Recovering Geometric Representations – Depths and points

D.A. Forsyth,

University of Illinois at Urbana-Champaign

Options

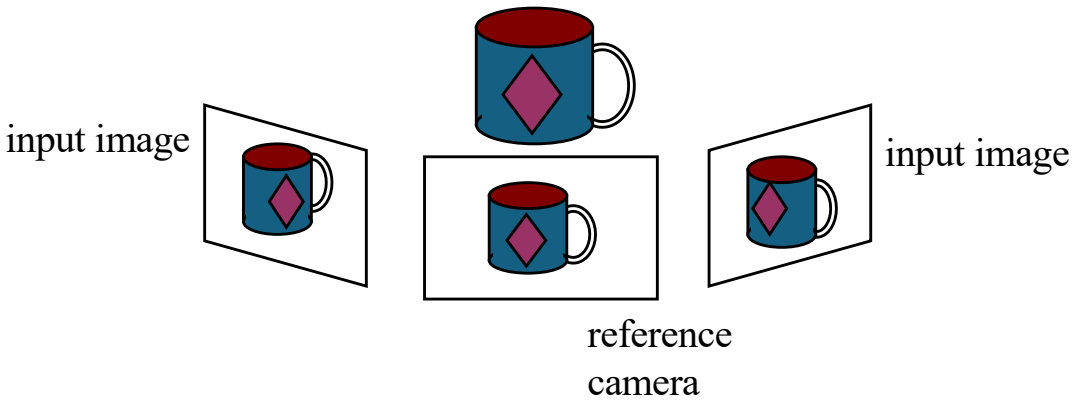
- Here:
  - Multiple depth maps
  - Point clouds
  - Voxel grids
  - Implicit functions
  - Density functions
- Others
  - CSG rep’ns
  - NURBS
  - etc.

Rep'n: Multiple depth maps

- One per image
  - or per cluster of images
  - consistent

- Advantages
  - straightforward to get
  - known how to pass to a triangle mesh (if dense enough)
- Disadvantages
  - resolution issues (eg zooming camera)
  - can be tricky to render cleanly
  - some geometric calculations are hard
    - volume

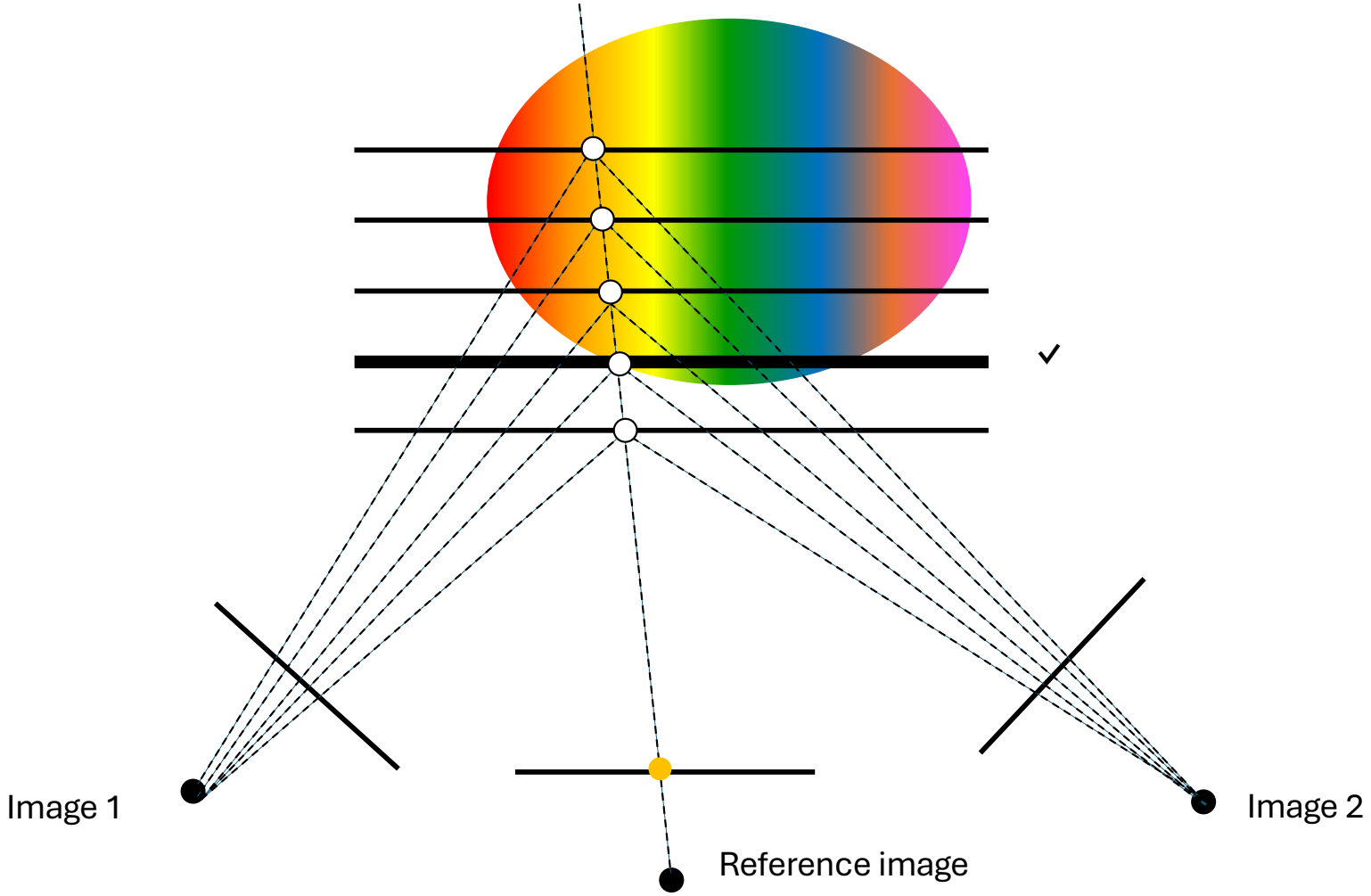
Plane sweep stereo



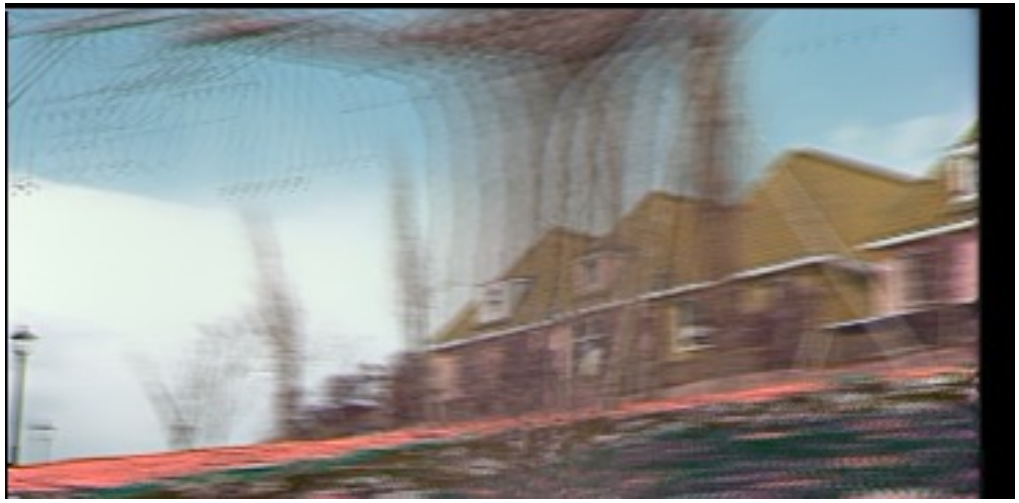
- Sweep plane across a range of depths w.r.t. a reference camera
- For each depth, project each input image onto that plane (homography) and compare the resulting stack of images

R. Collins, A space-sweep approach to true multi-image matching, CVPR 1996

Plane sweep stereo: Key idea



Plane sweep stereo: Fast implementation



- For each depth plane
  - Compute homographies projecting each image onto that depth plane
  - For each pixel in the composite image stack, compute the variance
- For each pixel, select the depth that gives the lowest variance

R. Yang and M. Pollefeys, Multi-Resolution Real-Time Stereo on Commodity Graphics Hardware, CVPR 2003

COLMAP MVS

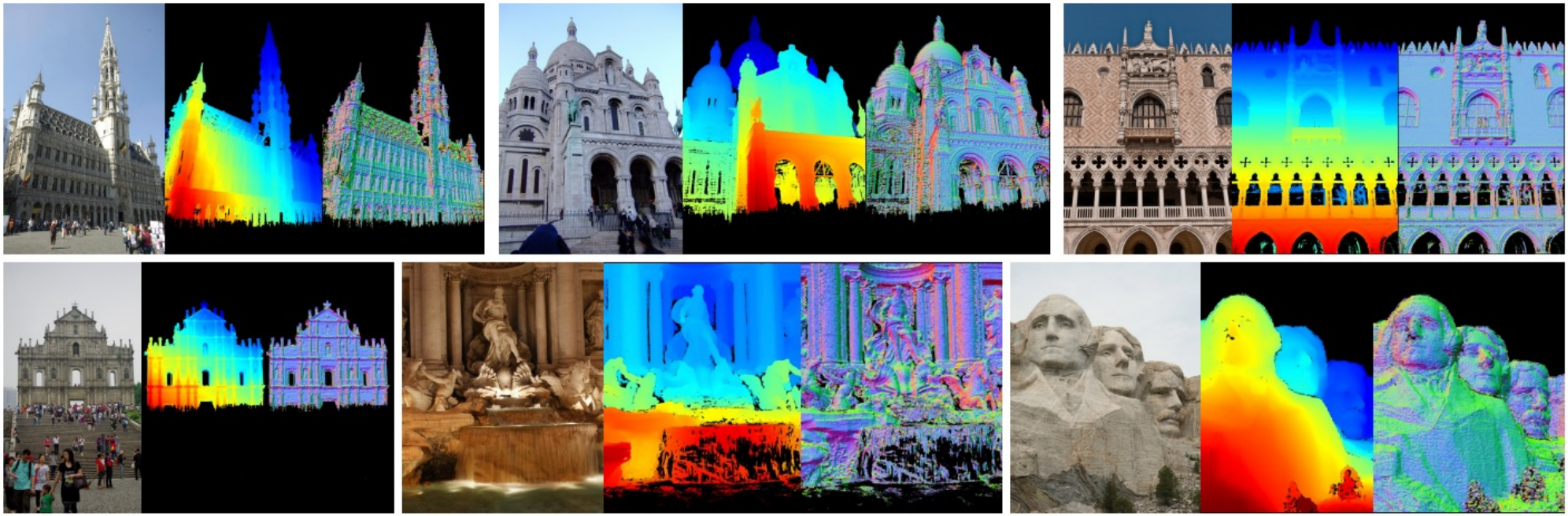


Fig. 6. Reference image with filtered depths and normals for crowd-sourced images.

J. Schonberger et al. Pixelwise View Selection for Unstructured Multi-View Stereo. ECCV 2016

Results video

DepthAnything V3 depth maps

https://github.com/ByteDance-Seed/Depth-Anything-3

Options

- Here:
  - Multiple depth maps
  - Point clouds
  - Voxel grids
  - Implicit functions
  - Density functions
  - Primitives
- Others
  - CSG rep’ns
  - NURBS
  - etc.

Rep’n: point cloud

- Large number of points in 3D
  - with attached color

- Advantages
  - easy to render
  - easy to construct
  - known how to pass to triangle mesh (if dense enough)
- Disadvantages
  - many geometric computations quite hard
    - volume, intersection, etc

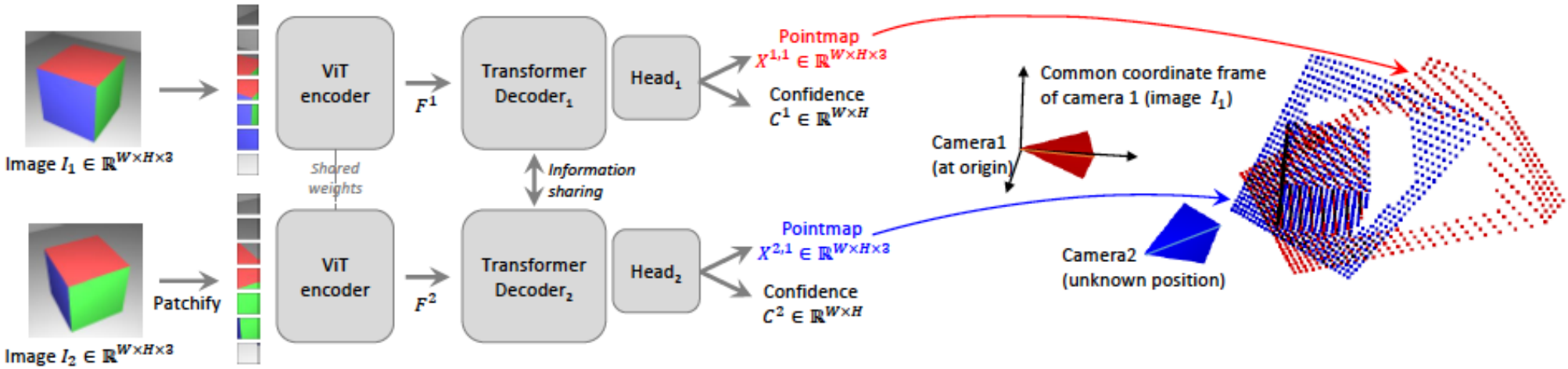
Patch-based multi-view stereo

1. Detect keypoints
2. Triangulate a sparse set of initial matches
3. Iteratively expand matches to nearby locations
   (because the surface is made of patches)
4. Use visibility constraints to filter out false matches

Y. Furukawa and J. Ponce, Accurate, Dense, and Robust Multi-View Stereopsis, CVPR 2007.
PMVS software

DUST3R – I

- Accept two images, make a point map for each



DUSt3R: Geometric 3D Vision Made Easy, Wang et al, 2024

Point maps

- one (3D) point per pixel
  - so WxHx3
- I1 is in canonical frame, both calibrated
  - point map at (i, j) with depth z(i, j) contains

$$\mathcal{K}^{-1} \begin{bmatrix} iz(i,j) \\ jz(i,j) \\ z(i,j) \end{bmatrix}$$

- point map for I2 is in I1 frame

Now train

- Simplest
  - accept image pair
  - produce pointmap
- Train w/ ground truth
  - recalling I2 pointmap is in I1’s frame
  - lots of data of form (image pair, depthmap pair)
- Loss:
  - scaled L2 regression loss between prediction, g.t.
  - weighted by confidence term

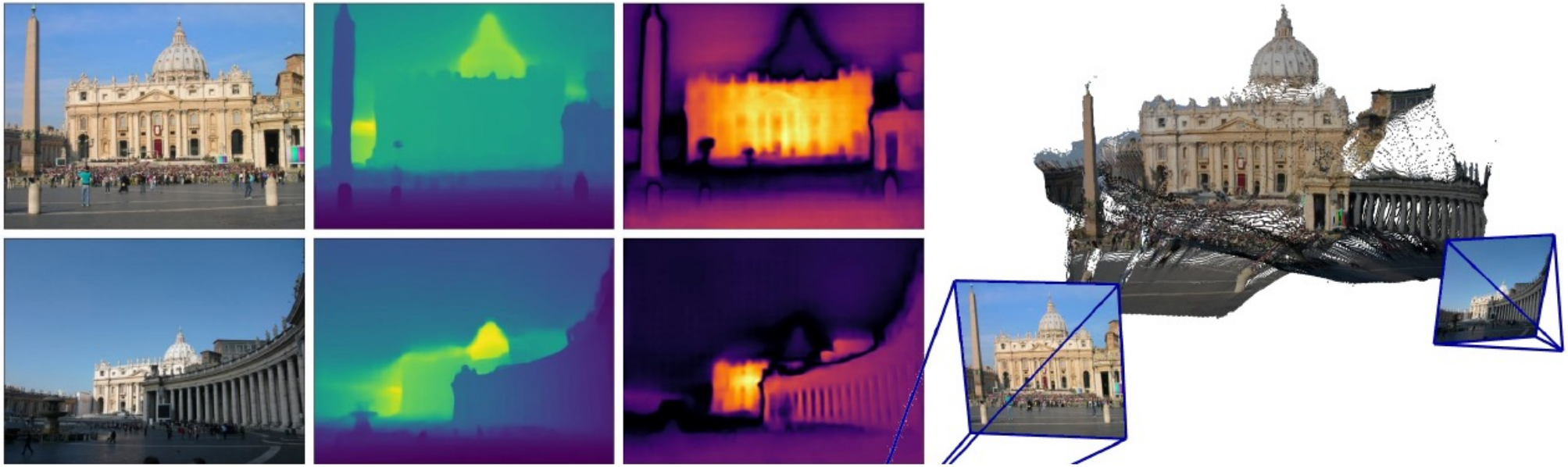


Image Depth Confidence Point cloud

DUSt3R: Geometric 3D Vision Made Easy, Wang et al, 2024

Downstream

- I1, I2 pixel correspondences
  - find nearest neighbors in pointmaps
- Camera intrinsics
  - pointmap is in I1’s frame
    - easy optimization
- Relative camera pose
  - predict twice using I1, I2 then I2, I1
  - register the two predictions
    - first in I1’s frame, second in I2’s frame
- Absolute pose
  - register

More than two frames

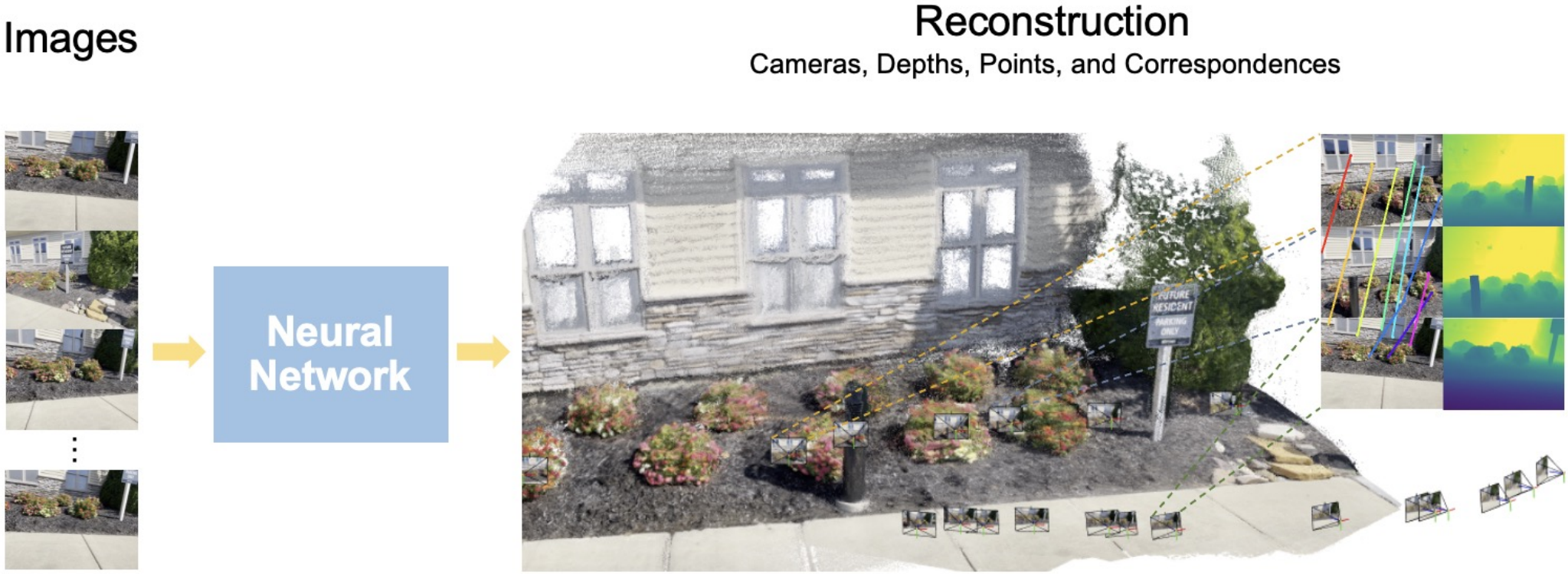
- Like SFM pipeline:
  - find pairs with many overlapping points
  - reconstruct for every such pair
  - register reconstructions

Image Depth Confidence

Point cloud

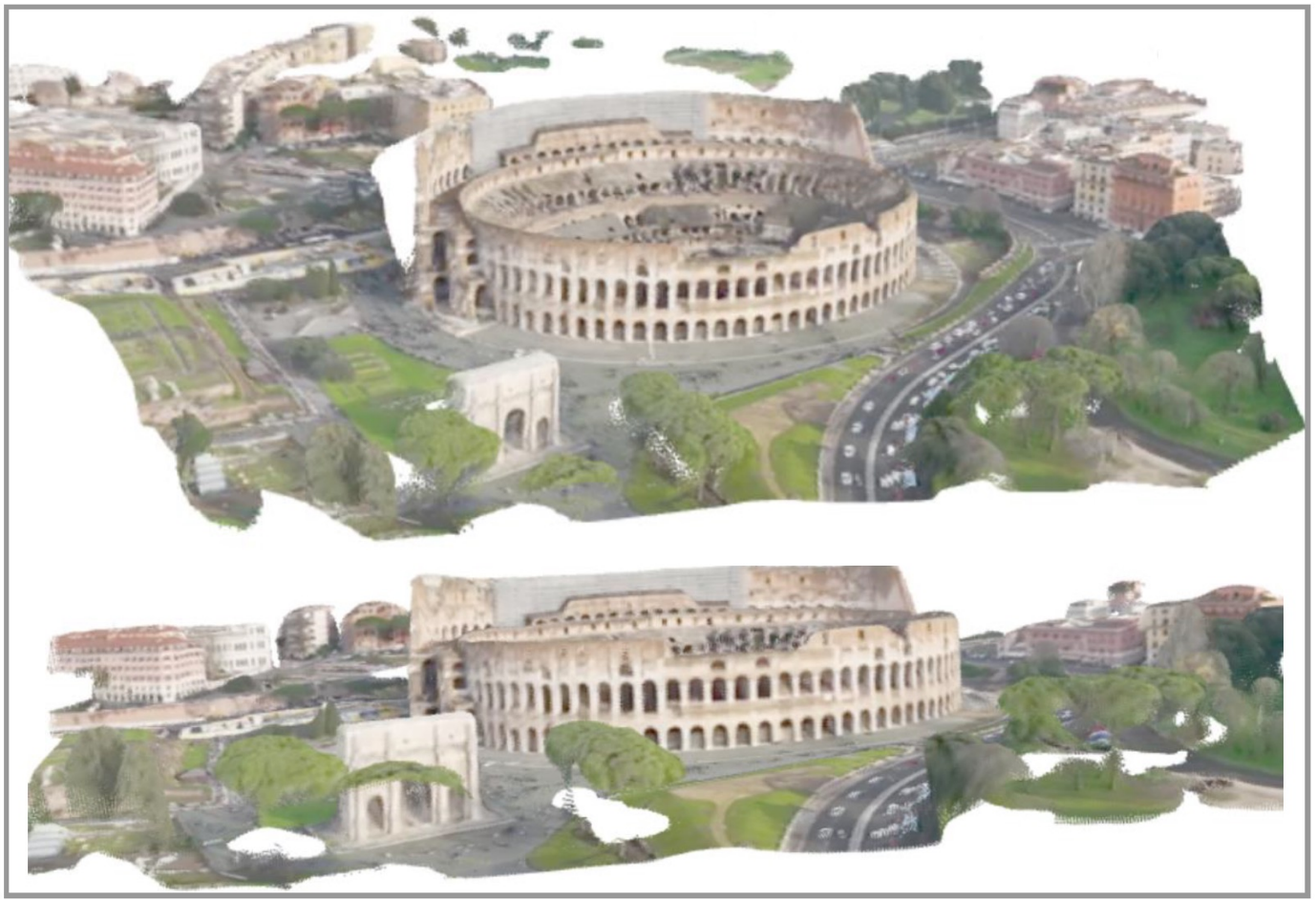
DUSt3R: Geometric 3D Vision Made Easy, Wang et al, 2024

VGGT



https://vgg-t.github.io

VGGT



https://vgg-t.github.io

https://github.com/ByteDance-Seed/Depth-Anything-3