LightGlue: Local Feature Matching at Light Speed (2023)

Philipp Lindenberger1
ETH Zurich

Paul-Edouard Sarlin
Microsoft Mixed Reality & AI Lab

Marc Pollefeys
Microsoft Mixed Reality & AI Lab

Presented By:
Adyasha Mahanta
17-06-2024

CONTENT

INTRODUCTION

What is feature detection, feature descriptor and feature matching ?

- **Feature point** is the point at which the direction of the boundary of the object changes abruptly or intersection point between two or more edge segments.
- **Feature descriptors** encode interesting information into a series of numbers and act as a sort of numerical “fingerprint” that can be used to differentiate one feature from another.
- **Features matching** is the task of establishing correspondences between two images of the same scene/object

Finding correspondences between two images is a fundamental building block of many computer vision applications like camera tracking and 3D mapping.

INTRODUCTION

SuperGlue

- A Deep network that considers both images at the same time to jointly match sparse points and reject outliers. It leverages the powerful Transformer model to learn to match challenging image pairs from large datasets.

- This yields robust image matching in both indoor and outdoor environments. SuperGlue is highly effective for visual localization in challenging conditions and generalizes well to other tasks like aerial matching, object pose estimation, and even fish re-identification.

Drawbacks

- Computationally expensive
- Transformer-based models, is notoriously hard to train, requiring computing resources that are inaccessible to many practitioners

INTRODUCTION

LightGlue
A deep network that is more accurate, more efficient, and easier to train than SuperGlue

Advantages

- LightGlue is adaptive to the difficulty of each image pair
- The inference is thus much faster on pairs that are intuitively easy to match than on challenging ones

Achieved by:

- predicting a set of correspondences after each computational blocks
- enabling the model to introspect them and predict whether further computation is required
- LigthGlue also discards at early stage points that are not matchable, thus focusing its attention on the covisible area

This opens up exciting prospects for deploying deep matchers in latency-sensitive applications like **SLAM** or **reconstructing larger scenes** from crowd-sourced data

PAST WORK

Matching

- Classical Methods used **Hand-crafted criteria**, gradient stats for feature matching
- then **CNNs** Improve accuracy, robustness
- Matching was done as **Nearest neighbor** search with Filtering: Lowe’s ratio test, mutual check, classifiers, geometric fitting
- Challenges: Non-matchable keypoints, imperfect descriptors
- **Deep Matchers** overcame classical limitations, enhanced performance

Deep Matchers

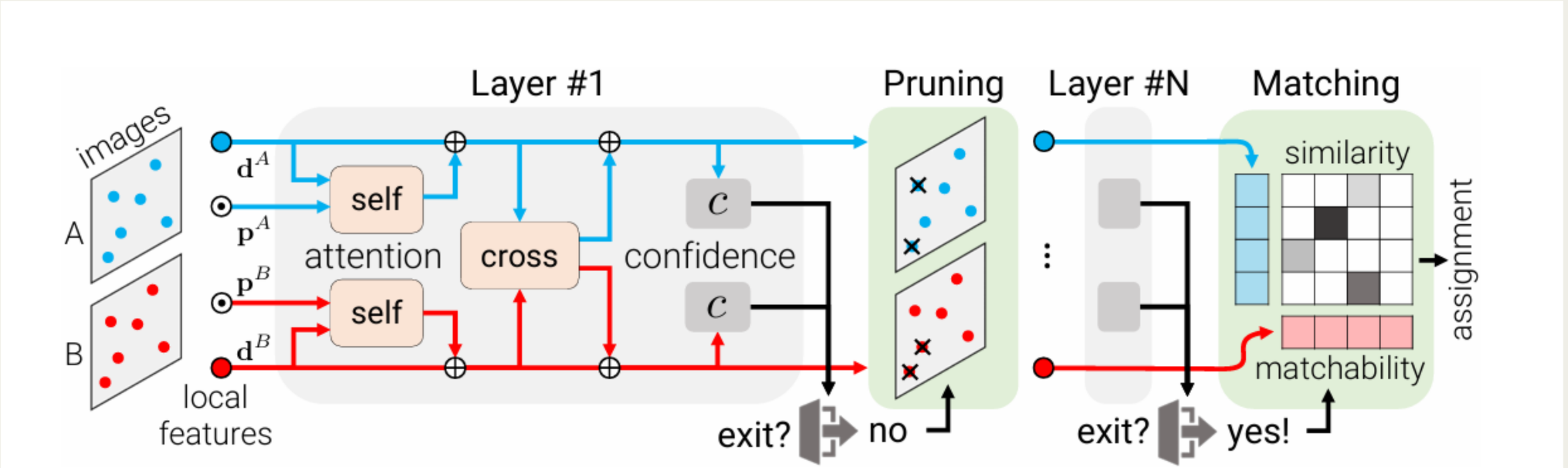
- Inheriting the limitations of early Transformers, SuperGlue is **hard to train** and its **complexity grows quadratically** with the number of key points. LightGlue dynamically adapts the network size instead of reducing its overall capacity.
- Dense matchers like **LoFTR** and follow-ups match points distributed on dense grids. This boosts the robustness to impressive levels but is generally much slower also limiting the resolution.

PROBLEM STATEMENT

Problem formulation: LightGlue predicts a partial assignment between two sets of local features extracted from images A and B, following SuperGlue. Each local feature i is composed of a 2D point position $\mathbf{p}_i := (x, y)_i \in [0, 1]^2$, normalized by the image size, and a visual descriptor $\mathbf{d}_i \in \mathbb{R}^d$. Images A and B have M and N local features, indexed by $\mathcal{A} := \{1, ..., M\}$ and $\mathcal{B} := \{1, ..., N\}$, respectively.

We design LightGlue to output a set of correspondences $\mathcal{M} = \{(i, j)\} \subset \mathcal{A} \times \mathcal{B}$. Each point is matchable at least once, as it stems from a unique 3D point, and some keypoints are unmatchable, due to occlusion or non-repeatability. As in previous works, we thus seek a soft partial assignment matrix $\mathbf{P} \in [0, 1]^{M \times N}$ between local features in A and B, from which we can extract correspondences.

ARCHITECTURE



TRANSFORMER BACKBONE

Initialization:

Each local feature's state is initialized with its visual descriptor.

$$\mathbf{x}_i^I \leftarrow \mathbf{d}_i^I$$

Layer Structure:

Consists of self-attention and cross-attention units.

Self-Attention:

- Each point attends to all points in the same image.
- Key and Query Vectors: States are decomposed into key and query vectors.
- Attention Score: Calculated using rotary encoding for relative positions:

$$a_{ij} = \mathbf{q}_i^\top \mathbf{R}(\mathbf{p}_j - \mathbf{p}_i)\ \mathbf{k}_j \quad ,$$

- Rotary Encoding:Captures relative positions using 2D subspaces and Fourier Features.
- Positional Encoding: Critical for addressing points based on relative positions.

TRANSFORMER BACKBONE

Cross-Attention:

- Each point in one image attends to all points in the other image.
- Key Vectors: Each element has a key vector but no query vector.
- Attention Score: Computed once for both directions using bidirectional attention:

$$a_{ij}^{IS} = \mathbf{k}_i^{I\top} \mathbf{k}_j^{S} \stackrel{!}{=} a_{ji}^{SI} \quad .$$

- Efficiency: Reduces computational complexity, no positional encoding added.

CORRESPONDENCE PREDICTION

Pairwise Matrix:

- Reflects the similarity between point pairs

$$\mathbf{S}_{ij} = \text{Linear}\left(\mathbf{x}_i^A\right)^\top \text{Linear}\left(\mathbf{x}_j^B\right) \quad \forall (i,j) \in \mathcal{A} \times \mathcal{B},$$

Matchability Score:

- Indicates likelihood of a point having a match

$$\sigma_i = \text{Sigmoid}\left(\text{Linear}(\mathbf{x}_i)\right) \in [0,1] \ .$$

Soft Partial Assignment Matrix (P):

- Combine similarity and matchability

$$\mathbf{P}_{ij} = \sigma_i^A \ \sigma_j^B \ \underset{k \in \mathcal{A}}{\text{Softmax}}(\mathbf{S}_{kj})_i \ \underset{k \in \mathcal{B}}{\text{Softmax}}(\mathbf{S}_{ik})_j$$

CORRESPONDENCE PREDICTION

Point Pruning:

- Exclude unmatchable points early to reduce search space and inference time.

Correspondence Selection:

A pair of points (i, j) yields a correspondence when both points are predicted as matchable and when their similarity is higher than any other point in both images. We select pairs for which $\mathbf{P}_{ij}$ is larger than a threshold τ and than any other element along both its row and column.

EFFICIENCY MECHANISM

Confidence Classifier

1. Contextual Descriptors:

- Augmentation: LightGlue adds context to input visual descriptors, making them reliable for easy image pairs with high overlap and minimal changes.

2. Confidence Score:

$$c_i = \text{Sigmoid}\left(\text{MLP}(\mathbf{x}_i)\right) \in [0, 1] \ .$$

2. Inference Halting:

- At the end of each layer, LightGlue assesses the confidence of each point.
- If predictions are confident, inference can be halted early, saving time.

EFFICIENCY MECHANISM

Exit Criterion:

Exit criterion: For a given layer ℓ, a point is deemed confident if $c_i > \lambda_\ell$. We halt the inference if a sufficient ratio α of all points is confident:

$$\text{exit} = \left(\frac{1}{N+M} \sum_{I \in \{A,B\}} \sum_{i \in \mathcal{I}} [\![c_i^I > \lambda_\ell]\!] \right) > \alpha \ . \quad (10)$$

Point Pruning:

- Points predicted as both confident and unmatchable are discarded at each layer.
- This reduces computation, as attention complexity is quadratic.
- Pruning does not impact accuracy since these points do not contribute to matching.

COMPARISON WITH SUPERGLUE

Aspect	SuperGlue	LightGlue
Positional Encoding	- Uses an MLP for absolute point positions.	- Uses relative positional encoding.
	- Fuses positions with descriptors early.	- Adds encoding in each self-attention unit.
	- Forgets positional information through layers.	- Improves accuracy of deeper layers.
Prediction Head	- Uses Sinkhorn algorithm for assignment prediction.	- Disentangles similarity and matchability.
	- Involves many compute- and memory-intensive iterations.	- Provides cleaner gradients and avoids issues with "dustbin."
	- Uses a "dustbin" to reject unmatchable points	
Deep Supervision	- Predicts and supervises only at the final layer due to computational expense.	- Allows predictions and supervision at each layer.
		- Accelerates convergence and enables early exit from inference.

EXPERIMENT

Homography Estimation:

features	+ matcher	R	P	AUC - RANSAC @1px	AUC - RANSAC @5px	AUC - DLT @1px	AUC - DLT @5px
dense	LoFTR	-	92.7	41.5	78.8	38.5	70.6
SuperPoint	NN+mutual	72.7	67.2	35.0	75.3	0.0	2.0
	SuperGlue	94.9	87.4	38.3	79.3	33.8	76.7
	SGMNet	**95.5**	83.0	**38.6**	79.0	31.7	76.0
	LightGlue	94.3	**88.9**	38.3	**79.6**	**35.9**	**78.6**

Table 1. **Homography estimation on HPatches.** LightGlue yields better correspondences than sparse matchers, with the highest precision (P) and a high recall (R). This results in accurate homographies when estimated by RANSAC or even a faster least-squares solver (DLT). LightGlue is competitive with dense matchers like LoFTR.

Relative Pose Estimation:

	features + matcher	RANSAC AUC 5° / 10° / 20°	LO-RANSAC AUC 5° / 10° / 20°	time (ms)
dense	LoFTR	52.8 / 69.2 / 81.2	66.4 / 78.6 / 86.5	181
	MatchFormer	53.3 / 69.7 / 81.8	66.5 / 78.9 / 87.5	388
	ASpanFormer	**55.3 / 71.5 / 83.1**	**69.4 / 81.1 / 88.9**	369
DISK	NN+ratio	38.1 / 55.4 / 69.6	57.2 / 69.5 / 78.6	7.4
	LightGlue	**43.5 / 61.0 / 75.3**	**61.3 / 74.3 / 83.8**	44.5
SuperPoint	NN+mutual	31.7 / 46.8 / 60.1	51.0 / 54.1 / 73.6	5.7
	SuperGlue	49.7 / 67.1 / **80.6**	65.8 / 78.7 / 87.5	70.0
	SGMNet	43.2 / 61.6 / 75.6	59.8 / 74.1 / 83.9	73.8
	LightGlue	**49.9** / 67.0 / 80.1	**66.7 / 79.3 / 87.9**	44.2
	↳ **adaptive**	49.4 / **67.2** / 80.1	66.3 / 79.0 / **87.9**	31.4

Table 2. **Relative pose estimation.** On the MegaDepth1500 dataset

EXPERIMENT

Outdoor Visual Localization:

SuperPoint + matcher	Day	Night	pairs per second
	(0.25m,2°) / (0.5m,5°) / (1.0m,10°)		
SuperGlue	88.2 / **95.5** / **98.7**	86.7 / 92.9 / **100**	6.5
SGMNet	86.8 / 94.2 / 97.7	83.7 / 91.8 / 99.0	10.2
ClusterGNN	**89.4** / **95.5** / 98.5	81.6 / **93.9** / **100**	13*
LightGlue	89.2 / 95.4 / 98.5	**87.8** / **93.9** / **100**	**17.2** / ***26.1***

Table 3. **Outdoor visual localization.** On the Aachen Day-Night dataset, LightGlue performs on par with SuperGlue but runs 2.5× faster, 4× when *optimized*. SGMNet and ClusterGNN are both slower and less robust on night-time images (*approximation).

ABLATION STUDY

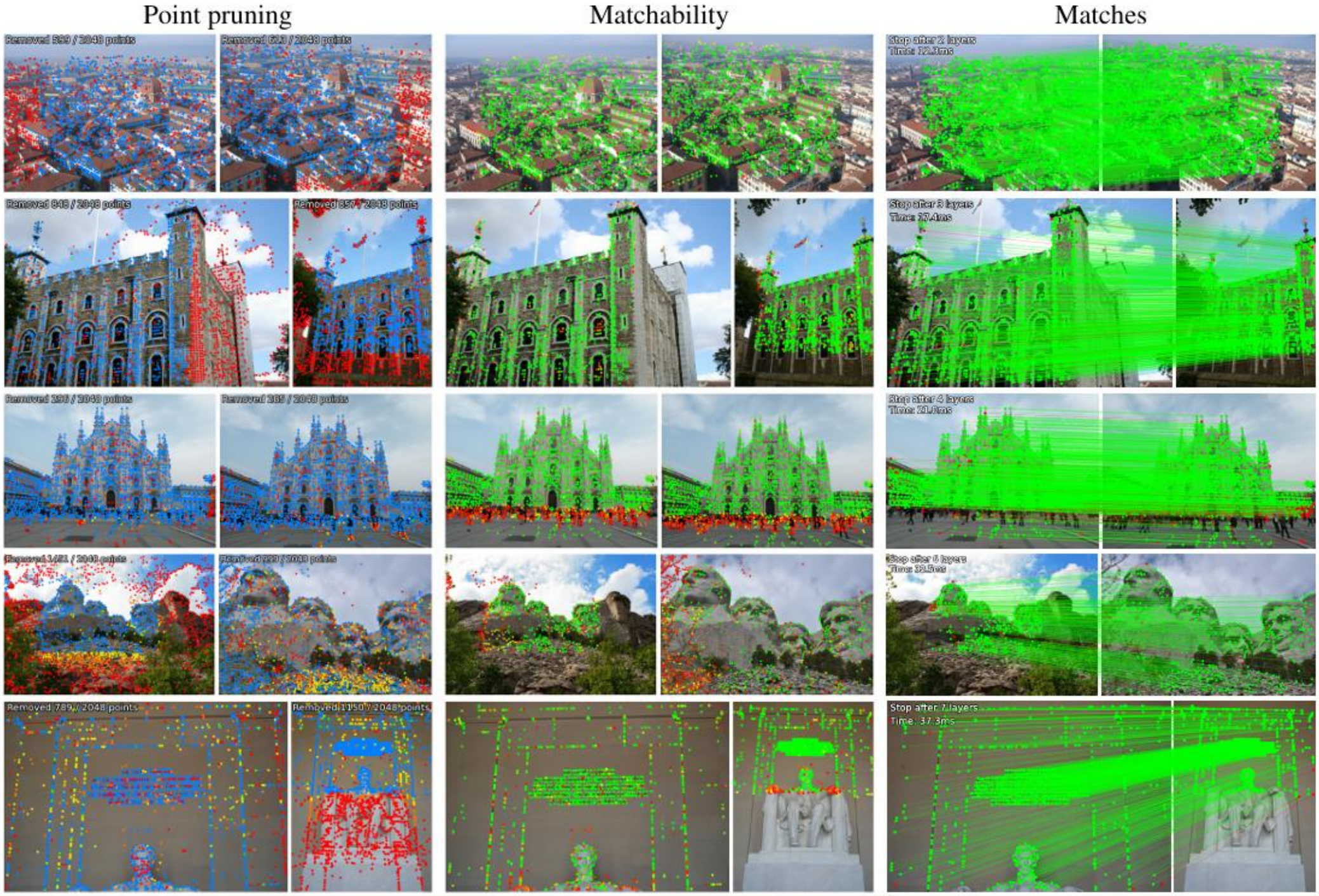
- **Matchability Classifier:** Crucial for filtering out erroneous matches, improving overall accuracy.
- **Positional Encoding:** Enhances geometric pattern recognition across images.
- **Efficiency:** Bidirectional cross-attention and deep supervision optimize performance without sacrificing accuracy, making LightGlue a robust choice for real-time applications in visual localization and matching tasks.

architecture	precision	recall	time (ms)
SuperGlue	74.6	90.5	29.1
LightGlue (full)	**86.8**	96.3	19.4
↳ a) no matchability	67.4	**97.0**	18.9
↳ b) absolute positions	84.2	94.7	18.7
↳ c) full cross-attention	86.6	96.1	22.8
↳ d) early layer (#5/9)	78.1	92.7	**11.9**

Table 4. **Ablation study on synthetic homographies.** a-b) Both matchability and positional encoding improve the accuracy without impact on the time. c) The bidirectional cross-attention is faster without drop of accuracy. d) Thanks to the deep supervision, early layers yield good predictions on pairs with low difficulty.

metric	difficulty			average
	easy	medium	hard	
average index of stopping layer ↓	4.7	5.5	6.9	5.7
ratio of unmatchable points (%) ↑	19.8	23.4	27.9	23.7
speedup over non-adaptive ↑	1.86	1.33	1.16	1.45

Table 5. **Impact of adaptive depth and width.** Early stopping helps most on smaller scenes, where the network stops after just half the layers. On harder scenes, the network requires more layers to converge, but smaller view overlap between image pairs allows the network to more aggressively prune the width of the network. Overall, adaptive depth- and width- pruning reduces the run time by 33% and is particularly effective on easy pairs.


Point pruning
Matchability
Matches
Removed 589 / 2048 points
Removed 613 / 2048 points
Stop after 2 layers
Time: 12.3ms
Removed 848 / 2048 points
Removed 859 / 2048 points
Stop after 3 layers
Time: 17.4ms
Removed 393 / 2048 points
Removed 382 / 2048 points
Stop after 4 layers
Time: 21.8ms
Removed 1481 / 2048 points
Removed 1291 / 2048 points
Stop after 6 layers
Time: 32.5ms
Removed 789 / 2048 points
Removed 1150 / 2048 points
Stop after 7 layers
Time: 37.3ms


SIFT+LightGlue
SuperPoint+LightGlue
DISK+LightGlue
Matches: 376
Precision: 77.66
Matches: 672
Precision: 87.65
Matches: 778
Precision: 89.01
Matches: 187
Precision: 87.17
Matches: 275
Precision: 87.27
Matches: 514
Precision: 97.67

THANK YOU