NeurIPS 2022 Spotlight

# ElasticMVS: Learning elastic part representation for self-supervised multi-view stereopsis


**Jinzhi Zhang[1,2*], Ruofan Tang[1,3], Zheng Cao[4], Jing Xiao[5], Ruqi Huang[2§] and Lu Fang[1§]**

[1]Department of Electronic Engineering, Tsinghua University
[2]Tsinghua Shenzhen International Graduate School
[3]Dept. of Automation, Tsinghua University, [4]BirenTech Research, [5]Pingan Group

*zjz22@mails.tsinghua.edu.cn

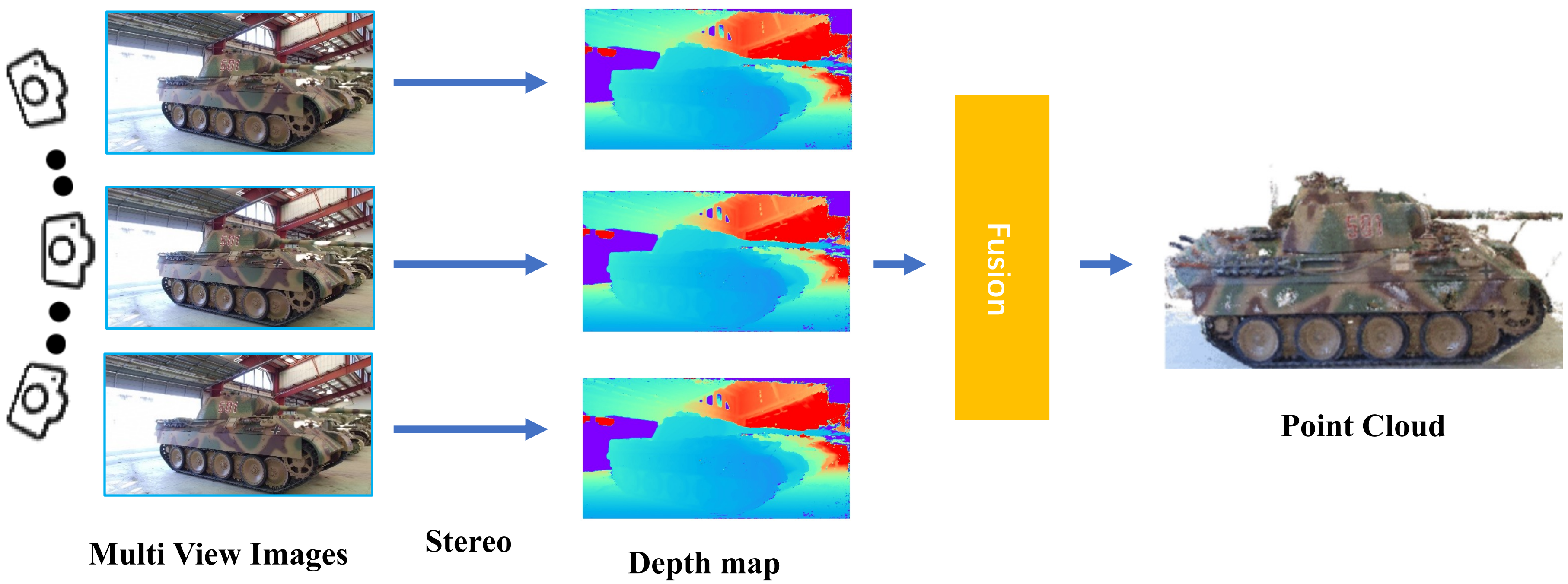

http://www.luvision.net

Tsinghua University

tSinghua vIsual intelliGence and coMputational imAging lab

NEURAL INFORMATION PROCESSING SYSTEMS

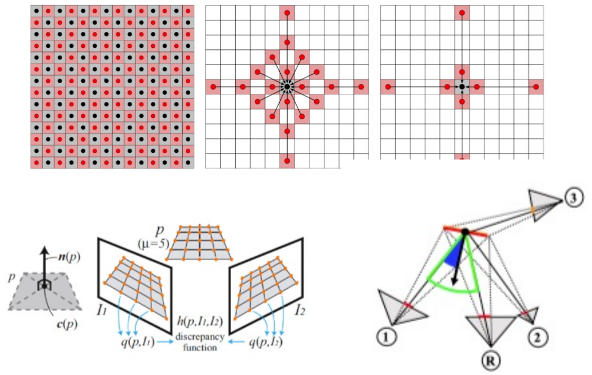
# Background: Multi-view stereopsis (MVS)



# Previous works in MVS

## Traditional MVS

- Calculate photometric consistency
  - Measures on patches locally
  - Robust similarity function (NCC)
- Random sampling and propagation

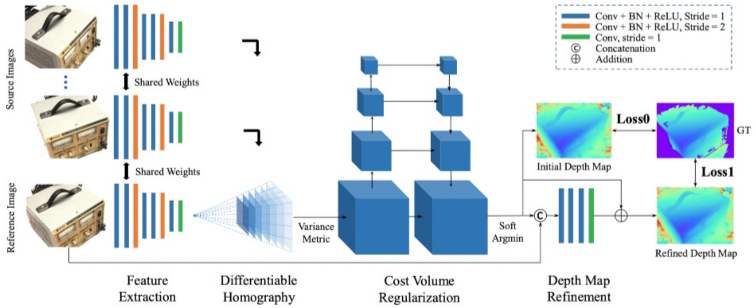


Gipuma [Galliani et al. 2015]

COLMAP [Schönberger et al 2016]

## Supervised MVS

- Construct 3D volume
- Project 2D features to 3D
- Learning through supervision

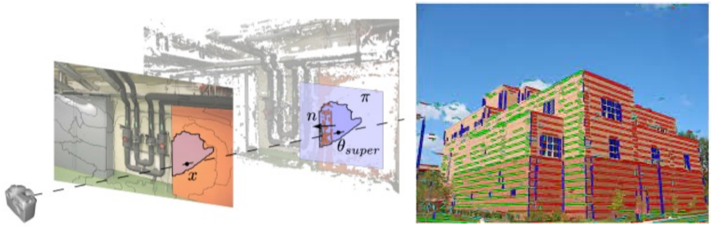



MVSNet [Yao et al. 2018]

Consistency [Khot et al 2019]

## Semantic MVS

- Handcraft semantic detection
  - Superpixel
  - Line\plane detection
- RANSAC primitive fitting



TAPAMVS [Romanoni et al. 2019]

Urban [Micusík et al 2010]

**Handcraft** **or** **data-drive**: susceptible to textureless patterns or geometry variations

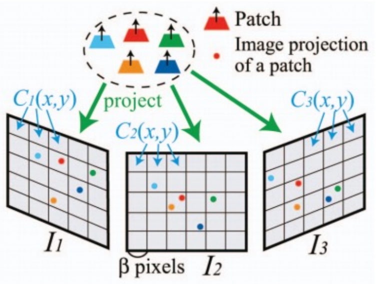
# Bottleneck

## Geometric consistency

- **Local region**
- **No shape prior**

## Semantic segmentation

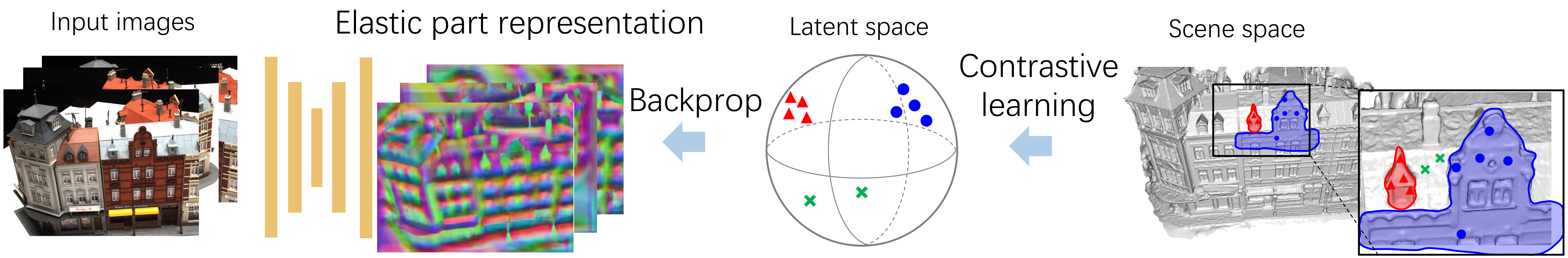
- **No geometric cues**:
  - Scales, shapes and boundaries
- **Lack of training data**




**Our work: Bridge the gap between the two areas.**

# ElasticMVS

A novel elastic part representation encoding part segmentations




- **Geometry-aware**: Encode geometric connectedness, smoothness and boundaries
- **Elastically**: Represent elastically-varying scales, shapes and boundaries
- **Self-supervised**: Learn the representation and estimate per-view depth iteratively

# Problem definition

- Definition
  - **Geometry**: Given an image $x$, find the best depth and normal $(d_p, n_p)$ on each pixel $p \in x$.
  - **Segmentation**: Given a set of images $X$, learn the segmentation $\Pi_\Theta(x)$.
- Optimization goal
  - **Geometry**: Make the photo-consistency loss as lower as possible ($M_s$).
  - **Segmentation**: Make the surface in each segment as smoother as possible ($M_g$).

$$\Theta^{optim} = \mathrm{argmin}_\Theta \sum_{x \in X} \min_{d,n,\Pi} \left\{ \sum_p \left[ \underbrace{M_s\left(d_p, n_p \mid x\right)}_{\text{Photo-consistency loss, used in traditional MVS.}} + \underbrace{M_g\left(d_p, n_p \mid \Pi(x)\right)}_{\text{Surface smoothness loss}} \right] \right\}$$

Intuively, In each segment from the segmented image, the depth is smooth and photometric consistent

# Representation & Learning

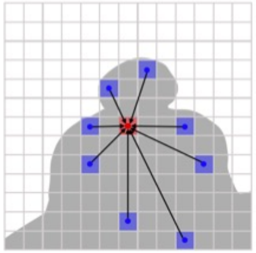
- Elastic Part Representation
  - Find geometrically concentrated areas $S_p$.
  - Representation $z_p$ in the latent feature space is close enough in these areas.
- Learning
  - Compact the representation in the geometric concentrated part.
  - Contrast the representation otherwise.
  - **Training by contrastive learning.**

Fixed

$$\Theta^{optim} = \text{argmin}_{\Theta} \sum_{x \in X} \min_{d,n,\Pi} \left\{ \sum_p \left[ \underbrace{M_s\left(d_p, n_p | \Pi(x)\right)}_{\text{Photo-consistency loss, used in traditional MVS.}} + \underbrace{M_g\left(d_p, n_p | \boxed{\Pi(x)}\right)}_{\text{Surface smoothness loss}} \right] \right\}$$

$$- \sum_p \log \frac{\sum_{p^+ \in S_p} \exp\left(\langle z_p, z_{p^+} \rangle / \tau\right)}{\sum_{q \neq p} \exp\left(\langle z_p, z_q \rangle / \tau\right)}$$

# Inference



- Part-aware propagation
  - gather hypotheses $T_p$ from the same physical surface part.
  - Use our representation to identify these parts.
  - Representation $z_p$ in the latent feature space is close enough in these areas.

$$\mathcal{T}_p = \left\{ q \in R^2 \middle| \|z_p - z_q\| \le \eta, c_q \ge \xi \right\}$$

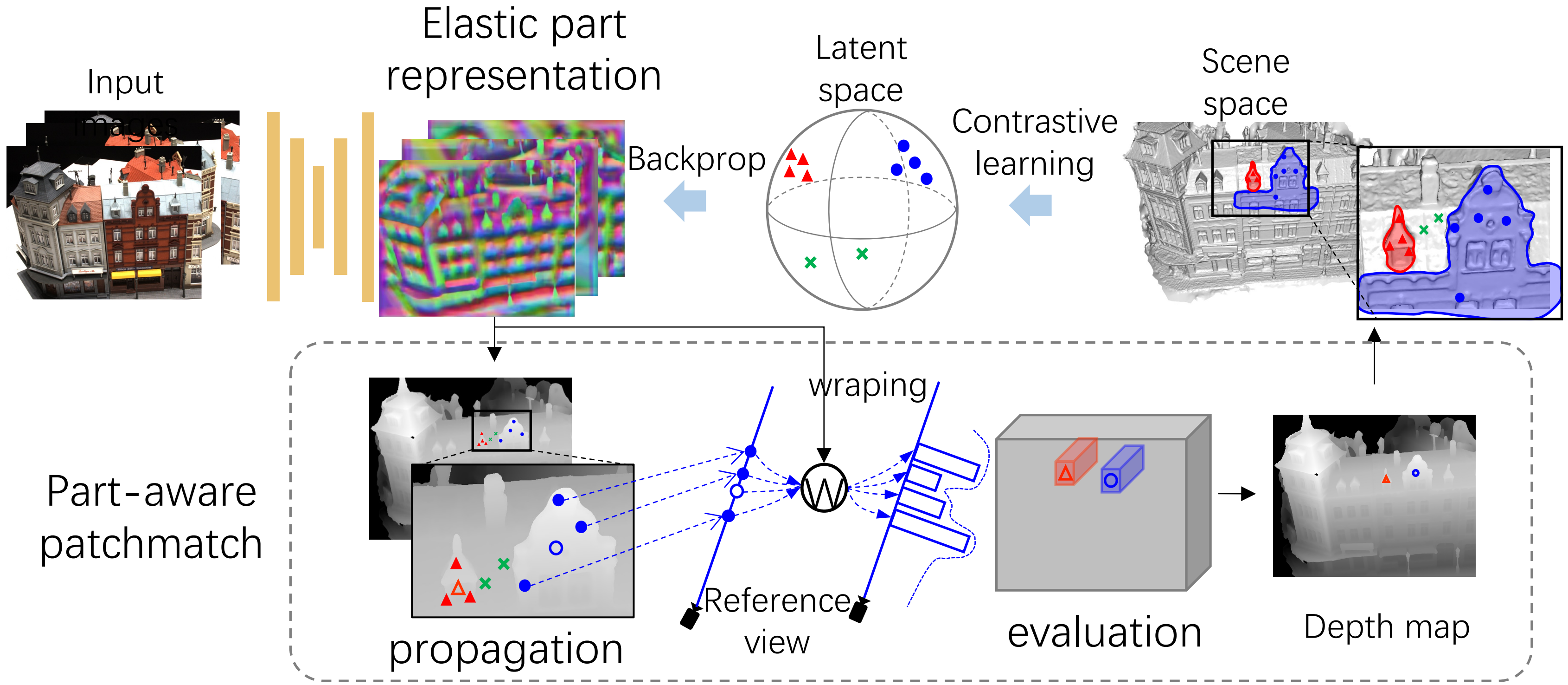
- Part-aware losses
  - ***Part-aware correspondence***: check the photo & representation consistency $M_s(d_p, n_p|x, z)$
  - ***Part smoothness loss***: piecewise smoothness using L1 median loss $M_g\left(d_p, n_p \mid z\right) = \sum_{q \in T_p} \omega_q \left\|e_p - e_q\right\|$

$$\Theta^{optim} = \operatorname{argmin}_{\Theta} \sum_{x \in X} \min_{d,n,\Pi} \boxed{\left\{ \sum_{p} \left[ \underbrace{M_s\left(d_p, n_p \mid x\right)}_{\text{Photo-consistency loss, used in traditional MVS.}} + \underbrace{M_g\left(d_p, n_p \mid \boxed{\text{Fixed}}\right]}_{\text{Surface smoothness loss}} \right\}}$$

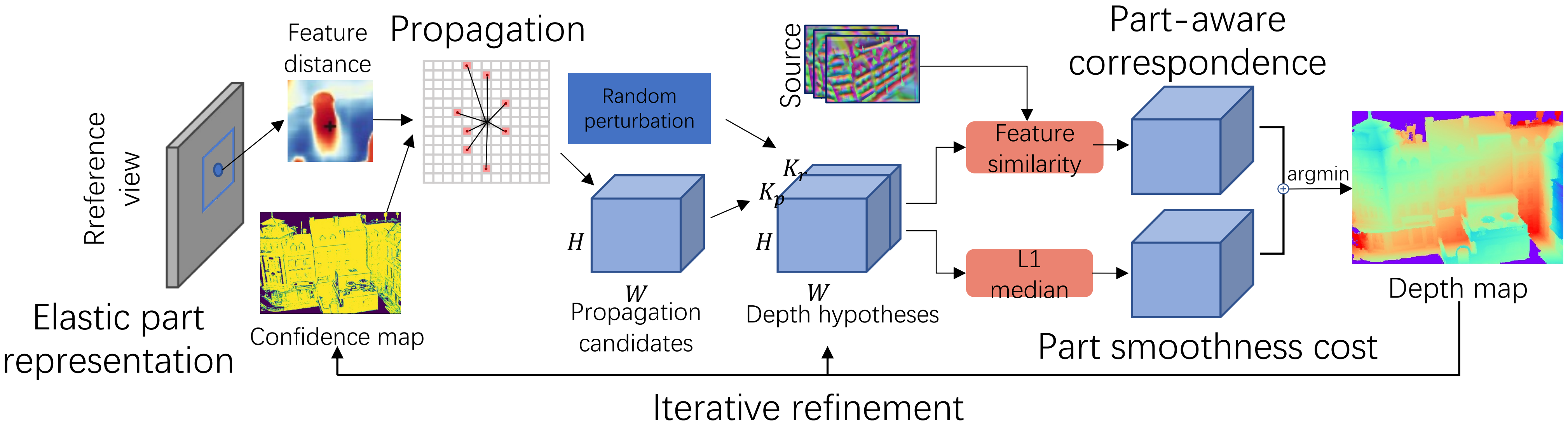
Solved using discretely sampling $\longrightarrow$ $\left(d_p^{\text{opt}}, n_p^{\text{opt}}\right) = \underset{d_p^*, n_p^*}{\operatorname{argmin}} \left\{ M_s(d_p^*, n_p^*|x, z) + \alpha_g \cdot M_g(d_p^*, n_p^* \mid z) \right\}$

# Pipeline

## Training

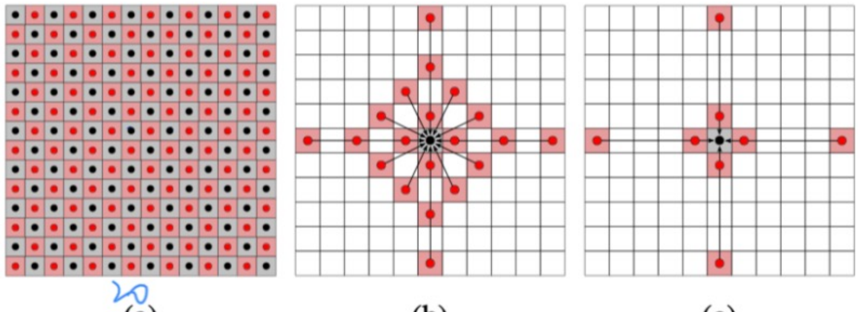



## Inference



# Inference: Different strategy during propagation

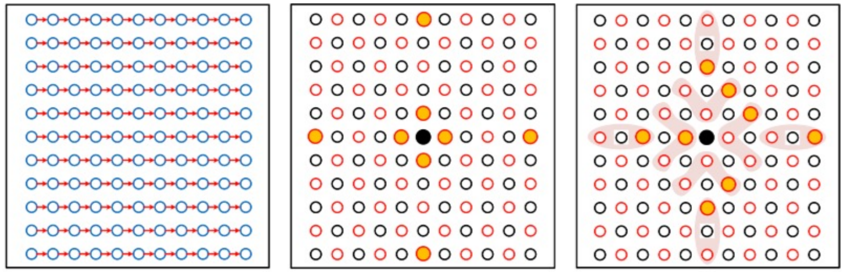
Gipuma: Fixed



(a) (b) (c)

[Galliani et al. 2015]

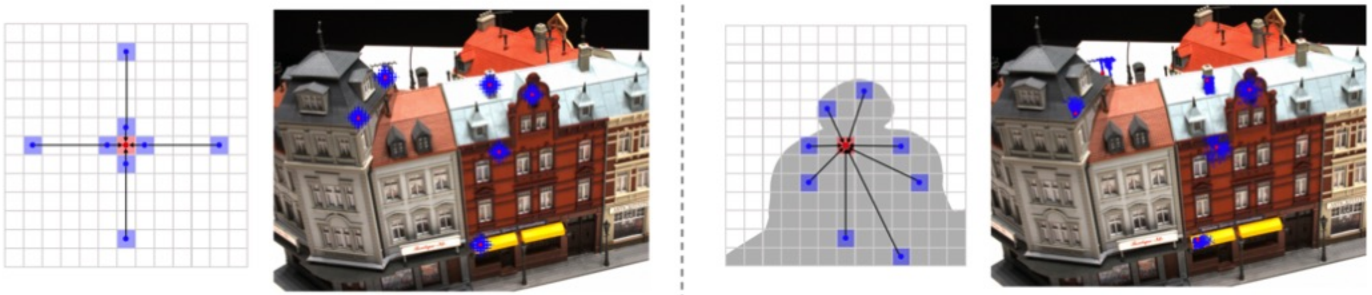
ACMM: Heuristic



(a) (b) (c)

[Xu. CVPR 2019]

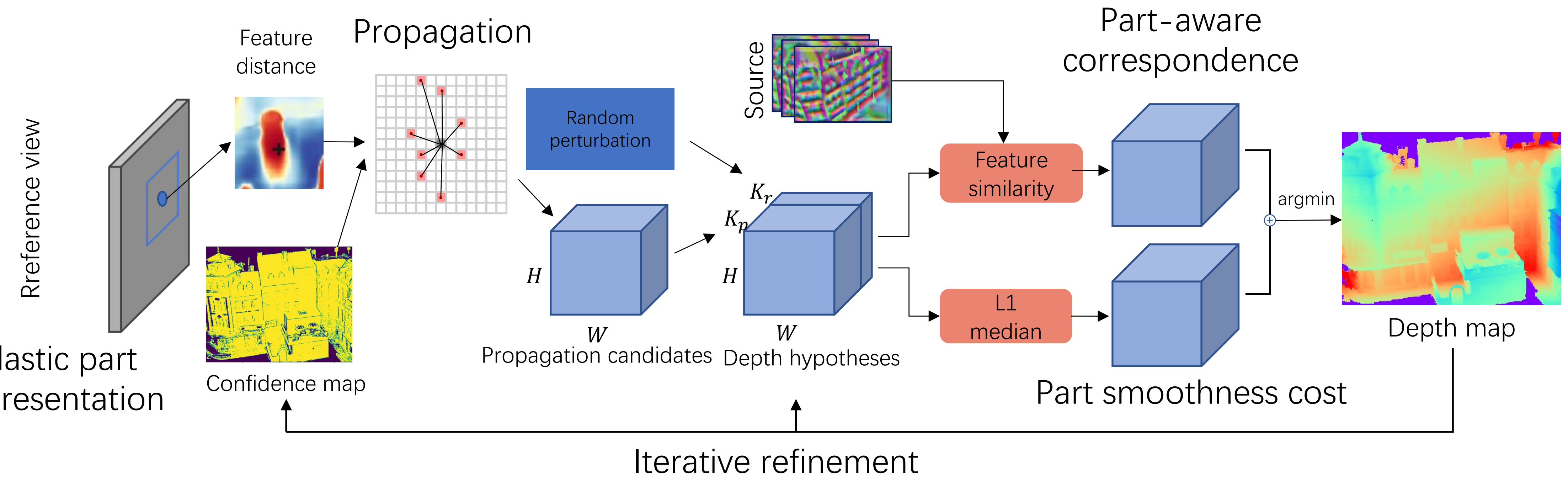
Ours: adaptive



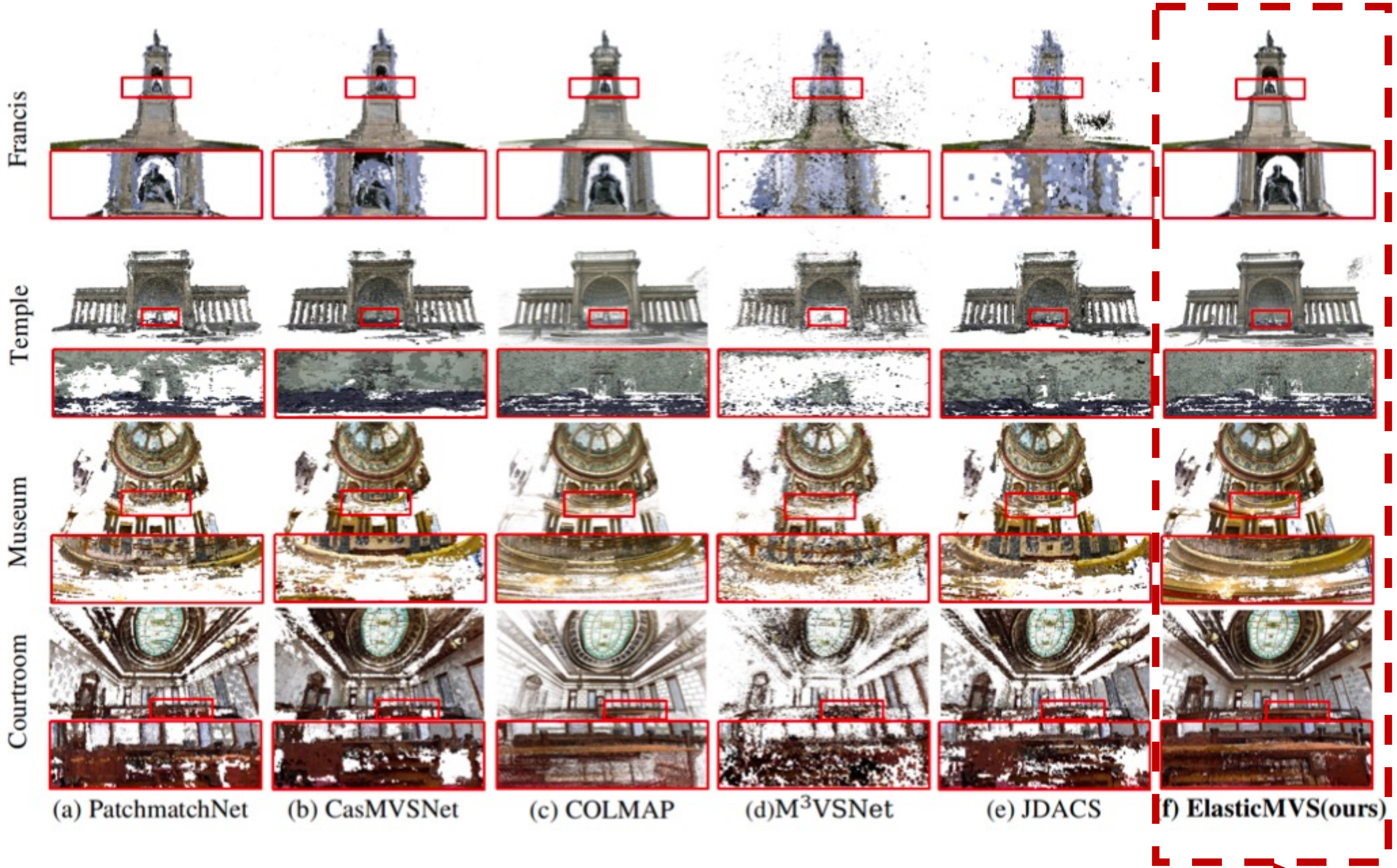
(a) Standard propagation (b) Part-aware propagation

# Inference: Detailed pipeline

Propagation → Depth hypothesis → Score → Depth map



# Results on T&T

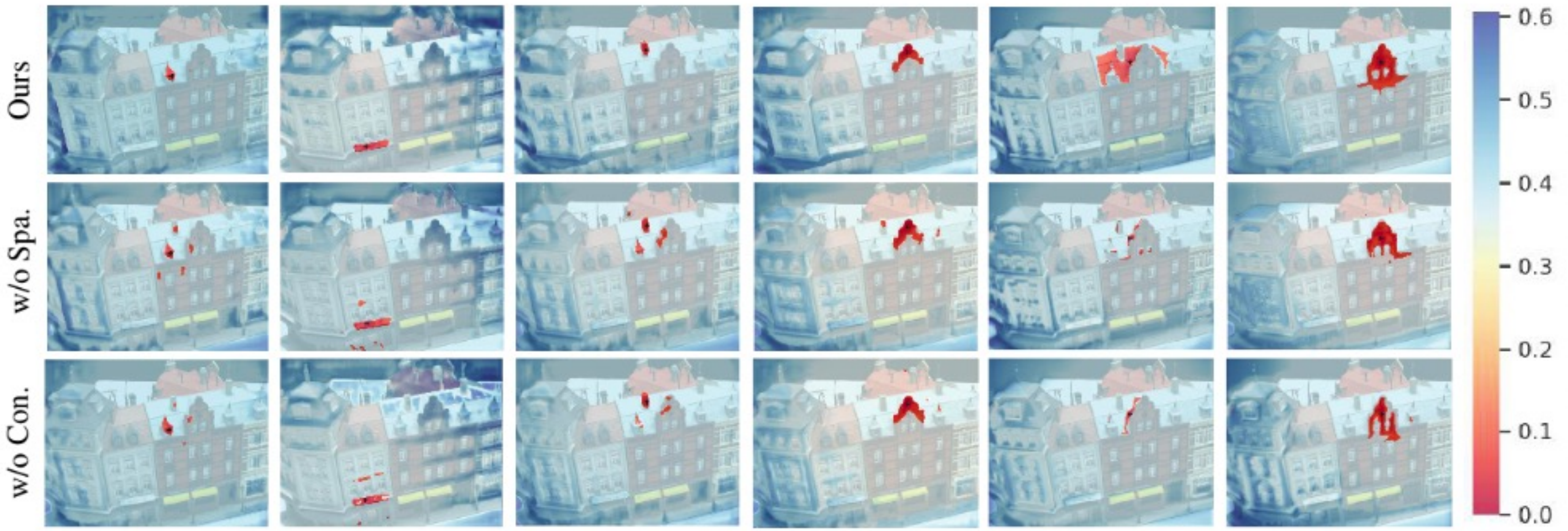



| Method | Intermediate | Advanced |
|---|---|---|
| MVSNet [47] | 43.48 | - |
| CasMVSNet [15] | **56.84** | 31.12 |
| UCSNet [10] | 54.83 | - |
| PVAMVSNet [49] | 54.46 | - |
| SurfaceNet+ [19] | 49.38 | - |
| R-MVSNet [48] | 50.55 | 29.55 |
| Point-MVSNet [7] | 48.27 | - |
| PatchmatchNet [39] | 53.15 | **32.31** |
| Patchmatch-RL [26] | 51.81 | 31.78 |
| $MVS^2$ [11] | 37.21 | - |
| $M^3VSNet$ [16] | 37.67 | - |
| SurRF [51] | 54.36 | - |
| JDACS [43] | 45.48 | - |
| COLMAP[34] | 42.14 | 27.24 |
| **ElasticMVS (ours)** | **57.88** | **37.81** |

**Ours**

# Visualization

$z_p$




$$\mathcal{T}_p = \left\{ q \in R^2 \middle| \|z_p - z_q\| \leq \eta, c_q \geq \xi \right\}$$

# Reference


Silvano Galliani, Katrin Lasinger, and Konrad Schindler. 2015. Massively parallel multiview stereopsis by surface normal diffusion. In *IEEE International Conference on Computer Vision*. 873–881.

Johannes L Schönberger, Enliang Zheng, Jan-Michael Frahm, and Marc Pollefeys. 2016. Pixelwise view selection for unstructured multi-view stereo. In *European Conference on Computer Vision*. Springer, 501–518.

Yao Yao, Zixin Luo, Shiwei Li, Tian Fang, and Long Quan. 2018. MVSnet: Depth inference for unstructured multi-view stereo. In *Proceedings of the European Conference on Computer Vision (ECCV)*. 767–783.

Tejas Khot, Shubham Agrawal, Shubham Tulsiani, Christoph Mertz, Simon Lucey, and Martial Hebert. 2019. Learning unsupervised multi-view stereopsis via robust photometric consistency. *arXiv preprint arXiv:1905.02706* (2019).

Andrea Romanoni and Matteo Matteucci. 2019. TAPA-MVS: Textureless-Aware PAtchMatch Multi-View Stereo. 2019 IEEE/CVF International Conference on Computer Vision (ICCV) (2019), 10412–10421.

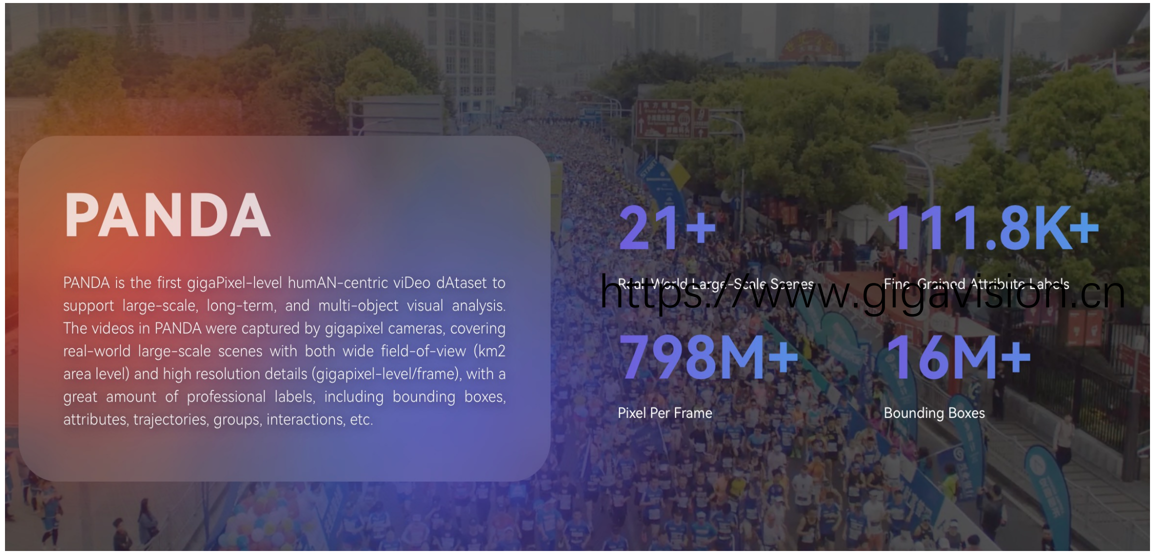
Branislav Micusík and Jana Kosecka. 2010. Multi-view Superpixel Stereo in Urban Environments. International Journal of Computer Vision 89 (2010), 106–119.

J. Zhang et al., "GigaMVS: A Benchmark for Ultra-Large-Scale Gigapixel-Level 3D Reconstruction," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 44, no. 11, pp. 7534-7550, 1 Nov. 2022, doi: 10.1109/TPAMI.2021.3115028.

J. Zhang, M. Ji, G. Wang, Z. Xue, S. Wang and L. Fang, "SurRF: Unsupervised Multi-View Stereopsis by Learning Surface Radiance Field," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 44, no. 11, pp. 7912-7927, 1 Nov. 2022, doi: 10.1109/TPAMI.2021.3116695.

# GIGAVISION When Gigapixel Videography Meets Computer Vision

## 2D




## 3D


**https://www.gigavision.cn**

**6 GigaVision challenges (GigaDetection, GigaMOT, GigaTrajectory, GigaReconstruction, GigaRendering and GigaCrowd) with lucrative awards.**

# Thank you!

## Welcome to our lab's website for more works !

tSinghua vIsual intelliGence and coMputational imAging lab

http://www.luvision.net

Github

https://github.com/THU-luvision