1 SUPPLEMENTARY APPENDIX

1.1 AACHEN DAY-NIGHT LOCALIZATION PIPELINE.

The assessment of visual localization relies on the Aachen Day-Night dataset, specifically version 1.1. To comprehensively evaluate the performance of both keypoints detection and description, we employ a predefined visual localization pipeline[1] based on colmap provided by benchmark[2]. This pipeline operates as follows: Initially, custom features extracted from the database's images are employed to construct a structure-from-motion model. Subsequently, the query images are registered within this model using the same custom features. For keypoints matching, we utilize the mutual nearest neighbor approach to effectively filter out outliers. The pipeline quantifies the percentage of successfully localized images based on three error tolerances: (0.25m, 2°), (0.5m, 5°), and (5m, 10°), with the first threshold relating to position and the second to orientation accuracy. Importantly, all comparison methods undergo evaluation using this identical pipeline configuration.

1.2 COMPARISONS ON INFERENCE SPEED.

We assessed the running speed of various methods using open-source code. In Table 1, our approach demonstrated exceptionally competitive performance while maintaining a fast inference speed among many lightweight methods.

Table 1: Comparisons on the Inference Speed. The speed is calculated as the average feature extraction inference speed on HPatches (480×640) with the same setting

Methods	Superpoint	D2-Net	SFD2	MTLDesc	R2D2	SAMFeat
Inference Speed	31FPS	6FPS	11FPS	24FPS	8FPS	21FPS

1.3 EDGE LEARNING GUIDED BY SAM.

As mentioned in Section 3.2, SAMFeat learns edge maps from SAM and utilizes Edge Attention Guidance (EAG) to further enhance the precision of local feature detection and description by encouraging the network to prioritize attention to the edge region. Figure 1 demonstrates the learning outcome of SAMFeat. With the fine-grained object boundaries from SAM, SAMFeat is able to learn clear object edges. This illustrates two things: first, the encoded feature that is used to generate the edge map contains rich edge information, and second, with a clear and accurate generated edge map and EAG, SAMFeat is able to better capture the details of edge areas and improve the robustness of local descriptors.

1.4 MORE ABLATION TESTS.

Table 2: Ablation test on the Pixel Semantic Relational Distillation (PSRD). When distilling the image feature from the SAM encoder, two approaches are tested: the first is our proposed PSRD and the second is direct semantic feature distillation (DSFD) between SAM's encoded feature and SAMFeat's encoded feature. The MMA @3 on HPatches are recorded for the baseline model with PSRD only

	MMA @3
DSFD	76.9
PSRD	**78.6**

[1] https://github.com/GrumpyZhou/image-matching-toolbox

[2] https://www.visuallocalization.net

Figure 1: Left: Random images selected from HPatches. Right: Learned edge boundaries from SAMFeat under the guide of SAM.

Table 3: Ablation test on edge feature enhancement. The obtained edge map can be incorporated two multi-levels of the encoded feature. The MMA @3 on HPatches are recorded to show the effectiveness on edge map when incorporating different levels of encoded features.

	MMA @3
C_1	81.3
C_2	81.7
C_3	**82.1**
C_4	81.8

Table 4: Ablation test on hyper-parameters on WCS. M is a margin parameter used to protect distinctiveness within semantic groupings, and T means the temperature coefficient. The MMA @3 on HPatches are recorded to show the effectiveness on different hyper-parameter values. As shown, the $\mathrm{M} = 0.07$ and $\mathrm{T} = 5$ brings the best result.

(M, T)	MMA @3
0, 1	80.9
0.03, 1	81.2
0.05, 1	80.3
0.07, 1	**81.3**
0.09, 1	80.8
0.11, 1	80.5
0.07, 3	81.6
0.07, 5	**82.1**
0.07, 7	82.0
0.07, 9	81.8

1.5 Detailed Explanation on Pixel Semantic Relational Distillation.

The "Pixel Semantic Relational Distillation" (PSRD) is designed to distill relationship matrices $\mathcal{R}$ and $\mathcal{R}'$ from encoded image features in a pair of images.

Algorithm 1 Pixel Semantic Relational Distillation

Input: Image pair I_1, I_2;$H = W = 400$; $C = 256$; SAMFeat's encoder E; SAM's encoder E'.
Output: SAMFeat's Relationship Matrix $\mathcal{R}$; SAM's Relationship Matrix $\mathcal{R}'$.

1: Given I_1, I_2, an encoded image feature $\mathcal{F} \in \mathbb{R}^{\frac{1}{8}H \times \frac{1}{8}W \times C}$ can be obtained via E.
2: Given I_1, I_2, an encoded image feature $\mathcal{F}' \in \mathbb{R}^{64 \times 64 \times C}$ can be obtained via E'.
3: Downsample $\mathcal{F}'$ to $\mathcal{F}'_{down} \in \mathbb{R}^{\frac{1}{8}H\frac{1}{8}W \times C}$
4: Flatten $\mathcal{F}$ and $\mathcal{F}'_{down}$ then calculate mean on $dim = C$ to obtain $\mathcal{F}_{flatten} \in \mathbb{R}^{\frac{1}{8}H\frac{1}{8}W}$ and $\mathcal{F}'_{flatten} \in \mathbb{R}^{\frac{1}{8}H\frac{1}{8}W}$
5: Construct Relationship Matrix $\mathcal{R} \in \mathbb{R}^{\frac{1}{8}H\frac{1}{8}W \times \frac{1}{8}H\frac{1}{8}W}$ and $\mathcal{R}' \in \mathbb{R}^{\frac{1}{8}H\frac{1}{8}W \times \frac{1}{8}H\frac{1}{8}W}$ from $\mathcal{F}_{flatten}$ and $\mathcal{F}'_{flatten}$ respectively, where $\mathcal{R}(i,j) = \frac{\mathcal{F}(i)\cdot\mathcal{F}(j)}{|\mathcal{F}(i)||\mathcal{F}(j)|}$, and same applied for $\mathcal{R}'$
6: **return** $\mathcal{R}$ and $\mathcal{R}'$.

Method	Source	Images	MMA@3
SuperPoint	COCO	80,000	64.5
D2Net	MegaDepth	617,774	42.9
R2D2	Aachen and Web images	**12,083**	68.6
ASLFeat	GL3D	1600,000	72.3
MTLDesc	MegaDepth	23,600	78.7
SFD2	Aachen and Web images	**12,083**	70.6
TRR	COCO + Image Matching Challenge	106,000	79.8
SAMFeat	MegaDepth	23,600	**82.1**

Table 5: Comparisons on the number of Training Samples. *MMA*@3 is the mean matching accuracy evaluated on the HPatches datasets. SAMFeat achieves an outstanding mean matching accuracy *MMA*@3 of 82.1, surpassing other methods, despite utilizing a relatively small training dataset of 23,600 images. This highlights the effectiveness of SAMFeat in achieving superior performance with limited training samples, underscoring its efficiency and robustness compared to alternative methods.

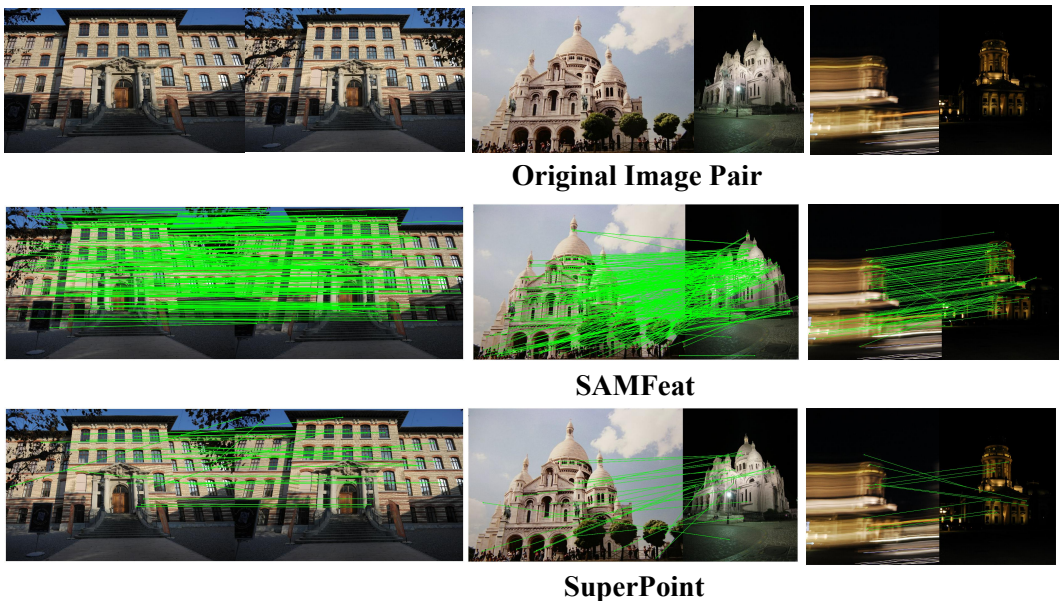

Figure 2: More Visualizations. Top: Original mage pair. Mid: SAMFeat matching visualization. Bot: SuperPoint Matching visulization

1.6 Local Feature Learning Meets the Visual Foundation Model.

Visual foundation models trained on large-scale datasets exhibit superb zero-sample generalization and performance motivation potential for downstream tasks. Applying visual foundation models to local feature learning is an interesting and meaningful research topic. Visual foundation models are mainly categorized into five types including segmentation foundation models (e.g. SAM Kirillov et al. (2023)), visual pre-training models (e.g. DINOv2 Oquab et al. (2023), MAE He et al. (2022), EVA Fang et al. (2023)), visual language models (e.g. CLIP Radford et al. (2021)), generative foundation models (e.g. Stable Diffusion Ramesh et al. (2022)), and multimodal foundation models (e.g. BLIPv2 Li et al. (2023)).

(1) **Segmentation foundation models**: Our proposed SAMFeat provides an in-depth exploration of the introduction of segmentation base models into local feature learning, and the results show that the visual base model has a significant facilitating effect on local feature learning.

Table 6: A quantitative analysis on the overhead training time costs for adding the extra loss functions. ✓means denotes applied loss components. Note that our method only requires training for 6 hours using two Nvidia RTX 3090 GPUs. Compared to other work like ASLFeat (42 hours on a single NVIDIA RTX 2080Ti) and TRR (30 hours for training with two NVIDIA-A100 GPUs), this demonstrates a totally reproducible cost for individual researchers. Each loss function will inevitably cause extra training costs, however, the tradeoff between a minimal incremental in time and the improvement in accuracy is reasonable, and the final training cost is acceptable. This further demonstrates the lightweight nature of our approach in the field of feature learning and description, making it easily implementable and resource-efficient.

PSRD	EAG	WCS	Training time in Hours
			3.6
✓			4.7
✓	✓		5.1
✓	✓	✓	6.0

Table 7: Detailed Ablation Study on SAMFeat. ✓means denotes applied components. The results of MMA@ 3 on HPatches of removing each component individually in addition to applying the components sequentially are reported.

PSRD	EAG	WCS	MMA @3
			75.7
✓			78.6
✓	✓		80.9
✓	✓	✓	**82.1**
✓		✓	81.2
	✓	✓	79.4

(2) **Visual pre-training models**: Since contrast learning is also one of the essential elements of visual pretraining model training, they have a natural adaptation to local feature learning. Thus making local feature learning benefit from with training models is one of our future work.

(3) **Visual language models**: While the visual language model has image-level text alignment capabilities, it lacks localized awareness of the Therefore it may not be able to be applied with local feature learning, on the contrary it has significant advantages for image-level representation learning tasks such as image retrieval and visual position regression (VPR).

(4) **Generative foundation models**: Generative base models can be used to improve scene generalization for local feature learning by being used to synthesize data. And how to generate geometrically consistent high-quality synthetic datasets for application to local feature learning tasks is one of our future work.

(5) **Multimodal foundation models**: The multimodal base model usually transforms images into tokens through an encoder to feed into a large predictive model (LLM), while its strength is the high-level semantic understanding and reasoning capabilities. Therefore, it is difficult for local feature learning to benefit directly from multimodal base models.

Table 8: Ablation test on adjusting the loss weight of EAG without WCS. The MMA @3 on HPatches are recorded, showing that it is difficult to achieve the effect of imposing WCS by only adjusting the loss weights.

Weights of PSRD	Weights of EAG	MMA @3
1.0	0.5	80.7
1.0	1.0	80.9
1.0	1.5	81.0

REFERENCES

Yuxin Fang, Quan Sun, Xinggang Wang, Tiejun Huang, Xinlong Wang, and Yue Cao. Eva-02: A visual representation for neon genesis. *arXiv preprint arXiv:2303.11331*, 2023.

Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 16000–16009, 2022.

Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. *arXiv preprint arXiv:2304.02643*, 2023.

Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. *arXiv preprint arXiv:2301.12597*, 2023.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pp. 8748–8763. PMLR, 2021.

Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 1(2):3, 2022.