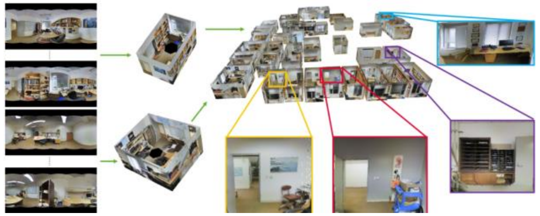
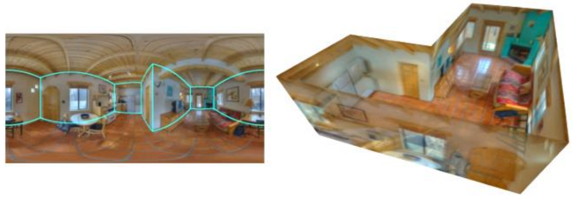
Room modeling

Introduction

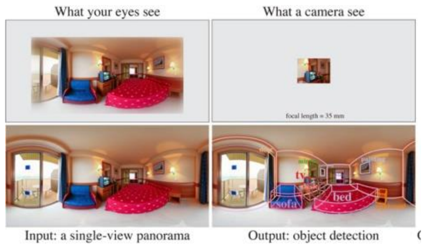
- Goal: reconstruction of individual spaces
  - commonly identified with rooms
- Input: images associated with the room
  - One or few panoramic images
- Output: 3D room model or pixel-wise information
  - Single scene
    - Walls, ceilings, floor
  - Multi-modal information for specific tasks
    - Editing, VR exploration, etc.
  - Optional: multi-room integration
- Peculiarity of panoramic images
  - Effective reconstruction even from a single image



MVlayoutNet – Hu ACM MM2022



HorizonNet – Sun CVPR2019



Zhang et al. ECCV 2014

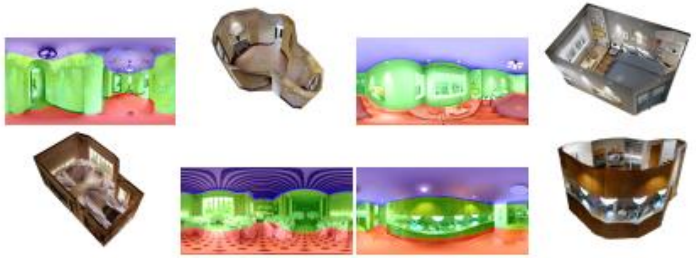
JUNE 18-22, 2023 CVPR VANCOUVER, CANADA

Fundamental tasks

- Pixel-wise information from a single image
  - **Depth**, semantics, normals
  - 2D – 2D task (at least 3D features of the visible point cloud)
  - Visible scene, dense reconstruction
  - Also needed for **model integration** (see next Section)

Structured3D ECCV 2020

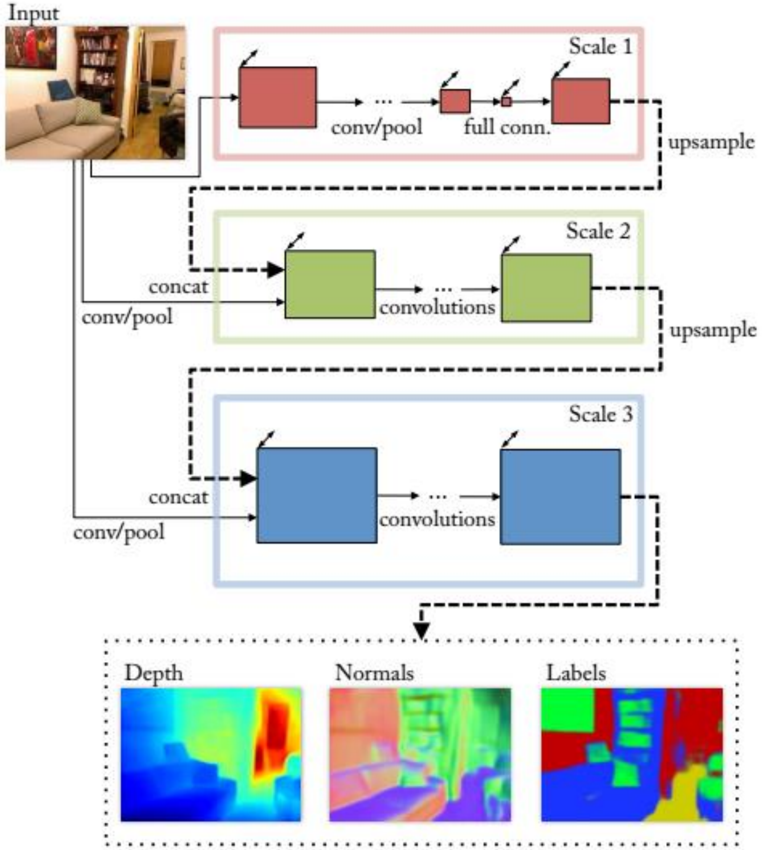
- Underlying structure reconstruction: **3D layout**
  - Geometry of the bounding permanent surfaces
    - i.e., ceiling, floor, walls
  - 2D – 3D task (corners, edges, planes, meshes)
  - Dealing with occlusions, sparse approximation



AtlantaNet ECCV 2020

Pixel-wise reconstruction

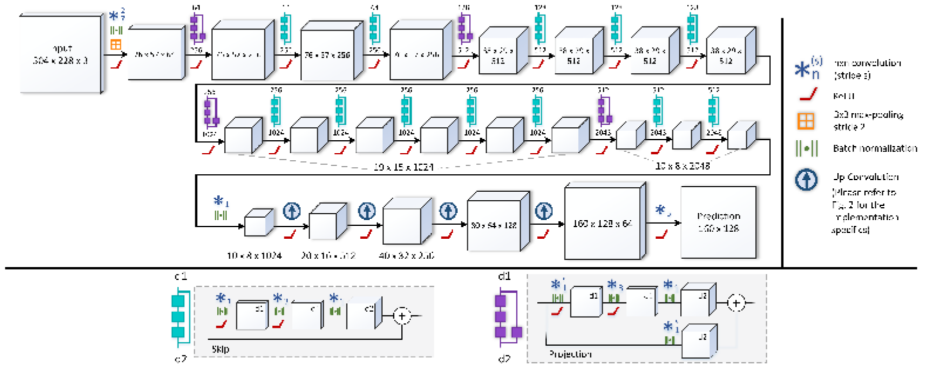
- Most common: depth reconstruction
- Naturally fits deep learning approach
  - Often supervised
- Standard solution: encoder-decoder scheme
  - Changing latest layer and activation function
    - Depth
    - Semantic
    - Normals
    - …

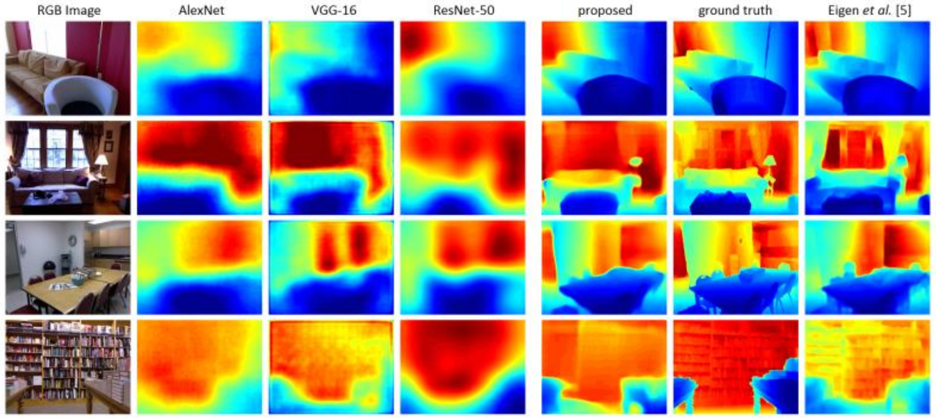


Eigen et al. ICCV2015

Depth estimation from a single image

- Early perspective solution: FCRN
  - Baseline of much of the following work
  - Fully convolutional
  - Residual architecture
  - Huber loss
- Generic encoder-decoder solution
  - No indoor priors
  - No spherical assumptions
- Simply adaption leads to poor results for indoor panoramic scenes

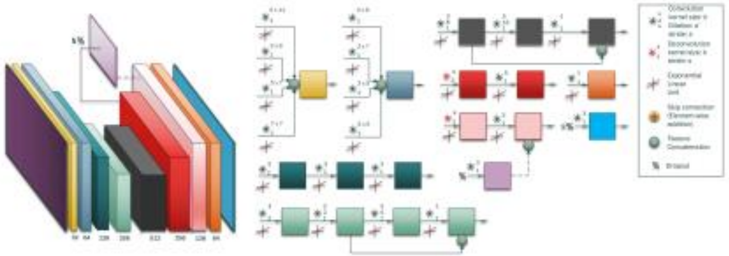




FCRN – Laina 3DV2016

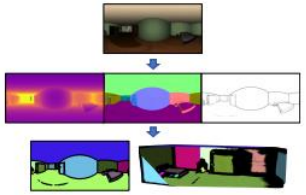
Depth estimation from indoor panorama

- Distortion-aware convolutions: OmniDepth
  - FCRN baseline extended with specialized kernels
  - Targeted for equirectangular images



OmniDepth - Zioulis ECCV2018

- Indoor priors in the loss function: panoPopups
  - Planar assumption
    - Principal curvature
  - Depth and normal map prediction



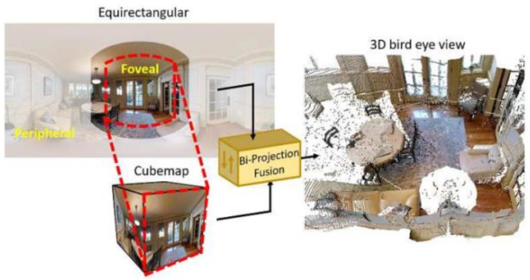
$$P(x) = xe^{-||c^*||}$$

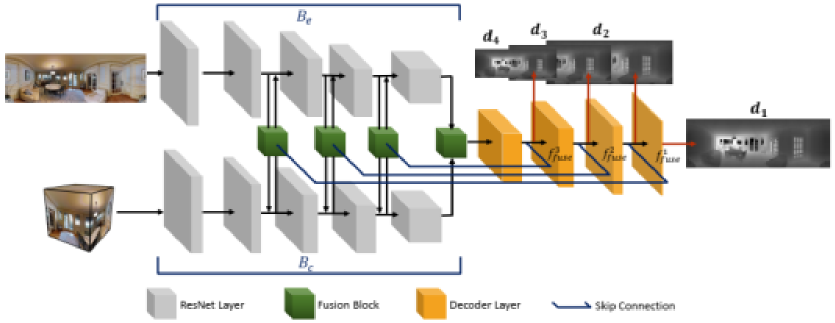
$$L_z = \frac{1}{|\mathcal{M}|} \sum_{i \in \mathcal{M}} P\left(B\left(z_i - z_i^*\right)\right)$$

Pano Popups - Eder CVPR2019

Depth estimation from indoor panorama

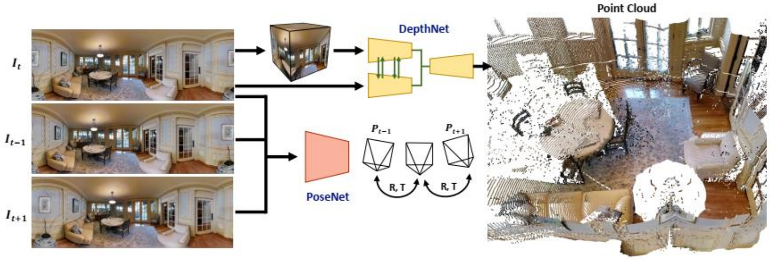
- BiFuse: fully-supervised dual branch network
  - Equirectangular branch: capturing wide context
  - Cubemap branch: aiming to minimize distortion
  - Multiple-fusion at feature-level
  - Computationally expensive





BiFuse - Wang CVPR 2020

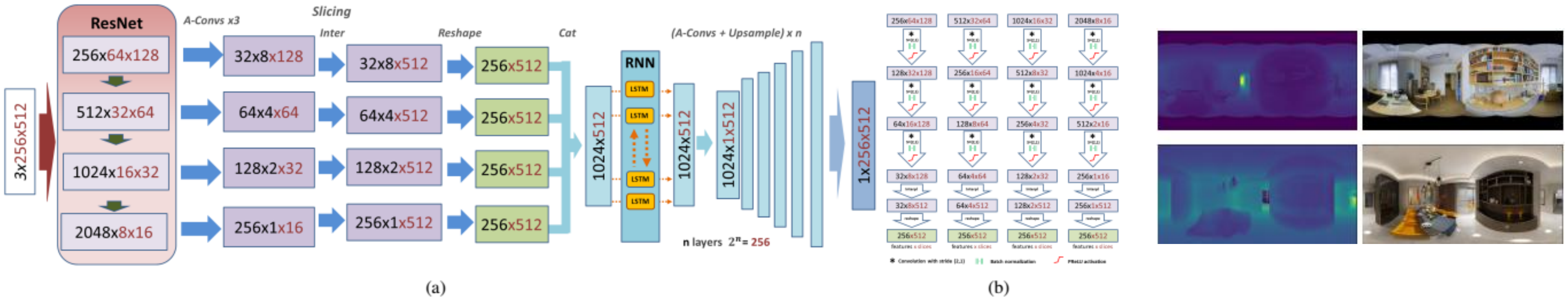
- BiFuse++: self-supervised dual branch network
  - Photo consistency from 3 adjacent panoramas
  - Improves BiFuse performance
- **No specific indoor assumption adopted**



BiFuse++ - Wang TPAMI 2022

Depth estimation from indoor panorama

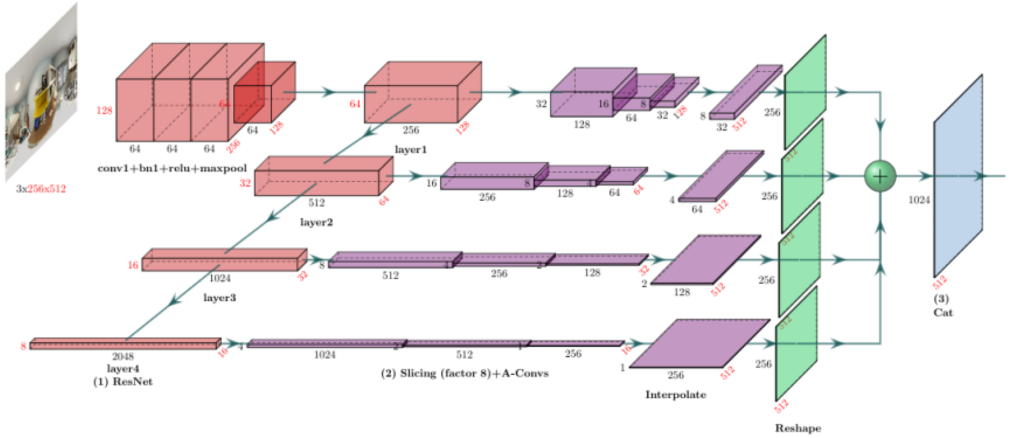
- Exploiting indoor features: SliceNet
  - Gravity aligned features encoded as sequences of compressed features
    - Typical of man-made structures
  - Spherical image compact representation
    - Asymmetric contractive convolutions (along vertical directions – sphere vertical slices)
    - Vertical slices correspond to vertically compressed features



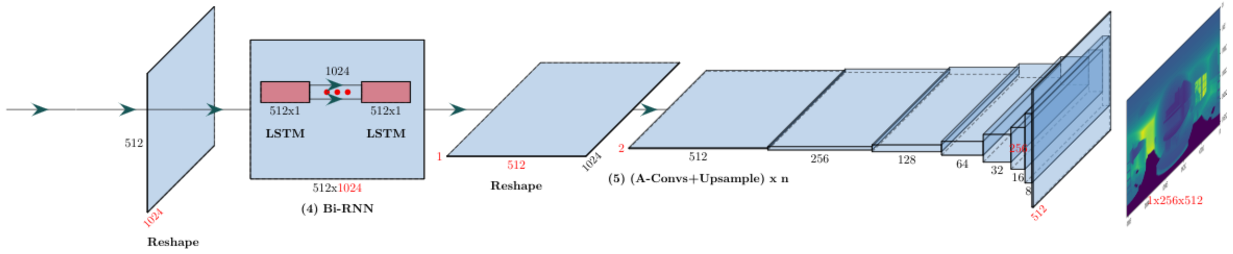
SliceNet - Pintore CVPR 2021

Depth estimation from indoor panorama

- Encoder: contractive convolution only along vertical direction
  - Return a sequence of compressed features along image width (W x s)
- Decoder: LSTM (transformer-like) + vertical decompression
  - LSTM recovering long and short term spatial relationships between slices
  - Decompression: asymmetric convolution (same of encoding) and upsampling



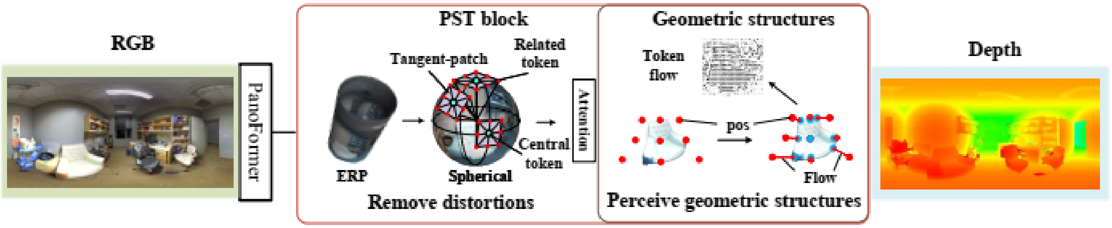
SliceNet encoder



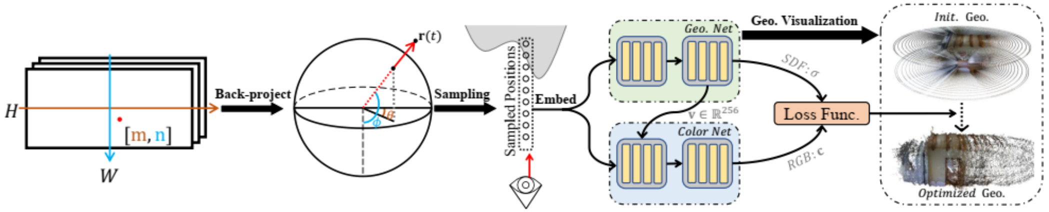
SliceNet dencoder

Depth estimation from indoor panorama

- Using transformer: PanoFormer
  - Improving small details recovery
  - Combining self-attention and tangent images (Eder CVPR2020)
- Neural scene representation
  - Multi-view: 3 panorama required
  - Better than common SfM
    - But not comparable on large scale datasets
  - NeRF drives self-supervised learning
  - Manhattan World weights initialization



PanoFormer - Shen ECCV2022

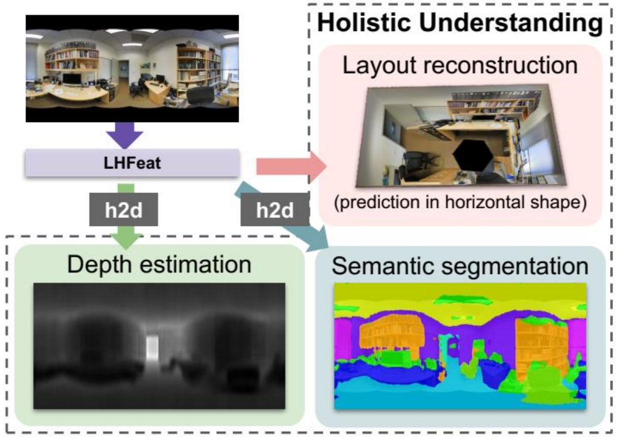


Chan et al. CVPR2023

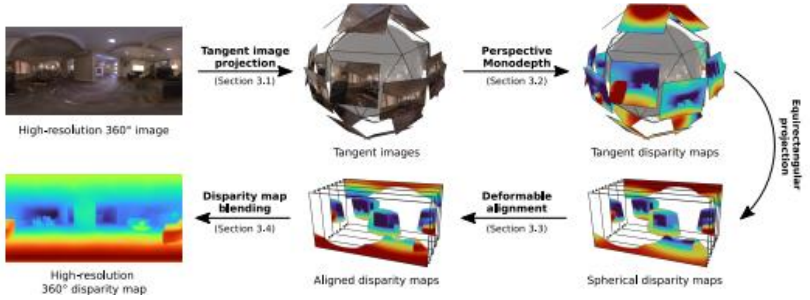
Pixel-wise reconstruction: summary

- Fundamental task to create high-level models
  - Holistic and semantic understanding
  - Supporting single and multi-view reconstruction

- Open problems
  - Resolution
  - Computational scalability
  - Self-supervised methods limits
    - Less performance for single view
  - Prediction consistency for multi-view
    - Limit to an effective integration



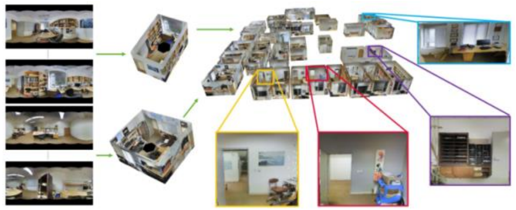
HoHoNet - Sun CVPR2021



360MonoDepth - Rey-Area CVPR2022

3D layout reconstruction

- Geometry of the bounding surfaces
  - Rooms from one or more images
    - combined to form a more complex structured model
- Common in panoramic world: room from a single image
  - Easier to capture (even by stitching with a mobile)
  - Avoid additional image-registration pipeline
    - Multi-view only to compose more complex environments
      - **See next Sec.**
  - A single 360 image contains enough information

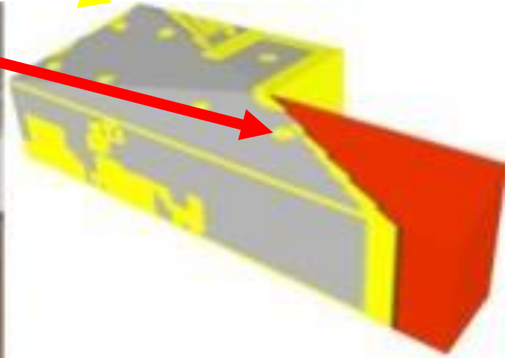


Hu et al. ACM MM2022

Zou et al. IJCV2022

Layout reconstruction: indoor issues

- Differences from depth estimation
  - Occlusions from clutter
  - Self-occlusions
- Ambiguities
  - Texture-poor surfaces
  - …
- Need for indoor priors!

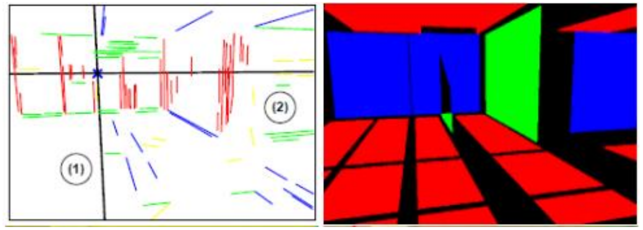


Layout from a single RGB image

- **Early approaches (perspective)**
  - Fundamental priors (see prev. sessions)

  - Geometric context (GC) for indoor scene
    - Cuboid (CB) prior
    - Room box and surface labels jointly estimated
      - floor, ceiling, wall, objects
  - Orientation maps (OM) from MW vanishing lines
    - Indoor World Model (IWM) geometric reasoning
    - Manhattan world planes bounding the room

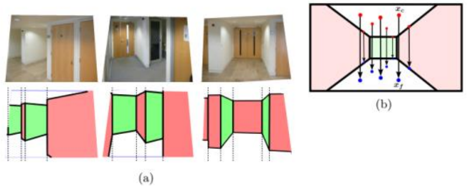
Hedau et al. ICCV 2009



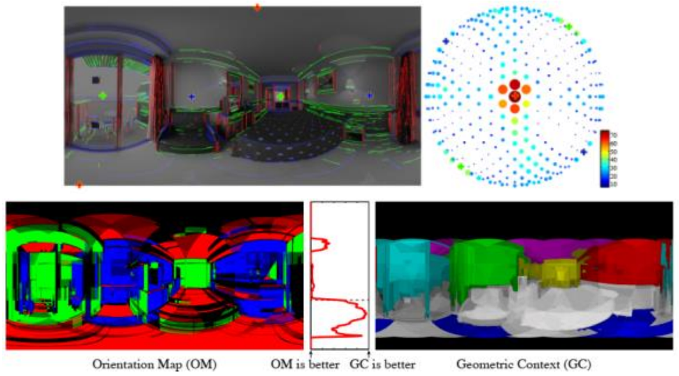
Lee et al. CVPR 2009

Layout from a single RGB image

- **Geometric context (GC) and Orientation Map (OM)**
  - Basis of indoor geometric reasoning
- Geometric reasoning on the IWM
  - Horizontal floor and ceiling related by a homography
- **Geometric reasoning on panorama: PanoContext**
  - Panoramic image converted into perspective images
    - e.g, cubemaps
  - GC and OM applied to perspective views
  - Results re-projected on the original panorama



Flint et al. ECCV 2010

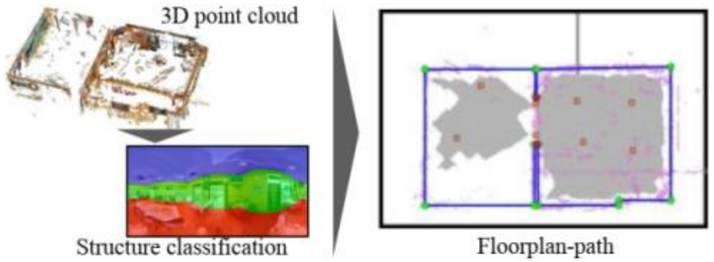


PanoContext - Zhang. ECCV2014

Layout from a single panorama

- **PanoContext**
  - Demonstrates advantages of panoramic vs. pine-hole view
  - Image preprocessing: GC+OM exploited for aligning to Manhattan World axis
    - Still adopted by many modern pipelines
    - Simplifying reconstruction
  - 3D room box fitting
- Room shape from panorama
  - Improving room shape estimation
  - Exploiting panoramic image to improve reconstruction from a point cloud

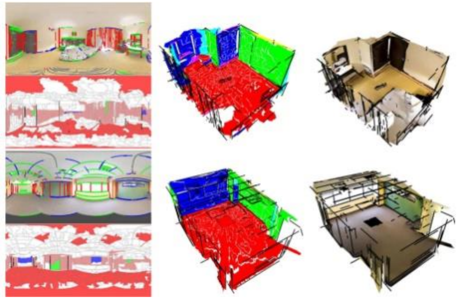
PanoContext - Zhang. ECCV2014



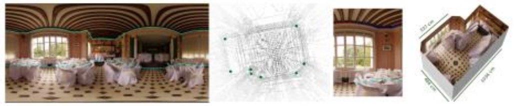
Cabral et al. CVPR2014

Layout from a single panorama

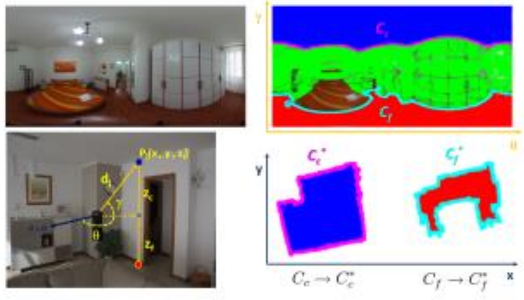
- Fully geometric reasoning approaches
  - Low-level features derived by GC e OM
    - Super-pixels and edges
  - Spatial transforms from indoor priors
    - Super-pixels and edges projected to floorplan
- ..untill the rise of deep-learning approaches
  - Data-driven features
  - Indoor priors still hold



Yang et al. CVPR2016



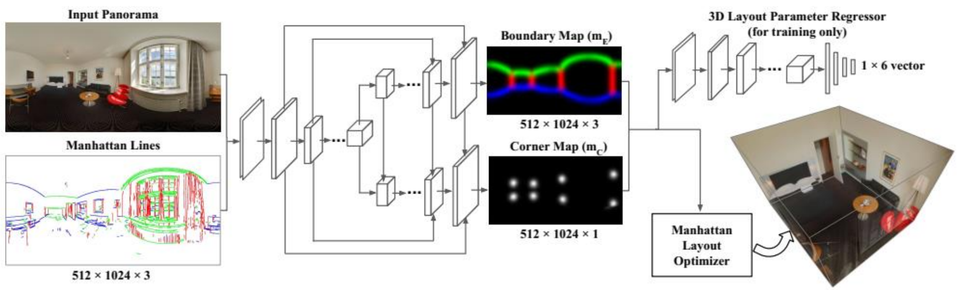
Pintore et al. WACV2016



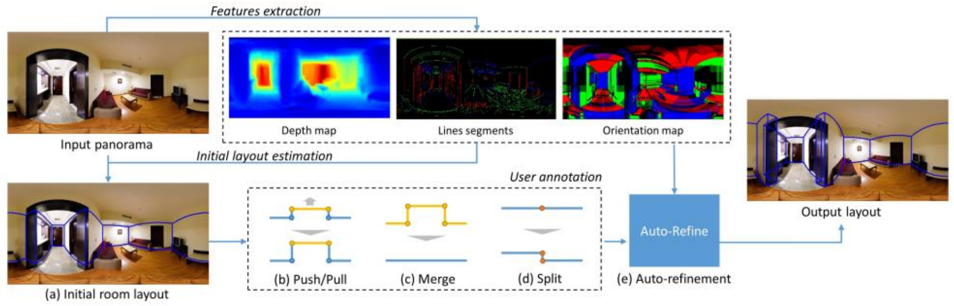
Pintore et al. C&G 2018

Data-driven layout from panorama

- Impressive results
  - Accuracy and speed
- Typical output
  - Corners and boundaries
- Large annotated datasets needed
  - Synthetic or real
  - Often rectified images for real-world data
    - GC+OM alignment
  - Abstract models
    - User annotation
    - Assisted tools needed



LayoutNet – Zou CVPR2018

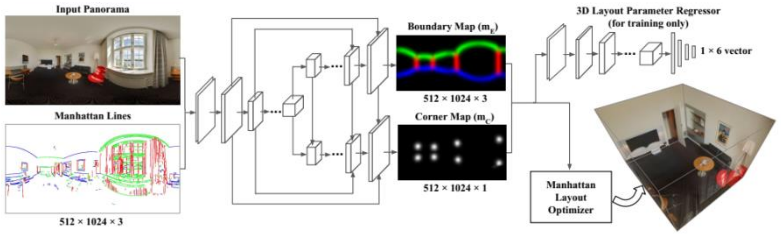


PanoAnnotator – Yang Siggraph Asia post. 2018

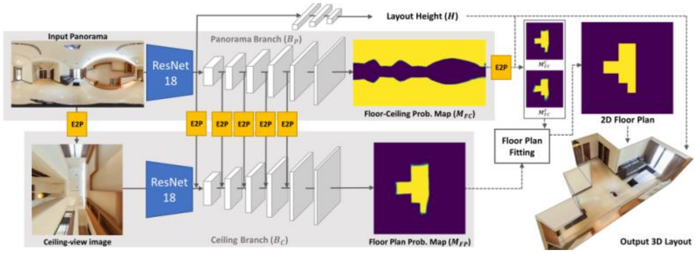
Speaker: Giovanni Pintore

Data-driven layout from panorama

- Common pipeline
  - Image pre-processing
    - Reduce 2D-3D error
  - Elements prediction
    - 2D image points: corners, boundaries, floor maps
  - Final model post-processing
    - Priors-guided regularization and 3D model generation



LayoutNet – Zou CVPR2018



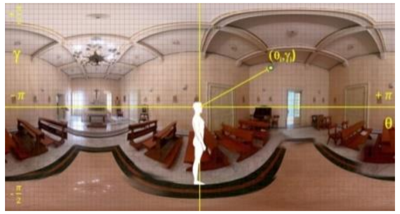
DulaNet – Yang CVPR2019

Method	Input			Pre-process	Network Architecture				Output			Post-processing	
					Encoder		Decoder						
	RGB in Equi-rectangular View	RGB in Ceiling View	Manhattan Lines Map		SegNet	ResNet	Equirectangular View	Ceiling View	Corner Position	Boudary Position	Floor Map	Equirectangular View	Ceiling View
LayoutNet	●		●	●	●		●		●	●		●	
DuLa-Net	●	●		●		●	●	●			●		●
HorizonNet	●			●		●	●		●	●			●

Zou et al. IJCV2021

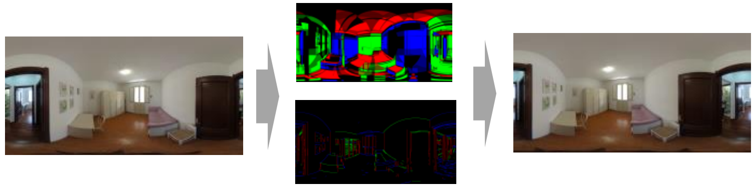
Data-driven layout: pre-processing

- Improve 2D-3D matching
  - Equirectangular to 3D space
  - Undistorted images
    - Warped panorama (usually by GC+OM)
      - Vanishing lines aligned with MW axis
        - LayoutNet, DulaNet, HorizonNet
      - **Computational expensive!**
    - E2P/A2P projections
      - Highlighting structures
        - DulaNet, AtlantaNet
      - Minimal computational cost but requires previous warping



$$G_h(\theta,\gamma) = \begin{cases} x = h/\tan\gamma * \cos\theta \\ y = h/\tan\gamma * \sin\theta \\ z = h \end{cases} \qquad d = \frac{h}{\tan\gamma}$$

Correspondence between angles and 3D points for Atlanta World scenes

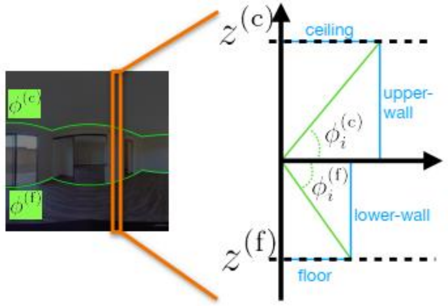


LayoutNet: warping to MW direction

AtlantaNet: A2P transform

Data-driven layout: pre-processing

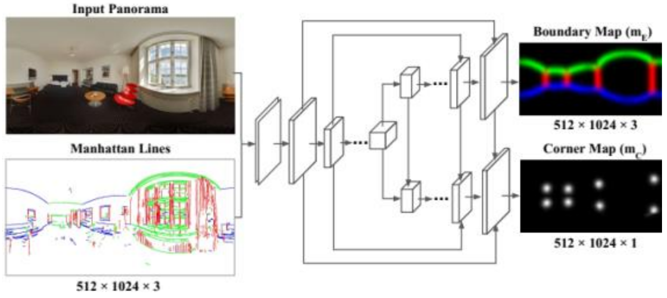
- Why alignment: indoor equirectangular case
  - Image corners and boundaries match 3D points only if image is aligned
- **Basic assumption: image aligned to gravity direction**
  - Less restrictive than Manhattan World alignment
  - Avoid full pre-processing
  - Commonly verified in almost all public dataset
  - Often automatically performed by capture hardware



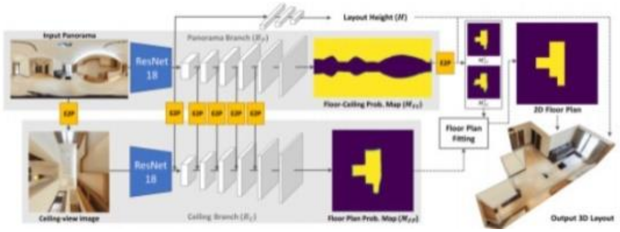
Bifuse++ – Wang TPAMI2022

Data-driven layout: prediction

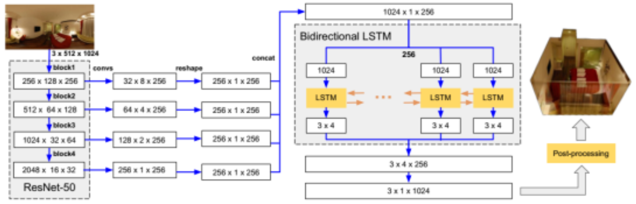
- Input
  - Aligned equirectangular image or its projections
- Encoder-decoder scheme
  - ResNet, SegNet, …
  - Bottleneck and expansion to original resolution
- Network output/ground truth
  - **LayoutNet 2018, HorizonNet 2019**
    - Corners position in image space
    - Wall-ceiling and wall-floor boundaries in image space
  - **DulaNet 2019, AtlantaNet 2020**
    - 2D shape on the floorplan + walls height



LayoutNet: corner and boundary map



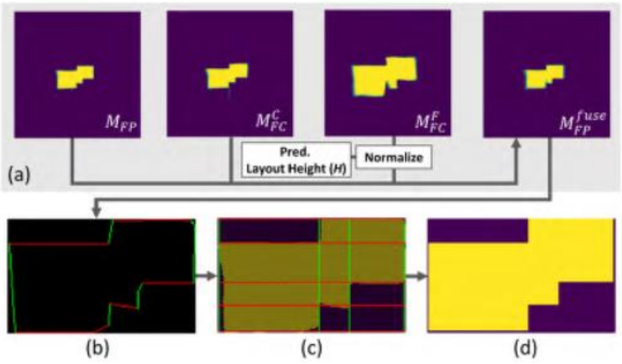
DulaNet: deep learning framework



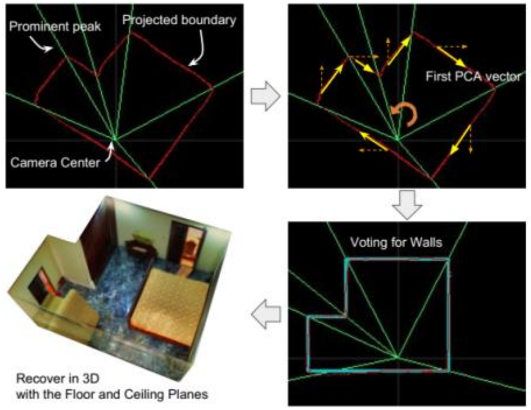
HorizonNet: corners position and W-C W-F edges

Data-driven layout: post-processing

- 2D regularization
  - **Noisy or uncomplete shapes**
  - Room shape on a 2D floorplan
  - Manhattan World prior
    - Walls are regressed and clustered into horizontal and vertical lines
    - Walls are fitted from corners position and ceiling/floor boudaries
  - **NB. Not data-driven! heuristic approach**
- 3D extrusion
  - Layout height
  - MW/ Atlanta World model : single height
  - Vertical walls model: multiple heights



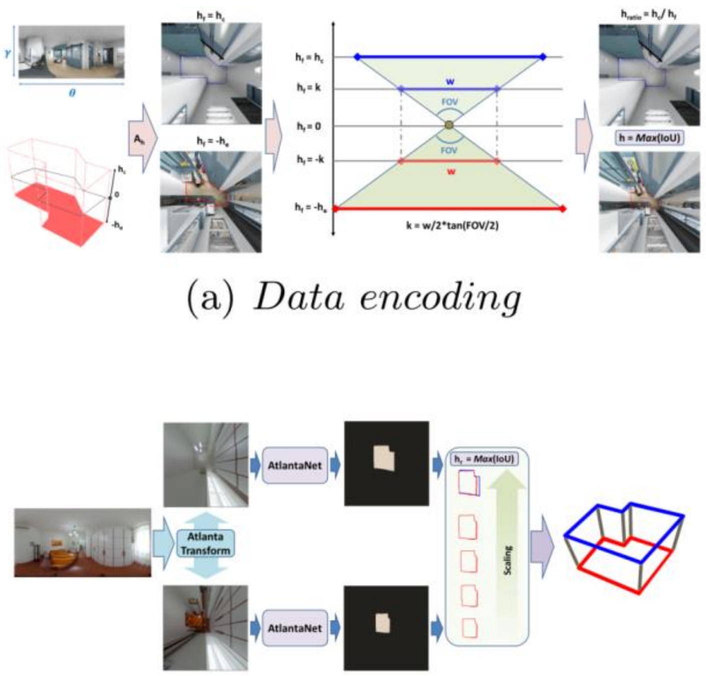
DulaNet: regression to H/V lines



HorizonNet: voting scheme

Layout prediction example: AtlantaNet

- No Manhattan World pre and post-processing
  - *Atlanta World* model
    - Admit non-right angles, curved walls, etc.
    - Constrains: Horizontal floor and ceiling
- Panoramic image -> two horizontal projections
  - Undistorted 2D room footprint from ceiling map
  - Height layout encoded into floor/ceiling IoU ratio



(a) *Data encoding*

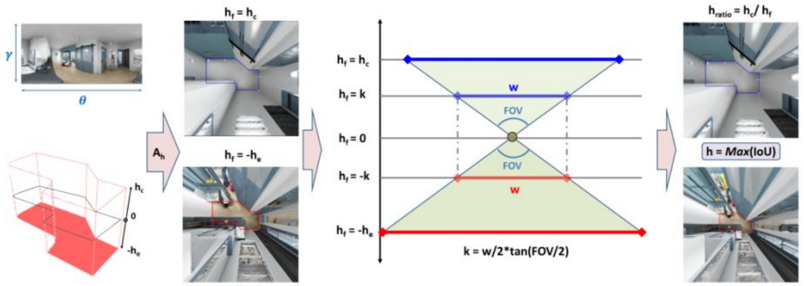
(b) *Layout recovery*

AtlantaNet – Pintore ECCV 2020

AtlantaNet: data encoding

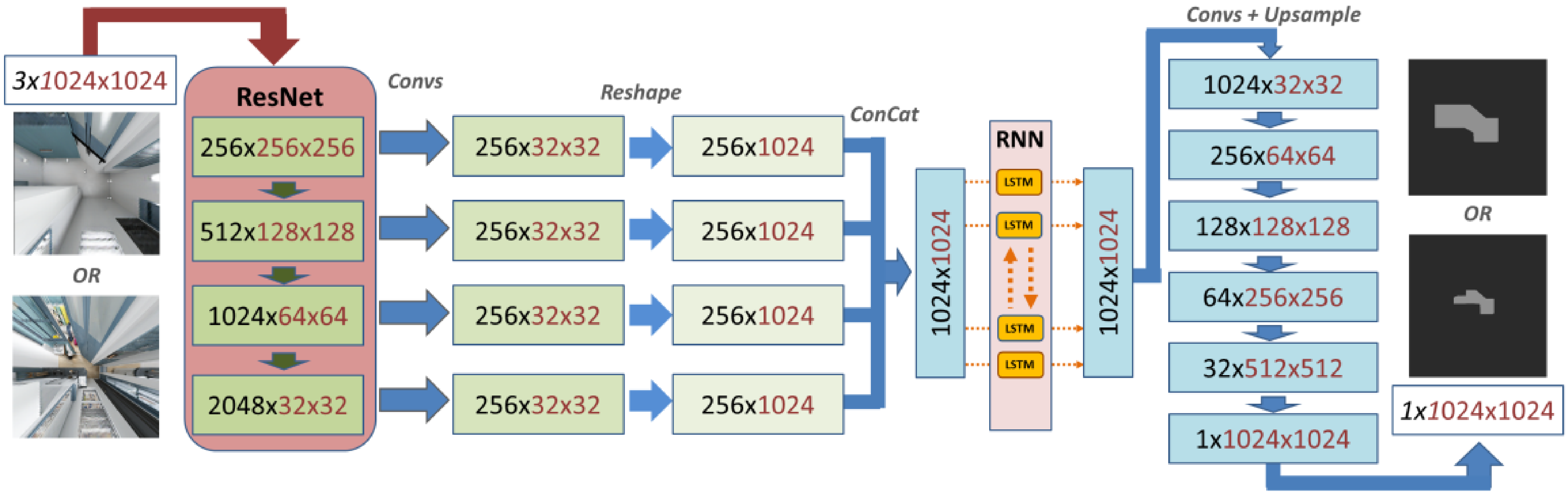
- Atlanta Transform $\mathbf{A_h}$
  - Maps image points in 3D space as if their height was $\mathbf{h_f}$
  - Generated two tensors
  - $\mathbf{h_f}$ can assume only two values
    - $\mathbf{h_e}$ (constant): floor plane distance
    - $\mathbf{h_c}$ (unknown): ceiling plane distance
- Room height (h_c- h_e) is determined by $\mathbf{h_r}$
  - Ratio between the ceiling and the floor shape
  - *V*alue that matches the floor shape with the ceiling shape

$$A_h(\theta, \gamma, h_f) = \begin{cases} x = h_f / \tan\gamma * \cos\theta \\ y = h_f / \tan\gamma * \sin\theta \\ z = h_f \end{cases}$$



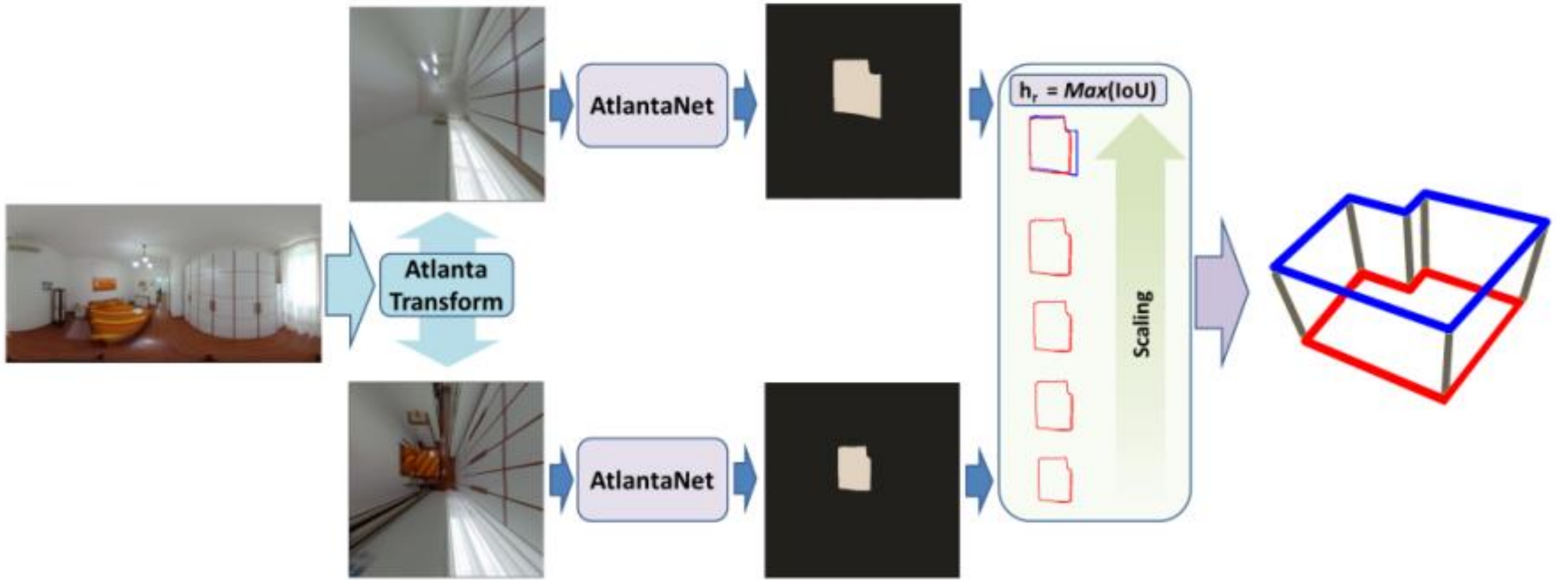
(a) *Data encoding*

AtlantaNet: Network architecture



- Ceiling and floor inferred separately (joined by training)
- Direct feature fusion from floor and ceiling not possible
  - Requires previous knowledge of the scale (h_r)

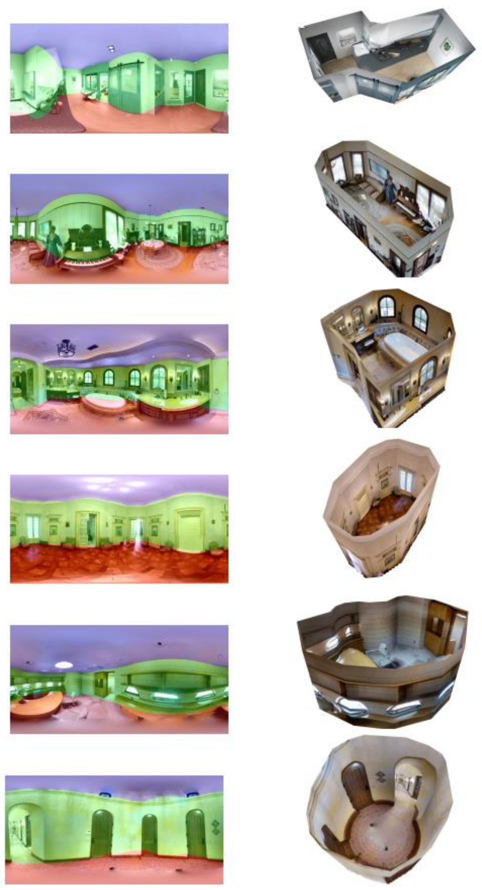
AtlantaNet: training and inference



- Binary cross entropy on mask and its gradient
- Random feeding of floor and ceiling: augmentation
- Inference: ceiling contour and layout height from floor/ceiling shape ratio

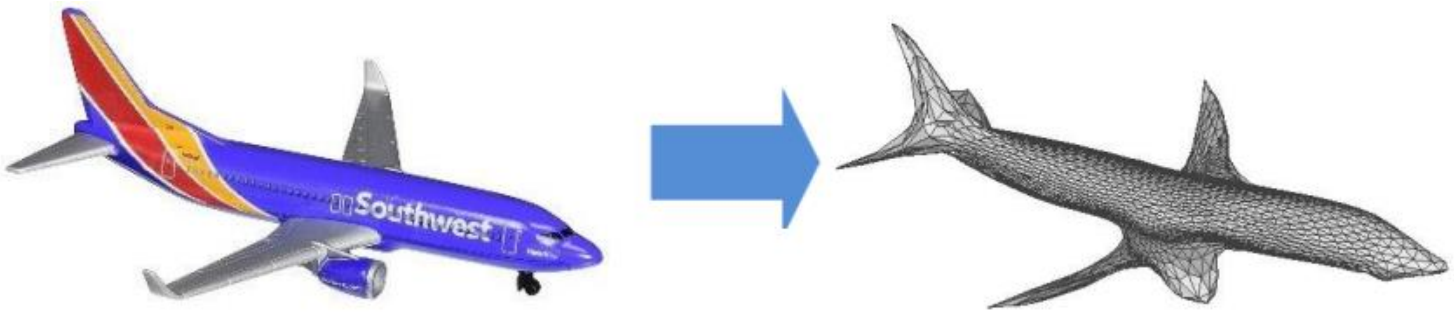
AtlantaNet: example results

- Public Datasets
  - PanoContext
  - Matterport3D
  - Stanford2D3DS
  - Structured3D
  - AtlantaLayout



Generic room shape reconstruction

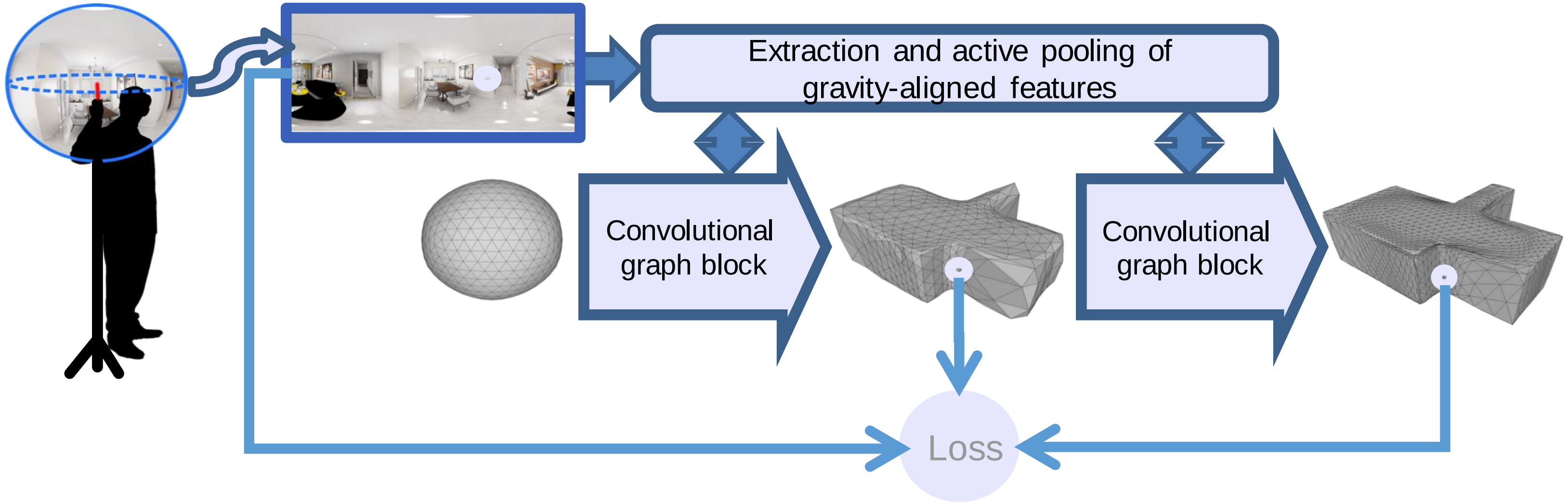
- Data-driven solutions: object surface reconstruction
  - e.g., Wang et al. [2018], Gkioxari et al. [2019], Smith et al. [2019]
  - Progressive deformation of a sphere according to image features



N. Wang et al., "Pixel2Mesh: 3D Mesh Model Generation via Image Guided Deformation," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 43, no. 10, pp. 3600-3613, 1 Oct. 2021, doi: 10.1109/TPAMI.2020.2984232.

Room shape as watertight mesh

- Indoor reconstruction target: modeling arbitrary shape rooms
- Deep3Dlayout: watertight 3D mesh representing room shape
  - Handling curved walls, sloped ceilings, domes

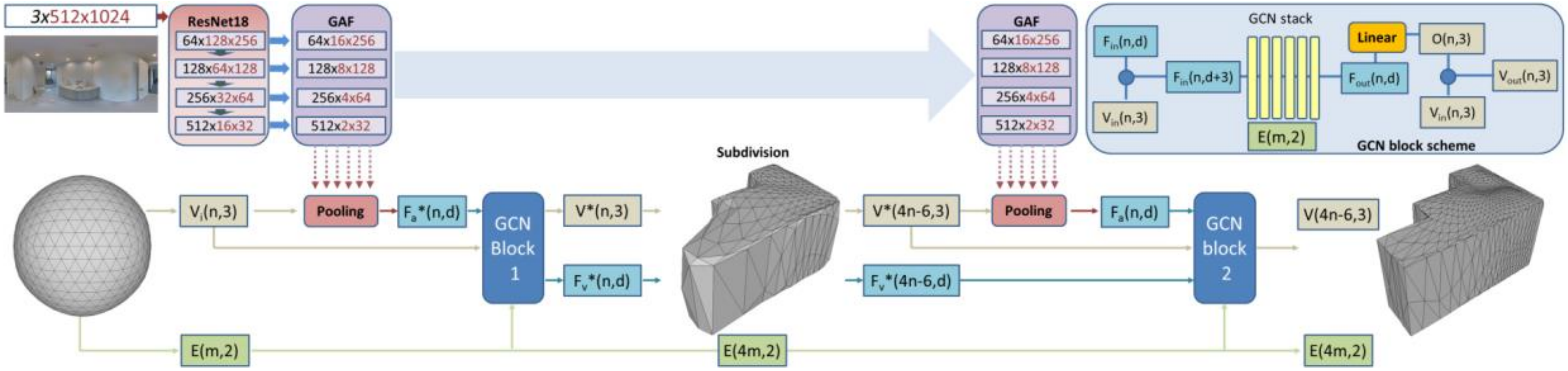


Deep3Dlayout – Pintore Siggraph Asia 2021 TOG

Speaker: Giovanni Pintore

Deep3DLayout: network architecture

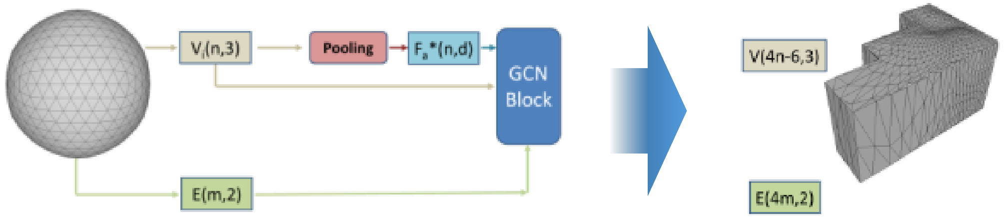
- Indoor layout as a 3D graph-encoded object
- Association of indoor panoramic features to 3D vertices
- Domain-specific loss function

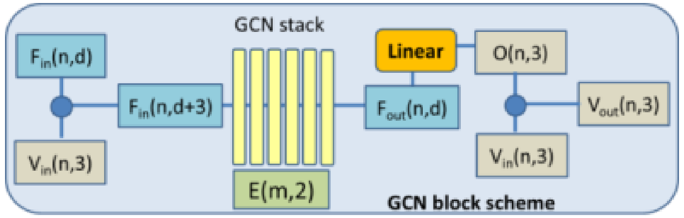


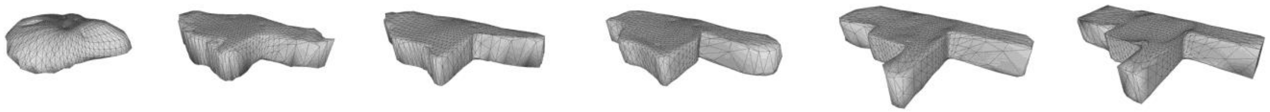
Speaker: Giovanni Pintore

Deep3DLayout:3D graph-encoded object

- Topological model
  - Closed 3D surface
    - Triangulated mesh
- 3D graph-encoded object
  - Vertices V(n,3) and features F(n,d)
  - Connectivity E(m,2)
- Layout by mesh deformation
  - Sequence of 2 GCN blocks
  - Driven by associating image feature
  - Coarse to fine approach



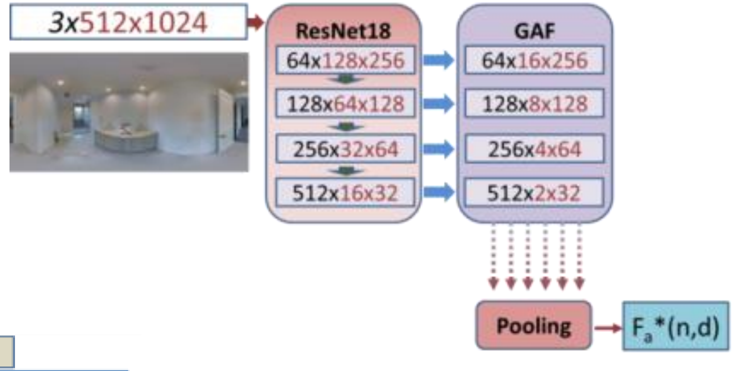


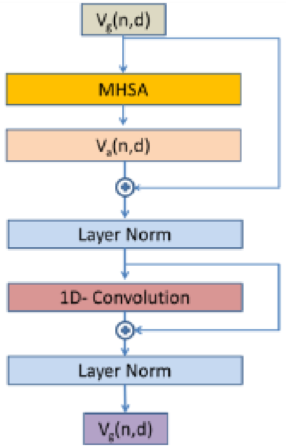


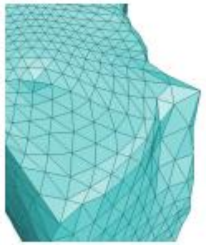
JUNE 18-22, 2023
CVPR
VANCOUVER, CANADA

Deep3Dlayout: features pooling

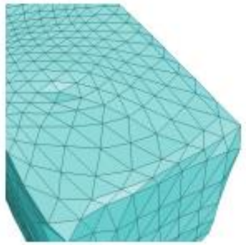
- Gravity aligned features
  - Anisotropic contractive encoding (see SliceNet)
    - Targeted to indoors
  - Maximize gathered information
  - Minimize interpolation effects
- Spherical pooling
  - Self-attention module (MHSA)
    - Short and long range relationships
    - Cope with major occlusion problems







without MHSA



with MHSA

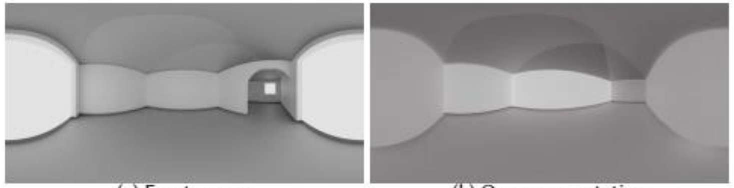
Deep3Dlayout: model and loss functions

- Combination of data and regularization terms
  - Plausible reconstruction

$$\mathcal{L}_{data} = \lambda_c \mathcal{L}_{pos} + \lambda_n \mathcal{L}_{norm} + \lambda_{sh} \mathcal{L}_{sharp}$$

$$\mathcal{L}_{reg} = \lambda_e \mathcal{L}_{edge} + \lambda_s \mathcal{L}_{smooth}$$

- Targeted model
  - Smooth surfaces joining at sharp edges
  - Less restrictive than common indoor priors
    - MWM, IWM, AWM
  - Allows other common structures
    - curved walls, vaults, domes



(a) Empty room (b) Our representation

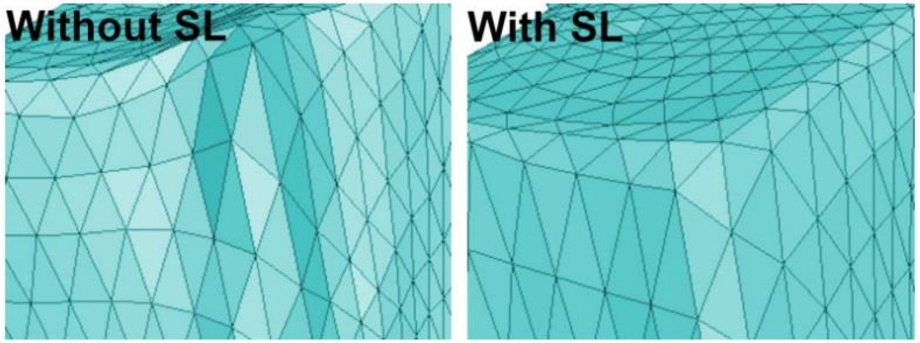
Deep3Dlayout: model and loss functions

- Data terms
  - Positional and orientation loss
  - Sharpness loss

$$\mathcal{L}_{pos} = |P|^{-1} \sum_{p \in P} \|p - N(Q,p)\|^2 + |Q|^{-1} \sum_{q \in Q} \|q - N(P,q)\|^2$$

$$\mathcal{L}_{norm} = -|P|^{-1} \sum_{p \in P} \left| n_p \cdot n_{N(Q,p)} \right|^2 - |Q|^{-1} \sum_{q \in Q} \left| n_q \cdot n_{N(P,q)} \right|^2$$

$$\mathcal{L}_{sharp} = |S_e|^{-1} \sum_{q \in S_e} \|q - N(P,q)\|^2$$



Difference in using or not the sharpness loss (SL)

Deep3Dlayout: model and loss functions

- Data terms
  - Positional and orientation loss
  - Sharp loss
- Regularization terms
  - Edge loss
  - Smooth loss

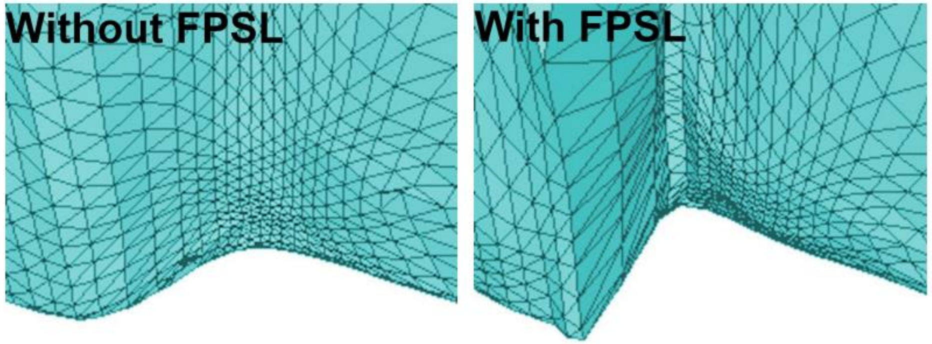
$$\mathcal{L}_{pos} = |P|^{-1} \sum_{p \in P} \|p - N(Q,p)\|^2 + |Q|^{-1} \sum_{q \in Q} \|q - N(P,q)\|^2$$

$$\mathcal{L}_{norm} = -|P|^{-1} \sum_{p \in P} \left| n_p \cdot n_{N(Q,p)} \right|^2 - |Q|^{-1} \sum_{q \in Q} \left| n_q \cdot n_{N(P,q)} \right|^2$$

$$\mathcal{L}_{sharp} = |S_e|^{-1} \sum_{q \in S_e} \|q - N(P,q)\|^2$$

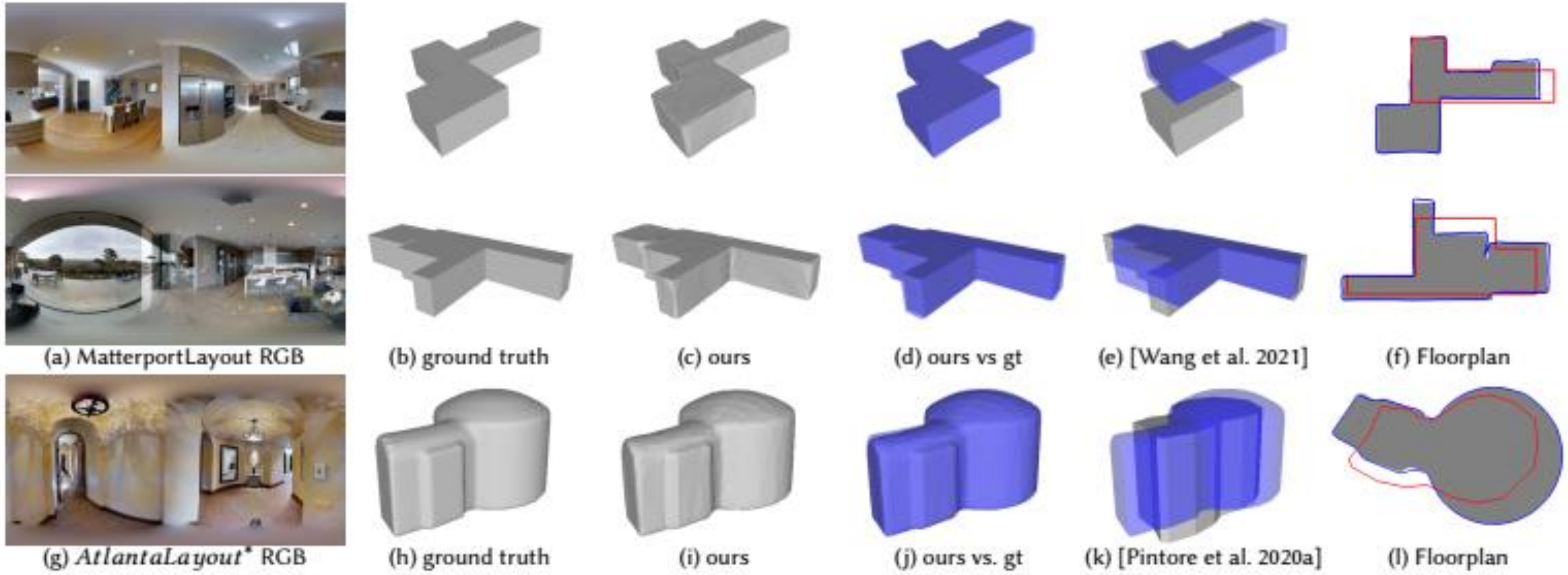
$$\mathcal{L}_{edge} = |E|^{-1} \sum_{(i,j) \in E} \|v_i - v_j\|^2$$

$$\mathcal{L}_{smooth} = |V|^{-1} \sum_{i \in V} e^{-|K_{H\,i}|} |K_{H\,i}|$$



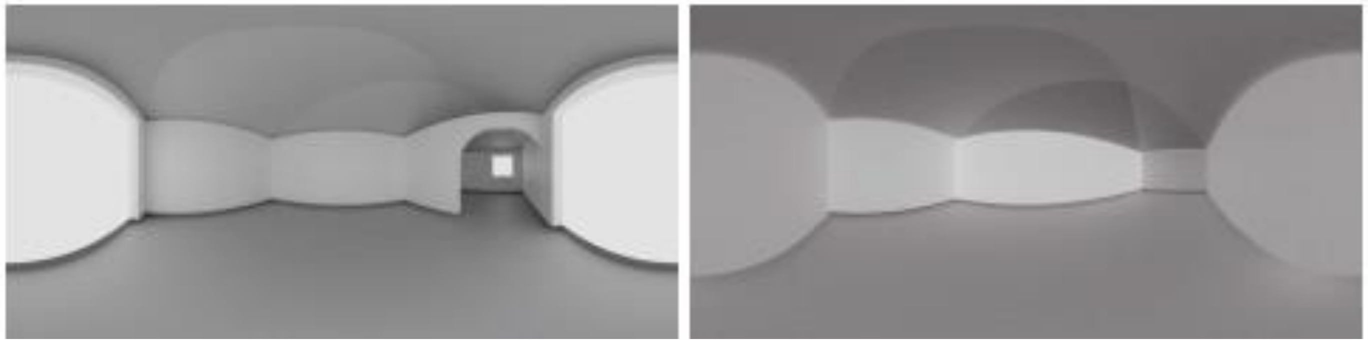
Difference in using or not the feature-preserving smoothness loss (FPSL)

Deep3Dlayout: example results



Room shape as watertight mesh: limitations

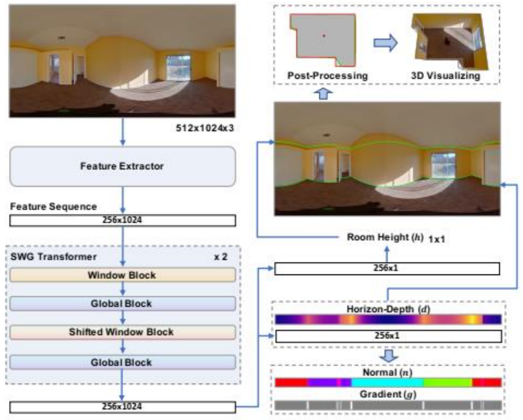
- Promising solution but important limitations
  - Indoor structure often are not closed, watertight mesh
  - Output is not really structured
  - GCN are not stable, transfer-learning limits



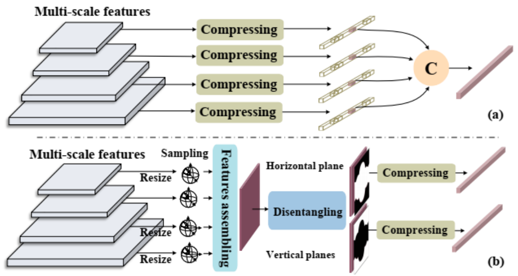
Empty room structure *Watertight mesh approximation*

Structured layout: latest trends

- Geometry-aware transformers
  - Vertical compression (as SliceNet, HoHoNet, Deep3DLayout)
  - Custom transformer improving LSTM or MHSA
    - Encoding local and global window blocks
  - Horizon depth and planar-aware losses (as LED2-Net)
- Disentangling Orthogonal Planes
  - Multi-scale features disentangled as horizontal and vertical MW planes before compression and gated fusion
  - Cross-scale Distortion Awareness (PanoFormer)
- MW pre post processing still adopted (same of HorizonNet)



LGT-Net Jiang CVPR2022

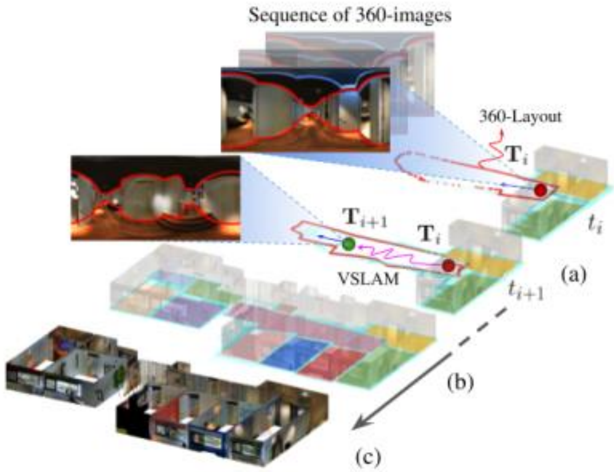


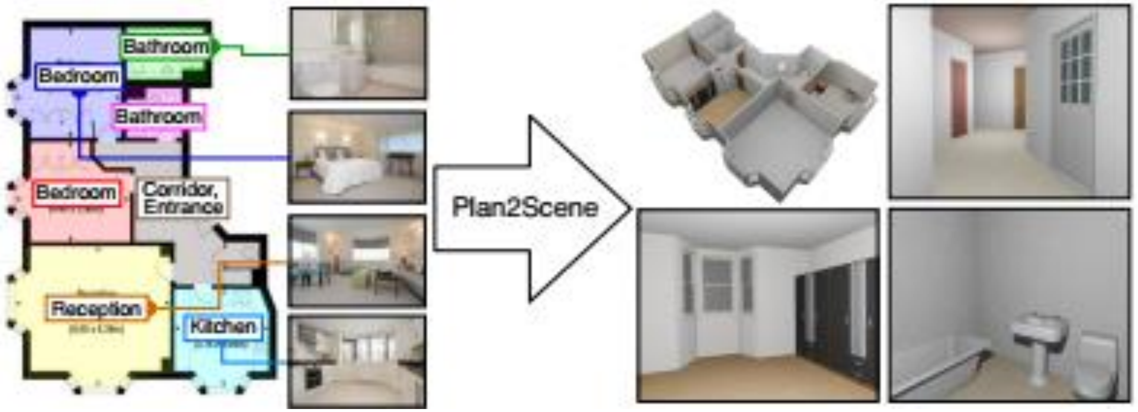
DOPNet Jiang CVPR2023

Room layout reconstruction: summary

- Geometry of the bounding permanent surfaces
  - 2D – 3D task (corners, edges, planes, meshes)
  - Common in panoramic world: room from a single 360 image
  - Rooms concept can be extended to larger and complex structures (next)
- Open problems
  - Occlusions: from clutter and from structure itself
    - Manhattan regularization and completion still common
  - Output model: truly representative of the structures and usable (e.g., CAD-like)
    - Corners and continous edges: very limiting
    - Closed meshes: difficult to structure (walls, floor, ceiling)

HoHoNet - Sun CVPR2021





Next session

Integrated indoor model