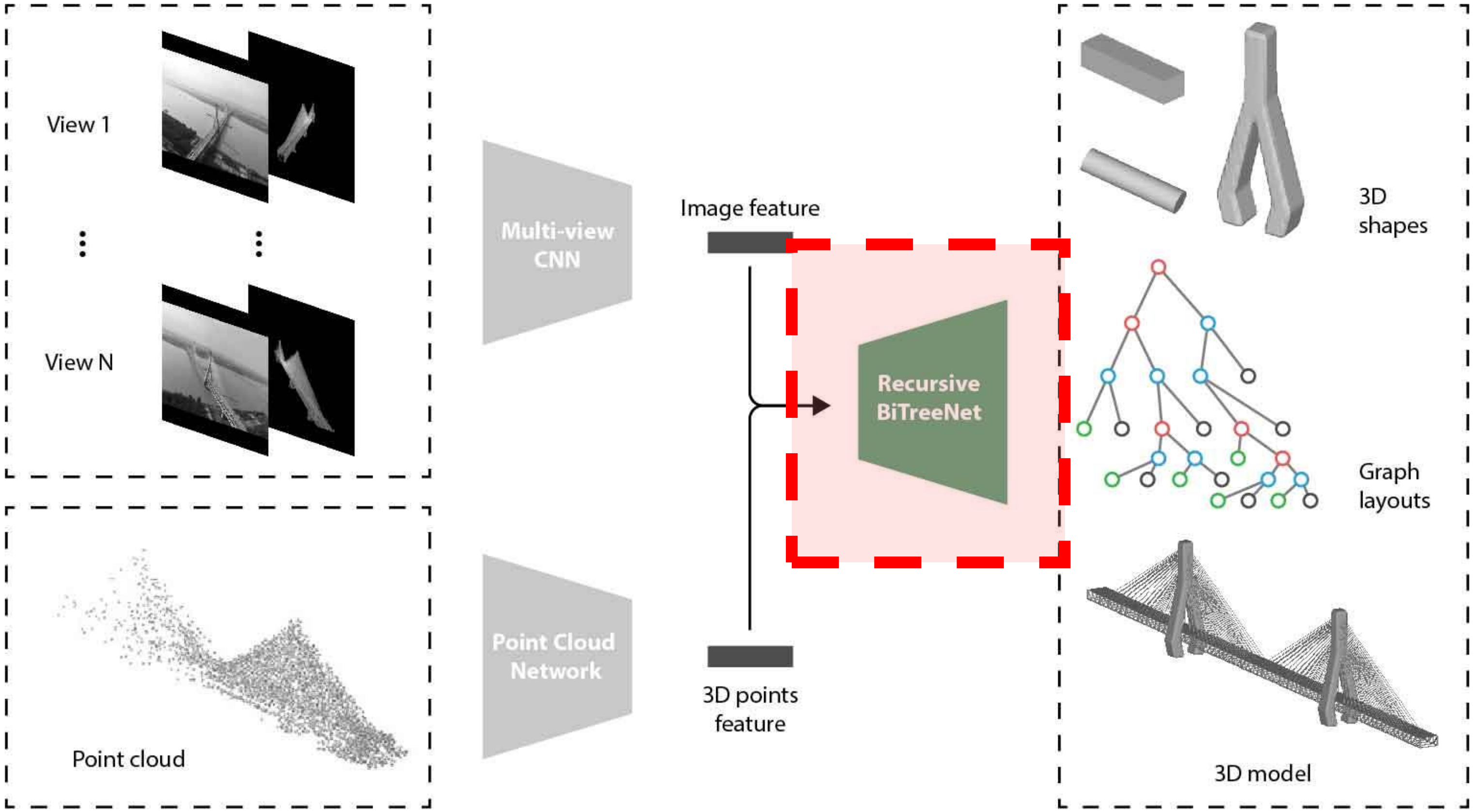
$X = \{P_i\}_{i=1}^{n}$: the point cloud

$P_i \in \mathbb{R}^3$: a 3D point in point cloud

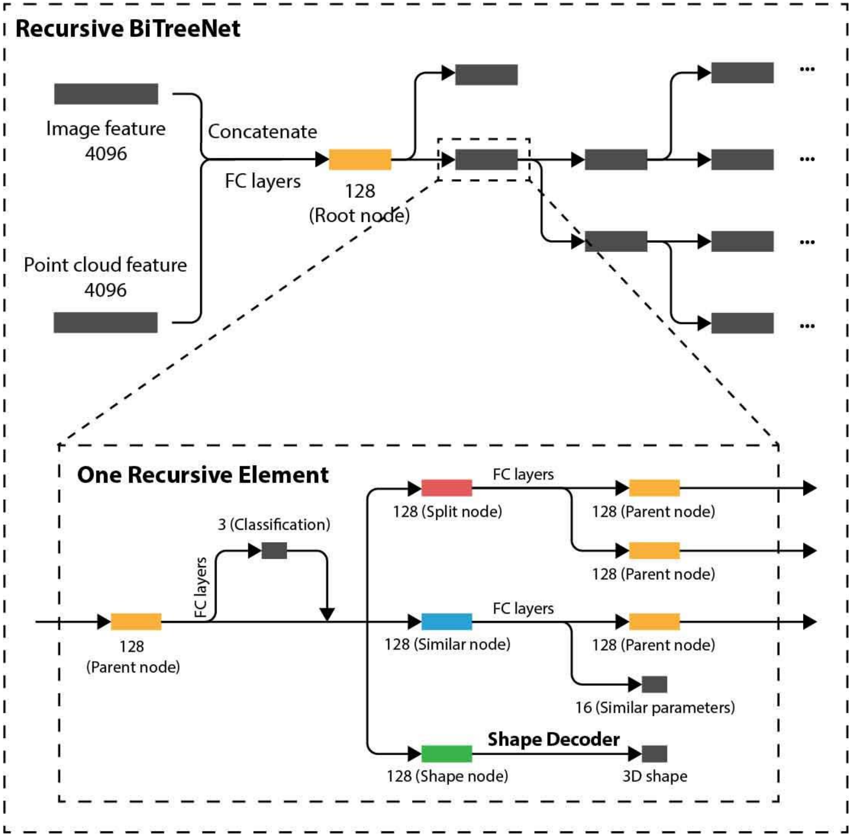
$h \in \mathbb{R}^{4096}$: the learned point cloud feature

$f : X \rightarrow h$: the point cloud network

Recursive binary tree network (Recursive BiTreeNet)



Recursive BiTreeNet



Algorithm 1

FUNCTION Recursion

PASS IN: the i-th node $X^i \in \mathbb{R}^{128}$, its node class $X^i_{cls} \in \{0,1,2,\mathrm{None}\}$

IF $X^i_{cls} = \mathrm{None}$ **THEN**

$X^i_{cls} \leftarrow \mathrm{NodeClassifer}(X^i)$

ENDIF

IF $X^i_{cls} = 0$ **THEN**

(Left child $X^{i+1}_{left} \in \mathbb{R}^{128}$, Right child $X^{i+1}_{right} \in \mathbb{R}^{128}$) $\leftarrow$ SplitNode(X^i)

PASS OUT: (Recursion(X^{i+1}_{left}, $X^{i+1}_{left\_cls}$), Recursion(X^{i+1}_{right}, $X^{i+1}_{right\_cls}$))

ELSE IF $X^i_{cls} = 1$

(Left child $X^{i+1}_{left} \in \mathbb{R}^{128}$, Right child $X^{i+1}_{right} \in \mathbb{R}^{16}$) $\leftarrow$ SimilarNode(X^i)

PASS OUT: (Recursion(X^{i+1}_{left}, $X^{i+1}_{left\_cls}$), X^{i+1}_{right})

ELSE

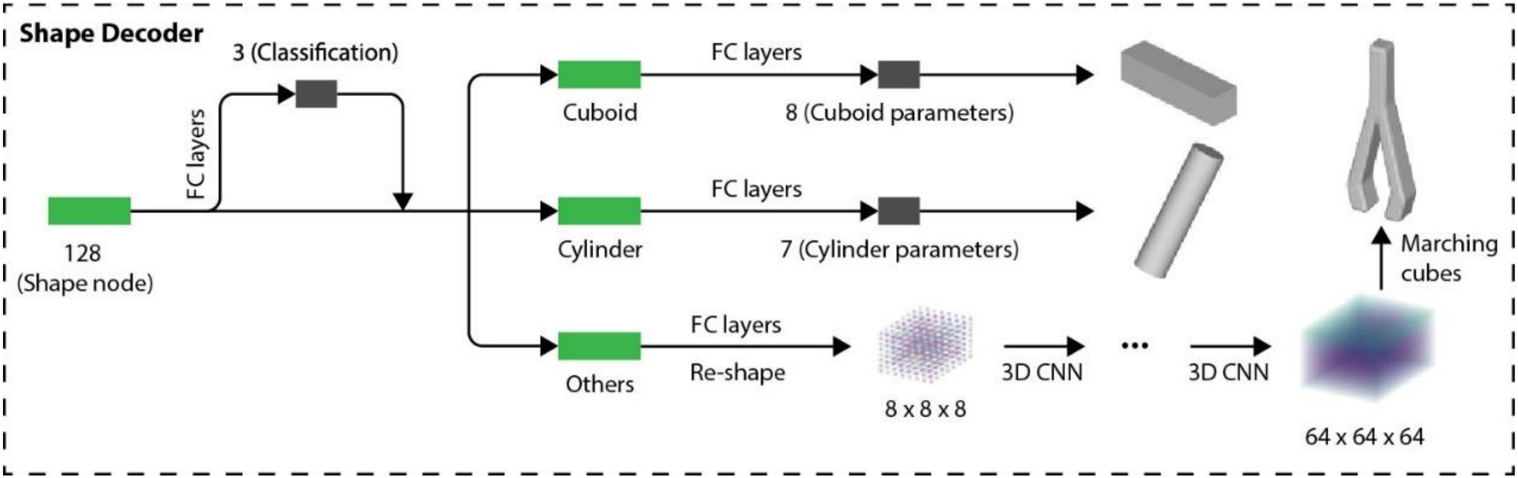
Child $S^{i+1} \leftarrow$ ShapeNode(X^i)

PASS OUT: S^{i+1}

ENDIF

ENDFUNCTION

Shape decoder



Loss functions

Node classification loss

$$L_{cls} = -\sum_{i \in \{1,\dots,n\}} \sum_{x \in X} p_i(x) \log q_i(x)$$

3D shapes loss

$$L_{shape} = \sum_{i \in \{1,\dots,n\}} dist(M_i^{gt}, M_i^{pred})$$

Shapes distance

$$dist(M_1, M_2) = \sum_{v_1 \in M_1} \min_{v_2 \in M_2} \| v_1 - v_2 \|_2^2 + \sum_{v_2 \in M_2} \min_{v_1 \in M_1} \| v_1 - v_2 \|_2^2$$

Similar parameters loss

$$L_{sim} = \sum_{i \in \{1,\dots,n\}} \| S_i^{pred} - S_i^{gt} \|_2$$

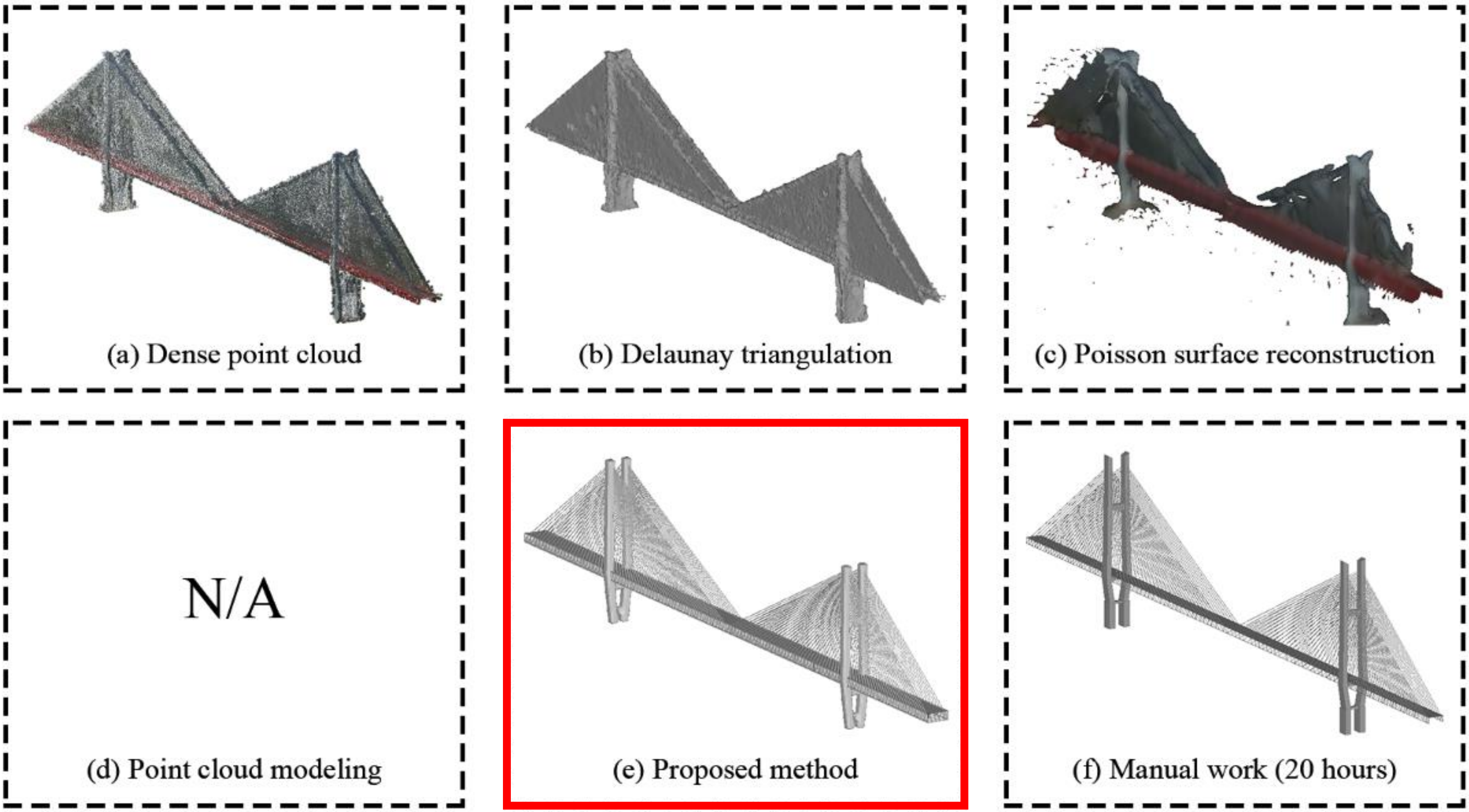
Overall loss

$$L = \lambda_{cls} L_{cls} + \lambda_{shape} L_{shape} + \lambda_{sim} L_{sim}$$

Part IV

The results

Results – Beipanjiang Bridge



(a) Dense point cloud

(b) Delaunay triangulation

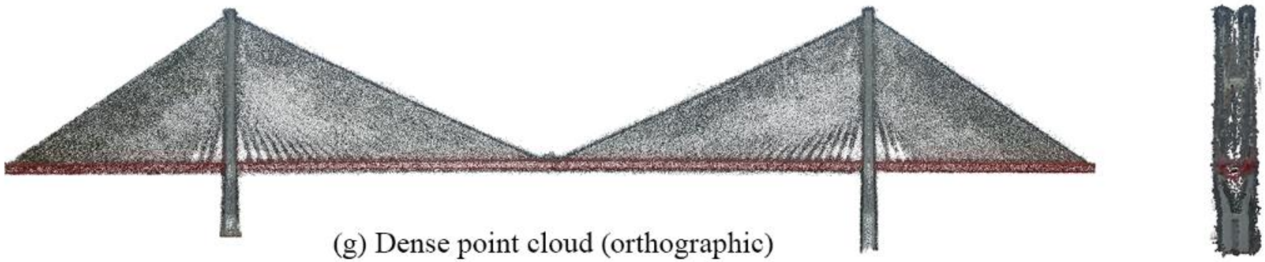
(c) Poisson surface reconstruction

(d) Point cloud modeling

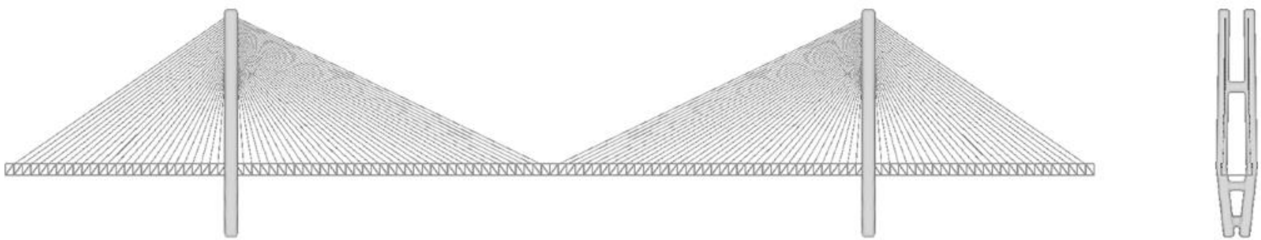
(e) Proposed method

(f) Manual work (20 hours)

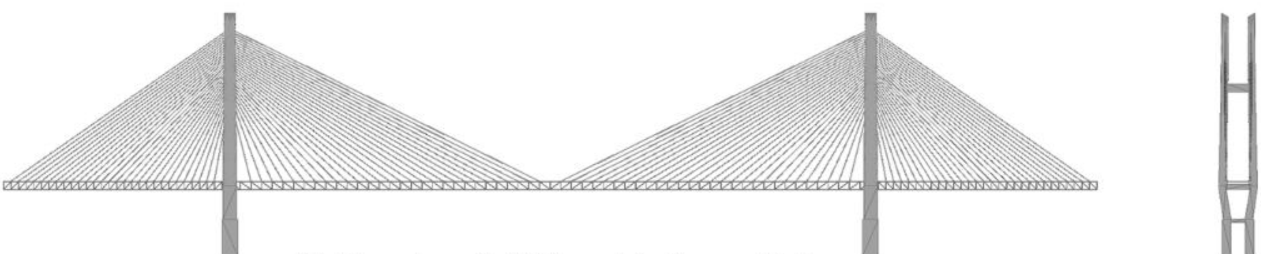
Results – Beipanjiang Bridge



(g) Dense point cloud (orthographic)

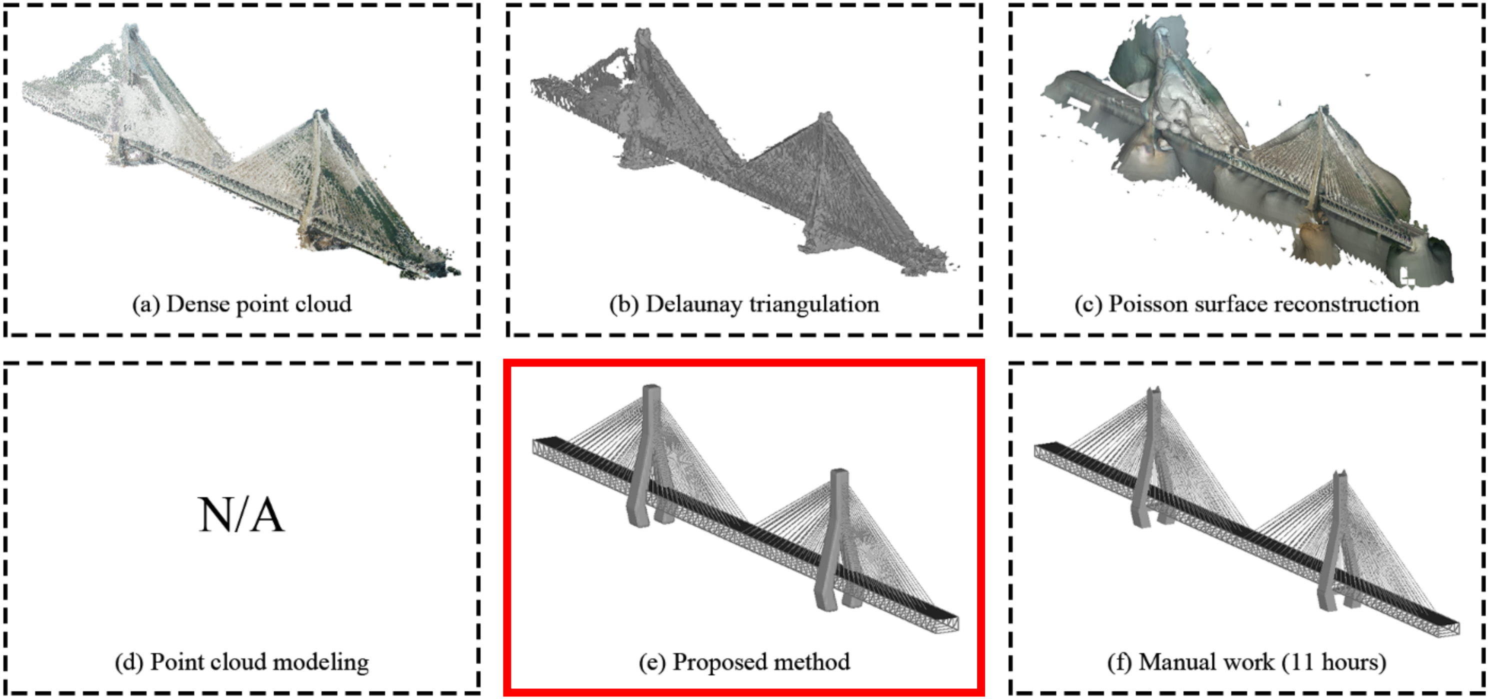


(h) Proposed method (orthographic)

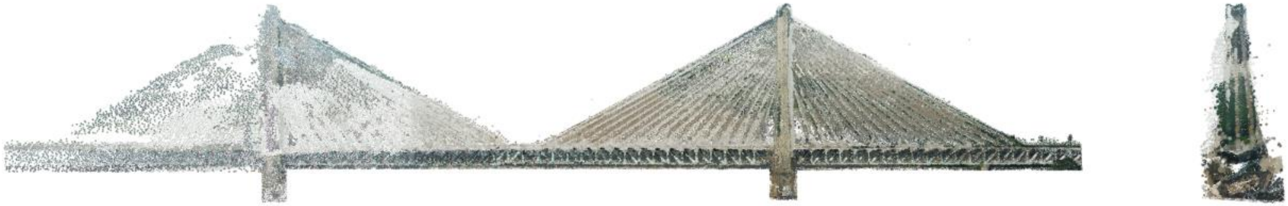


(i) Manual work (20 hours) (orthographic)

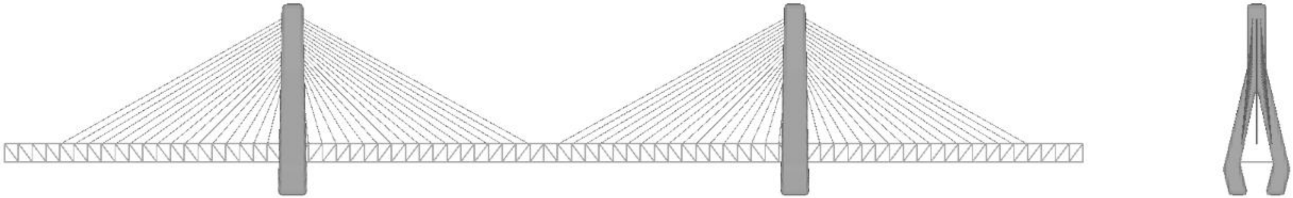
Results – Wuhan Tianxingzhou Bridge



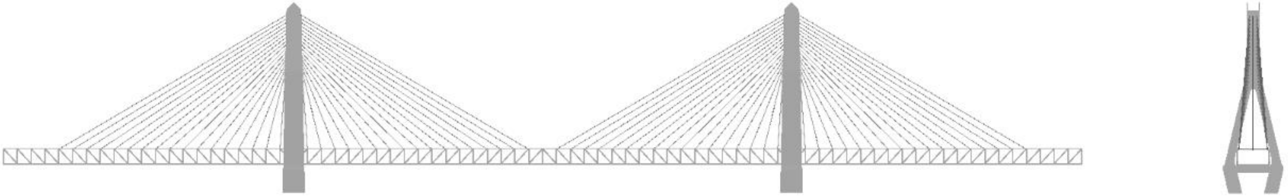
Results – Wuhan Tianxingzhou Bridge



(g) Dense point cloud (orthographic)

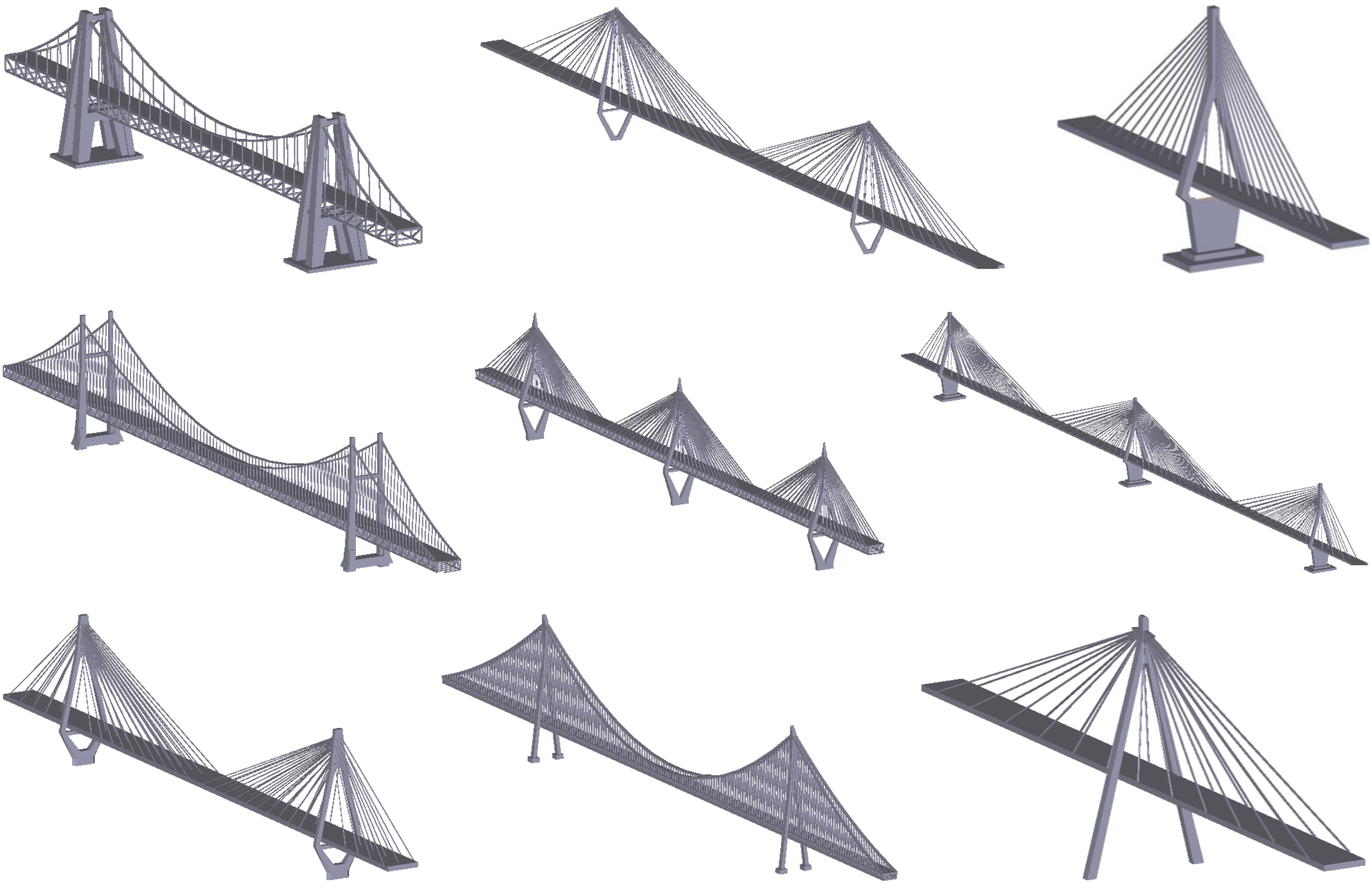


(h) Proposed method (orthographic)



(i) Manual work (11 hours) (orthographic)

Some of the training data



Summary and conclusions

- **A revisit to local points based methods**
  - No structural priors are introduced in these methods.
  - Point clouds suffer from noise and uneven distribution.
  - Surface reconstruction and point cloud modeling methods failed.
- **A learning based 3D reconstruction method**
  - Structural relations and topological properties are considered.
  - 3D information is considered in contrast to image based learning methods.

magicme.org

Fangqiao Hu (胡芳侨)

| A first year Ph.D candidate in Harbin Institute of Technology (HIT)

| 3D machine learning; Computer vision; Computer graphics

| E-mail: hufangqiao@foxmail.com

| Homepage: magicme.org