From DINO over CroCo to CroCoM: a novel object matching pipeline for ZS6D

Stefan Lechner
stefan.lechner@tuwien.ac.at

***Abstract*—6D object pose detection is increasingly extending beyond controlled laboratory environments into real-world settings which have a vast array of unique objects. Currently, one can not include every worldly object in a training set. To address this issue, Zero-Shot-6D (ZS6D) was developed, which leverages features from a self-supervised Vision Transformer (ViT) to identify objects that were not included in the training set. In this paper, we modified the ViT to one trained on self-supervised Cross View Completion (CroCo), through which it learned a three-dimensional understanding. We then tested this modified version to evaluate its performance. Additionally, we propose a new pipeline called Cross Completion Match (CroCoM), which utilizes the original training task as an alternative approach. Our results show an improvement in accuracy over the original pipeline but with the caveat of a significant increase in computational complexity.**

August 13, 2024

I. Introduction

Accurate 6D object pose estimation in real-world scenarios is vital for advanced robot tasks. However, most detection pipelines rely on fine-tuned machine-learning models for specific objects. To circumvent this problem, Ausserlechner et al. [1] proposed ZS6D. It is an example of a zero-shot pose estimation method which does not rely on object-specific training. Others are CNOS [2] for zero-shot detection and FoundPose [3] for zero-shot 6D pose estimation. ZS6D is based on the research of Amir et al. [4], where deep features are extracted from a pre-trained Vision Transformer (ViT). They showcased zero-shot segmentation with the help of dense visual descriptors by using DINO-ViT [5] a self-supervised ViT model.

To determine the position of an object using ZS6D, a three-dimensional object model must be available beforehand. This model is used to generate uniformly distributed templates and corresponding descriptors. Since the environment is simulated, the object's position in relation to the image plane is already known. Therefore, if one can match the best pair to an extracted segmented object of an image scene, one can determine its pose by obtaining 2D-3D correspondences from the template's corresponding NOCS maps or UV coordinate map images. This is achieved by segmenting the input image, extracting the objects, calculating the descriptors, finding the best template descriptor, and using local correspondence estimation to estimate the pose of the object. Figure 1 visualizes the similarity between descriptors of a segmented object and its perfect template match. The segmentation task is not part of the ZS6D codebase. For this, third-party techniques are available such as Segment Everything [6]. Ausserlechner et al. demonstrated improvements over results obtained by task-specific fine-tuned CNNs with the ”Benchmark for 6d-object pose estimation” (BOP) [7], indicating a new and interesting research field.

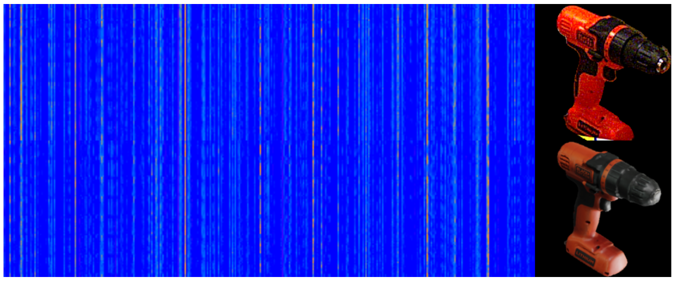

Fig. 1. Descriptor comparisons between a segmented object and its template counterpart

It is important to note that simply improving the entire pipeline does not necessarily lead to increased accuracy. The crucial factor lies in the ViT that extracts the features, as it can make or break the results. Therefore, we were concentrating on swapping out the DINO-ViT with a ViT that was trained using self-supervised Cross View Completion (CroCo) [8]. This approach has shown promising results in grasping three-dimensionality. We anticipate that the extracted features are useful for matching templates and the segmented object.

II. Methods

The features extracted from DINO originate from the Key, Query, and Value vectors within the ViT architecture. Amir et al. [4] demonstrated that the Key vector provides the most relevant representation of important information in an image. As a result, ZS6D utilizes the Key vector of the ViT. Layer eleven is used for template matching, while layer nine is used for extracting local correspondences for RANSAC [9]. It is also advantageous that these can be analyzed with cosine-similarity to match up two different descriptors.

As we see ZS6D builds on DINO to extract useful features. A way of improving upon this pipeline is to exchange the ViT. For this task, we opted to use CroCo as mentioned in Section I. It was trained with the goal of reconstructing a randomly masked scene image utilising another input image from a different angle. Thus this ViT has two inputs and one output. CroCo is available in several sizes, from CroCov1 being the smallest to CroCov2 being the most capable one.

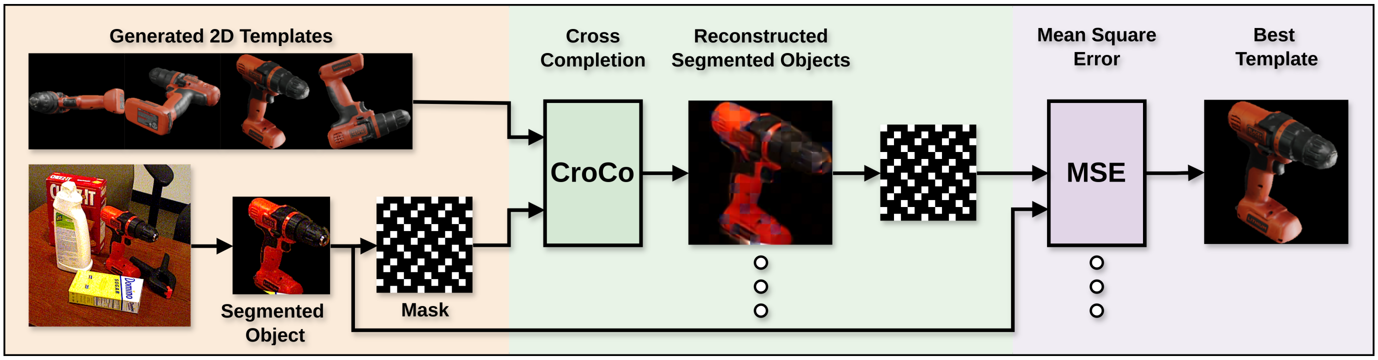

Fig. 2. CroCoM pipeline leveraging the original Cross View Completion task to match reconstructed segmented objects with the segmented object, utilizing mean squared error as the comparison metric

The implementation of CroCo in ZS6D was fairly straightforward, having used a version of this ViT tailored for use for monocular tasks. Additionally, this reduced CroCo allows for faster inference. During our testing of the descriptors from the best layer-token combination, we determined they were unsuitable for the given task. However, this discovery led us to propose a better-fitting zero-shot 6D pipeline, Cross Completion Match (CroCoM), tailored around the original training task of CroCo.

The pipeline of CroCoM is visualized in Figure 2. Same as ZS6D, it starts with a scene image from which a segmented object is needed and several pre-generated 2D images derived from a 3D representation of this object. The segmented object is then masked using a predefined mask (white squares mark the place where the mask is applied) and subsequently input into the CroCo model along with each template. The underlying principle is that the best match will produce the most accurate reconstruction. After processing all templates, we compare each masked reconstructed image to the segmented object using Mean Square Error (MSE) to identify the best match.

For testing purposes and to save on computing time the BOP YCB-V dataset was reduced to 12 scene images consisting of 55 objects. All three methods were analysed based on the Average Recall (AR) score. AR measures the mean proportion of true positive detections across various precision thresholds. For each detected object meeting the visibility criterion, a normalized error is calculated by dividing the translation error by the object's diameter. This normalized error is then used to determine individual AR scores.

III. Results

The layer analysis of CroCo showed that early layers perform best because they differentiate better between various descriptors. Specifically, the highest accuracy was achieved using the Key vector from the fifth layer. Regarding the size of the ViT: CroCov2 is simply a deeper network with more layers, offering no gain in accuracy. This is why we selected CroCov1 as the replacement ViT in the original pipeline. In Table I, one can see the performance metric of the original pipeline with DINO/CroCo and CroCoM/CroCoMv2. CroCo is just mentioned for completion. The metric showcases a rather dire performance. On the other hand, CroCoM excels compared to DINO. It can detect more objects, has a lower rotation error and a higher AR score. Only the translation error is higher, though the difference is smaller in percentage terms compared to the reduction of the rotation error. CroCoM, with the largest CroCov2, outperforms ZS6D in every metric.

Regarding the masking step, a relatively uniform mask, which is rotation invariant, performed best. Also, applying the mask to the reconstructed segmented object enhanced the accuracy because the reconstructed squares are often colour-shifted.

Metric	**DINO**	**CroCo**	**CroCoM**	**CroCoMv2**
Total Objects	55	55	55	55
Detected Objects	50	4	**51**	50
Detection Rate	0.91	0.07	**0.93**	0.91
Rotation Error (rad)	1.63	3.07	1.27	**1.20**
Translation Error (mm)	161.69	344.32	187.43	**148.66**
AR Score	0.1939	0	0.2485	**0.4030**

TABLE I
Comparison of DINO, CroCo, CroCoM, and CroCoMv2 across various metrics

IV. Discussion

The failure of CroCo in the original ZS6D pipeline can be attributed to the fact that CroCo never learned to reconstruct fine features during the training step. This limitation led ZS6D to consistently select overly simplified views lacking detailed features, a phenomenon clearly demonstrated when testing the original task using a black reference image against both low- and high-feature input object views. Amazingly, this task works quite well for the new pipeline because the fine features can be reconstructed when provided with a template that accurately matches the segmented object.

V. Conclusion

The implementation of CroCo as a replacement for the DINO-ViT in the ZS6D pipeline was successful. However, significant challenges emerged when evaluating the accuracy of descriptor matching, indicating that the learned three-dimensionality of CroCo is not suitable for this pipeline. In contrast, when CroCo is used in the new proposed pipeline CroCoM, first tests show that it outperforms ZS6D. Unfortunately, this is accompanied by a significant increase in processing complexity. This outcome paves the way for two promising research directions: conducting large-scale testing of CroCoM and reducing processing complexity through more efficient use of CroCo, specifically by optimizing which template/object combinations are actually worth reconstructing.

Acknowledgment

I am grateful to my supervisor, Philipp Ausserlechner, for his invaluable help and guidance in exploring a different ViT pipeline in his novel project ZS6D.

References

[1] Philipp Ausserlechner, David Haberger, Stefan Thalhammer, Jean-Baptiste Weibel, and Markus Vincze. Zs6d: Zero-shot 6d object pose estimation using vision transformers. *arXiv preprint arXiv:2309.11986*, 2023.

[2] Van Nguyen Nguyen, Thibault Groueix, Georgy Ponimatkin, Vincent Lepetit, and Tomas Hodan. Cnos: A strong baseline for cad-based novel object segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2134–2140, 2023.

[3] Evin Pınar Örnek, Yann Labbé, Bugra Tekin, Lingni Ma, Cem Keskin, Christian Forster, and Tomas Hodan. Foundpose: Unseen object pose estimation with foundation features. *arXiv preprint arXiv:2311.18809*, 2023.

[4] Shir Amir, Yossi Gandelsman, Shai Bagon, and Tali Dekel. Deep vit features as dense visual descriptors. *arXiv preprint arXiv:2112.05814*, 2(3):4, 2021.

[5] 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.

[6] 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. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4015–4026, 2023.

[7] Tomas Hodan, Frank Michel, Eric Brachmann, Wadim Kehl, Anders GlentBuch, Dirk Kraft, Bertram Drost, Joel Vidal, Stephan Ihrke, Xenophon Zabulis, Caner Sahin, Fabian Manhardt, Federico Tombari, Tae-Kyun Kim, Jiri Matas, and Carsten Rother. Bop: Benchmark for 6d object pose estimation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, September 2018.

[8] Weinzaepfel, Philippe and Leroy, Vincent and Lucas, Thomas and Brégier, Romain and Cabon, Yohann and Arora, Vaibhav and Antsfeld, Leonid and Chidlovskii, Boris and Csurka, Gabriela and Revaud Jérôme. CroCo: Self-Supervised Pre-training for 3D Vision Tasks by Cross-View Completion. In *NeurIPS*, 2022.

[9] Martin A Fischler and Robert C Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, 1981.