CVPR JUNE 17-21, 2024 SEATTLE, WA

Learning Structure-from-Motion with Graph Attention Networks

Lucas Brynte

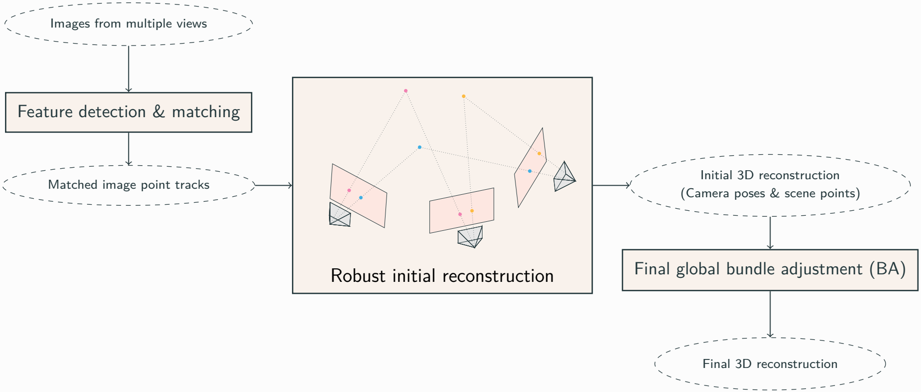
Joint work with José Pedro Iglesias, Carl Olsson, and Fredrik Kahl

Chalmers University of Technology

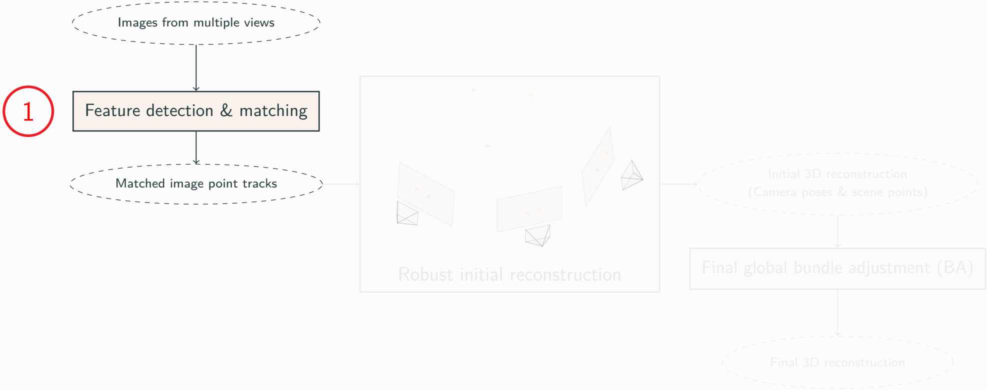
CHALMERS

WASP WALLENBERG AI, AUTONOMOUS SYSTEMS AND SOFTWARE PROGRAM

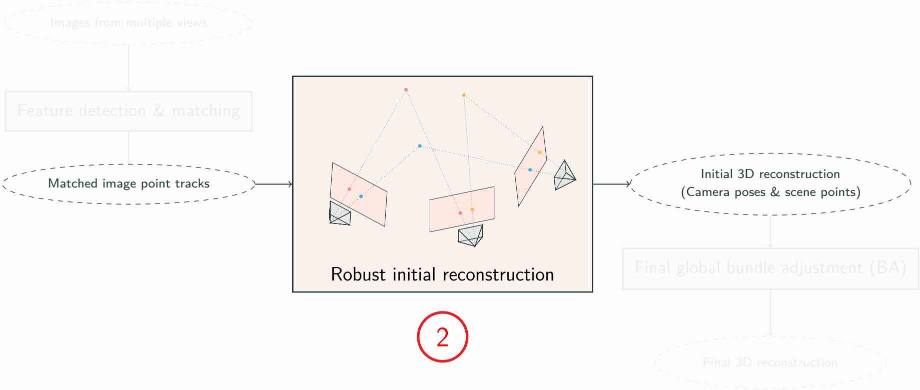
Learning Structure-from-Motion

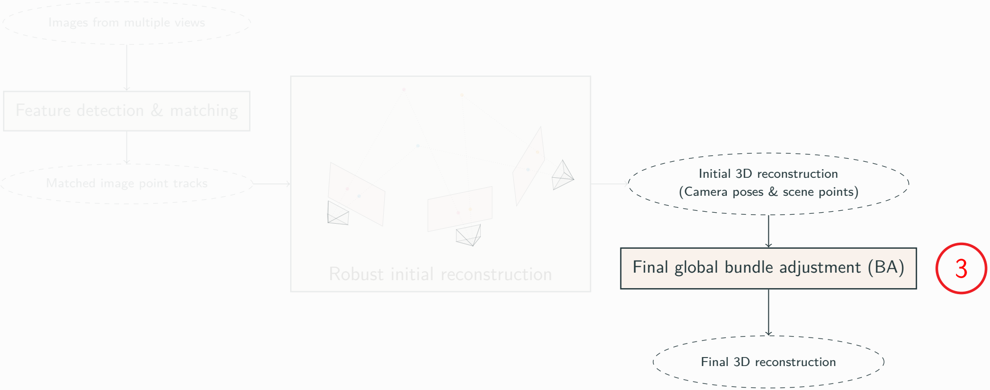


Learning Structure-from-Motion



Learning Structure-from-Motion

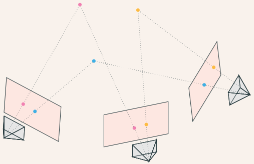



Images from multiple views
Feature detection & matching
Matched image point tracks
Robust initial reconstruction
Initial 3D reconstruction
(Camera poses & scene points)
Final global bundle adjustment (BA)
3
Final 3D reconstruction

Images from multiple views

Feature detection & matching

Matched image point tracks



Robust initial reconstruction

Learn this!

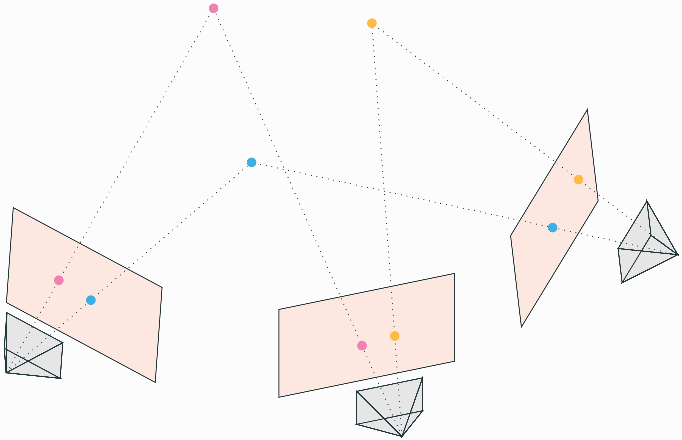
2

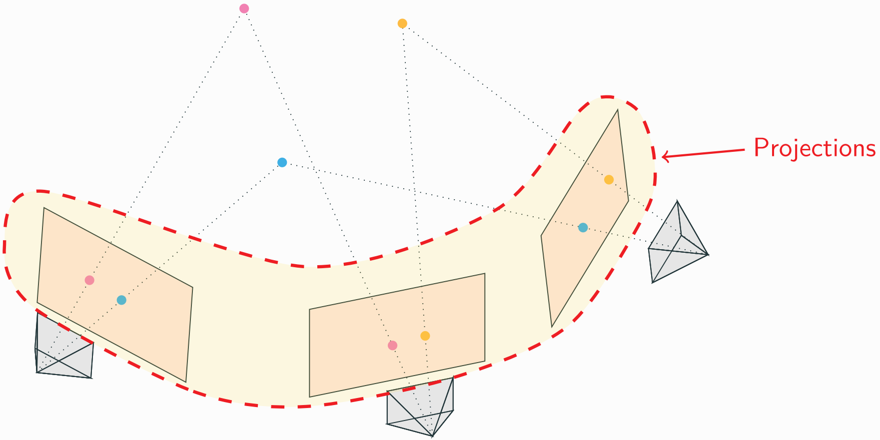
Initial 3D reconstruction
(Camera poses & scene points)

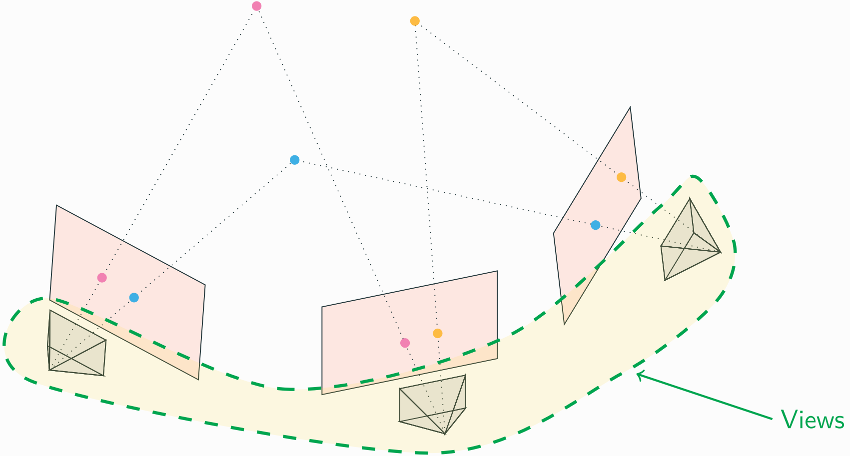
Final global bundle adjustment (BA)

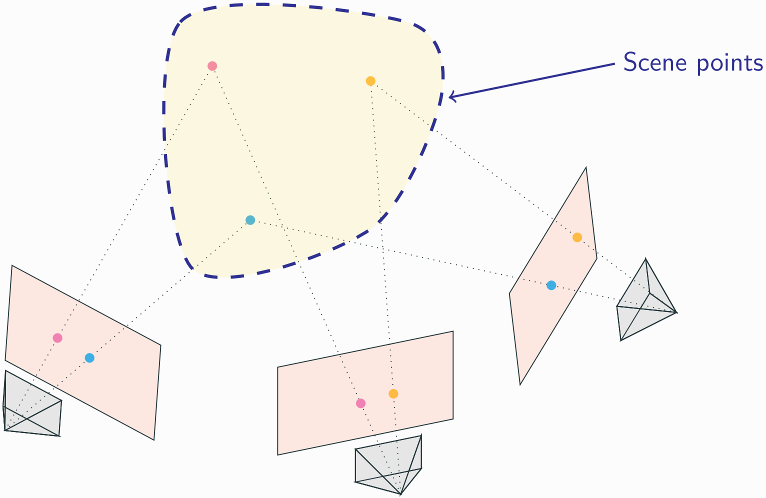
Final 3D reconstruction

Elements of Multi-View Geometry




Projections


Views


Scene points

Model Architecture: Feature Types

- Goal: A neural network model operating on the geometric elements of SfM.
- Organize all geometric entities (camera views, projections, ...) in vectors / matrices.
  - Rows & columns $\Longleftrightarrow$ camera views & scene points, respectively.
- Let each geometric entity carry information in a feature vector.

Model Architecture: Feature Types

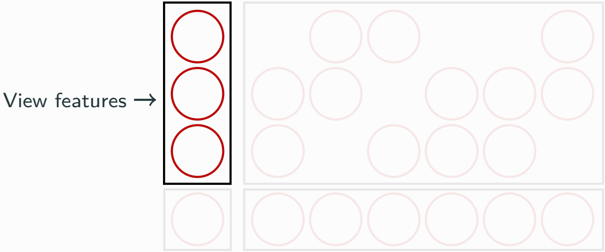
- Goal: A neural network model operating on the geometric elements of SfM.
- Organize all geometric entities (camera views, projections, . . .) in vectors / matrices.
  - Rows & columns $\iff$ camera views & scene points, respectively.
- Let each geometric entity carry information in a feature vector.

Model Architecture: Feature Types

- Goal: A neural network model operating on the geometric elements of SfM.
- Organize all geometric entities (camera views, projections, . . .) in vectors / matrices.
  - Rows & columns $\Longleftrightarrow$ camera views & scene points, respectively.
- Let each geometric entity carry information in a feature vector.

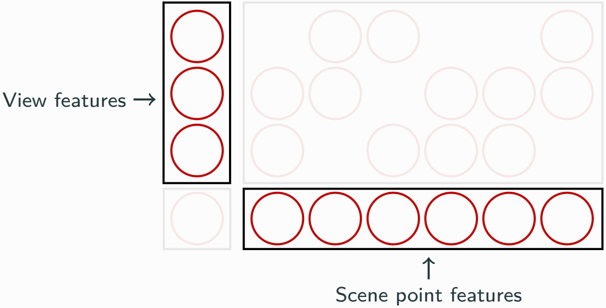
Model Architecture: Feature Types

- Goal: A neural network model operating on the geometric elements of SfM.
- Organize all geometric entities (camera views, projections, . . .) in vectors / matrices.
  - Rows & columns $\Longleftrightarrow$ camera views & scene points, respectively.
- Let each geometric entity carry information in a feature vector.



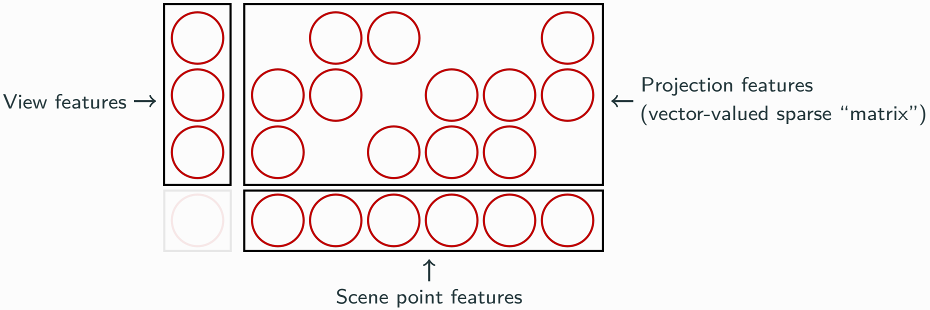
Model Architecture: Feature Types

- Goal: A neural network model operating on the geometric elements of SfM.
- Organize all geometric entities (camera views, projections, . . .) in vectors / matrices.
  - Rows & columns $\Longleftrightarrow$ camera views & scene points, respectively.
- Let each geometric entity carry information in a feature vector.



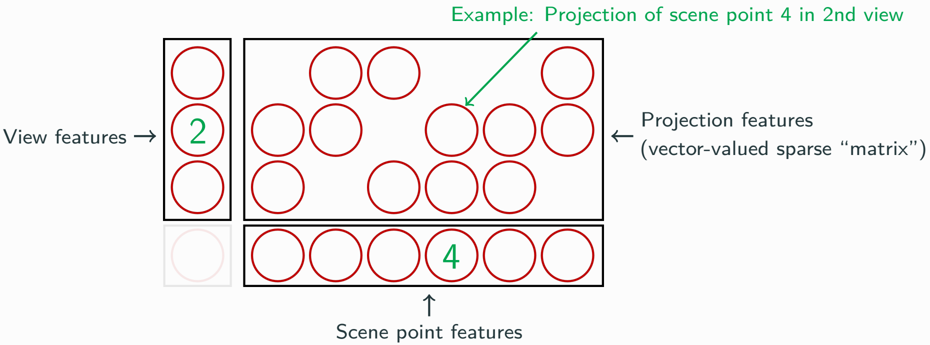
Model Architecture: Feature Types

- Goal: A neural network model operating on the geometric elements of SfM.
- Organize all geometric entities (camera views, projections, ...) in vectors / matrices.
  - Rows & columns $\iff$ camera views & scene points, respectively.
- Let each geometric entity carry information in a feature vector.



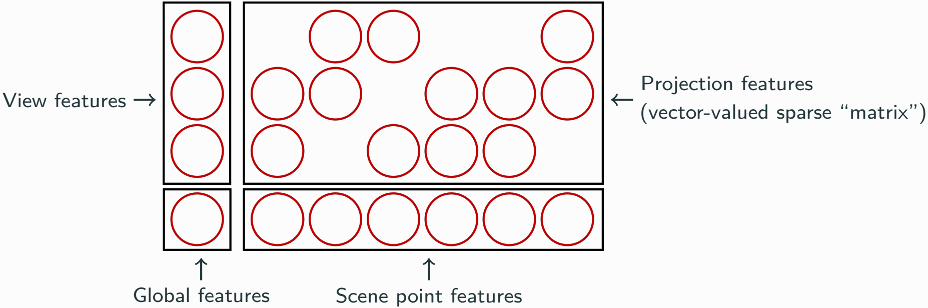
Model Architecture: Feature Types

- Goal: A neural network model operating on the geometric elements of SfM.
- Organize all geometric entities (camera views, projections, ...) in vectors / matrices.
  - Rows & columns $\Longleftrightarrow$ camera views & scene points, respectively.
- Let each geometric entity carry information in a feature vector.

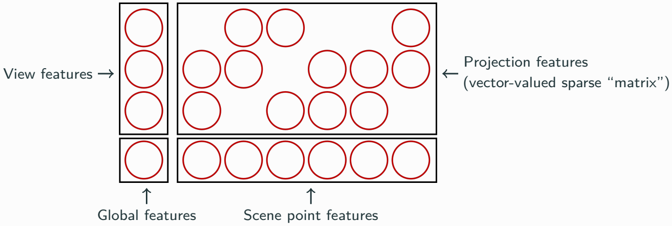


Model Architecture: Feature Types

- Goal: A neural network model operating on the geometric elements of SfM.
- Organize all geometric entities (camera views, projections, ...) in vectors / matrices.
  - Rows & columns $\iff$ camera views & scene points, respectively.
- Let each geometric entity carry information in a feature vector.



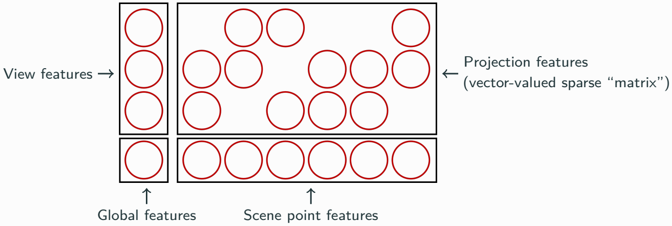
Model Architecture: Feature Types



"Connecting the dots"

- Sparse patterns $\implies$ Use graph neural networks (GNN).
- Possibility of outliers $\implies$ Use attention (we use GATv2).
- A single fully connected graph (i.e. self-attention) would have drawbacks:
  - Quadratic complexity.
  - Ignores feature type differences.
- Proposal: Aggregate features via cross-attention from one feature type to another.
  - (Bi-)linear complexity.

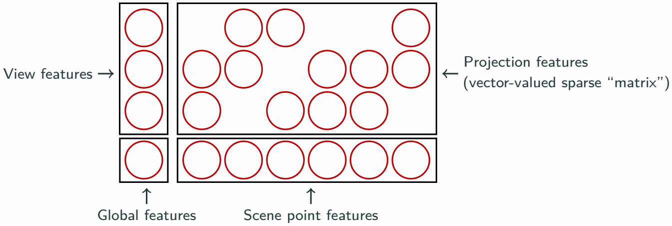
Model Architecture: Feature Types



"Connecting the dots"

- Sparse patterns $\implies$ Use graph neural networks (GNN).
- Possibility of outliers $\implies$ Use attention (we use GATv2).
- A single fully connected graph (i.e. self-attention) would have drawbacks:
  - Quadratic complexity.
  - Ignores feature type differences.
- Proposal: Aggregate features via cross-attention from one feature type to another.
  - (Bi-)linear complexity.

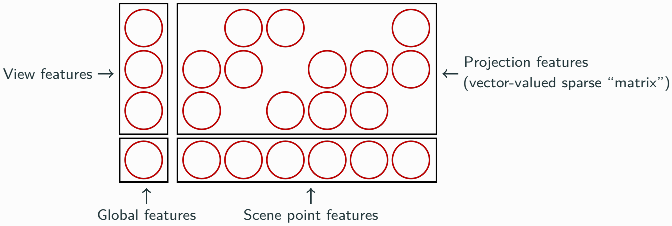
Model Architecture: Feature Types



“Connecting the dots”

- Sparse patterns $\implies$ Use graph neural networks (GNN).
- Possibility of outliers $\implies$ Use attention (we use GATv2).
- A single fully connected graph (i.e. self-attention) would have drawbacks:
  - Quadratic complexity.
  - Ignores feature type differences.
- Proposal: Aggregate features via cross-attention from one feature type to another.
  - (Bi-)linear complexity.

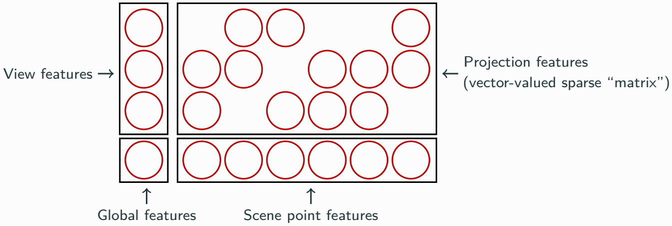
Model Architecture: Feature Types



"Connecting the dots"

- Sparse patterns $\implies$ Use graph neural networks (GNN).
- Possibility of outliers $\implies$ Use attention (we use GATv2).
- A single fully connected graph (i.e. self-attention) would have drawbacks:
  - Quadratic complexity.
  - Ignores feature type differences.
- Proposal: Aggregate features via cross-attention from one feature type to another.
  - (Bi-)linear complexity.

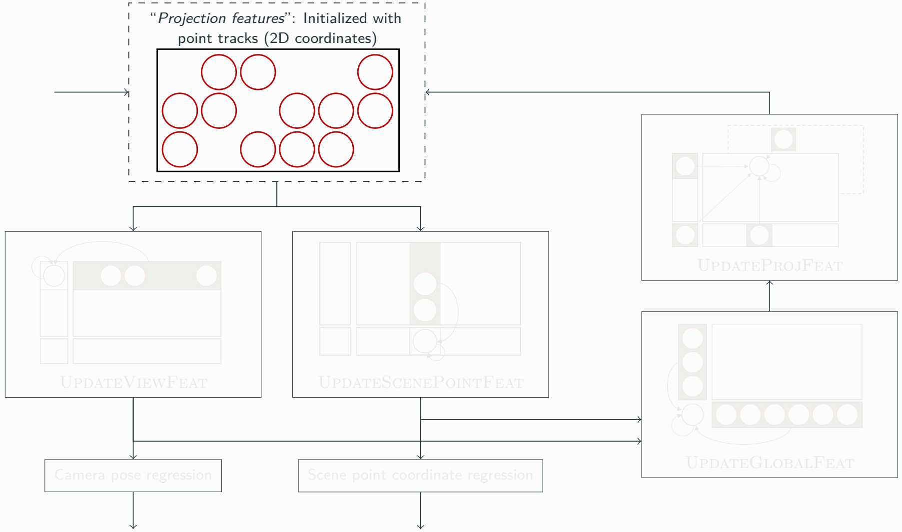
Model Architecture: Feature Types



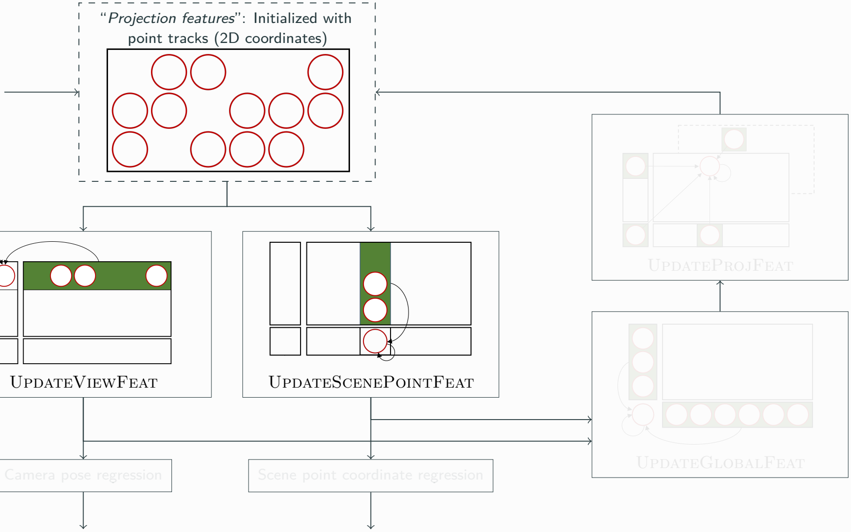
"Connecting the dots"

- Sparse patterns $\Longrightarrow$ Use graph neural networks (GNN).
- Possibility of outliers $\Longrightarrow$ Use attention (we use GATv2).
- A single fully connected graph (i.e. self-attention) would have drawbacks:
  - Quadratic complexity.
  - Ignores feature type differences.
- Proposal: Aggregate features via cross-attention from one feature type to another.
  - (Bi-)linear complexity.

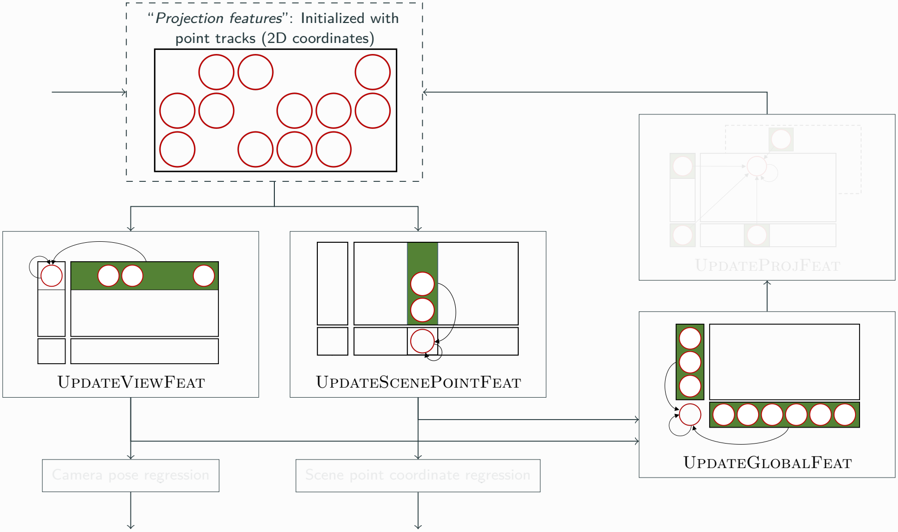
Model Architecture: Overview



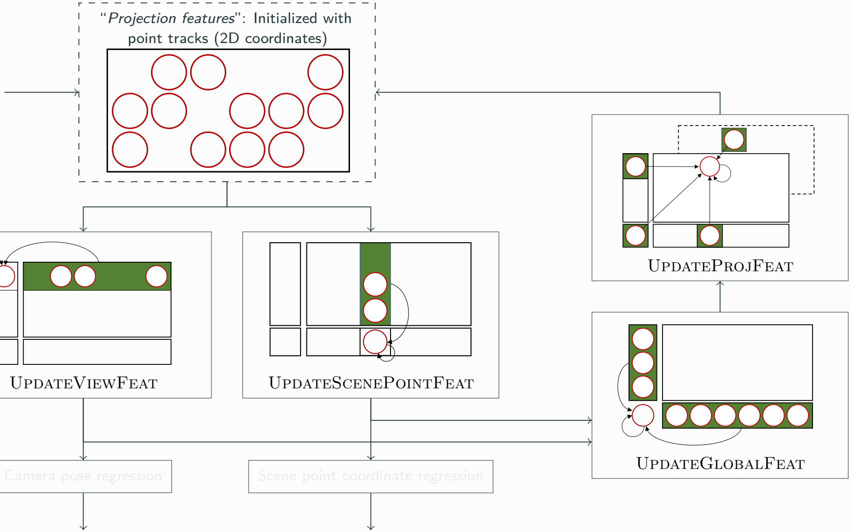
Model Architecture: Overview



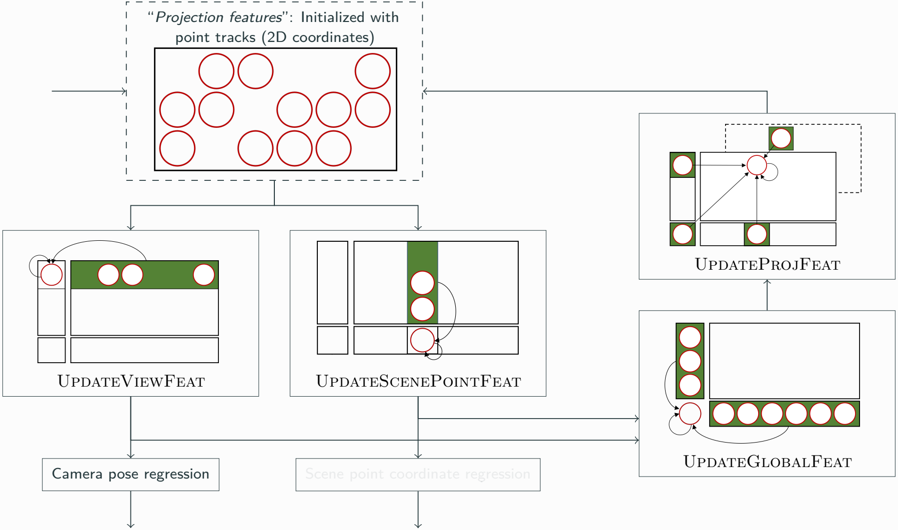
Model Architecture: Overview



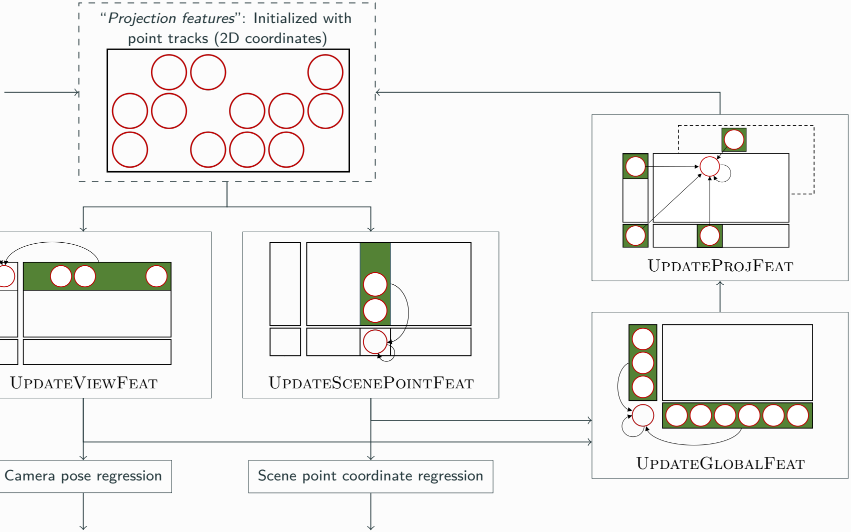
Model Architecture: Overview



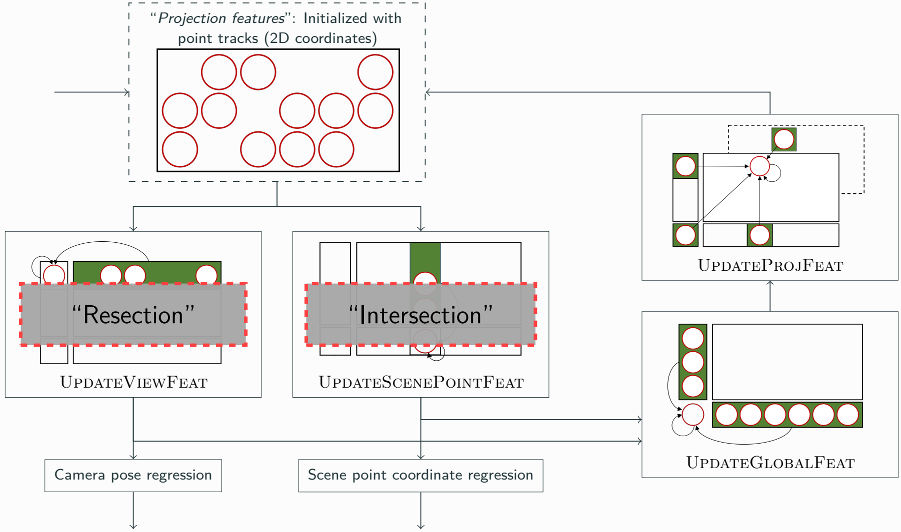
Model Architecture: Overview



Model Architecture: Overview



Model Architecture: Overview

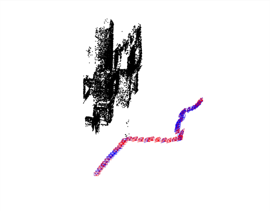


Implementation Details

- Loss Function: Average reprojection error.
- Model trained on 12 scenes (SfM reconstructions, outlier-free correspondences).
  - Will present initial experiments with outliers as well.

Implementation Details

- Loss Function: Average reprojection error.
- Model trained on 12 scenes (SfM reconstructions, outlier-free correspondences).
  - Will present initial experiments with outliers as well.



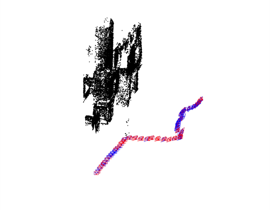
“Some Cathedral in Barcelona”

“Alcatraz Courtyard”

“Smolny Cathedral St Petersburg”

Implementation Details

- Loss Function: Average reprojection error.
- Model trained on 12 scenes (SfM reconstructions, outlier-free correspondences).
  - Will present initial experiments with outliers as well.



"Some Cathedral in Barcelona"

"Alcatraz Courtyard"

"Smolny Cathedral St Petersburg"

Results: Reconstruction of Novel Test Scenes

Scene	Ours (GASFM) Inference	Ours (GASFM) +BA	DPESFM[a] Inference	DPESFM[a] +BA	Colmap[b]
Alcatraz Courtyard	36.01	**0.81**	92.37	0.92	**0.81**
Alcatraz Water Tower	87.67	0.88	2831.94	10.16	**0.55**
Drinking Fountain Somewhere in Zurich	219.75	**0.31**	234.90	6.73	**0.31**
Nijo Castle Gate	61.41	0.88	68.19	0.89	**0.73**
Porta San Donato Bologna	52.15	0.76	84.46	0.75	0.75
Round Church Cambridge	29.80	**0.39**	59.54	1.49	**0.39**
Smolny Cathedral St Petersburg	85.38	**0.81**	87.81	**0.81**	**0.81**
Some Cathedral in Barcelona	125.68	**0.89**	687.83	16.77	**0.89**
Sri Veeramakaliamman Singapore	83.50	2.13	166.68	9.30	**0.71**
Yueh Hai Ching Temple Singapore	25.60	**0.65**	51.35	0.73	**0.65**
Average	80.69	0.85	436.51	4.86	**0.66**

Table 1: Avg. reprojection error (px) on 10 novel test scenes, with and without BA, compared to DPESFM[a] and Colmap[b].

[a] Moran et al., *Deep Permutation Equivariant Structure from Motion*, ICCV (2021).
[b] Schönberger et al., *Structure-from-Motion Revisited*, CVPR (2016).

Execution Time

Scene	#Views	#Points	Time (seconds) Inference	BA	Colmap	Speedup
Alcatraz Courtyard	133	23 674	0.24	45.54	286.0	6.3×
Alcatraz Water Tower	172	14 828	0.13	31.11	130.0	4.2×
Drinking Fountain Somewhere In Zurich	14	5 302	0.06	1.98	16.0	7.8×
Nijo Castle Gate	19	7 348	0.09	3.97	21.0	5.2×
Porta San Donato Bologna	141	25 490	0.18	27.02	170.0	6.3×
Round Church Cambridge	92	84 643	0.43	56.47	229.0	4.0×
Smolny Cathedral St Petersburg	131	51 115	0.49	86.09	516.0	6.0×
Some Cathedral In Barcelona	177	30 367	0.24	47.05	451.0	9.5×
Sri Veeramakaliamman Singapore	157	130 013	0.63	115.80	583.0	5.0×
Yueh Hai Ching Temple Singapore	43	13 774	0.08	8.54	106.0	12.3×

Table 2: Runtime per scene compared to Colmap.

Initial Results: Training / Evaluation With Outliers

Scene	Corrupted subset(s):	Ours (GASFM) Train	Ours (GASFM) Train+Test	DPESFM Train	DPESFM Train+Test
Alcatraz Courtyard		**47.74**	**52.99**	85.81	94.24
Alcatraz Water Tower		**35.96**	**37.89**	72.84	83.55
Drinking Fountain Somewhere in Zurich		**52.08**	**46.65**	1012.14	1453.31
Nijo Castle Gate		**46.48**	**62.52**	72.99	126.18
Porta San Donato Bologna		**53.12**	**65.08**	88.02	94.72
Round Church Cambridge		**36.09**	**48.63**	63.72	90.63
Smolny Cathedral St Petersburg		**47.28**	**59.52**	91.03	98.05
Some Cathedral in Barcelona		**109.86**	**123.75**	397.75	462.28
Sri Veeramakaliamman Singapore		**63.60**	**70.90**	169.63	146.98
Yueh Hai Ching Temple Singapore		**26.83**	**36.69**	51.41	57.59
Average		**51.91**	**60.46**	210.53	270.75

Table 3: Training with artificial outliers: Avg. reprojection error (px) for inference on 10 novel test scenes (no BA). With or w/o outliers for test scenes as well. (N.B.: In loss fcn. & eval. metric, targets remain uncorrupted.)

Future Work

- Consider many more training scenes.
- Train and evaluate performance on real outlier matches.
- Incorporate equivariance (break dependence on arbitrary global reference frame).

Future Work

- Consider many more training scenes.
- Train and evaluate performance on real outlier matches.
- Incorporate equivariance (break dependence on arbitrary global reference frame).

Future Work

- Consider many more training scenes.
- Train and evaluate performance on real outlier matches.
- Incorporate equivariance (break dependence on arbitrary global reference frame).

Thank You!

/ https://github.com/lucasbrynte/gasfm

arXiv / https://arxiv.org/abs/2308.15984

Poster #90