VP Lab: a PEFT-Enabled Visual Prompting Laboratory for Semantic Segmentation

Niccolo Avogaro[1,2] , **Thomas Frick**[1] , **Yagmur G. Cinar**[1] , **Daniel Caraballo**[1] , **Cezary Skura**[1] , **Filip M. Janicki**[1] , **Piotr Kluska**[1] , **Brown Ebouky**[1,2] , **Nicola Farronato**[1,2] , **Florian Scheidegger**[1] , **Cristiano Malossi**[1] , **Konrad Schindler**[2] , **Andrea Bartezzaghi**[1] , **Roy Assaf**[1] , **Mattia Rigotti**[1]

[1]IBM Research, [2]ETH Zürich

Abstract

Large-scale pretrained vision backbones have transformed computer vision by providing powerful feature extractors that enable various downstream tasks, including training-free approaches like visual prompting for semantic segmentation. Despite their success in generic scenarios, these models often fall short when applied to specialized technical domains where the visual features differ significantly from their training distribution. To bridge this gap, we introduce VP Lab, a comprehensive iterative framework that enhances visual prompting for robust segmentation model development. At the core of VP Lab lies E-PEFT, a novel ensemble of parameter-efficient fine-tuning techniques specifically designed to adapt our visual prompting pipeline to specific domains in a manner that is both parameter- and data-efficient. Our approach not only surpasses the state-of-the-art in parameter-efficient fine-tuning for the Segment Anything Model (SAM), but also facilitates an interactive, near-real-time loop, allowing users to observe progressively improving results as they experiment within the framework. By integrating E-PEFT with visual prompting, we demonstrate a remarkable 50% increase in semantic segmentation mIoU performance across various technical datasets using only 5 validated images, establishing a new paradigm for fast, efficient, and interactive model deployment in new, challenging domains. This work comes in the form of a demonstration[1]

1 Introduction

Foundation Models have revolutionized machine learning by shifting from task-specific models to generalist ones pretrained on large, diverse datasets and fine-tuned for various downstream tasks [et al., 2022]. In computer vision, models like Segment Anything Model (SAM) [Kirillov *et al.*, 2023], CLIP [Radford *et al.*, 2021], and DINOv2 [Oquab *et al.*, 2024] enable powerful functionalities such as semantic segmentation and classification in a zero-shot manner. Their versatile capabilities are central to many applications, including innovative approaches in applied computer vision, e.g. for visual inspection [Rigotti *et al.*, 2023]. This work focuses on semantic image segmentation, a key computer vision task essential for applications in medical imaging, autonomous driving, and visual inspection. We aim to develop a human-computer interaction workflow for few-shot open-world segmentation that efficiently addresses real-world, out-of-domain use cases.

We build on the existing visual prompting literature [Liu *et al.*, 2024b], [Frick *et al.*, 2024], [Avogaro *et al.*, 2025], which introduced pipelines relying on foundation models - in particular DINOv2 for feature matching and SAM to segment objects of interests based on minimal user annotations of reference images. While these pipelines support accurate visual prompting in generic vision domains [Frick *et al.*, 2024], they often struggle with specialized technical applications. Especially SAM, despite its broad capabilities, faces challenges in generating coherent segmentation masks for technical, domain-specific objects. This paper focuses on addressing these challenges, proposing a solution for scenarios where visual prompting alone is inadequate and lacks scalability.

To address this challenge, as our main innovation, we introduce E-PEFT, a scalable ensemble of parameter-efficient fine-tuning techniques for the SAM that ensures fast convergence and delivers state-of-the-art results for test-time training when integrated into a visual prompting pipeline. To fully leverage this technique in an interactive scenario, we introduce a complementary label correction workflow, which provides efficient mask refinement capabilities. Together, these developments notably enhance performance in complex, real-world scenarios where visual prompting based on generic pretrained models alone is inadequate. With the integration of E-PEFT into a base framework like [Frick *et al.*, 2024], we enable an unparalleled level of adaptability to novel domains without significantly sacrificing the interactivity between the user and the model, allowing users to see progressively better results as they continue experimenting within the new framework. This evolution transforms a visual prompting framework into a *Visual Prompting Laboratory* (VP Lab), empowering users to explore and flexibly address new challenging semantic segmentation use cases.

[1] Demonstration can be found in the official project page.

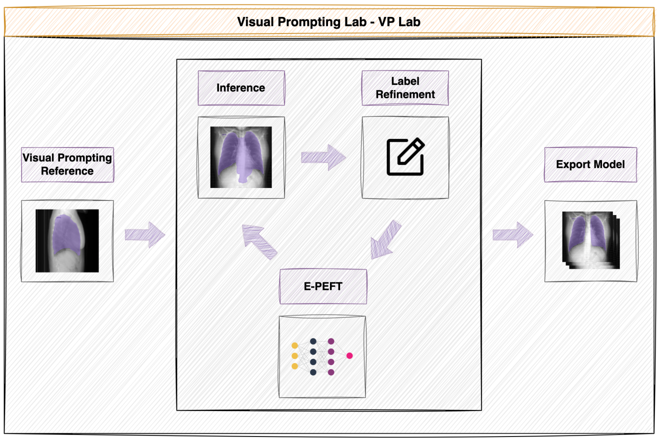

Figure 1: VP Lab Workflow: Users prompt and validate a reference image, which guides predictions on target datasets. They can refine these predictions with a labeling tool, and feed them into a parameter-efficient fine-tuning process of the underlying model. After iterative improvements, the optimized model can be exported and deployed as needed.

2 Related Work

Parameter-efficient fine-tuning techniques have become essential for adapting large pretrained foundation models to specific tasks with minimal computational resources. One of the pioneering techniques introduced by [Houlsby *et al.*, 2019] consists of adding trainable adapter layers inside the backbone, instead of fine-tuning the model parameters. The Low-Rank Adaptation (LoRA) framework introduced by [Hu *et al.*, 2021] achieves this by injecting low-rank trainable matrices into the attention layers of Transformer architectures, allowing for efficient fine-tuning without modifying the entire model. Building upon LoRA, QLoRA [Dettmers *et al.*, 2023] combines model quantization with low-rank adaptation, enabling fine-tuning of large language models on limited hardware by reducing memory usage.

Visual Prompt Tuning (VPT) [Jia *et al.*, 2022], adapts pretrained vision models to new tasks by learning a set of visual prompts, i.e. tokens which effectively condition the model without altering its original parameters. Additionally, the IA3 method [Liu *et al.*, 2022] offers a parameter-efficient approach by learning task-specific adapters, further enhancing the adaptability of large models to diverse tasks.

Several methods have been proposed for fine-tuning SAM [Kirillov *et al.*, 2023], each targeting different aspects of image segmentation. Notably, HQ-SAM [Ke *et al.*, 2023] focuses on improving segmentation quality by prioritizing high-fidelity outputs. BOFT [Liu *et al.*, 2024a] offers a parameter-efficient solution through orthogonal fine-tuning, optimizing segmentation performance while reducing computational costs and [Zhong *et al.*, 2024] proposes an hybrid solution adding low-rank convolution adapter layers to the SAM backbone.

3 Visual Prompting Lab

VP Lab (Figure 1) is a scalable segmentation framework that enhances visual prompting to address complex technical use cases. The workflow consists of the following steps: i) **Visual prompting and validating a reference image** – the user provides points on the reference image to indicate the object class of interest. These input serve as a sparse prompt for SAM, which generates a reference mask; ii) **Generating pseudolabels** – the SoftMatcher [Frick *et al.*, 2024] visual prompting algorithm computes pseudolabels for the target dataset; iii) **Refining labels** – labels are refined using an annotation tool in the web interface and validated by the human in the loop; iv) **Refining model** – the E-PEFT algorithm is used to fine-tune the SAM mask decoder in a matter of minutes, enhancing the segmentation capabilities of the full visual prompting pipeline and enabling it to handle new use cases not covered by the foundation model's internet-scale training dataset. The refined model can be exported for production or used to further refine predictions by restarting from step ii). This loop can be repeated as needed.

The key innovations of our framework, compared to approaches like Softmatcher [Frick *et al.*, 2024], focus on creating a system capable of generating models for real-world technical use cases within minutes. Achieving this requires enabling users to make substantial changes to the model through efficient fine-tuning of the visual prompted foundation model. To promptly address challenging use cases, the fine-tuning process must rely on accurate labels, execute quickly, and consume minimal resources. The first step for enabling this capability is the integration of a **label refinement tool** within the interface. This tool is essential for allowing users to contribute directly to the pipeline. By leveraging the interaction with the visual prompting framework, the labeling process becomes highly efficient, as users only need to refine existing outputs rather than annotate from scratch. Then, the main enabler and innovation of this pipeline is an efficient, **parameter-efficient fine-tuning technique** designed for test-time adaptation. To support for rapid iterations between the model and user, the fine-tuning procedure must be fast, scalable, and resource-efficient. This key introduction serves as pivotal component driving the pipeline's strong performance on OOD tasks. The following section delves further into its details.

Method. Most existing approaches for fine-tuning SAM

Table 1: Performance of the proposed E-PEFT compared to the single PEFT baselines and to the SOTA HQ-SAM on Kvasir-Seg and HQ44k datasets.

Model	Params	Kvasir-Seg	HQ-44k
SAM	0	72.88	84.49
Adapter	33.1K	84.60	87.17
IA3	5.6K	85.63	88.50
D-VPT	12.8K	86.65	88.49
LORA	144.4K	87.93	90.17
HQ-SAM	5.1M	87.97	89.95
E-PEFT	201.1K	**88.97**	**90.50**

rely on complex pipelines that add millions of parameters to the models. These methods typically involve high-capacity tuning processes that require hours to complete and often require multiple GPUs. To address this issue, we introduce **E-PEFT** (Ensemble of Parameter-Efficient Fine-Tuning), a scalable and efficient method optimized for the few-shot scenario. E-PEFT integrates multiple parameter-efficient techniques to enhance model adaptation when it comes to out-of-distribution tasks. The method targets SAM's decoder head, where parameter tuning has the greatest impact, enabling effective adaptation in low-data scenarios. E-PEFT combines LoRA, IA3, prompt tuning, and adapter modules, leveraging their orthogonal properties for improved performance.

The motivation behind E-PEFT stems from the fact that each method targets different parts of the model architecture, enabling seamless integration. This approach increases model capacity while maintaining a low parameter count, essential for fast convergence in low-data regimes. Specifically, LoRA decomposes the weight matrices of the decoder's linear layers into low-rank approximations. IA3 scales attention mechanisms and MLPs using three learned parameters per linear layer. Prompt tuning enhances learning by adding trainable memory tokens to the input sequence, which, in this case, are prepended to both of the decoder's input sequences. We also introduce a lightweight adapter module that shares the average image representation with the prompt through an MLP, in order for it to have a better initialization. E-PEFT leverages the complementary strengths of these techniques – LoRA's rapid convergence, IA3's flexibility in scaling activations, and the capacity-enhancing effects of prompt tuning and the adapter module – to optimize performance. The **synergy** between these methods allows E-PEFT to fine-tune only a small carefully selected subset of parameters that best maximize model performance on a given task. This is crucial for reducing computational costs and memory footprint, making it feasible to train on limited data or in resource-constrained environments.

Experiments. We validate our framework through experiments presented in Table 1. We highlight that E-PEFT outperforms the state-of-the-art HQ-SAM on the Kvasir-Seg [Jha *et al.*, 2019] and HQ-44k [Ke *et al.*, 2023] datasets, notably with three orders of magnitude fewer parameters. These results further demonstrate that the PEFT techniques are orthogonal, and their combination results in better performance compared to using each technique individually. As summarized in Table 2, our model excels in extremely limited data scenarios and shows a significant performance gap with respect to the state-of-the-art HQ-SAM, which fails to achieve good performance in 5-shot and 1-shot tuning, further demonstrating the effectiveness of our approach in low-data settings.

Table 2: Semantic segmentation performance of E-PEFT with respect to HQ-SAM on Kvasir-Seg when considering 1- and 5-shot scenarios.

Model	1-shot	5-shot
HQ-SAM	59.30	75.71
E-PEFT	**72.34**	**82.25**

Table 3: Evaluation of the Softmatcher visual prompting method fine-tuned with E-PEFT across varying number of tuning shots. "0-shot" denotes the baseline without fine-tuning.

Dataset	0-shot	5-shot	10-shot	40-shot
Kvasir-Inst.	40.28	63.44	63.33	65.92
PaxRay	36.39	48.61	51.19	50.97
DeepCrack	11.96	19.27	21.64	23.71
Corrosion CS	4.06	7.26	8.14	7.98
Average	**23.17**	**34.64**	**36.07**	**37.15**

We evaluate the effectiveness of E-PEFT when integrated into an existing visual prompting pipeline (SoftMatcher) on four out-of-distribution datasets: Kvasir-Inst. [Jha *et al.*, 2021], PaxRay [Seibold *et al.*, 2022], DeepCrack [Liu *et al.*, 2019], and Corrosion CS [Bianchi and Hebdon, 2021], which cover technical engineering and medical use-cases. Results in Table 3 show that the visual prompting algorithm alone performs poorly in these technical domains. However, when integrated with E-PEFT, performance improves drastically. With just 5 images for training—**completed in under a minute on a NVIDIA V100 GPU**—average performance increases by 50%, with some datasets showing up to an **80% improvement**. As more images are added, performance continues to improve. Tuning with 40 images takes only 2.5 minutes, showcasing the method's efficiency and effectiveness. This significant performance boost is made possible by the unique design choices in E-PEFT, which maximize both efficiency and capacity.

Deployed service and frontend. The interactive web interface is the core of VP Lab. It features a scalable architecture with an Angular-based [Jain *et al.*, 2014] frontend communicating via REST API with a Python backend, which organizes and submits compute tasks to PyTorch-based [Paszke *et al.*, 2017] inference and training services, leveraging the available GPU resources. When using the service, users can mark objects of interest on images using points, prompting the visual pipeline to generate precise segmentation masks for similar objects across target images. The initial results act as a support to iteratively build the final model. Users can manually refine masks using a set of annotation tools and use the resulting ground truth to fine-tune the model in a matter of minutes with extremely modest resources. This iterative process allows users to develop a deeper understanding of the model's behavior. By identifying its strengths and limitations, users can collaborate more effectively with the model, leading to improved outcomes.

Demonstration. We present a fully interactive user-model workflow, showing its effectiveness in an example of real-world medical use case. The user begins by prompting the first image from a randomly sampled subset of the PaxRay dataset. This image then serves as input for the visual prompting pipeline, which returns predictions across the entire dataset. These predictions are not always sufficiently good, therefore the user refines them slightly using the annotation tool. The refined annotations are then used to tune the model, which in turn generates better results. The entire process takes less than a minute. Finally, a second round of label refinement and fine-tuning leads to virtually flawless results.

Acknowledgments

This work is partially funded by the European Union's Horizon Europe research and innovation program under grant agreement No. 101070408 (SustainML).

References

[Avogaro *et al.*, 2025] Niccolo Avogaro, Thomas Frick, Mattia Rigotti, Andrea Bartezzaghi, Filip Janicki, Cristiano Malossi, Konrad Schindler, and Roy Assaf. Show or tell? effectively prompting vision-language models for semantic segmentation, 2025.

[Bianchi and Hebdon, 2021] Eric Bianchi and Matthew Hebdon. Corrosion Condition State Semantic Segmentation Dataset, 10 2021.

[Dettmers *et al.*, 2023] Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. Qlora: Efficient finetuning of quantized llms, 2023.

[et al., 2022] Rishi Bommasani et al. On the opportunities and risks of foundation models, 2022.

[Frick *et al.*, 2024] Thomas Frick, Cezary Skura, Filip Janicki, Roy Assaf, Niccolo Avogaro, Daniel Caraballo, Yagmur Cinar, Brown Ebouky, Ioana Giurgiu, Takayuki Katsuki, Piotr Kluska, A. Cristiano I. Malossi, Haoxiang Qiu, Tomoya Sakai, Florian Scheidegger, Andrej Simeski, Daniel Yang, Andrea Bartezzaghi, and Mattia Rigotti. Probabilistic feature matching for fast scalable visual prompting, 2024.

[Houlsby *et al.*, 2019] Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin de Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. Parameter-efficient transfer learning for nlp, 2019.

[Hu *et al.*, 2021] Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models, 2021.

[Jain *et al.*, 2014] Nilesh Jain, Ashok Bhansali, and Deepak Mehta. Angularjs: A modern mvc framework in javascript. *Journal of Global Research in Computer Science*, 5(12):17–23, 2014.

[Jha *et al.*, 2019] Debesh Jha, Pia H. Smedsrud, Michael A. Riegler, Pål Halvorsen, Thomas de Lange, Dag Johansen, and Håvard D. Johansen. Kvasir-seg: A segmented polyp dataset, 2019.

[Jha *et al.*, 2021] Debesh Jha, Sharib Ali, Krister Emanuelsen, Steven A. Hicks, Vajira Thambawita, Enrique Garcia-Ceja, Michael A. Riegler, Thomas de Lange, Peter T. Schmidt, Håvard D. Johansen, Dag Johansen, and Pål Halvorsen. Kvasir-instrument: Diagnostic and therapeutic tool segmentation dataset in gastrointestinal endoscopy. In Jakub Lokoč, Tomáš Skopal, Klaus Schoeffmann, Vasileios Mezaris, Xirong Li, Stefanos Vrochidis, and Ioannis Patras, editors, *MultiMedia Modeling*, pages 218–229, Cham, 2021. Springer International Publishing.

[Jia *et al.*, 2022] Menglin Jia, Luming Tang, Bor-Chun Chen, Claire Cardie, Serge Belongie, Bharath Hariharan, and Ser-Nam Lim. Visual prompt tuning, 2022.

[Ke *et al.*, 2023] Lei Ke, Mingqiao Ye, Martin Danelljan, Yifan Liu, Yu-Wing Tai, Chi-Keung Tang, and Fisher Yu. Segment anything in high quality, 2023.

[Kirillov *et al.*, 2023] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollár, and Ross Girshick. Segment anything, 2023.

[Liu *et al.*, 2019] Yahui Liu, Jian Yao, Xiaohu Lu, Renping Xie, and Li Li. Deepcrack: A deep hierarchical feature learning architecture for crack segmentation. *Neurocomput.*, 338(C):139–153, April 2019.

[Liu *et al.*, 2022] Haokun Liu, Derek Tam, Mohammed Muqeeth, Jay Mohta, Tenghao Huang, Mohit Bansal, and Colin Raffel. Few-shot parameter-efficient fine-tuning is better and cheaper than in-context learning, 2022.

[Liu *et al.*, 2024a] Weiyang Liu, Zeju Qiu, Yao Feng, Yuliang Xiu, Yuxuan Xue, Longhui Yu, Haiwen Feng, Zhen Liu, Juyeon Heo, Songyou Peng, Yandong Wen, Michael J. Black, Adrian Weller, and Bernhard Schölkopf. Parameter-efficient orthogonal finetuning via butterfly factorization, 2024.

[Liu *et al.*, 2024b] Yang Liu, Muzhi Zhu, Hengtao Li, Hao Chen, Xinlong Wang, and Chunhua Shen. Matcher: Segment anything with one shot using all-purpose feature matching, 2024.

[Oquab *et al.*, 2024] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Mahmoud Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Hervé Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. Dinov2: Learning robust visual features without supervision, 2024.

[Paszke *et al.*, 2017] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. 2017.

[Radford *et al.*, 2021] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision, 2021.

[Rigotti *et al.*, 2023] Mattia Rigotti, Diego Antognini, Roy Assaf, Kagan Bakirci, Thomas Frick, Ioana Giurgiu, Klára

Janoušková, Filip Janicki, Husam Jubran, Cristiano Malossi, Alexandru Meterez, and Florian Scheidegger. Towards workflows for the use of ai foundation models in visual inspection applications. *ce/papers*, 6(5):605–613, 2023.

[Seibold *et al.*, 2022] Constantin Seibold, Simon Reiß, Saquib Sarfraz, Matthias A. Fink, Victoria Mayer, Jan Sellner, Moon Sung Kim, Klaus H. Maier-Hein, Jens Kleesiek, and Rainer Stiefelhagen. Detailed annotations of chest x-rays via ct projection for report understanding, 2022.

[Zhong *et al.*, 2024] Zihan Zhong, Zhiqiang Tang, Tong He, Haoyang Fang, and Chun Yuan. Convolution meets lora: Parameter efficient finetuning for segment anything model, 2024.