Patch-wise Retrieval: An Interpretable Instance-Level Image Matching

Wonseok Choi[1] Sohwi Lim[2] Nam Hyeon-Woo[1] Moon Ye-Bin[1]
Dong-Ju Jeong[3] Jinyoung Hwang[3] Tae-Hyun Oh[2]
[1]POSTECH [2]KAIST [3]Samsung Research

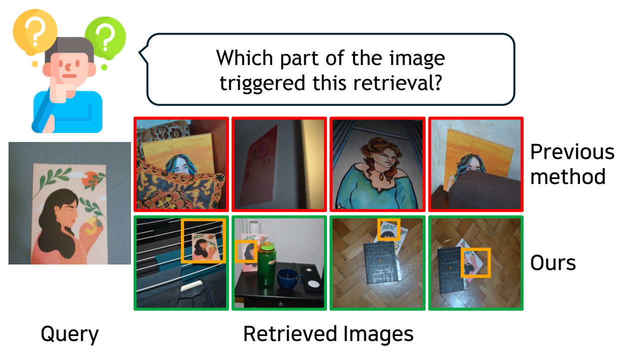
Abstract

Instance-level image retrieval aims to find images containing the same object as a given query, despite variations in size, position, or appearance. To address this challenging task, we propose Patchify, a simple yet effective patch-wise retrieval framework that offers high performance, scalability, and interpretability without requiring fine-tuning. Patchify divides each database image into a small number of structured patches and performs retrieval by comparing these local features with a global query descriptor, enabling accurate and spatially grounded matching. To assess not just retrieval accuracy but also spatial correctness, we introduce LocScore, a localization-aware metric that quantifies whether the retrieved region aligns with the target object. This makes LocScore a valuable diagnostic tool for understanding and improving retrieval behavior. We conduct extensive experiments across multiple benchmarks, backbones, and region selection strategies, showing that Patchify outperforms global methods and complements state-of-the-art reranking pipelines. Furthermore, we apply Product Quantization for efficient large-scale retrieval and highlight the importance of using informative features during compression, which significantly boosts performance.

1. Introduction

Instance retrieval aims to find images in a database containing the same visual instance as a query image, regardless of variations in perspective, scale, lighting, or background [12, 33]. It is widely applicable in scenarios such as personal photo search and product or landmark retrieval [6, 19, 26, 29, 36, 39].

Robust feature representations are essential due to appearance changes of the same instance. Recent methods [14, 30] achieve high accuracy by fine-tuning large encoders on domain-specific datasets or reranking with hundreds of dense local descriptors per image. However, these require costly annotations or incur heavy memory and computation overhead. Efficient solutions without fine-tuning or dense local matching remain underexplored.



	Previous method	Ours
Interpretability	✗	✓
Performance	✗	✓
Scalability	✗	✓

Figure 1. **Overview of our patch-wise retrieval framework.** Given a query image, global methods [14, 21, 38] often retrieve visually similar images without indicating what triggered the match. Our approach retrieves correct instances with spatial interpretability by identifying the most relevant regions.

In this work, we propose Patchify, a simple yet effective patch-wise retrieval pipeline that uses only a few local features per image. By combining multi-scale grid patches with pretrained visual encoders, Patchify enables efficient, interpretable, and accurate retrieval. Experiments show that patch-wise features consistently outperform global descriptors, and our simple grid-based method matches the performance of more complex SOTA approaches. We also investigate various design choices for patch-wise retrieval as a set of practical "bag-of-tricks," which are presented in the supplementary material due to space constraints.

We summarize our main contributions as follows:

- A patch-wise retrieval framework operating without fine-tuning or region proposals.
- LocScore, a localization-aware metric for evaluating retrieval accuracy and spatial alignment.
- Extensive experiments across backbones and region sampling strategies, showing competitive performance with greater simplicity.

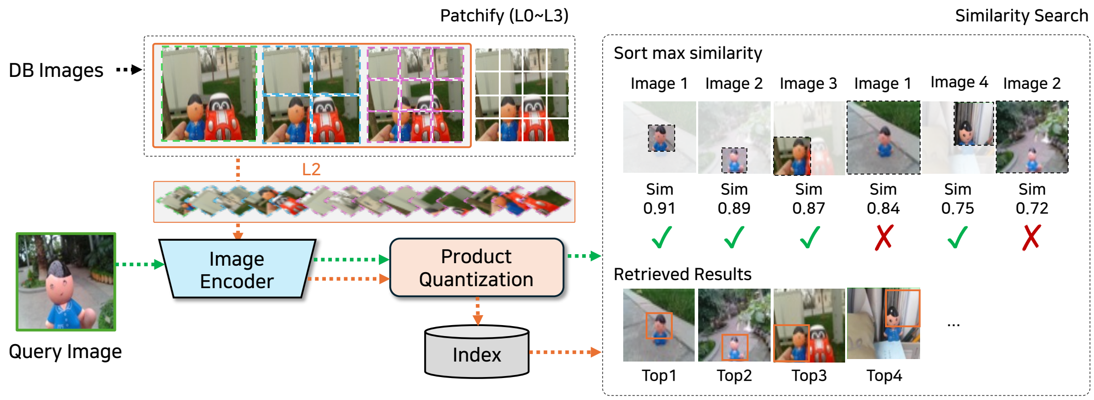


Figure 2. **Overview of our Patchify retrieval pipeline.** The L2 configuration extracts multi-scale grid patches (1×1, 2×2, and 3×3), which are individually encoded for feature extraction and indexing. Retrieval computes patch-level similarities across the database, ranking each image by its highest-scoring patch.

2. Method

Our goal is to design an instance-level image retrieval framework that not only achieves high accuracy and efficiency, but also offers strong interpretability to reveal *why* a result is retrieved. We propose **Patchify**, a single-stage retrieval pipeline that explicitly enables spatial alignment and interpretable retrieval outcomes. We further introduce **LocScore**, a localization-aware metric that quantifies whether retrieval decisions are based on the correct image regions, providing a principled way to assess interpretability.

2.1. Patchify: Patch-wise Retrieval Pipeline

Patchify integrates local spatial cues into retrieval without complex region proposals. Each database image is divided into multi-scale, non-overlapping grid patches (e.g., 1×1, 2×2, 3×3), producing a compact set of local descriptors using a frozen image encoder such as CLIP or DINOv2. A query image is encoded as a single global descriptor for scalability. For each database image, we identify the patch with the highest similarity to the query and use its score for ranking.

This design achieves strong performance while significantly reducing the number of descriptors compared to dense local matching, leading to lower memory and computation costs. The use of structured patches also makes the retrieval process interpretable, as it reveals which specific region triggered the match. Efficiency enhancements such as descriptor compression are described in the supplementary material.

2.2. LocScore: Localization-aware Metric

To evaluate the effectiveness of local features in instance retrieval, we introduce a localization-aware metric, LocScore, which quantifies not only whether the correct image is retrieved, but also how accurately the retrieved region aligns with the target object. Given a query image, the system returns a ranked list of retrieved images along with their most similar local patch.

To reflect not just the presence of correct retrievals but also their order in the ranked list, we weight each prediction by its retrieval precision. This allows us to evaluate localization performance in a retrieval-aware manner, rewarding predictions that are both accurate and highly ranked.

Let $B_{\text{gt}}^{n,i}$ and $B_{\text{pred}}^{n,i}$ denote the ground-truth and predicted bounding boxes, respectively, for the i-th ground-truth image of the n-th query. Suppose this ground-truth image is retrieved at rank $r^{n,i}$, and let $h^{n,i}$ denote the number of ground-truth positives retrieved within the top-$r^{n,i}$ positions.

We first define the localization score for a single query n as:

$$\text{LocScore}^{(n)} = \frac{1}{I_n} \sum_{i=1}^{I_n} \frac{h^{n,i}}{r^{n,i}} \cdot \text{IoU}(B_{\text{gt}}^{n,i}, B_{\text{pred}}^{n,i}), \quad (1)$$

where I_n is the number of ground-truth positives for query n. If a ground-truth image is not retrieved for a given query, its IoU is treated as zero.

The overall LocScore across all queries is then computed by averaging the per-query scores:

$$\text{LocScore} = \frac{1}{N} \sum_{n=1}^{N} \text{LocScore}^{(n)}, \quad (2)$$

where N is the total number of queries.

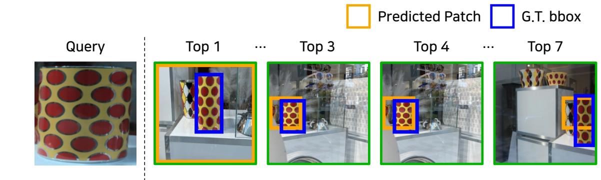
We further introduce a thresholded variant of the metric to provide a binary notion of successful localization:

$$\text{LocScore}^{(n)}(\delta) = \frac{1}{I_n} \sum_{i=1}^{I_n} \frac{h^{n,i}}{r^{n,i}} \cdot \mathbb{I}\left[\text{IoU}(B_{\text{gt}}^{n,i}, B_{\text{pred}}^{n,i}) \geq \delta\right], \quad (3)$$

$$\text{LocScore}(\delta) = \frac{1}{N} \sum_{n=1}^{N} \text{LocScore}^{(n)}(\delta), \quad (4)$$

where $\mathbb{I}[\cdot]$ is the indicator function.

To reduce sensitivity to the choice of threshold δ, we av-



Threshold	Top 1	Top 3	Top 4	Top 7	LocScore
$\delta = 0.3$	✗	✓	✓	✓	0.50
$\delta = 0.4$	✗	✗	✓	✓	0.33
$\delta = 0.5$	✗	✗	✓	✗	0.19

✓: Pass (IoU $\geq \delta$), ✗: Fail (IoU $< \delta$)

Figure 3. **Example of thresholded LocScore** (δ)**.** LocScore variation with different IoU thresholds δ. A retrieved result is marked correct if the predicted patch overlaps the ground-truth box by at least δ.

erage over a set of thresholds $\mathcal{T}$:

$$\text{mLocScore} = \frac{1}{|\mathcal{T}|} \sum_{\delta \in \mathcal{T}} \text{LocScore}(\delta), \tag{5}$$

where $\mathcal{T} = \{0.3, 0.4, 0.5\}$ in our experiments.

This formulation enables both fine-grained and thresholded interpretations of localization performance. Our proposed metrics evaluate two critical aspects: whether the correct image is retrieved, and whether the retrieval is based on the correct local region. As illustrated in Figure 3, the thresholded variant further allows evaluation at different granularities by varying the IoU threshold. Even when the correct image is retrieved, a low LocScore indicates that the retrieval relied on spatially irrelevant content. This makes LocScore a powerful and interpretable diagnostic tool, akin to class activation maps (CAM) in explainable AI.

3. Experiment

We evaluate our Patchify method (Figure 2) on two standard instance retrieval benchmarks, INSTRE [34] and ILIAS [14], reporting both mean Average Precision (mAP) and our localization-aware metric, LocScore, which reflects retrieval accuracy and spatial alignment. In the main paper, we present results using the continuous LocScore formulation, and provide analyses with thresholded variants in the supplementary material (Section S9).

3.1. Effectiveness of Patch-wise Representation

First, we compare Patchify-based local representations with conventional global descriptors using two pretrained encoders, DINOv2 [21] and CLIP [11].

Global vs. Local: A Comparative Analysis As summarized in Table 1, local features consistently outperform global ones across different datasets, encoders, and metrics. This highlights the importance of spatial granularity in improving both retrieval performance and localization quality.

Table 1. mAP (%) and LocScore (%) of global and local features on INSTRE and ILIAS using DINOv2 and CLIP encoders.

Encoder	Type	INSTRE		ILIAS	
		mAP	LocScore	mAP	LocScore
DINOv2	global	57.70	15.07	40.56	12.18
	local	72.54	22.22	57.52	18.49
CLIP	global	73.84	18.10	31.60	9.18
	local	87.57	30.37	53.35	17.95

We observe local features provide a substantial boost, particularly in challenging scenarios such as occluded or off-center objects. We hypothesize local features complement global ones by capturing fine-grained, spatially grounded cues that may be lost in global representations.

Qualitative result As shown in Figure 4, we compare the retrieved images of the Global and Local methods. We observe that Patchify can identify the small and non-centered objects. These qualitative results coincide with Figure S1. Therefore, local features play a crucial role in instance retrieval, particularly when dealing with diverse visual conditions affecting the target objects.

3.2. Method Comparison

In this section, we investigate how different region selection strategies affect instance retrieval performance, and compare our Patchify method against recent state-of-the-art approaches, both as a standalone feature extractor and within reranking pipelines.

3.2.1. Different region selection strategies

We investigate how different region selection strategies influence instance retrieval, comparing our grid-based Patchify approach with sliding windows and region proposal methods using strong encoders such as SigLIP. Detailed descriptions of each approach are provided in Section S5. As shown in Table 2, sliding window and proposal-based methods generally achieve higher mAP and LocScore than Patchify. We attribute these improvements to their finer spatial coverage or stronger semantic alignment, which increases the likelihood of capturing the target instance accurately. However, these benefits come at the cost of additional computation and memory, whereas Patchify maintains competitive performance with much greater efficiency. Additional results and discussion are presented in the supplementary material.

3.2.2. Comparison with SOTA Methods

We evaluate Patchify against state-of-the-art instance retrieval methods on the mini-ILIAS [14], an extension of ILIAS with challenging distractors from YFCC100M [32]. The comparison covers two setups: single-stage global retrieval and two-stage reranking frameworks.

Single-Stage Retrieval. Table 3 reports results for global feature baselines, the SigLIP linear adaptation from IL-

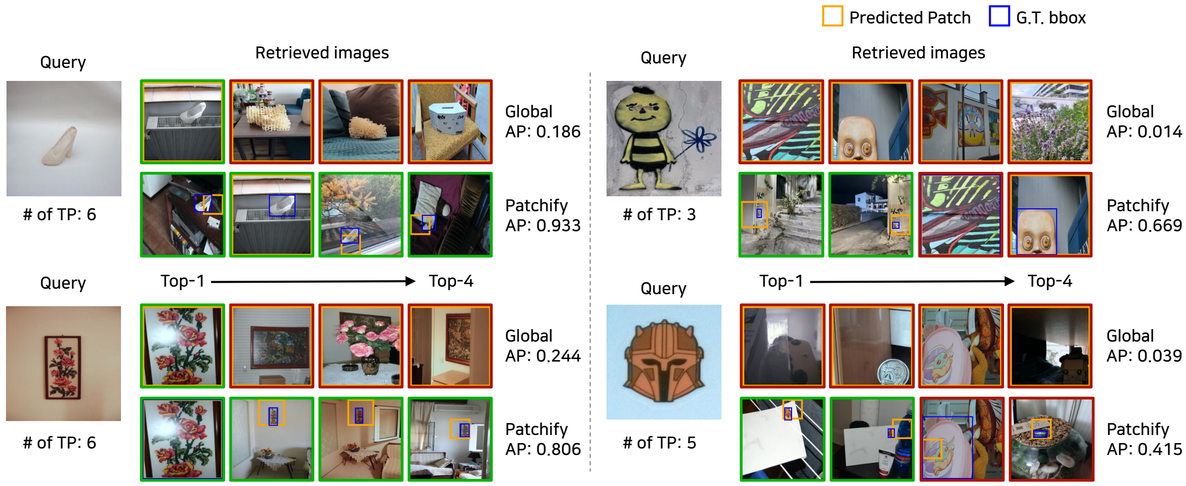


Figure 4. **Qualitative comparison between Global and Patchify on ILIAS.** While global features often fail to retrieve small or non-centered instances, Patchify successfully localizes and retrieves correct instances with better spatial grounding.

Table 2. Comparison of different localization strategies in terms of mAP (%) and LocScore (%) on INSTRE and ILIAS.

Method	**INSTRE**		**ILIAS**	
	mAP	**LocScore**	**mAP**	**LocScore**
Patchify	87.01	24.29	65.16	19.75
Sliding Window (0.5)	89.26	27.50	68.24	22.36
Sliding Window (0.25)	90.62	29.87	70.02	24.57
Region Proposal	92.96	71.44	84.19	68.23

IAS, and our Patchify variants. Patchify consistently surpasses all baselines, achieving competitive zero-shot performance without fine-tuning. Stronger backbones such as DINOv3 [28] further boost its effectiveness, indicating that the patch-based design benefits directly from improved encoders.

Integration into Reranking Frameworks. To assess compatibility with reranking systems, we integrate Patchify into AMES [30], a state-of-the-art local alignment method. Patchify enhances AMES performance and even approaches or exceeds prior SOTA results, showing that it serves effectively as both a standalone retriever and a first-stage feature representation within reranking pipelines.

Method	mAP@1k	DB mem. [GB]
Global feature		
DINOv2†	18.80	9.55
OpenCLIP†	22.90	9.55
SigLIP2†	37.30	9.55
Reranking (AMES)		
DINOv2†	26.50	1536
OpenCLIP†	32.90	1536
Global feature		
DINOv3	21.80	19.09
+ Patchify (L3)	42.98	267.29
SigLIP	20.41	19.09
+ Patchify (L2)	50.27	267.29
SigLIP†	33.86	9.55
+ Patchify (L3)	40.48	286.38
Reranking (AMES)		
DINOv3	29.86	1536
+ Patchify (L3)	50.72	1536
SigLIP	26.99	1536
+ Patchify (L2)	56.54	1536
SigLIP†	40.91	1536
+ Patchify (L3)	48.21	1536

Table 3. **Performance on mini-ILIAS in terms of mAP@1k and DB memory.** A dagger (†) denotes the linear adaptation baseline from the ILIAS [14], where the SigLIP encoder is fine-tuned on UnED [35] with a linear probing strategy. Patchify markedly improves over the global baselines, reaching or even surpassing the reranking performance of AMES.

4. Conclusion

We introduced **Patchify**, a lightweight patch-wise retrieval framework that achieves high accuracy, scalability, and strong interpretability without fine-tuning. By identifying the most relevant regions for each retrieval, Patchify makes the decision process transparent and provides deeper insight into retrieval behavior. Coupled with **LocScore**, our localization-aware metric, it offers quantitative and spatial evaluations that reveal when and why a retrieval succeeds or fails, and achieves competitiveness against more complex approaches across multiple benchmarks. While the main paper presents core findings, the supplementary material reports extensive experiments on encoder choices, patch configurations, region selection strategies, and practical scalability techniques such as PQ training recipes, offering actionable guidance for deploying interpretable and scalable retrieval in large-scale scenarios.

Acknowledgment

This work was supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government(MSIT) (No. 2022-0-00124, No.RS-2022-II220124, Development of Artificial Intelligence Technology for Self-Improving Competency-Aware Learning Capabilities; Artificial Intelligence Graduate School Program(POSTECH))

References

[1] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool. Speeded up robust features. In *Computer Vision – ECCV 2006*, pages 404–417. Springer, 2006. 1

[2] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9650–9660, 2021. 1

[3] Junsuk Choe, Seong Joon Oh, Seungho Lee, Sanghyuk Chun, Zeynep Akata, and Hyunjung Shim. Evaluating weakly supervised object localization methods right, 2020. 2

[4] Junsuk Choe, Seong Joon Oh, Sanghyuk Chun, Seungho Lee, Zeynep Akata, and Hyunjung Shim. Evaluation for weakly supervised object localization: Protocol, metrics, and datasets, 2021. 2

[5] Amil Dravid, Yossi Gandelsman, Alexei A. Efros, and Assaf Shocher. Rosetta neurons: Mining the common units in a model zoo, 2023. 2

[6] Cristopher Flagg and Ophir Frieder. Reconstruction of artifacts from digital image repositories. *J. Comput. Cult. Herit.*, 16(1), 2022. 1

[7] Yossi Gandelsman, Alexei A. Efros, and Jacob Steinhardt. Interpreting clip's image representation via text-based decomposition, 2024. 2

[8] Yossi Gandelsman, Alexei A. Efros, and Jacob Steinhardt. Interpreting the second-order effects of neurons in clip, 2025. 2

[9] Peng Gao, Shijie Geng, Renrui Zhang, Teli Ma, Rongyao Fang, Yongfeng Zhang, Hongsheng Li, and Yu Qiao. Clip-adapter: Better vision-language models with feature adapters. *International Journal of Computer Vision*, 132(2): 581–595, 2024. 1

[10] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition, 2015. 3

[11] Gabriel Ilharco, Mitchell Wortsman, Ross Wightman, et al. Openclip: An open-source implementation of clip. https://github.com/mlfoundations/open_clip, 2021. 3, 1

[12] Tomas Jenicek and Ondřej Chum. No fear of the dark: Image retrieval under varying illumination conditions, 2019. 1

[13] Herve Jégou, Matthijs Douze, and Cordelia Schmid. Product quantization for nearest neighbor search. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(1):117–128, 2011. 2, 6

[14] Giorgos Kordopatis-Zilos, Vladan Stojnic, Anna Manko, Pavel Suma, Nikolaos-Antonios Ypsilantis, Nikos Efthymiadis, Zakaria Laskar, Jiri Matas, Ondrej Chum, and Giorgos Tolias. Ilias: Instance-level image retrieval at scale. In *CVPR*, 2025. 1, 3, 4, 5

[15] Feng Liang, Bichen Wu, Xiaoliang Dai, Kunpeng Li, Yinan Zhao, Hang Zhang, Peizhao Zhang, Peter Vajda, and Diana Marculescu. Open-vocabulary semantic segmentation with mask-adapted clip, 2023. 1

[16] Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Qing Jiang, Chunyuan Li, Jianwei Yang, Hang Su, Jun Zhu, and Lei Zhang. Grounding dino: Marrying dino with grounded pre-training for open-set object detection, 2024. 5

[17] Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. A convnet for the 2020s. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11976–11986, 2022. 3

[18] David G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004. 1

[19] Shiyang Lu, Haonan Chang, Eric Pu Jing, Abdeslam Boularias, and Kostas Bekris. Ovir-3d: Open-vocabulary 3d instance retrieval without training on 3d data. In *Conference on Robot Learning*, pages 1610–1620. PMLR, 2023. 1

[20] Huaishao Luo, Lei Ji, Ming Zhong, Yang Chen, Wen Lei, Nan Duan, and Tianrui Li. Clip4clip: An empirical study of clip for end to end video clip retrieval, 2021. 1

[21] 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. 1, 3

[22] Yash Patel, Lluis Gomez, Marçal Rusiñol, Dimosthenis Karatzas, and CV Jawahar. Self-supervised visual representations for cross-modal retrieval. In *Proceedings of the 2019 on international conference on multimedia retrieval*, pages 182–186, 2019. 1

[23] Filip Radenović, Giorgos Tolias, and Ondřej Chum. Fine-tuning cnn image retrieval with no human annotation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(7):1655–1668, 2019. 1

[24] Alec Radford, Jong Wook Kim, et al. Learning transferable visual models from natural language supervision. In *ICML*, 2021. 1

[25] Yongming Rao, Wenliang Zhao, Guangyi Chen, Yansong Tang, Zheng Zhu, Guan Huang, Jie Zhou, and Jiwen Lu. Denseclip: Language-guided dense prediction with context-aware prompting, 2022. 1

[26] Taichi Sakaguchi, Akira Taniguchi, Yoshinobu Hagiwara, Lotfi El Hafi, Shoichi Hasegawa, and Tadahiro Taniguchi. Object instance retrieval in assistive robotics: Leveraging fine-tuned simsiam with multi-view images based on 3d semantic map. In *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 7817–7824. IEEE, 2024. 1

[27] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition, 2015. 3

[28] Oriane Siméoni, Huy V. Vo, Maximilian Seitzer, Federico Baldassarre, Maxime Oquab, Cijo Jose, Vasil Khalidov, Marc Szafraniec, Seungeun Yi, Michaël Ramamonjisoa, Francisco Massa, Daniel Haziza, Luca Wehrstedt, Jianyuan Wang, Timothée Darcet, Théo Moutakanni, Leonel Sentana, Claire Roberts, Andrea Vedaldi, Jamie Tolan, John Brandt, Camille Couprie, Julien Mairal, Hervé Jégou, Patrick Labatut, and Piotr Bojanowski. Dinov3, 2025. 4

[29] Ivan Sipiran, Patrick Lazo, Cristian Lopez, Milagritos Jimenez, Nihar Bagewadi, Benjamin Bustos, Hieu Dao, Shankar Gangisetty, Martin Hanik, Ngoc-Phuong Ho-Thi, Mike Holenderski, Dmitri Jarnikov, Arniel Labrada, Stefan Lengauer, Roxane Licandro, Dinh-Huan Nguyen, Thang-Long Nguyen-Ho, Luis A. Perez Rey, Bang-Dang Pham, Minh-Khoi Pham, Reinhold Preiner, Tobias Schreck, Quoc-Huy Trinh, Loek Tonnaer, Christoph von Tycowicz, and The-Anh Vu-Le. Shrec 2021: Retrieval of cultural heritage objects•. *Comput. Graph.*, 100(C):1–20, 2021. 1

[30] Pavel Suma, Giorgos Kordopatis-Zilos, Ahmet Iscen, and Giorgos Tolias. Ames: Asymmetric and memory-efficient similarity estimation for instance-level retrieval. In *European Conference on Computer Vision*, pages 307–325. Springer, 2024. 1, 4, 5

[31] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions, 2014. 3

[32] Bart Thomee, David A Shamma, Gerald Friedland, Benjamin Elizalde, Karl Ni, Douglas Poland, Damian Borth, and Li-Jia Li. Yfcc100m: The new data in multimedia research. *Communications of the ACM*, 59(2):64–73, 2016. 3

[33] Akihiko Torii, Relja Arandjelović, Josef Sivic, Masatoshi Okutomi, and Tomas Pajdla. 24/7 place recognition by view synthesis. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1808–1817, 2015. 1

[34] Yang Wang, Sheng Yang, Tianzhu Qi, Alan Yuille, and Hong Tang. Instre: A new benchmark for instance-level object retrieval and recognition. In *ACM MM*, 2015. 3, 5

[35] Nikolaos-Antonios Ypsilantis, Kaifeng Chen, Bingyi Cao, Mário Lipovský, Pelin Dogan-Schönberger, Grzegorz Makosa, Boris Bluntschli, Mojtaba Seyedhosseini, Ondřej Chum, and André Araujo. Towards universal image embeddings: A large-scale dataset and challenge for generic image representations. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 11290–11301, 2023. 4

[36] Jiangbo Yuan, An-Ti Chiang, Wen Tang, and Antonio Haro. eproduct: A million-scale visual search benchmark to address product recognition challenges, 2021. 1

[37] Xiaohua Zhai, Alexander Kolesnikov, Neil Houlsby, and Lucas Beyer. Scaling vision transformers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12104–12113, 2022. 3

[38] Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 11975–11986, 2023. 1, 3

[39] Xunlin Zhan, Yangxin Wu, Xiao Dong, Yunchao Wei, Minlong Lu, Yichi Zhang, Hang Xu, and Xiaodan Liang. Product1m: Towards weakly supervised instance-level product retrieval via cross-modal pretraining, 2021. 1

[40] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization, 2015. 2

[41] Jinghao Zhou, Chen Wei, Huiyu Wang, Wei Shen, Cihang Xie, Alan Yuille, and Tao Kong. ibot: Image bert pre-training with online tokenizer. *arXiv preprint arXiv:2111.07832*, 2021. 1

[42] Kaiyang Zhou, Ziwei Liu, Yu Qiao, Tao Xiang, and Chen Change Loy. Domain generalization: A survey. *IEEE transactions on pattern analysis and machine intelligence*, 45(4):4396–4415, 2022. 3

S10. Qualitative Results

Figure S10. Qualitative results of Global and Patchify on ILIAS

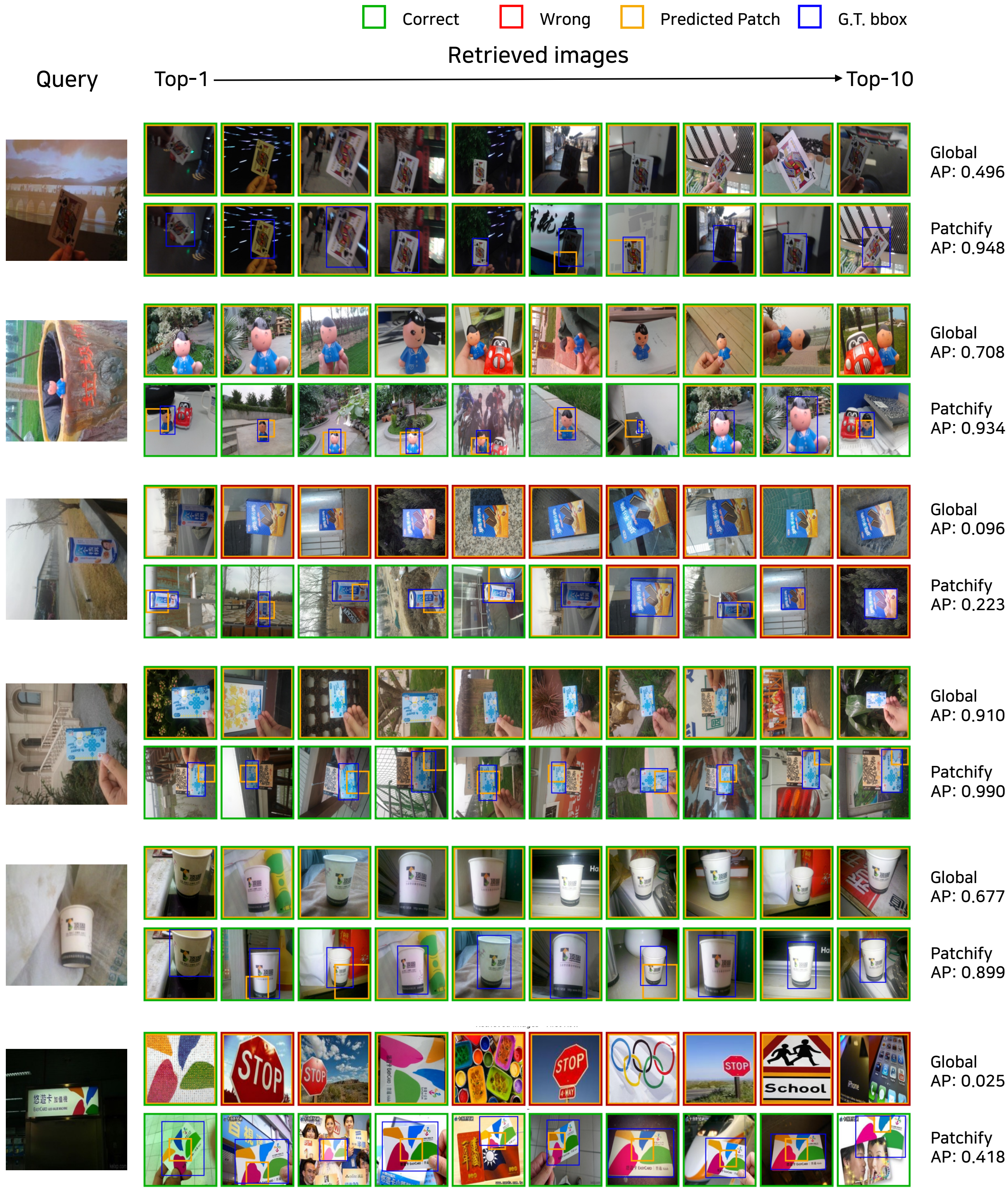


Figure S11. Qualitative results of Global and Patchify on INSTRE