

---

# Visual Understanding of the Physical World

---



Guanqi Zhan  
Mansfield College  
University of Oxford

A thesis submitted for the degree of  
*Doctor of Philosophy*

Michaelmas 2025

# Abstract

We are living in a 3D physical world, and a first step towards artificial general intelligence is to enable machines to understand the physical world. This is the goal of the thesis, and it is structured around three themes: (1) understanding occlusion and occlusion handling, (2) understanding the 3D and physical properties of the scene, and (3) bridging visual understanding with language. *For all the themes, we build our method on top of large-scale pre-trained models or their representations.*

For occlusion understanding and handling, we design a tri-layer plugin for conventional pre-trained object detectors to improve the performance of object detection and instance segmentation under occlusion. As an additional contribution on occlusion, we advance the amodal completion model to recover the complete shape of occluded objects, by utilising the prior of pre-trained Stable Diffusion model.

For 3D physical understanding, we start with static 3D physical properties in images. To this end, we set up a protocol to probe large-scale pre-trained visual foundation models for the understanding of such properties. Additionally, we also study dynamic 3D physical properties in videos, and explore predicting these properties from different types of large-scale pre-trained video foundation models.

For visual-language understanding, we focus on improving visual-language foundation models. On the CLIP-like large-scale pre-trained models, we improve their performance for text-to-image retrieval by introducing a learnable prompt for the visual encoder conditioned on the text; On the ChatGPT-like large-scale pre-trained models, we improve their performance and efficiency for visual grounding by equipping a small model with multi-modal reasoning capability.

**Keywords** – scene understanding, occlusion handling, 3D physical properties, foundation models, visual-language models

This thesis is submitted to the Department of Engineering Science, The University of Oxford, in fulfilment of the requirements for the degree of Doctor of Philosophy. This thesis is entirely my own work, and except where otherwise stated, describes my own research.

Guanqi Zhan, December 2025.

# Acknowledgement

First and foremost to my supervisor Andrew Zisserman, thank you for your guidance, advice, encouragement and support along this journey. I feel extremely fortunate to have worked with you. You are my role model in so many aspects. Similarly to Weidi Xie, my wonderful co-supervisor, thank you for your suggestions and in-depth guidance. The support from you two makes this experience much more enjoyable.

To all the VGG members: thank you for making VGG such a ‘Very Good Group’, of which I feel privileged to be a member. In particular to Max Bain, Liliane Momeni, Sagar Vase, Vadim Tschernezki and Yan Xia for the great times we spent together. To Chuanxia Zheng, Kai Han and Xianzheng Ma for the great collaboration. To Honglie Chen, Tengda Han, Shangzhe Wu, Chuhan Zhang, Yash Bhargat, Ragav Sachdeva, Junyu Xie, Jianyuan Wang and Minghao Chen for the insightful discussions. Especially to Alyosha Efros, who inspires me to hike from Abingdon to Oxford, thank you for the great hiking together and the insightful discussion about psychology, computer vision and the world. I also thank Jaesung Huh, Charig Yang, Eldar Insafutdinov, Luke Melas-Kyriazi, Vladimir Iashin and other VGG members for their support and camaraderie. Importantly, I thank Ashish Thandavan for answering my endless server questions, and Andrea Vedaldi, Christian Rupprecht, João Henriques, Iro Laina for their insights on my PhD projects.

I also thank professors and friends outside of VGG: Thank professors Bill Freeman, David Forsyth, Jitendra Malik, Derek Hoiem, Anand Bhattad, Liang Zheng and Mike Shou, for the encouragement, discussions and insights on my PhD projects. I am deeply grateful to my NVIDIA supervisor, Prof. Song Han, and the entire NVIDIA Efficient AI team for their invaluable guidance and support during my

internship. Thank Jonathan Young, Mengmeng Li and Jenny Smallwood from Mansfield, thank you so much for your continuous companionship, encouragement, and raising me up every time I am down. Thank Jindong Gu from Oxford, Jialei Huang from Tsinghua, Tianyuan Zhang from MIT, Baifeng Shi from Berkeley, Yang You from Stanford, Xiaohang Zhan from Adobe, for the insightful discussions on life, computer vision and AI.

Finally, I thank my parents for their understanding and support.

# Contents

<b>1</b>	<b>Introduction</b>	<b>10</b>
1.1	Motivation and Overview . . . . .	10
1.2	Background and Related Work . . . . .	11
1.3	Thesis Outline and Contributions . . . . .	13
1.4	Publications . . . . .	15
<b>I</b>	<b>Occlusion Handling</b>	<b>16</b>
<b>2</b>	<b>A Tri-Layer Plugin to Improve Occluded Detection</b>	<b>17</b>
2.1	Introduction . . . . .	19
2.2	Related Work . . . . .	20
2.3	Detector Architecture and Application . . . . .	22
2.3.1	Architecture . . . . .	22
2.4	Data Preparation . . . . .	25
2.4.1	Amodal Completion . . . . .	26
2.4.2	Occlusion Reasoning . . . . .	26
2.4.3	Occluded COCO & Separated COCO for Evaluation . . . . .	27
2.5	Experiments . . . . .	28
2.5.1	Datasets and Implementation Details . . . . .	28

2.5.2	Ablation Study . . . . .	29
2.5.3	Comparison with State-of-the-Art . . . . .	30
2.5.4	Qualitative Results . . . . .	31
2.6	Conclusion and Future Work . . . . .	32
<b>3</b>	<b>Amodal Ground Truth and Completion in the Wild</b>	<b>34</b>
3.1	Introduction . . . . .	36
3.2	Related Work . . . . .	39
3.3	The MP3D-Amodal Dataset . . . . .	40
3.3.1	An Overview of the Dataset . . . . .	40
3.3.2	Generating Amodal Ground Truth from 3D . . . . .	41
3.4	Architectures for Amodal Prediction . . . . .	44
3.4.1	Two-Stage Architecture – OccAmodal . . . . .	44
3.4.2	One-stage Architecture – SDAmodal . . . . .	45
3.4.3	Training . . . . .	46
3.5	Experiments . . . . .	47
3.5.1	Experimental Details . . . . .	47
3.5.2	Ablation Study of Different Architectures . . . . .	47
3.5.3	Comparison with State-of-the-Art . . . . .	49
3.5.4	Effectiveness of Different Training Data . . . . .	50
3.5.5	Qualitative Results . . . . .	50
3.6	Conclusion and Extensions . . . . .	51
<b>II</b>	<b>3D Physical Understanding</b>	<b>53</b>
<b>4</b>	<b>A General Protocol to Probe Large Vision Models for 3D Physical Understanding</b>	<b>54</b>
4.1	Introduction . . . . .	56

4.2	Related Work . . . . .	58
4.2.1	Exploration of Pre-trained Models . . . . .	58
4.2.2	Exploitation of Pre-trained Models . . . . .	59
4.2.3	3D Physical Scene Understanding . . . . .	60
4.3	Method – Properties, Datasets, and Classifiers . . . . .	60
4.3.1	Properties and Questions . . . . .	60
4.3.2	Datasets . . . . .	61
4.3.3	Property Probing . . . . .	65
4.4	Experiments . . . . .	66
4.4.1	Implementation Details and Evaluation Metric . . . . .	66
4.4.2	Results for Stable Diffusion . . . . .	67
4.4.3	Results for CLIP/DINO/VQGAN Features . . . . .	68
4.4.4	Comparison of Different Features Trained at Scale . . . . .	69
4.5	Discussion and Future Work . . . . .	70
<b>5</b>	<b>Inferring Dynamic Physical Properties from Video Foundation Models</b>	<b>73</b>
5.1	Introduction . . . . .	75
5.2	Related Work . . . . .	76
5.3	Problem Scenario and The <i>PhysVid</i> Datasets . . . . .	77
5.3.1	The <i>PhysVid</i> Datasets . . . . .	78
5.4	Inferring Physical Properties . . . . .	81
5.4.1	Oracle Estimation . . . . .	81
5.4.2	Video Generative and Self-Supervised Models . . . . .	83
5.4.3	Multimodal Large Language Models . . . . .	85
5.5	Experiments . . . . .	87
5.5.1	Results for Relative Value Comparison . . . . .	87
5.5.2	Results for Absolute Value Prediction . . . . .	88

5.5.3	Qualitative Results . . . . .	89
5.6	Conclusion . . . . .	90
<b>III Improving Visual-Language Foundation Models</b>		<b>93</b>
<b>6</b>	<b>ELIP: Enhanced Visual-Language Foundation Models for Image Retrieval</b>	<b>94</b>
6.1	Introduction . . . . .	96
6.2	Related Work . . . . .	98
6.3	Preliminaries . . . . .	100
6.4	The ELIP Architecture . . . . .	101
6.4.1	Text-Guided MLP Mapping Network . . . . .	101
6.4.2	Training and Testing ELIP-C/ELIP-S . . . . .	102
6.4.3	Training and Testing ELIP-B . . . . .	103
6.5	Evaluation Datasets . . . . .	104
6.5.1	Standard Benchmarks . . . . .	104
6.5.2	Out-of-Distribution Benchmarks . . . . .	104
6.6	Experiment . . . . .	105
6.6.1	Results on COCO and Flickr Benchmarks . . . . .	106
6.6.2	Results on OOD Benchmarks . . . . .	108
6.7	Conclusion . . . . .	111
<b>7</b>	<b>EGM: Efficient Visual Grounding Language Models</b>	<b>113</b>
7.1	Introduction . . . . .	115
7.2	Related Work . . . . .	117
7.3	Why Small VLMs Fall Behind? . . . . .	119
7.4	Method . . . . .	121
7.4.1	SFT Data Curation . . . . .	121

7.4.2	RL Data Curation . . . . .	123
7.4.3	Training Pipeline and Reward . . . . .	124
7.5	Experiments . . . . .	125
7.5.1	Implementation Details . . . . .	125
7.5.2	Evaluation Benchmarks and Metrics . . . . .	126
7.5.3	Vanilla Grounding . . . . .	126
7.5.4	Amodal Grounding . . . . .	128
7.5.5	Efficiency Comparison . . . . .	129
7.5.6	Qualitative Comparison . . . . .	130
7.6	Conclusion . . . . .	132
<b>8</b>	<b>Summary</b>	<b>134</b>
8.1	Achievement and Impact . . . . .	134
8.1.1	Part I: Occlusion Handling . . . . .	134
8.1.2	Part II: 3D Physical Understanding . . . . .	136
8.1.3	Part III: Improving Visual-Language Foundation Models . . . . .	137
8.2	Future Work . . . . .	139
	<b>References</b>	<b>142</b>
<b>A</b>	<b>Statement of Authorship</b>	<b>175</b>

# Chapter 1

## Introduction

### 1.1 Motivation and Overview

We are living in a 3D physical world, and as humans, we can understand the *3D physical properties* of the world *in the wild* and *connect it with natural language*. However, the major way of computer vision system to perceive the physical world is via the images and videos captured by the camera, which is a 2D projection of the 3D physical world. Conventional computer vision research focuses on the pixel-level understanding of the images and videos, so there is a gap between the machine vision system and human vision perception.

Mitigating such a gap can benefit the following aspects: First, an in-the-wild understanding makes the machine vision system more robust to out-of-distribution scenarios, *e.g.*, under occlusion or applied to a different domain. Second, understanding the 3D physical properties can help downstream application scenarios such as autonomous driving and robotics, as such a human-like perception is a prerequisite for interaction with the 3D physical world. Third, bridging visual understanding with language enables AI to better serve humans, for example, visual-language models like CLIP [Radford et al. 2021] and ChatGPT [Hurst et al. 2024] can satisfy human requirements for different aspects, including multi-modal information retrieval and language-instructed visual grounding.

In this thesis, therefore, we make efforts in the above directions to improve the visual understanding of the physical world of the computer vision systems. *A common methodology we use for all the chapters is that we develop our method on*

*top of large-scale pre-trained models or their representations.*

## 1.2 Background and Related Work

This section introduces the background and related work of the thesis in three themes: occlusion, physical understanding, and visual-language models, which serves as a background of the thesis.

### Part I: Occlusion

Occlusion naturally occurs during the projection of the 3D physical world into 2D images. It can be both a benefit and a problem. On the one hand, occlusion provides a beneficial cue that gives information about ordering and depth. This idea appeared in early literature of psychophysics and computer vision, where categorising an edge into an internal edge or an occlusion edge is used to determine the occlusion ordering and depth ordering in the ‘Blocks World’ [Huffman 1971; Clowes 1971]. On the other hand, occlusion leads to information being missing – a problem that influences many fundamental computer vision scene understanding tasks, such as object detection and instance segmentation, where the performance of detectors and segmentors degrade when the instance to detect or segment is occluded [Wang et al. 2020a; Yuan et al. 2021; Saleh et al. 2021]. For the human visual system, psychology studies of occlusion [Lee and Vecera 2005; Chen et al. 2016] show humans can imagine the completed shape of occluded objects. The terminology for such a capability in visual psychophysics is ‘Amodal Completion’ [Wikipedia contributors 2025]. Therefore, to solve the occlusion problem, the concept of ‘amodal’ is introduced in the computer vision community. In computer vision, the task of ‘Deocclusion’ or ‘Amodal Completion’ [Li and Malik 2016; Zhan et al. 2020] is to recover the occluded parts of objects in the 3D scene, which can benefit other computer vision tasks influenced by occlusion.

### Part II: Physical Understanding

Beyond recognising what the object is and where the object is, physical understanding means knowing about the underlying physical properties of the objects and physical laws of the world, such as understanding the mass of the objects,

the forces that are on each object, and the material that an object is made of. As humans, we can understand the physical properties of the world, including both static physical properties such as support relations [Baillargeon et al. 1992], material [Adelson 2001; Fleming 2014], and dynamic physical properties, such as viscosity of liquid [Kawabe et al. 2014], elasticity of a bouncing ball [Warren Jr et al. 1987], and friction between an object and a surface [Wu et al. 2015]. A hypothesis was made for human perception and understanding of physical properties [Battaglia et al. 2013; Wu et al. 2015] – humans understand the physical properties by mental simulation, *i.e.*, assuming the values of the physical properties and then comparing the simulated process with the actual process. Therefore, inspired by this, Galileo [Wu et al. 2015] introduced an approach to estimate the physical properties such as mass and friction via a physics engine. Later work [Zhang et al. 2024] inherited the idea of ‘analysis by synthesis’ and used pre-trained video generative models as the ‘mental simulator’. Beyond the hypothesis of understanding by simulating, in Galileo [Wu et al. 2015] another recognition network is further trained to map the visual observation to physical properties directly, without the generative process. The process of training the recognition network corresponds to the sleep cycle of humans, as stated in the paper [Wu et al. 2015]. So another direction in computer vision would be to explore obtaining the physical properties directly from large-scale pre-trained models, which have seen a large number of images and videos during their pre-training stage (‘wake cycle for humans’).

### **Part III: Visual-Language Models**

Language is an important medium for humans to understand and describe the visual world [Tanenhaus et al. 1995; Lupyan 2012]. There were (single modality) vision models and language models for years, before the two modalities were brought together by DeVISE [Frome et al. 2013], where a joint visual-language embedding space is first introduced. This marks the start of aligning visual representations with language representations directly. Later, Show and tell [Vinyals et al. 2015] feeds the visual features into a language model for image caption generation, which is a prototype of the ChatGPT-like visual-language models nowadays. More recently, CLIP [Radford et al. 2021] scaled up the joint embedding approach of DeVISE using contrastive learning, and serves as the basis of modern approaches

of aligning visual representation with language representation. On the other hand, VirTex [Desai and Johnson 2021] demonstrated that generative pre-training, *i.e.*, generating image captions during training, is an effective way to learn high-quality visual representations. Nowadays, both threads of models have been continuously advanced, and have enabled numerous applications of bridging visual understanding with language. The representative works for CLIP-like visual-language foundation models include SigLIP [Zhai et al. 2023] and SigLIP-2 [Tschannen et al. 2025]; the representative works for ChatGPT-like visual-language foundation models include ChatGPT [Hurst et al. 2024], Gemini [Comanici et al. 2025], Claude [Anthropic 2025], QwenVL [Alibaba 2025] and InternVL [Wang et al. 2025b].

### 1.3 Thesis Outline and Contributions

This section provides an outline of the following thesis chapters and summarises the key contributions. The thesis is grouped into three themes building on top of the large-scale pre-trained models or their representations: occlusion handling, 3D physical understanding, and improving visual-language foundation models.

#### Part I: Occlusion Handling

The challenges we aim to address here are how to make the pre-trained models more robust under occlusion, and how to use the pre-trained models to help better understand occlusion.

In this theme, Chapter 2 aims to improve the performance of conventional *pre-trained* object detectors under occlusion. We propose a simple ‘plugin’ module for the detection head of two-stage object detectors to improve the recall of partially occluded objects. The module predicts a tri-layer of segmentation masks for the target object, the occluder and the occludee, and we show that by doing so the model is able to better predict the mask of the target object. In the process of generating the training data for the plugin, we use off-the-shelf ‘amodal completion’ model to help identify the occluder and occludee of the target object. Further, in Chapter 3, we focus on advancing the amodal completion model, and show properly utilising the prior from *pre-trained* Stable Diffusion model can help us develop a better amodal completion model that can generalise better for new scenario and

new domain.

## Part II: 3D Physical Understanding

The challenges we aim to address here are how to get the prior knowledge about or how to predict the 3D physical properties from large-scale pre-trained models or their representations.

In this theme, Chapter 4 focuses on the static 3D physical properties in images and Chapter 5 focuses on the dynamic physical properties in videos. More specifically, Chapter 4 studies scene geometry, material, support relations, shadow, occlusion and depth, and provides insights of how to get the prior information of these properties from *pre-trained* visual foundation models, such as DINO and Stable Diffusion; Chapter 5 studies elasticity, viscosity and friction, and provides insights of how to predict these properties from *pre-trained* video foundation models, such as DynamiCrafter, V-JEPA and Gemini.

## Part III: Improving Visual-Language Foundation Models

The challenges we aim to address here are how to improve large-scale pre-trained visual-language models for different multi-modal tasks.

More specifically, Chapter 6 studies CLIP-like large-scale *pre-trained* visual-language models and focuses on the task of text-to-image retrieval. To this end, we enhance the pre-training of visual-language foundation models, such as CLIP and SigLIP, to better discriminate between hard negatives and the ground truth when guided by the query text prompt. In addition, the data curation and novel architecture contribute to better zero-shot generalisation to occlusion scenario and other unusual domains, as well as efficient training with academic resources. Chapter 7, on the other hand, studies ChatGPT-like large-scale *pre-trained* visual-language models and focuses on the task of visual grounding. To this end, we boost the visual grounding capability of small visual-language models with multi-modal reasoning to compete with large visual-language models, therefore achieving more accurate and efficient visual grounding at inference time.

## 1.4 Publications

Chapter 2 to 7 each contains a research paper. Most of them have been peer-reviewed and accepted for publication at a conference or workshop, with the exception that Chapter 7 is on arXiv. For each paper, we also provide a statement of authorship in Appendix A. The papers included in the thesis are listed below.

Chapter 2: *A Tri-Layer Plugin to Improve Occluded Detection* is published at *British Machine Vision Conference (BMVC), 2022*. **Guanqi Zhan**, Weidi Xie, Andrew Zisserman.

Chapter 3: *Amodal Ground Truth and Completion in the Wild* is published at *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2024*. **Guanqi Zhan**, Chuanxia Zheng, Weidi Xie, Andrew Zisserman.

Chapter 4: *A General Protocol to Probe Large Vision Models for 3D Physical Understanding* is published at *Advances in Neural Information Processing Systems (NeurIPS), 2024*. **Guanqi Zhan**, Chuanxia Zheng, Weidi Xie, Andrew Zisserman.

Chapter 5: *Inferring Dynamic Physical Properties from Video Foundation Models* is accepted at the VGBE Workshop of *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2026*. **Guanqi Zhan**, Xianzheng Ma, Weidi Xie, Andrew Zisserman.

Chapter 6: *ELIP: Enhanced Visual-Language Foundation Models for Image Retrieval* is published at *IEEE International Conference on Content-Based Multimedia Indexing (CBMI), 2025* as a Best Paper Nominee. **Guanqi Zhan**, Yuanpei Liu, Kai Han, Weidi Xie, Andrew Zisserman.

Chapter 7: *EGM: Efficient Visual Grounding Language Models* is on *arXiv, 2026*. **Guanqi Zhan**, Changye Li, Zhijian Liu, Yao Lu, Yi Wu, Song Han, Ligeng Zhu.

# Part I

## Occlusion Handling

## Chapter 2

# A Tri-Layer Plugin to Improve Occluded Detection

The paper was published at the British Machine Vision Conference (BMVC), 2022.

# A Tri-Layer Plugin to Improve Occluded Detection

Guanqi Zhan<sup>1</sup>, Weidi Xie<sup>1,2</sup>, Andrew Zisserman<sup>1</sup>

<sup>1</sup>VGG, University of Oxford,

<sup>2</sup>CMIC, Shanghai Jiao Tong University

## Abstract

Detecting occluded objects still remains a challenge for state-of-the-art object detectors. The objective of this work is to improve the detection for such objects, and thereby improve the overall performance of a modern object detector.

To this end we make the following four contributions: (1) We propose a simple ‘plugin’ module for the detection head of two-stage object detectors to improve the recall of partially occluded objects. The module predicts a tri-layer of segmentation masks for the target object, the occluder and the occludee, and by doing so is able to better predict the mask of the target object. (2) We propose a scalable pipeline for generating training data for the module by using amodal completion of existing object detection and instance segmentation training datasets to establish occlusion relationships. (3) We also establish a COCO evaluation dataset to measure the recall performance of partially occluded and separated objects. (4) We show that the plugin module inserted into a two-stage detector can boost the performance significantly, by only fine-tuning the detection head, and with additional improvements if the entire architecture is fine-tuned. COCO results are reported for Mask R-CNN with Swin-T or Swin-S backbones, and Cascade Mask R-CNN with a Swin-B backbone.



**Figure 2.1. Improving occluded object detection and instance segmentation.** Left: Occlusion is very common in the 3D world, where one object is in front of another and a portion of the scene disappears behind the non-transparent object that is closer to the viewer. Right: For this example from COCO val, a Swin-T + Mask R-CNN Baseline incorrectly detects and segments the target object (the middle of the three cases). However, if the detection head is replaced with our tri-layer plugin, a correct segmentation mask is obtained. The plugin is tasked with predicting the masks of the target object occluder and occludee, and this leads to a better modelling for occluded detection.

## 2.1 Introduction

Occlusion frequently occurs in images of real scenes and is both a benefit and a problem. It is a benefit, because it reveals depth orderings and so contributes to a 3D perception of the scene [Gibson 1979]. However, it is a problem because despite the continual increase in performance of object detectors over the last decade, detection of occluded objects is still a significant deficiency [Wang et al. 2020a; Yuan et al. 2021; Ke et al. 2021].

In this paper, our objective is to improve the detection for objects under occlusion, and thereby improve the overall performance of the object detector. Specifically, we develop a lightweight ‘plugin’ module, that can be inserted into the detection head of any two-stage object detector, *e.g.* Mask R-CNN [He et al. 2017], to improve the recall of occluded objects. The module simultaneously infers *three* segmentation ‘layers’: the mask for the target object; the mask of the occluder (the object in front that occludes the target); and the mask of the occludee (the object behind that is occluded by the target) within the same detection box. This *tri-layer prediction head* forces the detection model to explicitly understand the existence of occlusion relationships, and thereby is better able to predict the target object mask. Additionally, we show that the process can be iterated, using the better prediction of the target mask to adapt the detection box for the next tri-layer prediction.

One challenge for training our proposed plugin module lies in the lack of suitable training data – almost all large-scale detection datasets provide annotations on

the visible part of objects, but no occlusion information is available. To this end, we propose a scalable pipeline for automatically discovering occluded objects (and their occluders), by running an amodal completion model on publicly available detection datasets, *e.g.*, COCO. Additionally, to verify the occlusion relationships between objects, *i.e.*, occluder or occludee, we adopt an off-the-shelf monocular depth estimator [Ranftl et al. 2020] and determine the occlusion relationship based on their relative depth. With this pipeline, we acquire a reliable large-scale data source for training our tri-layer model. We also establish an evaluation dataset to measure the recall performance for occluded objects, distinguishing between the two cases where the target object is *partially occluded* but the segmentation mask is connected, or where the target object segmentation mask is *separated* into distinct regions by the occluder.

To summarise, in this paper, we make the following four contributions: (i) We propose a simple ‘plugin’ module that can be inserted into two-stage object detectors, to improve their performance on occluded objects. (ii) We establish a scalable pipeline to determine occlusion relationships between objects, which is used to train the plugin module on publicly available detection datasets. (iii) We set up an evaluation benchmark with real images to measure the recall performance of partially occluded and separated objects. (iv) We show that the plugin module inserted into two-stage detectors can boost the performance significantly, by only fine-tuning the detection head, and with additional improvements if the entire architecture is fine-tuned. We show detection results for Mask R-CNN with Swin-T, Swin-S backbones, and Cascade Mask R-CNN with a Swin-B backbone.

## 2.2 Related Work

**Object Detection & Instance Segmentation.** Methods for object detection and instance segmentation have made great progress by training deep neural networks on large-scale datasets [Everingham et al. 2010; Lin et al. 2014; Kuznetsova et al. 2020]. Two-stage detectors like Faster R-CNN [Ren et al. 2015] and Mask R-CNN [He et al. 2017] first train region proposal networks to propose candidate objects. These candidates are then further refined in their location with a regression head, and categorised by a classifier head. With the development of stronger

transformer-based [Liu et al. 2021; Liu et al. 2022] architectures, replacing the original CNN backbones, there have been further improvements in the detection performance [Liu et al. 2021]. There has also been a similar development in single-stage detectors such as [Carion et al. 2020; Zhu et al. 2021; Cheng et al. 2022] but we do not consider those in this work. However, both single and two-stage detectors are still suffering when detecting occluded objects [Saleh et al. 2021].

**Amodal Segmentation & Occlusion-Related Datasets.** Amodal segmentation refers to the task of segmenting the object as whole, including the portions that are partially occluded [Li and Malik 2016; Zhan et al. 2020; Sun et al. 2022]. In the recent literature, various datasets have been collected to study occlusion [Zhu et al. 2017; Qi et al. 2019; Chen et al. 2020b; Cai et al. 2020; Wen et al. 2020]. However, most of these datasets either focus on certain domains/tasks or do not provide a large number of images for training and evaluating models in terms of occlusion for a large variety of categories.

**Layered Scene Understanding.** Layered representation was originally proposed by Wang and Adelson [Wang and Adelson 1994] to represent a video as a composition of layers with simpler motions. Since then, layered representations have been widely adopted in computer vision, *e.g.*, to decompose videos into layers [Brostow and Essa 1999; Jovic and Frey 2001], to improve scene segmentation by explicitly modelling occlusions, temporal consistency, depth ordering [Winn and Shotton 2006; Kumar et al. 2008; Yang et al. 2010; Yang et al. 2011], to estimate optical flow [Sun et al. 2012; Sun et al. 2013; Wulff and Black 2014; Wulff and Black 2015], and for novel view synthesis [Zitnick et al. 2004].

**Occlusion Handling.** To improve object detection and instance segmentation under occlusion scenarios, novel architectures, *e.g.*, compositional networks [Wang et al. 2020a; Yuan et al. 2021], have been proposed. Rather than develop a new architecture, ASN [Qi et al. 2019] and ORCNN [Follmann et al. 2019] ask modern detectors to output both modal and amodal masks of target objects. Additionally, BCNet [Ke et al. 2021] exploits two-layer graph neural networks to modern detectors for better instance segmentation under occlusion, inferring both the target object and the surrounding objects, while in this paper, we go one step further, and propose a simple, automatic pipeline for estimating objects' occlusion order, which

enables training of the tri-layer plugin with explicit supervision on the occlusion ordering for objects.

## 2.3 Detector Architecture and Application

In this section we describe our lightweight plugin module, and its application within an object detector. The module is designed to improve the detection performance on occluded objects. We start by introducing the standard two-stage detector, and then describe the tri-layer plugin architecture and functionality.

**Two-stage detector.** Given an image detection dataset,  $\mathcal{D} = \{(I_1, y_1), \dots, (I_n, y_n)\}$ , a standard two-stage Mask R-CNN detector can be parametrized as:

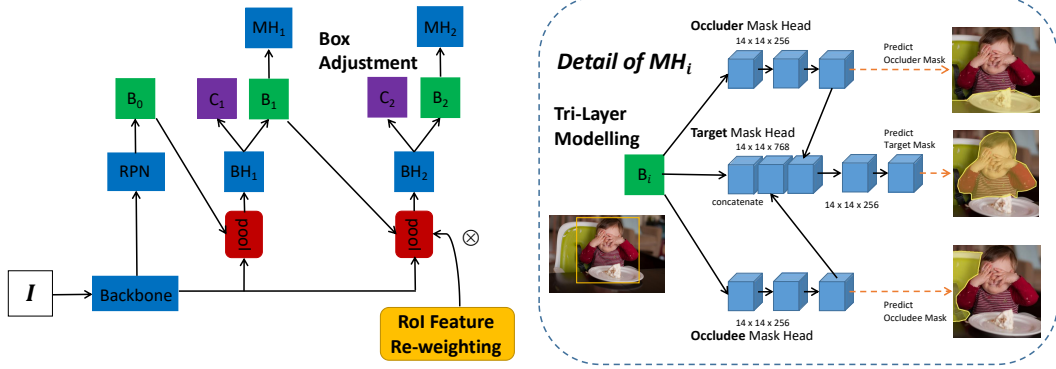
$$y_j = \{(b_j, c_j, m_j)\}^K = \Phi_{\{\text{CLS+BOX;SEG}\}} \circ \Phi_{\text{ALIGN}} \circ \Phi_{\text{RPN}} \circ \Phi_{\text{ENC}}(I_j) \quad (2.1)$$

where an input image ( $I_j \in \mathbb{R}^{H \times W \times 3}$ ) with a total of  $K$  objects is sequentially processed by a set of operations: an image encoder,  $\Phi_{\text{ENC}}(\cdot)$ ; a region proposal network,  $\Phi_{\text{RPN}}(\cdot)$ ; a region of interest feature alignment,  $\Phi_{\text{ALIGN}}(\cdot)$ ; after predicting the class and box offset ( $\Phi_{\text{CLS+BOX}}(\cdot)$ ), a binary mask for each RoI is also predicted,  $\Phi_{\text{SEG}}(\cdot)$ . As a result,  $y_j = \{(b_j, c_j, m_j)\}^K$  denotes the box coordinates ( $b_j^k \in \mathbb{R}^4$ ), category label ( $c_j^k \in \mathbb{R}^{\mathcal{C}}$ ), and **modal** segmentation mask ( $m_j^k \in [0, 1]^{H \times W}$ ) of the object, which has been converted from the spatial resolution of RoI align to the original image.

In the following, we refer to a pair of objects that have an occlusion relationship as **occluder** (the object that is in front of and thus occludes the other one), and **occludee** (the object at the back and being occluded), as illustrated in Figure 2.1. Note that, the role of each object is often relative, thus it can be both occluder and occludee at the same time, depending on its paired objects. In Section 2.4, we detail the procedure for acquiring the occlusion information, including their estimated **amodal** segmentation masks and the occlusion orderings.

### 2.3.1 Architecture

In this section, we introduce the architecture details of our tri-layer plugin for better detecting objects under occlusion. Specifically, we first augment the detection



**Figure 2.2. Architecture and function of the plugin module.** There are three functions: (a) The tri-layer mask head ( $MH_i$  shown in detail on the right) predicts the mask of the target object (the infant wiping food on their face), the occluder (the dining table), and the occludee (the chair) within the detection box  $B_i$ . The feature embeddings of the occluder/occludee branch are concatenated to the target mask embedding as cues to help better predict the target object mask; (b) As shown on the left, the process of predicting the target mask is iterated (index  $i$ ), such that the second iteration is able to adjust the initial box predictions and better detect partially occluded / separated objects; (c) After the first iteration, RoI features are pooled according to the predicted target mask to guide the model to focus more on the partially occluded / separated object itself. The notation used is: “ $I$ ” for the input image, and “BH”, “MH”, “B”, “C” refer to bbox head, mask head, bounding box, and classification respectively.

module to simultaneously output both occluder and occludee, along with the target object itself; we then introduce the idea of box adjustment, that facilitates the model to reason about the full object coverage with only partial observation, due to partial occlusion or separation; and lastly we improve the RoI pooling procedure by weighting the feature map with the inferred object mask, effectively preventing the model from concentrating on the occluder/occludee. Figure 2.2 illustrates the architecture of the plugin.

### Tri-Layer Modelling

Here, we augment the instance segmentation head with a tri-layer module that accepts the RoI aligned feature map as input, and outputs the modal segmentation masks of the occluder and occludee for the target object respectively (denoted as **Occluder Mask Head** and **Occludee Mask Head**). Specifically, we pass the RoI aligned feature maps into three different mask prediction heads:

$$\{\hat{b}_j, \hat{c}_j, \hat{m}_j, \hat{m}_{j1}, \hat{m}_{j2}\}^K = \Phi_{\{\text{CLS+BOX;SEG}\}}(\mathcal{F}_j) \quad (2.2)$$

where  $\mathcal{F}_j$  refers to the feature map from RoI align, and  $\hat{m}_{j1}, \hat{m}_{j2}$  refer to the inferred modal segmentation masks for the object’s occluder and occludee respectively. Note that, an object may not have occluder or occludee, and the predicted mask in this case is all zero.

As shown in Figure 2.2 (right), the feature embeddings from occluder/occludee branches are further integrated into the target object segmentation branch ( $\hat{m}_j$ ), providing cues to better infer the modal mask of the target object. To act as a proper layering model, we distinguish the order of occluder and occludee by concatenating the feature embeddings in order. Note that, this is in contrast to the previous approach [Ke et al. 2021], where the embeddings are simply element-wise added, leading to ambiguous occlusion ordering from the commutative rule.

### Box Adjustment

After predicting the box and segmentation, we carry out a second iteration, to refine the initial predictions. Intuitively, if the network can detect part of an occluded object in the first iteration, an extra iteration will provide the opportunity to adjust the box accordingly, to include any part of the occluded object missed in the first iteration. For the  $k$ th instance in image  $j$ ,

$$\{\dot{b}_j, \dot{c}_j, \dot{m}_j, \dot{m}_{j1}, \dot{m}_{j2}\}^k = \Phi_{\{\text{CLS+BOX;SEG}\}}(\Phi_{\text{ALIGN}}(\mathcal{V}_j, \hat{b}_j^k)), \forall k \in [1, K] \quad (2.3)$$

where  $\mathcal{V}_j$  denotes the feature map from the visual encoder. The inferred object box ( $\hat{b}_j^k$ ) from the first iteration is used for RoI align to generate refined boxes ( $\dot{b}_j^k$ ), and then we can do tri-layer modelling in the refined boxes.

### RoI Feature Re-weighting

To guide the model to focus more on the target objects, rather than the occluder-s/occludees that may take up a large proportion of the box, we also introduce RoI feature re-weighting with the inferred object mask from the previous iteration:

$$\{\dot{b}_j, \dot{c}_j, \dot{m}_j, \dot{m}_{j1}, \dot{m}_{j2}\}^k = \Phi_{\{\text{CLS+BOX;SEG}\}}(\Phi_{\text{ALIGN}}(\mathcal{V}_j, \hat{b}_j^k) \otimes \hat{m}_j^k), \forall k \in [1, K] \quad (2.4)$$

where  $\hat{m}_j^k$  denotes the inferred segmentation mask from the previous iteration,  $\otimes$  denotes the element-wise product between the RoI aligned feature map and the object segmentation from the previous iteration, and  $\hat{b}_j, \hat{c}_j, \hat{m}_j$  refer to the output bounding box, object category and instance mask, respectively.

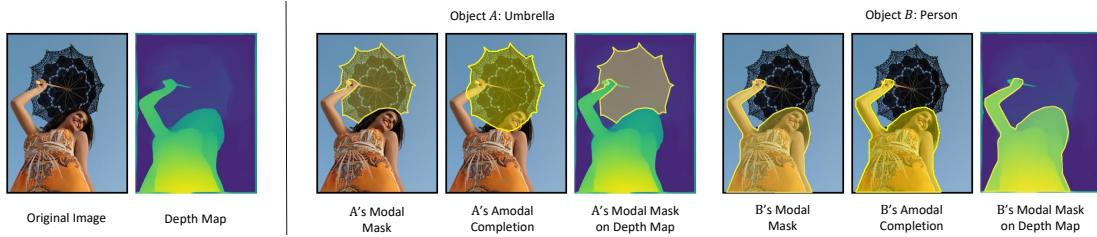
## Training

We start from publicly released pre-trained models. The fine-tuning procedure is conducted progressively from “Tri-Layer Modelling” to “BBox Adjustment” and then “RoI Feature Re-weighting”, *i.e.*, proposed modules are gradually added after the previous one converges. For tri-layer modelling, both occluder and occludee mask heads use the same structures as the main mask head, except that they are class-agnostic while the main mask head is not. They are trained with binary cross-entropy loss using the inferred ground truth masks, as will be detailed in Section 2.4.2. The implementation and training are based on MMDet [Chen et al. 2019b]. Please refer to the appendix for more training details.

## 2.4 Data Preparation

To properly train our plugin, ground truth occluder/occludee masks for each object are required, as discussed in Section 2.3.1. In this section, we describe the procedure for determining objects’ occlusion order based on their **amodal** segmentation and depth ordering.

In general, acquiring amodal segmentations can be very costly, due to the requirement of annotating the occluded parts of the object. To our knowledge, none of the existing large-scale datasets, *e.g.* COCO [Lin et al. 2014], LVIS [Gupta et al. 2019], provide amodal segmentation masks. Here, we start by describing a simple, automatic, thus scalable pipeline to approximate object amodal segmentation masks (Section 2.4.1); in Section 2.4.2, we further exploit these amodal segmentations to infer the occlusion relationships between paired objects; and in Section 2.4.3, we detail two different types of occlusion, namely, separation and partial occlusion, and define an evaluation benchmark for occlusion, based on the COCO2017 val set.



**Figure 2.3. The process of occlusion reasoning via amodal completion and depth estimation.** Left: The original image and its depth map. Right: the amodal completion and depth maps for the two objects: Umbrella and Person. In this case we conclude that the Person occludes the Umbrella since: (i) the amodal umbrella has an overlap with the modal person, but no overlap vice versa; and (ii) the average depth map indicates that the depth of the umbrella is greater than that of the person. In this way, we predict that **“The umbrella is occluded by the person”**.

### 2.4.1 Amodal Completion

Amodal completion aims to infer the amodal mask for an object, given its modal mask, and we want to do amodal completion on COCO. Here, we adopt a similar approach to that of [Zhan et al. 2020], but re-train the amodal completion model on COCO to reduce the domain gap. In detail, instance masks are selected from the COCO dataset and randomly pasted onto training images in order to create artificial occlusions in the images. The amodal completion model is then trained on these images with artificial occlusions to predict the occluded parts, conditioned on the partial (modal) observation. During inference time, for each object, we slightly dilate its modal mask to reduce the gap between connected COCO masks, and the model can thus infer the object’s mask to its full extent, *i.e.*, amodal mask.

While evaluating for amodal completion on different datasets, as shown in Table 2.1, our model brings a significant improvement over previous approach on COCO. We refer the readers to appendix for more evaluation details.

Model	Eval Dataset	mIoU
[Zhan et al. 2020]	COCOA val	81.35
[Zhan et al. 2020]	COCO2017 val	69.32
Ours	COCO2017 val	<b>81.55</b>

**Table 2.1. Amodal Completion on COCO:** The comparison between our model and [Zhan et al. 2020]

### 2.4.2 Occlusion Reasoning

Once we get the amodal mask for each object, the occlusion ordering between a pair of connected objects is inferred in two stages (most easily understood by first



**Figure 2.4. Examples in generated training and testing datasets.** Left: Example of a target object with its occluder and occludee. Right: Examples of automatically picked objects for Separated COCO and Occluded COCO. More examples are in appendix.

looking at the example in Figure 2.3): *First*, we compute the intersection between one object’s amodal segmentation and the other object’s modal segmentation. Specifically, for each pair of connected objects  $A$  and  $B$ , the intersection of  $A$ ’s amodal mask with  $B$ ’s modal mask can be denoted as  $I_A$ , and the intersection of  $B$ ’s amodal mask with  $A$ ’s modal is denoted as  $I_B$ . Then  $A$  is likely to be occluded by  $B$ , if  $I_A > I_B$ . *Second*, we verify the results by depth estimation, *i.e.*, the occludee should have a greater depth than the occluder. We adopt an off-the-shelf depth estimator [Ranftl et al. 2021]. After conducting inference on all COCO images, we compute the average depth of each object over all pixels. Denoting object  $A$ ’s average depth as  $d_A$ ,  $B$ ’s average depth as  $d_B$ . If  $d_A > d_B$  also holds, the result of occlusion ordering is then verified. In this way, we determine the occlusion relationship between objects with greater confidence. For those cases with inconsistent amodal completion and depth verification, we do not assign any occluder-occludee relationship.

With such occlusion reasoning, for each confirmed object pair we have the pseudo ground truth relative occlusion order and ground truth modal masks, to train the tri-layer module (described in Section 2.3.1). Note that, when there are multiple occluders/occludees for one target object, we merge all their modal masks to form the final occluder/occludee mask.

### 2.4.3 Occluded COCO & Separated COCO for Evaluation

To monitor progress of object detection and instance segmentation under occlusion, we create a benchmark based on the COCO2017 val dataset. We start by defining two different splits for occluded objects, namely, **Separated COCO** and

**Occluded COCO**, consisting of **separated** and **partially occluded** objects respectively. Unlike previous work [Wang et al. 2020a] that manually selects occluded objects in COCO, we collect the data splits automatically, based on the occlusion reasoning. Specifically, for each object, we can easily check its ground truth modal mask for connectivity. If the mask is split into pieces, then the object is “separated” and should be in the Separated COCO split, for example, the car behind the horses in Figure 2.4; otherwise, the object is put into the Occluded COCO split as “partially occluded” object, if it is confirmed to be occluded by some other objects, *e.g.*, the airplane at the back in the last image of Figure 2.4. In this way, we can easily collect two datasets with different occlusion types. Detailed statistics are given in Table 2.2.

Dataset	# Total Objects
Separated COCO	3522
Occluded COCO	5550
Occluder Masks	345169
Occludee Masks	328561

**Table 2.2. Statistics of our generated training and evaluation datasets.**

## 2.5 Experiments

### 2.5.1 Datasets and Implementation Details

**Datasets:** We train all models on the COCO2017 training split, together with the occluder/occludee masks obtained in the way described in Section 2.4.2. We evaluate the performance to detect occluded objects on Separated COCO and Occluded COCO generated in Section 2.4.3, and the overall detection performance on both COCO2017 val and COCO2017 test-dev. As for evaluating generalisability, we also test on other datasets like KINS [Qi et al. 2019] (7517 images for testing), OVIS [Qi et al. 2022] and OpenImages [Kuznetsova et al. 2020] (details in appendix).

**Baselines:** We compare the following state-of-the-art architectures, Swin-T + Mask R-CNN, Swin-S + Mask R-CNN and Swin-B + Cascade Mask R-CNN, with or without our designed plugin. In addition, we also compare our tri-layer plugin with the bi-layer modelling approach (inspired by BCNet [Ke et al. 2021], set all

connected objects to be occluders), and compositional network [Yuan et al. 2021] that is specifically designed to handle object occlusion with instance masks.

**Evaluation Metrics:** In addition to the standard metric for COCO2017 val and COCO2017 test-dev, *i.e.*, mAP, we also calculate the recall on Occluded COCO and Separated COCO, *i.e.*, the number of partially occluded / separated objects that are recalled with a fixed number of RPN proposals. Specifically, we treat a partially occluded / separated object being recalled if and only if there is a detection whose confidence  $> 0.3$  and mask IOU  $> 0.75$  with it. mIOU on KINS is calculated, in accordance with that in [Yuan et al. 2021].

Model	Tri-Layer Modelling	BBox Adjustment	RoI Feature Re-weighting	Fine-tuning Whole Network?	Recall Occluded	Recall Separated	BBox mAP	Mask mAP
B1					3264(58.81%)	1125(31.94%)	46.0	41.6
B2		✓			3296(59.39%)	1141(32.40%)	47.9	42.2
B3	✓				3339(60.16%)	1157(32.85%)	46.0	41.9
B4	✓	✓			3400(61.26%)	1187(33.70%)	48.1	42.5
B5	✓	✓	✓		<b>3410(61.44%)</b>	<b>1208(34.30%)</b>	<b>48.2</b>	<b>42.8</b>
C1		✓		✓	3367(60.67%)	1170(33.22%)	48.3	42.5
C2	✓			✓	3360(60.54%)	1159(32.91%)	46.3	42.2
C3	✓	✓		✓	3434(61.87%)	1208(34.30%)	48.3	42.9
C4	✓	✓	✓	✓	<b>3441(62.00%)</b>	<b>1223(34.72%)</b>	<b>48.5</b>	<b>43.0</b>

**Table 2.3.** Ablation study for adding our plugin to Swin-T + Mask R-CNN.

## 2.5.2 Ablation Study

Here, we conduct thorough experiments to validate the effectiveness of different modules in our plugin. As shown in Table 2.3, we make the following observations: 1) **Tri-layer modelling:** brings a significant improvement in terms of Recall on Occluded COCO (B1 to B3, +75, when only fine-tuning the mask heads; B1 to C2, +96, when fine-tuning the whole network); 2) **Box adjustment:** gives a significant performance boost for BBox mAP, for example, from B1 to B2, +1.9 for only fine-tuning the head, and from B1 to C1, +2.3 for fine-tuning the network; 3) **RoI feature re-weighting:** further improves Mask mAP, Recall on Occluded COCO and Recall on Separated COCO (+0.3/+10/+21 for only fine-tuning the head and +0.1/+7/+15 for fine-tuning the whole network), which is shown by the models, from B4 to B5, and C3 to C4; 4) **Fine-tuning the whole network:** can generally bring improvement on all evaluation metrics, for example, comparing B2-B5 with C1-C4. Notably, only fine-tuning the head could already contribute the majority of the improvement, validating the effectiveness of our proposed module

as a general ‘plugin’, which can be inserted into pre-trained detectors, and give quick performance improvement.

### 2.5.3 Comparison with State-of-the-Art

**Comparison on COCO.** As shown in Table 2.4, we inject our plugin into a series of popular strong architectures, *i.e.* Mask R-CNN / Cascade Mask R-CNN with different Swin Transformers as backbone. In all cases, the plugin can always improve recalls for both Occluded COCO and Separated COCO, as well as detection performance on BBox and Mask mAP, sometimes by over 2.5/1.4 (val) and 2.4/1.4 (test-dev) mAP on box and mask predictions. In particular, when compared with bi-layer modelling, our plugin on Swin-T + Mask R-CNN can recall 126 more objects on Occluded COCO and 76 more on Separated COCO, and boost BBox/Mask mAP by 2.2/1.0 (val) and 2.2/1.1 (test-dev) respectively, which shows the effectiveness of our plugin.

Detector	Backbone	Plugin	Recall Occluded	Recall Separated	val mAP		test-dev mAP	
					BBox	Mask	BBox	Mask
Mask R-CNN	Swin-T	–	3264(58.81%)	1125(31.94%)	46.0	41.6	46.3	42.0
Mask R-CNN	Swin-T	bi-layer	3315(59.73%)	1147(32.57%)	46.3	42.0	46.5	42.3
Mask R-CNN	Swin-T	ours	<b>3441(62.00%)</b>	<b>1223(34.72%)</b>	<b>48.5</b>	<b>43.0</b>	<b>48.7</b>	<b>43.4</b>
Mask R-CNN	Swin-S	–	3393(61.14%)	1186(33.67%)	48.5	43.3	49.0	44.1
Mask R-CNN	Swin-S	ours	<b>3473(62.58%)</b>	<b>1261(35.80%)</b>	<b>50.3</b>	<b>44.2</b>	<b>50.6</b>	<b>44.9</b>
Cascade Mask R-CNN	Swin-B	–	3491(62.90%)	1279(36.31%)	51.9	45.0	52.6	45.6
Cascade Mask R-CNN	Swin-B	ours*	<b>3532(63.64%)</b>	<b>1299(36.88%)</b>	<b>52.1</b>	<b>45.4</b>	<b>52.7</b>	<b>45.9</b>

**Table 2.4. Comparison with state-of-the-art on different architectures.** The plugin gives a performance boost across all the architectures, even for the strongest detector (Swin-B + Cascade Mask R-CNN). \* Only Tri-Layer Modelling is applied as Cascade Mask R-CNN has already used multiple iterations.

**Comparison on other benchmarks.** Here, we directly evaluate the model on the KINS [Qi et al. 2019] dataset in terms of mIoU. To handle the problem that KINS classes and COCO classes are different, we make a mapping from COCO classes to KINS classes, as detailed in the appendix. Note that, these models are only trained on COCO, thus resembling a cross-domain generalisation. Specifically, while evaluating the baseline Swin-T + Mask R-CNN, it only achieves 66.6 mIoU, which is slightly lower than the CompositionalNet instance segmentation work [Yuan et al. 2021] with a 67.2 mIoU. However, with our plugin module, the performance can be largely improved, getting 68.5 mIoU and becoming better

than [Yuan et al. 2021] on KINS dataset. We refer the reader to the appendix for more detailed results on KINS, OVIS, and OpenImages.

Detector	Backbone	Tri-Layer Modelling	BBox Adjustment	RoI Feature Re-weighting	# Parameters	FLOPs
Mask R-CNN	Swin-T				47.8M	263.78G
Mask R-CNN	Swin-T	✓			54.2M	389.87G
Mask R-CNN	Swin-T	✓	✓		77.6M	583.33G
Mask R-CNN	Swin-T	✓	✓	✓	77.6M	583.33G
Mask R-CNN	Swin-S				69.1M	353.77G
Mask R-CNN	Swin-S	✓			75.5M	479.86G
Mask R-CNN	Swin-S	✓	✓		98.9M	673.32G
Mask R-CNN	Swin-S	✓	✓	✓	98.9M	673.32G
Cascade Mask R-CNN	Swin-B				145.0M	975.44G
Cascade Mask R-CNN	Swin-B	✓			164.3M	1353.68G

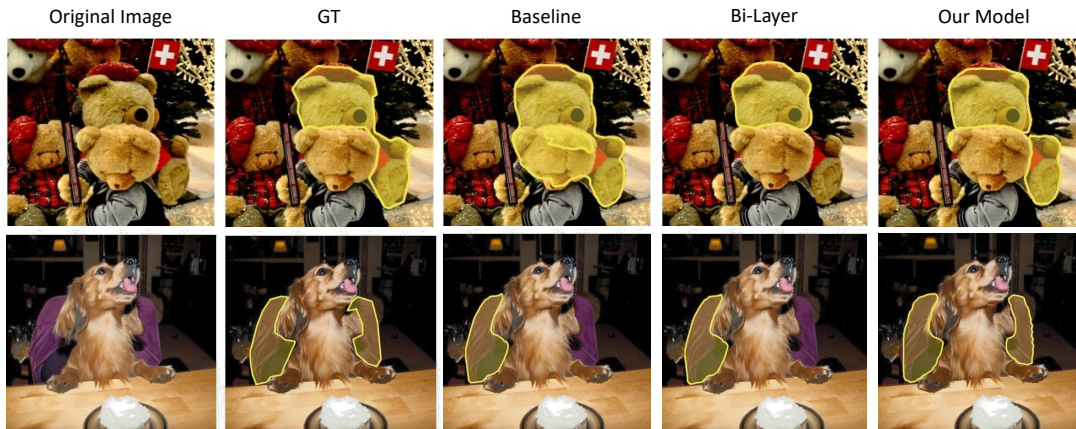
**Table 2.5. Comparison of number of parameters and FLOPs.** “Tri-Layer Modelling” introduces two extra mask heads, but only increases the number of parameters by a small proportion. The parameter increase brought by “BBox Adjustment” is mainly due to the extra bbox head (14M) in the extra iteration. The increase in FLOPs is approximately proportional to the increase in the number of parameters.

**Comparison of number of parameters and FLOPs.** Table 2.5 compares the number of parameters and FLOPs for different models with/without our plugin. “Tri-Layer Modelling” is a lightweight plugin, only introducing 9.3%, 13.4%, 13.3% more parameters for Swin-S + Mask R-CNN, Swin-T + Mask R-CNN, and Swin-B + Cascade Mask R-CNN, respectively. “BBox Adjustment” introduces more parameters for each model, where the majority of increase comes from the extra bbox head (14M) in the extra iteration. “RoI Feature Re-weighting” does not introduce any extra parameters since it only re-weights the RoI feature using the inferred segmentation mask from the previous iteration. When only fine-tuning the heads, only a small proportion of parameters need to be adjusted, so the training speed is fast.

## 2.5.4 Qualitative Results

In Figure 2.5, we show qualitative results for inserting our plugin into Swin-T + Mask R-CNN. As can be seen, the baseline model tends to fail in challenging occlusion cases, either over-segmenting the partially occluded (row 1) or under-segmenting the separated (row 2) objects. While our proposed model has largely improved the detection, for example, disambiguating the teddy bears (row 1), and inferring the two separated pieces of the chair that is heavily occluded by the

dog (row 2). See appendix for more examples.



**Figure 2.5. Qualitative results on COCO.** Please see the text for more discussion. More qualitative results are provided in appendix.

## 2.6 Conclusion and Future Work

We have proposed a simple ‘plugin’ module for two-stage object detectors that can improve their performance in detecting objects under challenging occlusions. Additionally, we describe a scalable pipeline for automatically identifying occluded and occluding objects in existing benchmarks, to provide training data and an evaluation dataset. Adding the module to a series of popular strong detectors, Mask R-CNN / Cascade Mask R-CNN with different Swin Transformer backbones, leads to consistent performance improvements.

A possible avenue of future work is to improve detection performance of occluded objects in videos, where multiple views of the objects and temporal cues are potentially available to help disambiguate the occlusions.

## Appendices

Appendices and future updates for this chapter can be found in the online version of the paper.<sup>1</sup>

## Statement of Authorship

A statement of authorship for this paper is provided in Appendix [A](#).

---

<sup>1</sup><https://arxiv.org/pdf/2210.10046>

## Chapter 3

# Amodal Ground Truth and Completion in the Wild

The paper was published at the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2024.

# Amodal Ground Truth and Completion in the Wild

Guanqi Zhan<sup>1</sup>, Chuanxia Zheng<sup>1</sup>, Weidi Xie<sup>1,2</sup>, Andrew Zisserman<sup>1</sup>

<sup>1</sup>VGG, University of Oxford,

<sup>2</sup>CMIC, Shanghai Jiao Tong University

## Abstract

This paper studies amodal image segmentation: predicting entire object segmentation masks including both visible and invisible (occluded) parts. In previous work, the amodal segmentation ground truth on real images is usually predicted by manual annotation and thus is subjective. In contrast, we use 3D data to establish an automatic pipeline to determine authentic *ground truth* amodal masks for partially occluded objects in real images. This pipeline is used to construct an amodal completion *evaluation* benchmark, *MP3D-Amodal*, consisting of a variety of object categories and labels. To better handle the amodal completion task in the wild, we explore two architecture variants: a two-stage model that first infers the occluder, followed by amodal mask completion; and a one-stage model that exploits the representation power of Stable Diffusion for amodal segmentation across many categories. Without bells and whistles, our method achieves a new state-of-the-art performance on Amodal segmentation datasets that cover a large variety of objects, including COCOA and our new *MP3D-Amodal* dataset. The dataset, model, and code are available at <https://www.robots.ox.ac.uk/~vgg/research/amodal/>.

### 3.1 Introduction

The vision community has rapidly improved instance segmentation performance over the last few years with a succession of powerful models, such as MaskRCNN [He et al. 2017], Mask2Former [Cheng et al. 2022], and Seg-Anything (SAM) [Kirillov et al. 2023]. However, despite this remarkable progress, these models only provide pixel-level *modal* segmentations for objects in the images, *i.e.*, the instance masks are for the *visible* pixels. The models lack the human ability to infer the full extent of the object in an image, when it is partially occluded. The prediction of *amodal masks*, which covers the full extent of the object, can assist several downstream tasks including object detection [Zhan et al. 2022], smart image editing [Zhan et al. 2020; Ling et al. 2020; Xu et al. 2024], 3D reconstruction from a single image [Kar et al. 2015; Zou and Hoiem 2020; Kanazawa et al. 2018; Wu et al. 2023b; Ozguroglu et al. 2024], object permanence in video segmentation [Xie et al. 2022; Van Hoorick et al. 2023; Hsieh et al. 2023], predicting support relationships between objects [Silberman et al. 2012; Zhuo et al. 2017], and more generally for planning in manipulation and navigational tasks where reasoning on the full extent of the object could lead to improvements [Kellman and Massey 2013; Varley et al. 2017; Kim and Likhachev 2016; Inagaki et al. 2019; Wu et al. 2023a].

Predicting amodal masks for objects in 2D images is challenging: this is because real scenes contain a vast collection of different types of objects, resulting in complex occlusions when they are projected to 2D images from a 3D physical world. To accurately complete the amodal shape of partially occluded objects requires a prediction of occlusion relations (in order to infer if and where the object is partially occluded), as well as predicting the shape of the occluded regions. This challenge is also reflected in the type of amodal datasets available which are mostly synthetic – for real images, amodal masks are almost always ‘imagined’ by human annotators providing the contour of the amodal mask, and there is no dataset available to evaluate amodal completions with *authentic* ground truth annotations for a large variety of object categories (see Table 3.1).

In this paper, our first contribution is to provide a new amodal benchmark evaluation dataset based on authentic ground truth amodal segmentation for real images,

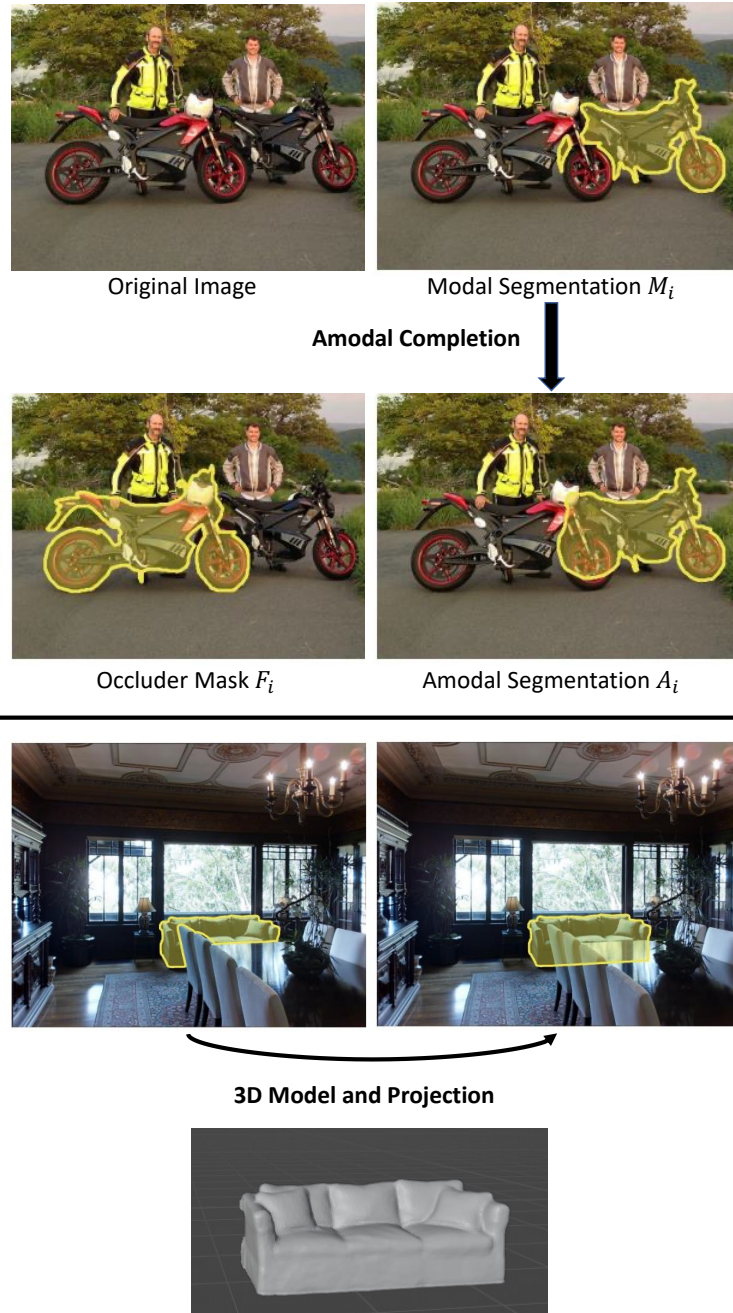
Dataset	Amodal GT	Image Type	# Categories	# Images	Type
COCOA [Zhu et al. 2017]	✗	Real	Multiple	5,073	General
COCOA-cls [Follmann et al. 2019]	✗	Real	80	3,499	General
KINS [Qi et al. 2019]	✗	Real	8	14,991	Vehicle
DYCE [Ehsani et al. 2018]	✓	Synthetic	79	5,500	Indoor
OLMD [Dhamo et al. 2019]	✓	Synthetic	40	13,000	Indoor
CSD [Zheng et al. 2021]	✓	Synthetic	40	11,434	Indoor
D2SA [Follmann et al. 2019]	✓	Synthetic	60	5,600	Industrial
KITTI-360-APS [Mohan and Valada 2022]	✗	Real	17	61,168	Vehicle
BDD100K-APS [Mohan and Valada 2022]	✗	Real	16	202	Vehicle
WALT [Reddy et al. 2022]	✓	Real	2	60,000	Vehicle
MUVA [Li et al. 2023c]	✓	Synthetic	20	26,406	Shopping
MP3D-Amodal (Ours)	✓	Real	427(40)	10,883	Indoor

**Table 3.1. Comparison of Different Amodal Datasets.** Amodal GT: whether the dataset provides ground truth amodal annotations or is manually guessed. # represents the number of the following types. Our MP3D-Amodal dataset (Sec. 3.3) has 427 different semantic labels mapped to 40 different MatterPort categories. Note, the WALT dataset consists of video frames from 12 camera viewpoints, mainly of vehicles moving.

and covering a large variety of objects. The new dataset is called *MP3D-Amodal*, and the amodal mask is obtained by projecting the 3D structure of objects in the scene onto the image (Figure 3.1 bottom). We build the dataset from MatterPort3D [Chang et al. 2017] that has both 3D structure and real images of indoor scenes. The dataset and generation method is described in Sec. 3.3.

In most prior work, amodal completion algorithms required the occluder mask to be specified [Zhan et al. 2020; Nguyen and Todorovic 2021] (Figure 3.1 top). Our second contribution is to propose two architecture variants that do not require the occluder mask to be supplied: *OccAmodal*, a two-stage model that first infers the occluder, followed by amodal mask completion; and *SDAmodal*, a one-stage model that uses the features of a pre-trained Stable Diffusion network, benefiting from its strong outpainting capabilities. These architectures are described in Sec. 3.4.

We achieve state-of-the-art amodal completion performance on both the public COCOA [Zhu et al. 2017] dataset, and on our own *MP3D-Amodal* benchmark. In particular, the one-stage model, *SDAmodal*, benefiting from the pre-trained Stable Diffusion model, is able to generalize to another dataset with objects from a different domain and different categories, demonstrating class-agnostic completion. Taken together, the handling of situations where occluder masks are not provided and the class-agnostic domain generalization, moves amodal completions towards an ‘in the wild’ capability.



**Figure 3.1. Amodal Ground Truth and Completion in the Wild.** Top: The task of amodal completion is to predict the amodal mask  $A_i$  for an object ‘ $i$ ’ in the image specified by the modal mask  $M_i$  (here the object is the rear motorbike). Previous methods [Zhan et al. 2020; Nguyen and Todorovic 2021] require the mask of the occluder  $F_i$  to be also provided to do the task; but our goal is to predict the amodal mask when the occluder mask is *not* provided and the occluded object is of *any category*. Bottom: We propose a novel method for generating amodal masks for real images: using 3D structure to produce *ground truth* modal and amodal masks for object instances. The method is used to generate a ground truth evaluation dataset for real images.

## 3.2 Related Work

**Amodal Datasets.** In the literature, there have been continuous efforts on creating datasets for amodal segmentation, for example, COCOA [Zhu et al. 2017; Follmann et al. 2019] builds on COCO [Lin et al. 2014], KINS [Qi et al. 2019] builds on KITTI [Geiger et al. 2012]. However, the ground truth amodal masks for both of these datasets are created based on the 2D images, thus inevitably requiring human imagination for the occluded regions. To improve the quality of ground truth amodal mask, the DYCE [Ehsani et al. 2018], OLMD [Dhamo et al. 2019], CSD [Zheng et al. 2021] and MUVA [Li et al. 2023c] datasets were created by rendering the whole scene and corresponding individual intact objects using synthetic 3D models. WALT [Reddy et al. 2022] collected objective amodal masks via time-lapse imagery, but their objects are limited to cars and humans. Table 3.1 provides a summary of the datasets currently available. In contrast, we are the first to collect a complex dataset that provides authentic amodal ground truth for the occluded objects of a large variety of categories in real scenes.

**Amodal Instance Segmentation.** Classical instance segmentation methods [Oliveira et al. 2015; He et al. 2017; Chen et al. 2019a; Cai and Vasconcelos 2019; Cheng et al. 2022; Kirillov et al. 2023] mainly focus on segmenting *visible* pixels, while amodal instance segmentation [Li and Malik 2016] aims to detect the objects as a whole, *i.e.*, both *visible* and *invisible* parts. These methods are usually trained on images [Qi et al. 2019; Follmann et al. 2019; Zheng et al. 2021; Xiao et al. 2021; Ke et al. 2021; Mohan and Valada 2022; Sun et al. 2022; Tran et al. 2022; Li et al. 2022; Gao et al. 2023; Li et al. 2023b] with manually annotated ground truth amodal masks in a fully-supervised manner. However, these methods are trained on datasets with limited number of object classes, *e.g.*, 80 categories for COCOA-cls, and are class dependent.

**Amodal Completion** is conceptually similar to amodal instance segmentation, except that here the *visible* mask for the target object is already provided as input. Most existing methods [Zhan et al. 2020; Nguyen and Todorovic 2021] assume the occluder mask is provided and cannot handle the situation where the object is occluded by an unknown occluder, *i.e.*, the occluder mask is not provided or the occluder is difficult to define. The methods are trained on COCOA, which covers

Split	# Scenes	# Images	# Instances	# MatterPort Categories	# Semantic Labels
Training	4	1,100	1,283	35	130
Evaluation	86	9,783	11,441	40	385
Total	90	10,883	12,724	40	427

**Table 3.2. Statistics of the generated *MP3D-Amodal* dataset.** Each instance has a semantic label as annotated in the MatterPort3D dataset, which is also mapped to a more general MatterPort category. Across the training and evaluation splits, there are 88 semantic labels in common, and 297 semantic labels in the evaluation split but not in the training split.

a large variety of categories, and are more class-agnostic than methods trained on COCOA-cls. Another work [Ling et al. 2020] uses VAE [Kingma and Welling 2014] to model the task of amodal completion, but can only handle limited categories in driving scenes.

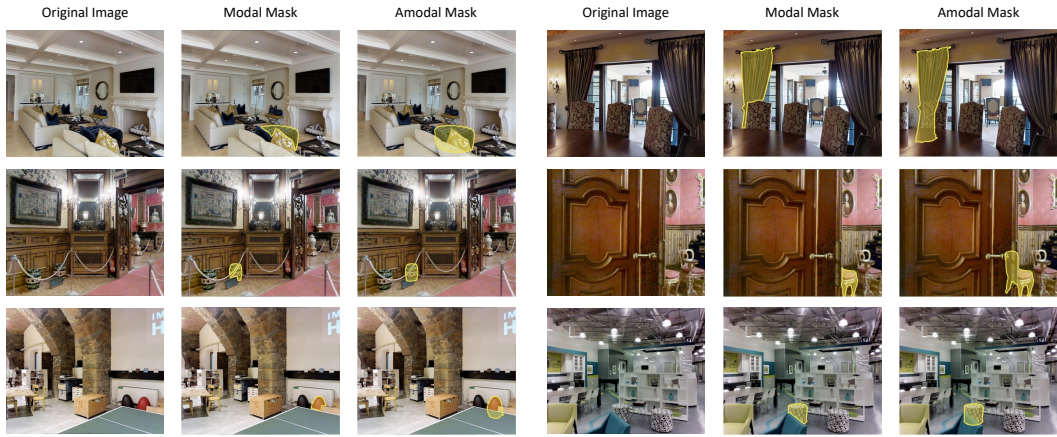
### 3.3 The MP3D-Amodal Dataset

In this section we describe the new amodal dataset *MP3D-Amodal*, that is constructed from the MatterPort3D [Chang et al. 2017] dataset. We first overview the contents of the dataset in Sec. 3.3.1, and then describe our method of generating ground truth amodal masks on real images from 3D data in Sec. 3.3.2.

#### 3.3.1 An Overview of the Dataset

The dataset contains 12,724 annotated amodal ground truth masks for over 10,883 real images. Since it is built from the MatterPort dataset, we use the classifications inherited from that dataset, where objects are described by their *category* and *semantic labels*. Note that, categories are more coarse-grained than semantic labels and one category may contain several different semantic labels, *e.g.*, the category ‘chair’ contains semantic labels ‘dining chair’, ‘sofa chair’ and ‘armchair’; and the category ‘appliances’ contains ‘refrigerator’, ‘oven’, and ‘washing machine’.

Table 3.2 gives the details for dataset splits. To have a better and more comprehensive evaluation, we make the evaluation split to have more scenes than the training split. Across the training and evaluation splits, there are 88 semantic labels in common, and 297 semantic labels in the evaluation split but not in the training split. A small part of the collected dataset is reserved for training, as this



**Figure 3.2.** Samples from the *MP3D-Amodal Dataset*. For each sample, the original image is shown together with the generated modal and amodal masks.

allows some domain adaptation for an algorithm. The scenes of the training set are disjoint from those of the evaluation set.

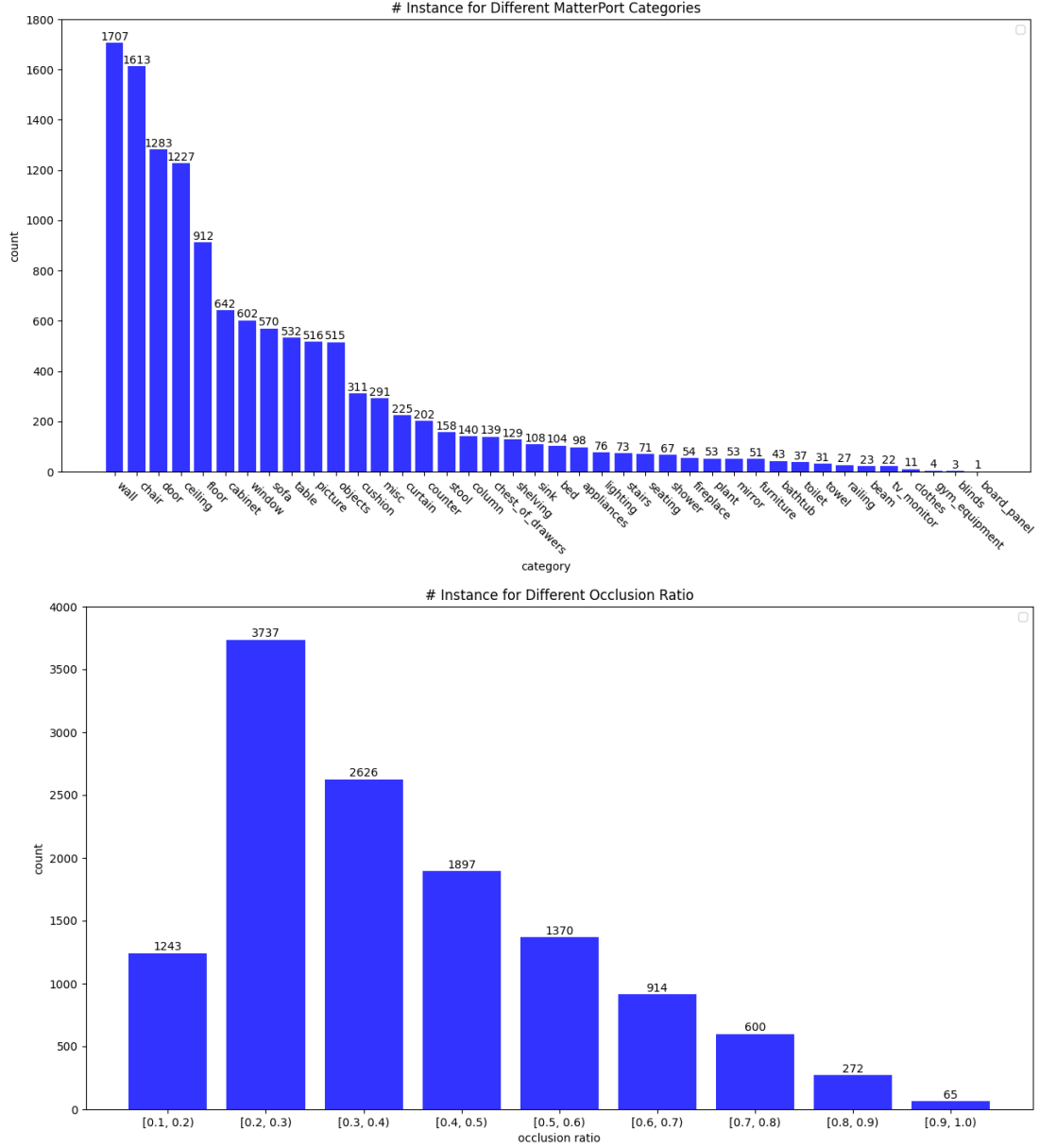
Samples from the dataset are displayed in Figure 3.2. The dataset contains diverse range of objects, with some categories not in the ‘general’ COCOA dataset, *e.g.*, the example in the bottom left of Figure 3.2 is a novel category. More examples of the dataset are in the appendix.

Figure 3.3 visualizes the distributions of the dataset in terms of the number of instances for each MatterPort category, and the number of instances for different occlusion ratios, where the occlusion ratio is the proportion of the object that is occluded (the difference between amodal and modal masks, divided by the area of the amodal mask). It is evident that there is a wide range of occlusion ratios, from slightly occluded to severely occluded.

### 3.3.2 Generating Amodal Ground Truth from 3D

We exploit the MatterPort3D [Chang et al. 2017] dataset, that is equipped with two essential elements: a 3D mesh for each object instance in the scene, and real images (and associated cameras) of the scene. In the following we detail the procedure for automatically generating amodal and modal masks of individual objects. The process is illustrated in Figure 3.4.

**Modal Mask Generation.** For a particular scene, we obtain 2D instance segmentations (a modal mask for each object) by projecting all objects with their instance labels together onto the image with the associated camera. If  $M_i$  and  $O_i$



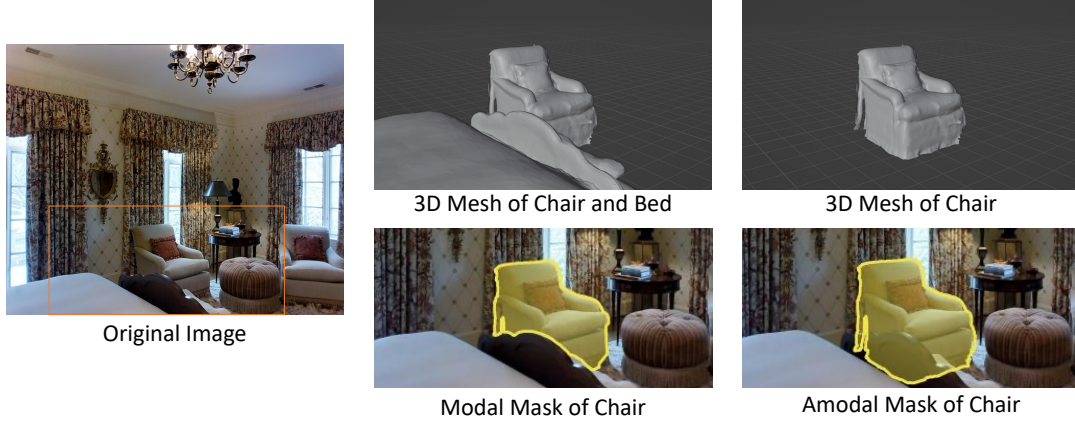
**Figure 3.3.** Distributions of the *MP3D-Amodal* Dataset in terms of the number of instances for each MatterPort category, and the number of instances for different occlusion ratios.

denote the modal mask and 3D mesh of the  $i$ -th object,  $\Phi$  refers to the projection of 3D meshes to the camera space, then the modal masks of the image are:

$$\{M_1, M_2, \dots, M_n\} = \Phi(O_1 \cup O_2 \cup \dots \cup O_n) \quad (3.1)$$

**Amodal Mask Generation.** The amodal mask  $A_i$  for each object  $i$  is simply obtained by projecting each object to the camera separately:

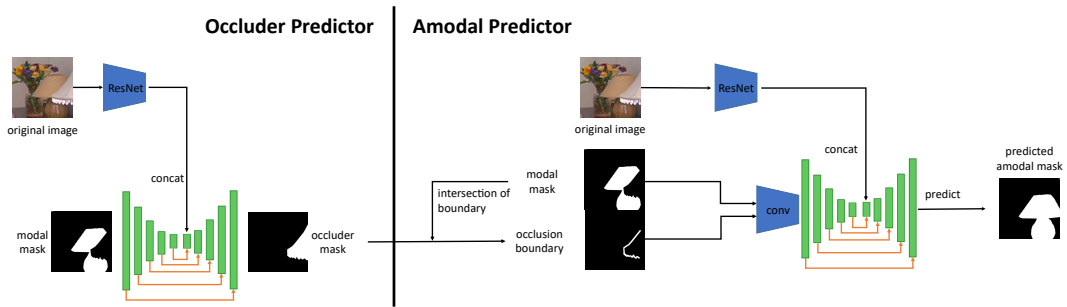
$$\{A_i\} = \Phi(O_i), \quad \forall i \in \{1, 2, 3, \dots, n\} \quad (3.2)$$



**Figure 3.4. Automated Generation of the *MP3D-Amodal* Ground Truth Dataset.** The dataset is automatically generated from the MatterPort3D [Chang et al. 2017] dataset, and provides ground truth modal and amodal masks for objects in real images. The generation process is illustrated here for the chair and proceeds in two steps: first, modal and amodal masks in a particular image are obtained for each object by projecting the object’s 3D mesh individually (for the amodal mask), and also by projecting the 3D mesh of all objects (for the modal mask). In this example, the 3D mesh of the bed occludes the chair when projected into the image. In the second step, an object is selected for the dataset if its amodal mask is larger than its modal mask by a threshold. In this case the chair is selected, but other objects such as the stool would not be selected since it is not occluded by other objects in this viewpoint, and so their modal and amodal masks would be the same.

**Occluded Object Selection.** Then partially occluded objects are identified and selected by comparing the modal and amodal mask. If the amodal mask of the object is larger than the modal mask, then there must be something occluding the object, and that object’s modal and amodal masks are candidates to be included in the dataset. Here we automatically include objects with  $S(A_i) > 1.2 S(M_i)$ , *i.e.*, the area of its amodal mask is more than 1.2 times larger than its modal mask. Take the chair in Figure 3.4 as an example, we first generate its modal and amodal mask using Equations 3.1 and 3.2. Because the amodal mask of the chair is larger than its modal mask (it is occluded by the bed), we select the chair in the dataset. In this way, we have an automatic method to collect ground truth amodal masks for occluded objects in real images without any manual guessing.

**Manual Selection.** However, not all generated modal and amodal masks are of very good quality as the 3D meshes in MatterPort3D can be incomplete or noisy sometimes. We thus apply a manual selection stage, where human annotators inspect and select the pairs with good-quality modal and amodal masks. Bad quality examples due to problems of MatterPort3D are categorized (*e.g.*, the modal mask does not contain all visible parts of the object, or the amodal mask is noisy / in-



**Figure 3.5. Two-Stage Architecture (OccAmodal) for Amodal Prediction.** *Left:* A lightweight U-Net based architecture is used to predict the occluder mask for each object. *Right:* The amodal predictor takes the predicted occluder mask, together with the modal mask and image as input to predict the amodal segmentation mask.

complete) and shown to the human annotators. The manual selection is described in full detail in the appendix.

### 3.4 Architectures for Amodal Prediction

Given a single image  $\mathcal{I} \in \mathbb{R}^{H \times W \times 3}$  and its corresponding modal (*visible*) mask  $M_i \in \mathbb{R}^{H \times W}$  for the  $i$ -th object, our goal is to predict the amodal (*full*) mask for the object,  $A_i \in \mathbb{R}^{H \times W}$ . Specifically, we explore two architecture variants:

- A two-stage architecture, as shown in Figure 3.5, consisting of an **occluder predictor** to first estimate the occluder mask, followed by an **amodal predictor** to infer the amodal mask, given the modal mask, estimated occluder, and image.
- A one-stage architecture, as shown in Figure 3.6, that exploits the strong representation power of the pre-trained stable diffusion model, and adapts it to infer the amodal mask from the given image and modal mask.

#### 3.4.1 Two-Stage Architecture – OccAmodal

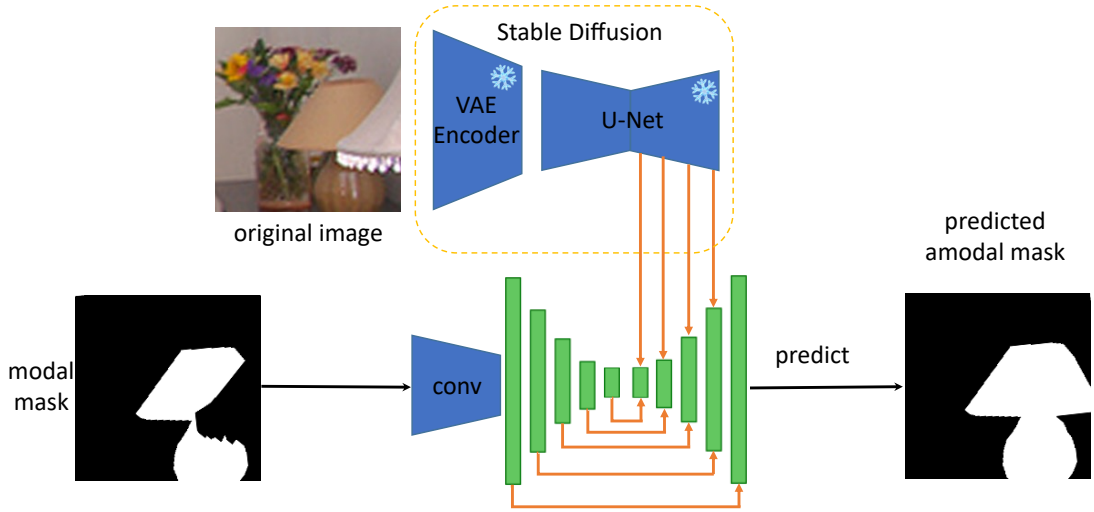
**Occluder Predictor.** Occlusion in an image occurs when an object hides a part of another object, referred to as occluder and occludee respectively. For amodal completion, having the occluder’s mask can largely simplify the task, as it provides information on which parts of one specified object should be completed [Zhan et al. 2020; Nguyen and Todorovic 2021]. In existing works [Zhan et al. 2020; Nguyen and Todorovic 2021], the occluder mask is often considered as a prior, and is directly fed into the model as input. One obvious limitation, however, is that the

occluder mask can be unavailable at inference time. For example, in large-scale datasets, *e.g.*, COCO [Lin et al. 2014] or LVIS [Gupta et al. 2019], not all objects in an image are annotated, resulting in a failure of amodal completion in existing works [Zhan et al. 2020; Nguyen and Todorovic 2021], *i.e.*, they cannot expand the modal mask at all if the segmentation of occluder mask is not annotated and provided. Here, instead of relying on an a-prior occluder mask, we consider a two-stage architecture, that first infers the occluder mask from the given image and the target object’s modal mask, and then generates an amodal completion with the occluder mask as guidance. Specifically, as shown in Figure 3.5 (left), the occluder predictor takes the original image and the object’s modal mask as input, to the ResNet and U-Net encoder respectively, and is then concatenated and upsampled to generate the prediction of the occluder mask for the object,  $F_i = \Psi_{\{\text{OCP}\}}(\mathcal{I}, M_i)$ , where  $F_i \in \mathbb{R}^{H \times W}$  denotes the binary mask of occluder, that can be completely empty (no occluder), or with the union of all occluders.

**Amodal Predictor.** Given the mask of the predicted occluder, we compute the occlusion boundary ( $B_i$ ), between the modal mask and occluder mask. We then feed the input image, object’s modal mask, and occlusion boundary to an amodal predictor, as shown in Figure 3.5 (right), similar to existing work [Nguyen and Todorovic 2021]. In detail, both the input modal mask ( $M_i$ ) and occlusion boundary ( $B_i$ ) are concatenated, and input to a U-Net for encoding and decoding, with skip connections. Additionally, we also encode the input image with a ResNet, and inject it into the U-Net’s bottleneck layer, providing visual conditioning for amodal completion. We denote the procedure as :  $A_i = \Psi_{\{\text{AMP}\}}(\mathcal{I}, M_i, F_i)$

### 3.4.2 One-stage Architecture – SDAmodal

In recent literature, generative models based on diffusion have demonstrated the ability to generate photorealistic images, with seemingly correct geometry, object semantics and shapes. Here, we investigate the possibility of exploiting the visual features in diffusion models for amodal completion – after all, the task is mainly about understanding the shape of certain objects. Specifically, as shown in Figure 3.6, we feed the image into a pre-trained Stable Diffusion model, and add noise onto the latent features after the VAE encoder. We extract the multi-scale features from the decoding layers of the U-Net at time step 181 following the investigation



**Figure 3.6. One-Stage Architecture (SDAmodal) for Amodal Prediction.** The image is fed into a pre-trained Stable Diffusion model to get multi-scale representations containing occlusion information. The image and modal mask features are concatenated and forwarded to multiple decoding layers for amodal prediction. The Stable Diffusion model is frozen during training.

about occlusion in [Zhan et al. 2023]. Then we concatenate the Stable Diffusion features with multi-scale features of the modal mask, and forward them to multiple decoding layers to generate the amodal mask prediction, the procedure can be denoted as :  $A_i = \Psi_{\{SD\}}(\mathcal{I}, M_i)$ .

### 3.4.3 Training

Training the first stage of OccAmodal requires ground truth occluder masks, while training both the second stage of OccAmodal and the SDAmodal requires the ground truth amodal masks. Both COCOA and our MP3D-Amodal provide ground truth amodal masks while only COCOA provides ground truth occluder masks.

**OccAmodal.** For training of the occluder predictor, the occlusion relationships annotated in COCOA [Zhu et al. 2017] are used to obtain the ground truth occluder mask, and then the pixel-level prediction of occluder mask is trained via cross-entropy loss. For training of the amodal predictor, the amodal mask prediction is supervised by the ground truth amodal mask (provided by COCOA or MP3D-Amodal) via an Uncertainty Weighted Segmentation Loss as mentioned in [Nguyen and Todorovic 2021].

**SDAmodal.** For training of the Stable Diffusion based architecture, the amodal

mask prediction is supervised by the ground truth amodal mask (provided by COCOA or MP3D-Amodal) via a cross-entropy loss.

## 3.5 Experiments

### 3.5.1 Experimental Details

**Datasets and Implementation Details.** We employ both COCOA [Zhu et al. 2017] and our collected MP3D-Amodal (Section 3.3) for training and evaluating our models. To ensure a fair comparison, we use the same training setting as in [Zhan et al. 2020; Nguyen and Todorovic 2021], which employs SGD with momentum, sets the learning rate to be  $1e^{-3}$ , and trains the model for 56K iterations with a batch size of 32. Models are trained on A6000 / A40 GPUs. More training details are given in the appendix.

**Baselines.** We compare with two existing amodal completion models [Zhan et al. 2020; Nguyen and Todorovic 2021], of which [Nguyen and Todorovic 2021] is the latest state-of-the-art method for amodal completion. [Zhan et al. 2020] has both one-stage and two-stage architectures, which we denote as Deocclusion (Single Stage) and Deocclusion (Two Stage). The default architecture in [Zhan et al. 2020] is Deocclusion (Two Stage), while Deocclusion (Single Stage) uses a ResNet to encode the input image and concatenate it with the features in the U-Net decoder (similar to Figure 3.5 left). Additionally, in the appendix, we compare with recent amodal instance segmentation methods such as VRSP [Xiao et al. 2021], A3D [Li et al. 2022], AISformer [Tran et al. 2022], C2F-Seg [Gao et al. 2023] and GIN [Li et al. 2023b].

**Evaluation.** Following [Zhan et al. 2020; Nguyen and Todorovic 2021], we compute mIOU between the ground truth and predicted amodal mask. Additionally, mIOU-inv is also used, which refers to the mIOU for only the occluded regions.

### 3.5.2 Ablation Study of Different Architectures

In Table 3.3, we ablate the importance of the occluder predictor and the number of skip connections for the OccAmodal architecture. As can be seen, the occluder mask is crucial for amodal mask prediction. This is evident from the results of

ID	Occluder Predictor	Final Skip Connection	COCO A	
			mIOU	mIOU-inv
A			69.9	0.006
B	✓		88.4	64.4
C	✓	✓	<b>89.4</b>	<b>66.2</b>

**Table 3.3. Ablation Study of OccAmodal.** Setting A is the setting of ASBU [Nguyen and Todorovic 2021]. All models are trained on COCO A.

ID	SD Backbone	Multi-Scale SD Feature	Final Skip Connection	COCO A	
				mIOU	mIOU-inv
A				88.0	63.8
B	✓			89.4	69.2
C	✓	✓		89.6	69.8
D			✓	89.2	66.4
E	✓		✓	90.5	71.1
F	✓	✓	✓	<b>90.7</b>	<b>71.6</b>

**Table 3.4. Ablation Study of SDAmodal.** Setting A is Deocclusion (Single Stage) in [Zhan et al. 2020]. All models are trained on COCO A.

Setting A, achieving only 69.9 mIOU on COCO A. In comparison, when the predicted occluder mask is incorporated, amodal completion can be boosted to 88.4 mIOU (Setting B) on COCO A, and the performance is further boosted when we include a skip connection for the final layer of the U-Net (Setting C). In the architecture of [Nguyen and Todorovic 2021] there are only 4 skip connections and we are adding the fifth. In Table 3.4, we ablate variations on the SDAmodal architecture. Replacing the original ResNet image encoder with the Stable Diffusion backbone brings a significant boost (+1.4/+5.4 in terms of mIOU and mIOU-inv for Setting A to B, +1.3/+4.7 in terms of mIOU and mIOU-inv for Setting D to E). If multiple layers of Stable Diffusion features at different resolutions are fed into the model (as shown in Figure 3.6) the performance is higher than if only a single layer feature is used (the second layer of the Stable Diffusion U-Net as in [Zhan et al. 2023]) (comparing Setting B/C and E/F). The performance can also be improved by adding a final layer skip connection for the U-Net (comparing Setting A/D, B/E and C/F). According to [Zhan et al. 2023], the features of other pre-trained models such as DINO [Caron et al. 2021; Oquab et al. 2023] and CLIP [Radford et al. 2021; Ilharco et al. 2021] perform worse than Stable Diffusion features on “occlusion” task. We have further trained our model using DINO and CLIP features. The results are given in the appendix, validating the superiority

ID	Comments	Occluder Mask Provided	COCOA		MP3D-Amodal	
			mIOU	mIOU-inv	mIOU	mIOU-inv
A	Deocclusion(Two Stage)	✓	88.2	65.3	-	-
B	ASBU(reproduced)	✓	88.9	65.3	-	-
C	ASBU(reported)	✓	89.9	-	-	-
D	Deocclusion (Two Stage)		69.9	0.006	64.4	0.004
E	ASBU		69.9	0.006	64.4	0.004
F	Deocclusion (Single Stage)		88.0	63.8	72.4	28.0
G	OccAmodal		89.4	66.2	72.9	27.5
H	SDAmodal		<b>90.7</b>	<b>71.6</b>	<b>76.4</b>	<b>38.5</b>

**Table 3.5. Comparison with State-of-the-Art Amodal Completion Methods.** Our SDAmodal model achieves the new state-of-the-art performance for amodal completion over a larger variety of categories. All models are trained on COCOA, and evaluated on both COCOA and MP3D-Amodal.

of Stable Diffusion features.

### 3.5.3 Comparison with State-of-the-Art

We compare our method with previous amodal completion state-of-the-art methods, Deocclusion [Zhan et al. 2020] and ASBU [Nguyen and Todorovic 2021], on both COCOA and MP3D-Amodal. Note that, ASBU [Nguyen and Todorovic 2021] and Deocclusion (Two Stage) [Zhan et al. 2020] require the occluder masks provided, while in both of the architectures we propose, the occluder masks are not necessary. The comparisons are given in Table 3.5. We can make the following observations: (1) SDAmodal outperforms the previous state-of-the-art methods (Setting C and H) even if the occluder mask is not provided for SDAmodal, but is for previous methods; (2) When the occluder mask is not provided, previous methods Deocclusion (Two Stage) [Zhan et al. 2020] and ASBU [Nguyen and Todorovic 2021] cannot expand the modal mask of the object and achieve poor performance for amodal completion (Setting D and E). In comparison, OccAmodal (Setting G), where the occluder mask is generated by our occluder predictor, has a high performance, demonstrating the effectiveness of the occluder mask prediction module. (3) Even though SDAmodal is only trained on COCOA, the Stable Diffusion backbone efficiently boosts the performance not only on COCOA, but also *zero-shot generalized* to MP3D-Amodal which contains objects from different domains and categories (compare Settings F and H where the difference is +4.0/+10.5 in terms of mIOU and mIOU-inv). Comparison with recent Amodal Instance Segmentation methods are given in the appendix.

ID	Architecture	COCOA	MP3D-Amodal	COCOA		MP3D-Amodal	
				mIOU	mIOU-inv	mIOU	mIOU-inv
A	OccAmodal	✓		<b>89.4</b>	66.2	72.9	27.5
B	OccAmodal	✓	✓	<b>89.4</b>	<b>66.4</b>	<b>73.8</b>	<b>29.6</b>
C	SDAmodal	✓		<b>90.7</b>	<b>71.6</b>	76.4	38.5
D	SDAmodal	✓	✓	<b>90.7</b>	<b>71.6</b>	<b>81.8</b>	<b>53.7</b>

**Table 3.6. Effectiveness of Different Training Data.** The performance of both models are boosted on MP3D-Amodal if extra training data from MP3D-Amodal is used.



**Figure 3.7. Qualitative Comparison on COCOA and MP3D-Amodal.** COCOA: Rows 1, 2 and 3; MP3D-Amodal: Rows 4 and 5. Please see the text for more discussion. More qualitative examples are provided in the appendix.

### 3.5.4 Effectiveness of Different Training Data

Table 3.6 shows the effectiveness of training with extra data from our MP3D-Amodal training split. Both OccAmodal and SDAmodal improve performance on MP3D-Amodal when they are also trained with MP3D-Amodal and there is no performance deterioration on COCOA.

### 3.5.5 Qualitative Results

In Figure 3.7, we show a qualitative comparison of different amodal completion methods on both the COCOA and MP3D-Amodal datasets. We observe that ASBU [Nguyen and Todorovic 2021] faces limitations in expanding the modal mask when the occluder mask is not provided (Column 3). Deocclusion (Single Stage) can partially complete the amodal mask when the occluder mask is not available but the prediction quality is not good (Column 4). In contrast, our

models, especially SDAmodal, can handle the situation where the occluder mask is not provided and significantly improve the accuracy of amodal mask predictions (Columns 5 and 6), even when the object to complete is from a different domain (Rows 4 and 5) when only trained on COCOA. More qualitative results are in the appendix.

### 3.6 Conclusion and Extensions

By utilising real 3D data, we have proposed an automatic pipeline to generate ground truth amodal masks for occluded objects in real images, and used this to create a new amodal segmentation evaluation benchmark for a large variety of instances. The pipeline has been applied to the MatterPort3D dataset, but can be applied to other suitable datasets such as ScanNet [Dai et al. 2017] that have real images together with the 3D structure for objects in the scene. Furthermore, we have developed two new state-of-the-art methods for amodal completion *in the wild* – *i.e.*, capable of handling situations where the occluder is unknown or undefined, and for a wide variety of object classes. The models, with a lightweight occluder predictor and Stable Diffusion representations, achieve superior performance on different domains and object categories.

## Appendices

Appendices and future updates for this chapter can be found in the online version of the paper.<sup>1</sup>

## Statement of Authorship

A statement of authorship for this paper is provided in Appendix [A](#).

---

<sup>1</sup><https://arxiv.org/pdf/2312.17247>

## Part II

### 3D Physical Understanding

## Chapter 4

# A General Protocol to Probe Large Vision Models for 3D Physical Understanding

The paper was published at the Advances in Neural Information Processing Systems (NeurIPS), 2024.

# A General Protocol to Probe Large Vision Models for 3D Physical Understanding

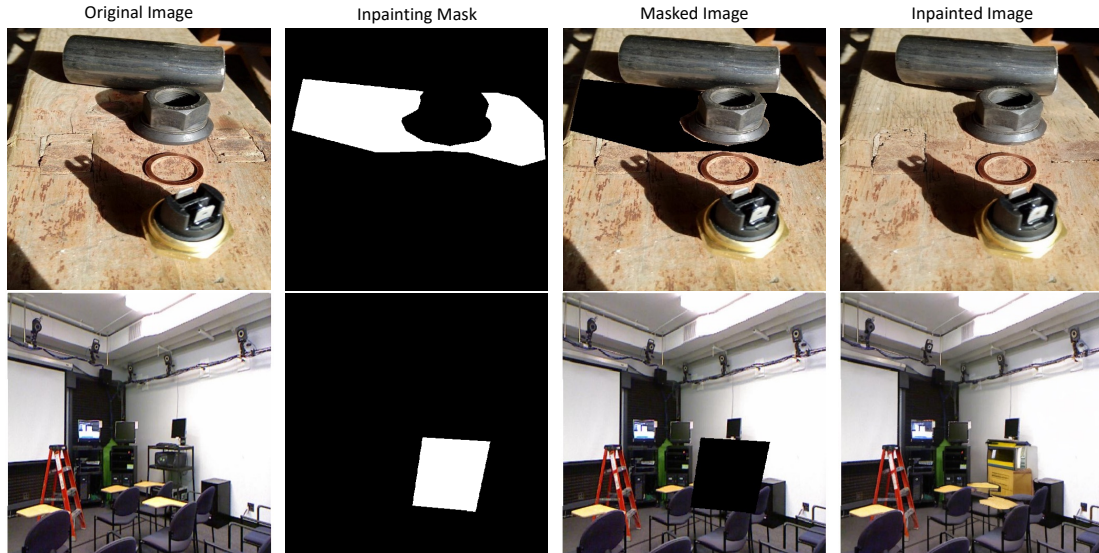
Guanqi Zhan<sup>1</sup>, Chuanxia Zheng<sup>1</sup>, Weidi Xie<sup>1,2</sup>, Andrew Zisserman<sup>1</sup>

<sup>1</sup>VGG, University of Oxford,

<sup>2</sup>SAI, Shanghai Jiao Tong University

## Abstract

Our objective in this paper is to probe large vision models to determine to what extent they ‘understand’ different physical properties of the 3D scene depicted in an image. To this end, we make the following contributions: (i) We introduce a *general* and *lightweight* protocol to evaluate whether features of an off-the-shelf large vision model encode a number of physical ‘properties’ of the 3D scene, by training discriminative classifiers on the features for these properties. The probes are applied on datasets of real images with annotations for the property. (ii) We apply this protocol to properties covering scene geometry, scene material, support relations, lighting, and view-dependent measures, and large vision models including CLIP, DINOv1, DINOv2, VQGAN, Stable Diffusion. (iii) We find that features from Stable Diffusion and DINOv2 are good for discriminative learning of a number of properties, including scene geometry, support relations, shadows and depth, but less performant for occlusion and material, while outperforming DINOv1, CLIP and VQGAN for all properties. (iv) It is observed that different time steps of Stable Diffusion features, as well as different transformer layers of DINO/CLIP/VQGAN, are good at different properties, unlocking potential applications of 3D physical understanding. Our project page is <https://www.robots.ox.ac.uk/~vgg/research/phy-sd/>.



**Figure 4.1. Motivation: What do large vision models know about the 3D scene?** We take Stable Diffusion as an example because Stable Diffusion is generative, and so its output is an image that can be judged directly for verisimilitude. The Stable Diffusion inpainting model is here tasked with inpainting the masked region of the real images. It correctly predicts a shadow consistent with the lighting direction (top), and a supporting structure consistent with the scene geometry (bottom). This indicates that the Stable Diffusion model generation is consistent with the geometry (of the light source direction) and physical (support) properties. These examples are only for illustration and we probe a general Stable Diffusion network to determine whether there are explicit features for such 3D scene properties. The appendix provides more examples of Stable Diffusion’s capability to predict different physical properties of the scene.

## 4.1 Introduction

The large-scale pre-trained vision foundation models have achieved great success in computer vision tasks, including classification (CLIP [Radford et al. 2021; Ilharco et al. 2021]), segmentation (DINO [Caron et al. 2021; Oquab et al. 2023]), and image generation (VQGAN [Esser et al. 2021], Stable Diffusion [Rombach et al. 2022]) with strong generalisation capabilities. However, they are mainly trained with 2D images, which are the projection of the 3D physical world. This naturally raises the question of to what extent these large-scale vision models have learned about the 3D scene depicted with only the 2D images. Our objective in this paper is to investigate this question, and we do this precisely by determining whether features from these large-scale pre-trained vision models can be used to estimate the true geometric and physical properties of the 3D scene. If they can, then that is evidence that the models encode the 3D properties. For example, as an indication that Stable Diffusion is 3D and physics aware, Figure 4.1 shows the result of the off-the-shelf Stable Diffusion model [Rombach et al. 2022] inpainting masked regions

in real images – it correctly predicts shadows and supporting structures.

To answer this question, we propose a *general* and *lightweight* evaluation protocol to *systematically* and *efficiently* ‘probe’ a pre-trained network on its ability to represent a number of ‘properties’ of the 3D scene and viewpoint. The protocol could be used for any network and any property of interest. We have probed properties including: 3D structure and material of the scene, such as surface layout; lighting, such as object-shadow relationships; and viewpoint dependent relations such as occlusion and depth.

The protocol involves three steps: *First*, a suitable real image evaluation dataset is selected that contains ground truth annotations for the property of interest, for example the SOBA dataset [Wang et al. 2020d] is used to probe the understanding of lighting, as it has annotations for object-shadow associations. This provides a train/val/test set for that property; *Second*, a grid search is carried out over the layers and time steps of the Stable Diffusion model, and transformer layers for other models, to select the optimal feature for determining that property. The selection involves learning the weights of a simple linear classifier for that property (*e.g.* ‘are these two regions in an object-shadow relationship or not’); *Third*, the selected feature (layer, time step) and trained classifier are evaluated on a test set, and its performance answers the question of how well the model ‘understands’ that property.

In short, we probe scene geometry, material, support relation, shadow, occlusion and depth, to answer the question “To what extent do large vision models encode 3D properties of the scene?”. We apply this protocol to a wide range of networks that are trained at large scale, including OpenCLIP [Radford et al. 2021; Ilharco et al. 2021], DINOv1 [Caron et al. 2021], DINOv2 [Oquab et al. 2023], and VQGAN [Esser et al. 2021]. This covers networks trained generatively (Stable Diffusion), with self-supervision (DINOv1 & DINOv2), with weak supervision (OpenCLIP), and by auto-regression (VQGAN).

From our investigation, we make three observations: *First*, the Stable Diffusion and DINOv2 networks have a good ‘understanding’ of the scene geometry, support relations, the lighting, and the depth of a scene, with Stable Diffusion and DINOv2 having a similar and high prediction performance for these properties. However,

their prediction of material and occlusion is poorer. *Second*, Stable Diffusion and DINOv2 generally demonstrate better performance for 3D properties than other networks trained at large scale: OpenCLIP, DINOv1, and VQGAN. *Third*, different time steps of Stable Diffusion features, as well as different transformer layers of DINO/CLIP/VQGAN, perform best for different 3D physical properties.

Why is an understanding of the networks’ ability to predict 3D properties useful? There are four reasons: (1) It begins to answer the scientific question of the extent to which these networks implicitly model the 3D scene; (2) The features we determined that are able to predict 3D physical properties can be used for this task, e.g. to predict shadow-object associations or support relations. This could either be carried out directly by incorporating them in a prediction network, in the manner of [Zhan et al. 2024b]; or they can be used indirectly as a means to train a feed forward network to predict the properties [Wu et al. 2023c; Wu et al. 2023d]; (3) By knowing what properties Stable Diffusion is not good at, we have a way to spot images generated by Stable Diffusion, as has been done by [Sarkar et al. 2024]; (4) It also reveals which properties the network could be trained further on to improve its 3D modelling, *e.g.*, via extra supervision for that property.

## 4.2 Related Work

### 4.2.1 Exploration of Pre-trained Models

Building on the success of large-scale vision networks, there has been significant interest from the community to understand what has been learned by these complex models. On discriminative models, for example, [Zeiler and Fergus 2014; Mahendran and Vedaldi 2015] propose inverse reconstruction to directly visualize the acquired semantic information in various layers of a trained classification network; [Zhou et al. 2016; Fong and Vedaldi 2017; Fong et al. 2019] demonstrate that scene classification networks have remarkable localization ability despite being trained on only image-level labels; and [Erhan et al. 2009; Simonyan et al. 2014; Selvaraju et al. 2017] use the gradients of any target concept, flowing into the final convolutional layer to produce a saliency map highlighting important regions in the image for predicting the concept. In the more recent literature, [Chefer et al. 2021] explores what has been learned in the powerful transformer model by visualizing

the attention map.

On generative models, researchers have mainly investigated what has been learned in GANs, for example, GAN dissection [Bau et al. 2019] presents an analytic framework to visualize and understand GANs at the unit-, object-, and scene-level; [Wu et al. 2021] analyse the latent style space of StyleGANs [Karras et al. 2019]. The most recent work [Sarkar et al. 2024] studies the 3D geometric relations in generated images, such as vanishing points and shadows, and notes that the errors made can be used to discriminate real from generated images.

There is concurrent work [Banani et al. 2024] exploring the capability of predicting depth, surface normal and geometric correspondence for visual foundation models. To probe the depth and surface normal capability, a prediction network is trained using frozen multi-layer features from the foundation models. The capability for computing geometric correspondence between images is investigated using the dense spatial feature maps directly. In contrast to their work, we have studied a wider range of properties, covering both 3D geometric properties and 3D physical properties. Additionally, we have proposed a simple, general, yet efficient protocol for any property and any model, and have investigated the performance of different time steps and layers of models for different properties.

### 4.2.2 Exploitation of Pre-trained Models

Apart from understanding the representation in pre-trained models, there has been a recent trend for exploiting models trained at large scale, to tackle a series of downstream tasks. For example, leveraging generative models for data augmentation in recognition tasks [Jahanian et al. 2022; He et al. 2023], utilising large vision models for semantic segmentation [Baranchuk et al. 2022; Xu et al. 2023], open-vocabulary segmentation [Li et al. 2023d], depth map estimation [Shi et al. 2022; Zhao et al. 2023; Ke et al. 2024; Zhang et al. 2023a; Patni et al. 2024; Yang et al. 2024a; Yang et al. 2024b], correspondence estimation [Oquab et al. 2023; Zhang et al. 2023b; Tang et al. 2023; Luo et al. 2023; Hedlin et al. 2023] and pose estimation [Zhang et al. 2023c; Goodwin et al. 2022]. More recently, [Bhattad et al. 2023] searched for intrinsic offsets in a pre-trained StyleGAN for a range of downstream tasks, predicting normal maps, depth maps, segmentations, albedo maps, and shading.

### 4.2.3 3D Physical Scene Understanding

There have been works studying different 3D physical properties for scene understanding, including shadows [Wang et al. 2020d; Wang et al. 2021], material [Upchurch and Niu 2022], occlusion [Zhan et al. 2022], scene geometry [Liu et al. 2019a], support relations [Silberman et al. 2012] and depth [Silberman et al. 2012]. However, these works focus on one or two physical properties, and most of them require training a model for the property in a supervised manner. In contrast, we use a single model to predict multiple properties, and do not train the features.

## 4.3 Method – Properties, Datasets, and Classifiers

Our goal is to examine the ability of large-scale vision models to understand different physical and geometrical properties of the 3D scene, including: scene geometry, material, support relations, shadows, occlusion and depth. Specifically, we conduct linear probing of the features from different layers and time steps of the Stable Diffusion model, and different transformer layers for other models including OpenCLIP, DINOv1, DINOv2 and VQGAN. First, we set up the questions for each property (Section 4.3.1); and then select real image datasets with ground truth annotations for each property (Section 4.3.2). We describe how a classifier is trained to answer the questions, and the grid search for the optimal time step and layer to extract a feature for predicting the property in Section 4.3.3.

### 4.3.1 Properties and Questions

Here, we study the large vision model’s ability to predict different *properties* of the 3D scene; the properties cover the 3D structure and material, the lighting, and the viewpoint. For each property, we propose *questions* that classify the relationship between a pair of *Regions*,  $A$  and  $B$ , in the same image, based on the features extracted from the large vision model. The properties and questions are:

- *Same Plane*: ‘Are Region  $A$  and Region  $B$  on the same plane?’
- *Perpendicular Plane*: ‘Are Region  $A$  and Region  $B$  on perpendicular planes?’
- *Material*: ‘Are Region  $A$  and Region  $B$  made of the same material?’

**Table 4.1. Overview of the datasets and training/evaluation statistics for the properties investigated.** For each property, we list the image dataset used, and the number of images for the train, val, and test set. 1000 images are used for testing if the original test set is larger than 1000 images. Regions are selected in each image, and pairs of regions are used for all the probe questions.

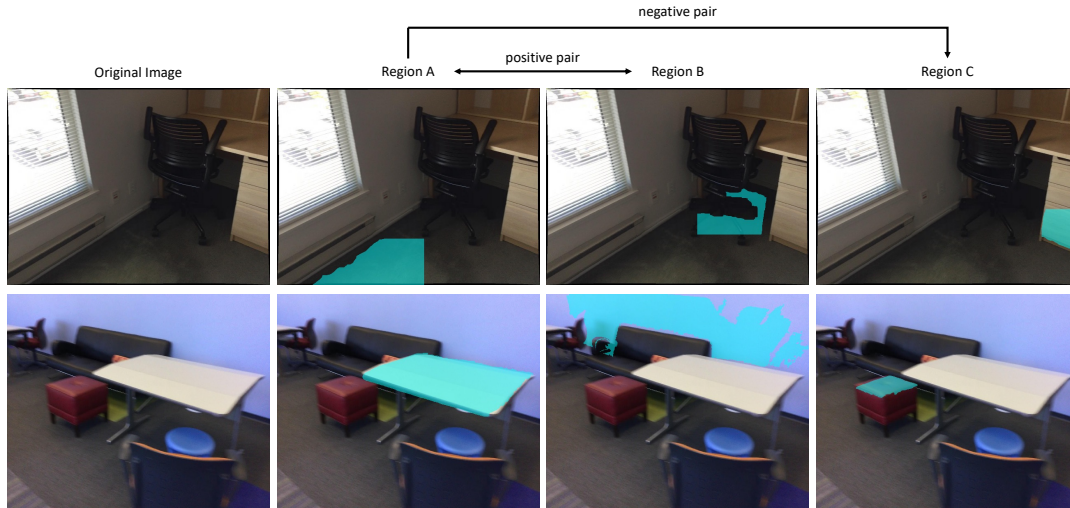
Property:	Same Plane	Perpendicular Plane	Material	Support Relation	Shadow	Occlusion	Depth
Dataset:	ScanNetv2	ScanNetv2	DMS	NYUv2	SOBA	Sep. COCO	NYUv2
Images	# Train	400	400	400	400	400	400
	# Val	100	100	100	100	100	100
	# Test	1000	1000	1000	654	160	983
Regions	# Train	7600	4493	4997	8943	3576	6799
	# Val	1844	1112	1180	1968	976	1677
	# Test	17159	10102	11364	13968	1176	16993
Pairs	# Train	14360	17530	18520	13992	7152	19238
	# Val	3498	4232	4284	2874	1952	4724
	# Test	32654	38640	41824	21768	2352	44266

- *Support Relation*: ‘Is Region  $A$  (object  $A$ ) supported by Region  $B$  (object  $B$ )?’
- *Shadow*: ‘Are Region  $A$  and Region  $B$  in an object-shadow relationship?’
- *Occlusion*: ‘Are Region  $A$  and Region  $B$  part of the same object but, separated by occlusion?’
- *Depth*: ‘Does Region  $A$  have a greater average depth than Region  $B$ ?’

We choose these properties as they exemplify important aspects of the 3D physical scene: the *Same Plane* and *Perpendicular Plane* questions probe the 3D scene geometry; the *Material* question probes what the surface is made of, *e.g.*, metal, wood, glass, or fabric, rather than its shape; the *Support Relation* question probes the physics of the forces in the 3D scene; the *Shadow* question probes the lighting of the scene; the *Occlusion* and *Depth* questions depend on the viewpoint, and probe the disentanglement of the 3D scene from its viewpoint.

### 4.3.2 Datasets

To study the different properties, we adopt various off-the-shelf real image datasets with annotations for the different properties, where the dataset used depends on the property. We repurpose each dataset to support probe questions of the form:  $\mathcal{D} = \{(R_A, R_B, y)_1, \dots, (R_A, R_B, y)_n\}$ , where  $R_A, R_B$  denote a pair of regions,



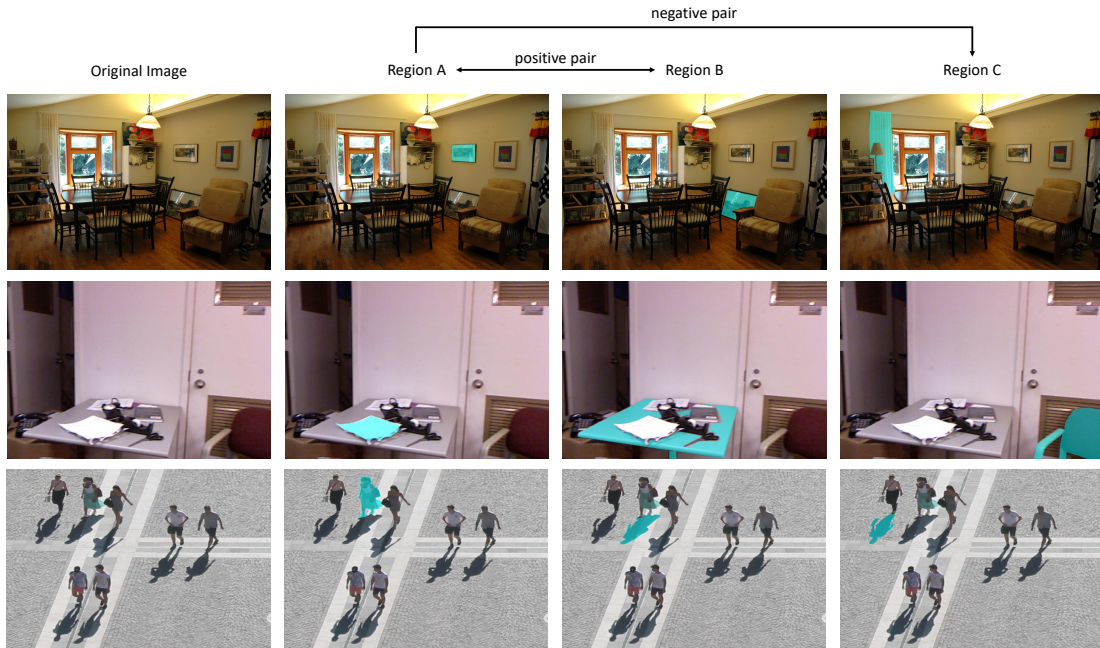
**Figure 4.2. Example images for probing *scene geometry*.** The first row shows a sample annotation for the *same plane*, and the second row is a sample annotation for *perpendicular plane*. Here, and in the following figures,  $(A, B)$  are a positive pair, while  $(A, C)$  are negative. The images are from the ScanNetv2 dataset [Dai et al. 2017] with annotations for planes from [Liu et al. 2019a]. In the first row, the first piece of floor ( $A$ ) is on the same plane as the second piece of floor ( $B$ ), but is not on the same plane as the surface of the drawers ( $C$ ). In the second row, the table top ( $A$ ) is perpendicular to the wall ( $B$ ), but is not perpendicular to the stool top ( $C$ ).

and  $y$  is the binary label indicating the answer to the considered question of the probed property. For each property, we create a train/val/test split from those of the original datasets, if all three splits are available. While for datasets with only train/test splits available, we divide the original train split into our train/val splits. Table 4.1 summarises the datasets used and the statistics of the splits used. We discuss each property and dataset in more detail next.

**Same Plane.** We use the ScanNetv2 dataset [Dai et al. 2017] with annotations of planes from [Liu et al. 2019a]. Regions are obtained via splitting plane masks into several regions. A pair of regions are *positive* if they are on the same plane, and *negative* if they are on different planes. The first row of Figure 4.2 is an example.

**Perpendicular Plane.** We use the ScanNetv2 dataset [Dai et al. 2017]. We use the annotations from [Liu et al. 2019a] which provide segmentation masks as well as plane parameters for planes in the image, so that we can obtain the normal of planes to judge whether they are perpendicular or not. A pair of regions are *positive* if they are on perpendicular planes, and *negative* if they are not on perpendicular planes. The second row of Figure 4.2 is an example.

**Material.** We adopt the recent DMS dataset [Upchurch and Niu 2022] to study the material property, which provides dense annotations of material category for



**Figure 4.3.** Example images for probing *material*, *support relation* and *shadow*. The first row is for *material*, the second row for *support relation*, and the third row for *shadow*. First row: the material images are from the DMS dataset [Upchurch and Niu 2022]. The paintings are both covered with glass (*A* and *B*) whereas the curtain (*C*) is made of fabric. Second row: the support relation images are from the NYUv2 dataset [Silberman et al. 2012]. The paper (*A*) is supported by the table (*B*), but it is not supported by the chair (*C*). Third row: the shadow images are from the SOBA dataset [Wang et al. 2020d]. The person (*A*) has the shadow (*B*), not the shadow (*C*).

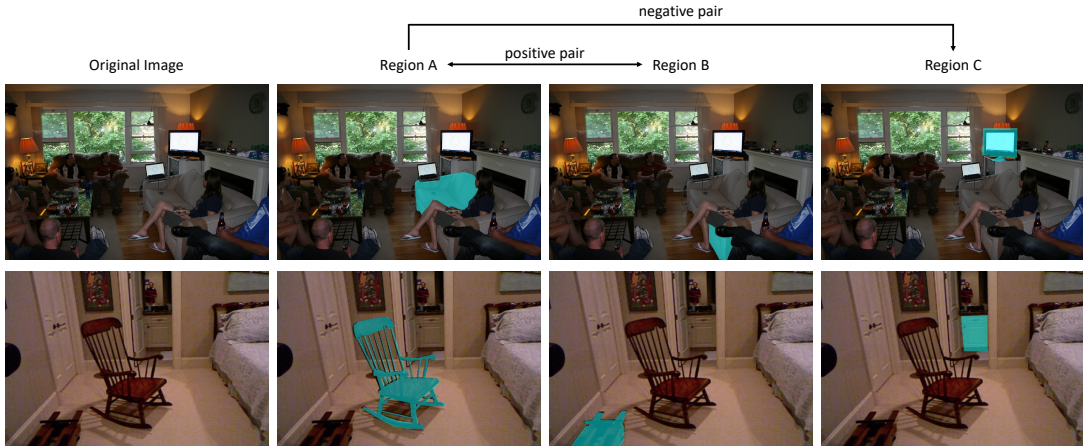
each pixel in the images. Therefore, we can get the mask of each material via grouping pixels with the same material label together. In total, there are 46 predefined material categories. Regions are obtained by splitting the mask of each material into different connected components, *i.e.*, they are simply groups with the same material labels, yet not connected. A pair of regions are *positive* if they are of the same material category, and *negative* if they are of different material categories. First row of Figure 4.3 is an example.

**Support Relation.** We use the NYUv2 dataset [Silberman et al. 2012] to probe the support relation. Segmentation annotations for different regions (objects or surfaces) are provided, as well as their support relations. Support relation here means an object is physically supported by another object, *i.e.*, the second object will undertake the force to enable the first object to stay at its position. Regions are directly obtained via the segmentation annotations. A pair of regions are *positive* if the first region is supported by the second region, and *negative* if the first region is not supported by the second region. Second row of Figure 4.3 is an

example.

**Shadow.** We use the SOBA dataset [Wang et al. 2020d; Wang et al. 2021] to study the shadows which depend on the lighting of the scene. Segmentation masks for each object and shadow, as well as their associations are provided in the dataset annotations. Regions are directly obtained from the annotated object and shadow masks. In a region pair, there is one object mask and one shadow mask. A pair of regions are *positive* if the shadow mask is the shadow of the object, and *negative* if the shadow mask is the shadow of another object. Third row of Figure 4.3 is an example.

**Occlusion.** We use the Separated COCO dataset [Zhan et al. 2022] to study the occlusion (object separation) problem. Regions are different connected components of objects (and the object mask if it is not separated), *i.e.*, groups of connected pixels belonging to the same object. A pair of regions are *positive* if they are different components of the same object separated due to occlusion, and *negative* if they are not from the same object. First row of Figure 4.4 is an example.



**Figure 4.4.** Example images for probing *viewpoint-dependent properties (occlusion & depth)*. The first row is for *occlusion* and the second row is for *depth*. First row: the occlusion images are from the Separated COCO dataset [Zhan et al. 2022]. The sofa (A) and the sofa (B) are part of the same object, whilst the monitor (C) is not part of the sofa. Second row: the depth images are from the NYUv2 dataset [Silberman et al. 2012]. The chair (A) is farther away than the object on the floor (B), but it is closer than the cupboard (C).

**Depth.** We use the NYUv2 dataset [Silberman et al. 2012], which provides mask annotations for different objects and regions, together with depth for each pixel. A pair of regions are *positive* if the first region has a greater average depth than the second region, and *negative* if the first region has a lower average depth than the second region. The average depth of a region is calculated via the average

of depth value of each pixel the region contains. Second row of Figure 4.4 is an example.

### 4.3.3 Property Probing

Take Stable Diffusion as an example, we aim to determine which features best represent different properties. To obtain features from an off-the-shelf Stable Diffusion network, we follow the approach of [Tang et al. 2023] used for DIFT, where noise is added to the input image in the latent space, and features are extracted from different layers and time steps of the model. While probing the properties, linear classifiers are used to infer the relationships between *regions*. The region representation is obtained by a simple average pooling of the diffusion features over the annotated region or object.

**Extracting Stable Diffusion Features.** We add noise  $\epsilon \sim \mathcal{N}(0, \mathbf{I})$  of time step  $t \in [0, T]$  to the input image  $x_0$ 's latent representation  $z_0$  encoded by the VAE encoder:

$$z_t = \sqrt{\alpha_t}z_0 + (\sqrt{1 - \alpha_t})\epsilon \quad (4.1)$$

and then extract features from the immediate layers of a pre-trained diffusion model,  $f_\theta(\cdot)$  after feeding  $z_t$  and  $t$  in  $f_\theta$  ( $f_\theta$  is a U-Net consisting of 4 downsampling layers and 4 upsampling layers):

$$F_{t,l} = f_{\theta_l}(z_t, t) \quad (4.2)$$

where  $f_{\theta_l}$  is the  $l$ -th U-Net layer. In this way, we can get the representation of an image  $F_{t,l}$  at time step  $t$  and  $l$ -th U-Net layer for the probe. We upsample the obtained representation to the size of the original image with bi-linear, then use the region mask to get a region-wise feature vector, by averaging the feature vectors of each pixel it contains, *i.e.*, average pooling.

$$v_{k,t,l} = \text{avgpool}(R_k \odot \text{upsample}(F_{t,l})) \quad (4.3)$$

where  $v_{k,t,l}$  is the feature vector of the  $k$ -th region  $R_k$ .  $\odot$  here is a per-pixel product of the region mask and the feature. For other models, including CLIP, DINOv1,

DINOv2 and VQGAN, we feed the image into the ViT/Transformer and extract features from different layers. Then use average pooling as in Equation 4.3 to obtain the feature for each region.

**Linear Probing.** After extracting features from large-scale vision models, we use a linear classifier (a linear SVM) to examine how well these features can be used to answer questions to each of the properties. Specifically, the input of the classifier is the difference or absolute difference between the feature vectors of Region  $A$  and Region  $B$ , *i.e.*,  $v_A - v_B$  or  $|v_A - v_B|$ , and the output is a Yes/No answer to the question. Denoting the answer to the question as  $Q$ , then since the questions about *Same Plane*, *Perpendicular Plane*, *Material*, *Shadow* and *Occlusion* are symmetric relations,  $Q(v_A, v_B) = Q(v_B, v_A)$ . However, the questions about *Support Relation* and *Depth* are not symmetric. Thus, we use  $|v_A - v_B|$  (a symmetric function) as input for the first group of questions, and  $v_A - v_B$  (non-symmetric) for the rest of questions. We train the linear classifier on the train set via the positive/negative samples of region pairs for each property; do a grid search on the validation set to find (i) the optimal time step  $t$  (only for Stable Diffusion), (ii) the U-Net layer  $l$  for Stable Diffusion and the Transformer layer  $l$  for other models, and (iii) the SVM regularization parameter  $C$ ; and evaluate the performance on the test set. The grid search is only feasible because our proposed protocol is lightweight, and can assess the effectiveness of features for different downstream tasks with minimal resource demands.

## 4.4 Experiments

In this section, we first give details of the grid search method in Section 4.4.1. We then give the linear probing grid search results on features from Stable Diffusion in Section 4.4.2 and from other networks trained at scale in Section 4.4.3. Finally, we compare all models on the test set in Section 4.4.4.

### 4.4.1 Implementation Details and Evaluation Metric

**Implementation Details.** For each property, we sample the same number of positive / negative pairs, to maintain a balanced evaluation set. In terms of the linear SVM, we tune the penalty parameter  $C$  on the val split to find the best

$C$  for each property. Therefore, we are grid searching 3 parameters on the val set, namely, Timestep  $t$  ranging from 1 to 1000 (only for Stable Diffusion), U-Net Layer  $l$  covering the 4 downsampling and 4 upsampling layers for Stable Diffusion and Transformer Layer  $l$  for other networks, and the SVM penalty parameter  $C$  ranging over 0.001, 0.01, 0.1, 1, 10, 100, 1000. The timestep is searched with a stride of 20 steps, since the difference in performance around the optimal value varies by less than 0.01 within 20 steps. In practice the  $C$  parameter is always between 0.1 and 1, so we carry out a finer search over values between 0.1 and 1.0 in steps of 0.1. The linear SVM is solved using the *libsvm* library [Chang and Lin 2011] with the SMO algorithm, to get the unique global optimal solution. Please refer to the appendix for more implementation details.

**Evaluation Metric.** All protocols are binary classification, therefore, we use ROC Area Under the Curve (AUC Score) to evaluate the performance of the linear classifier, as it is not sensitive to different decision thresholds.

#### 4.4.2 Results for Stable Diffusion



**Figure 4.5.** (a) **Nomenclature for the U-Net Layers.** We probe 4 downsampling encoder layers  $E_1$ - $E_4$  and 4 upsampling decoder layers  $D_1$ - $D_4$  of the Stable Diffusion U-Net. (b) **A prediction failure for *Material*.** In this example the model does not predict that the two regions are made of the same material (fabric). (c) **A prediction failure for *Occlusion*.** In this example the model does not predict that the two regions belong to the same object (the sofa).

**Table 4.2.** SVM grid search results of Stable Diffusion features. For each property, we train the linear SVM on the training set and grid search the best combination of time step, layer, and  $C$  on the validation set. The ROC AUC score (%) is reported on the validation set using the selected combination.

Property	Time Step	Layer	$C$	Val AUC
Same Plane	360	$D_3$	0.4	97.3
Perpendicular Plane	160	$D_3$	0.5	88.5
Material	20	$D_3$	0.5	81.5
Support Relation	120	$D_3$	1.0	92.6
Shadow	160	$D_3$	0.8	95.4
Occlusion	180	$D_3$	0.3	83.8
Depth	60	$D_3$	0.9	99.2

The results for grid search are shown in Table 4.2. For Stable Diffusion U-Net Layer,  $D_l$  means the  $l$ -th layer of the U-Net decoder, *i.e.*, upsampling layer, from outside to inside (right to left), and we provide an illustration of the layers in Figure 4.5(a).

We can make observations: First, the best time step for different properties varies, but the optimal time step is usually before 400. This is expected as a large time step adds too much noise, so not much information is contained about the image. Second, in terms of the layer, the best U-Net layer is always  $D_3$  in the decoder rather than the encoder. The optimal layer is in the middle, as  $D_1$  is too close to the noise space and  $D_4$  has just started decoding. Further explorations using Stable Diffusion features for downstream tasks could thus start from the U-Net decoder layers, especially  $D_3$ . Third, in terms of the performance on the test set, we find that Stable Diffusion can understand very well about scene geometry, support relations, shadows, and depth, but it is less performant at predicting material and occlusion. Examples of its failure are shown in Figure 4.5 (b) (c). As noted in [Zhan et al. 2022] and [Kirillov et al. 2023], grouping all separated parts of an object due to occlusion remains challenging even for state-of-the-art detection and segmentation models. The appendix gives grid search results at all time steps and layers.

### 4.4.3 Results for CLIP/DINO/VQGAN Features

In this section we show grid search results for OpenCLIP [Radford et al. 2021; Ilharco et al. 2021] pre-trained on LAION dataset [Schuhmann et al. 2022], DINOv1 [Caron et al. 2021] pre-trained on ImageNet dataset [Deng et al. 2009], DINOv2 [Oquab et al. 2023] pre-trained on LVD-142M dataset [Oquab et al. 2023], and VQGAN [Esser et al. 2021] pre-trained on ImageNet dataset [Deng et al. 2009]. We use the best pre-trained checkpoints available on their official GitHub – ViT-B for DINOv1, ViT-G for OpenCLIP and DINOv2, and the 48-layer transformer checkpoint for VQGAN. Similar to Stable Diffusion, for each of these models, we conduct a grid search on the validation set in terms of the ViT/Transformer layer and  $C$  for SVM, and use the best combination of parameters for evaluation on the test set.

Grid search results are reported in Table 4.3. It can be observed that different

**Table 4.3. SVM grid search results of CLIP/DINO/VQGAN features.** We train the linear SVM on the training set, and grid search the best combination of ViT/-Transformer layer and  $C$  on the validation set. The OpenCLIP and VQGAN models we use have 48 transformer layers, DINOv1 has 12 layers and DINOv2 has 40 layers. The  $i$ -th layer means the  $i$ -th transformer layer from the input side.

	Same Plane				Perpendicular Plane			
	OpenCLIP	DINOv1	DINOv2	VQGAN	OpenCLIP	DINOv1	DINOv2	VQGAN
Optimal Layer	27	8	24	12	27	9	22	12
Optimal C	0.7	0.7	0.8	1.0	1.0	0.2	0.6	0.6
Val AUC	94.5	93.2	96.0	82.6	72.9	70.9	84.9	62.8

	Material				Support Relation			
	OpenCLIP	DINOv1	DINOv2	VQGAN	OpenCLIP	DINOv1	DINOv2	VQGAN
Optimal Layer	30	8	23	11	32	9	40	14
Optimal C	0.3	0.2	0.6	0.3	0.3	0.3	0.6	0.4
Val AUC	77.5	77.4	81.3	65.8	92.0	91.5	93.6	85.4

	Shadow				Occlusion			
	OpenCLIP	DINOv1	DINOv2	VQGAN	OpenCLIP	DINOv1	DINOv2	VQGAN
Optimal Layer	28	2	29	8	31	3	29	2
Optimal C	1.0	0.8	1.0	1.0	0.2	0.2	0.3	1.0
Val AUC	94.6	92.4	96.6	88.7	80.6	77.0	84.4	77.4

	Depth			
	OpenCLIP	DINOv1	DINOv2	VQGAN
Optimal Layer	32	7	30	45
Optimal C	0.1	0.4	1.0	0.5
Val AUC	99.2	97.4	99.6	93.7

layers of different models are good at different properties.

#### 4.4.4 Comparison of Different Features Trained at Scale

We compare the state-of-the-art pre-trained large-scale vision models' representations on various downstream tasks in Table 4.4. It can be observed that the Stable Diffusion and DINOv2 representations outperform OpenCLIP, DINOv1 and VQGAN for all properties, indicating the potential of utilizing Stable Diffusion and DINOv2 representations for different downstream tasks with the optimal time steps and layers found in Section 4.4.2 and Section 4.4.3.

**Table 4.4. Comparison of different features trained at scale.** For each property, we use the best time step, layer and  $C$  found in the grid search for Stable Diffusion, and the best layer and  $C$  found in the grid search for other features. The performance is the ROC AUC on the test set, and ‘Random’ means a random classifier.

Property	Random	OpenCLIP	DINOv1	DINOv2	VQGAN	Stable Diffusion
Same Plane	50	92.3	91.4	94.5	81.3	<b>96.3</b>
Perpendicular Plane	50	71.8	71.3	82.1	62.0	<b>86.0</b>
Material	50	78.7	78.8	83.5	65.5	<b>83.6</b>
Support Relation	50	90.6	90.8	<b>92.8</b>	84.1	92.1
Shadow	50	94.9	92.2	<b>95.8</b>	89.0	95.4
Occlusion	50	81.2	79.9	84.4	78.4	<b>84.8</b>
Depth	50	99.2	97.1	<b>99.7</b>	91.8	99.6

## 4.5 Discussion and Future Work

In this paper, we have developed a general and lightweight protocol to efficiently examine whether models pre-trained on large scale image datasets, like CLIP, DINO, VQGAN and Stable Diffusion, have explicit feature representations for different properties of the 3D physical scene.

It is interesting to find that different time steps of Stable Diffusion and different layers of DINOv2 representations can handle several different physical properties at a state-of-the-art performance, indicating the potential of utilising the Stable Diffusion and DINOv2 models for different downstream tasks.

However, for some properties such as material and occlusion, the networks are not distilling the information in a manner that can be used by a linear probe. This could indicate that these properties are not modelled well by the network or that more than a linear probe is required to tease them out. We show examples of the prediction failures for these properties in Figure 4.5. In the appendix, we show that such prediction failures also occur in generated (i.e. synthetic) images. It is worth noting that occlusion is a challenge even for the powerful Segment Anything Model (SAM) [Kirillov et al. 2023], where the model ‘hallucinates small disconnected components at times’.

In the appendix, we provide preliminary results of using the probed Stable Diffusion feature for downstream tasks. We also provide examples of another use case of spotting Stable Diffusion generated images based on the properties that the model is not good at .

This paper has given some insight into answering the question: ‘To what extent

do large vision models understand the 3D scene' for real images. Of course, there are more properties that could be investigated in the manner proposed here. For example, contact relations [Fouhey et al. 2016] and object orientation [Xiang et al. 2018], as well as the more nuanced non-symmetric formulations of the questions. Please refer to arxiv version of the paper for future updates.

## Appendices

Appendices and future updates for this chapter can be found in the online version of the paper.<sup>1</sup>

## Statement of Authorship

A statement of authorship for this paper is provided in Appendix [A](#).

---

<sup>1</sup><https://arxiv.org/pdf/2310.06836>

## Chapter 5

# Inferring Dynamic Physical Properties from Video Foundation Models

The paper was accepted at the VGBE Workshop of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2026.

# Inferring Dynamic Physical Properties from Video Foundation Models

Guanqi Zhan<sup>1\*</sup>, Xianzheng Ma<sup>1\*</sup>, Weidi Xie<sup>1,2</sup>, Andrew Zisserman<sup>1</sup>

<sup>1</sup>VGG, University of Oxford      <sup>2</sup>Shanghai Jiao Tong University

## Abstract

We study the task of predicting dynamic physical properties from videos. More specifically, we consider physical properties that require *temporal* information to be inferred: elasticity of a bouncing object, viscosity of a flowing liquid, and dynamic friction of an object sliding on a surface. To this end, we make the following contributions: (i) We collect a new video dataset for each physical property, consisting of synthetic training and testing splits, as well as a real split for real world evaluation. (ii) We explore three ways to infer the physical property from videos: (a) an oracle method where we supply the visual cues that intrinsically reflect the property using classical computer vision techniques; (b) a simple read out mechanism using a visual prompt and trainable prompt vector for cross-attention on pre-trained video generative and self-supervised models; and (c) prompt strategies for Multi-modal Large Language Models (MLLMs). (iii) We show that a video foundation model trained in a generative (DynamiCrafter) or trained in a self-supervised manner (V-JEPA-2) achieve a generally similar performance, though behind that of the oracle, and that MLLMs are currently inferior to the other models, though their performance can be improved through suitable prompting. The dataset, model, and code are available at <https://www.robots.ox.ac.uk/~vgg/research/idpp/>.

---

\*Equal contribution.

## 5.1 Introduction

Humans are remarkably adept at intuitively estimating physical properties from visual observations. Without direct interaction, people can often estimate how bouncy a ball is, how thick a liquid seems, or how slippery a surface might be—simply by watching how objects move. While these estimations are not precise in a scientific sense, they are sufficiently accurate for guiding perception, prediction, and action. Bringing this capability to machines is an important step towards building more general and physically grounded artificial intelligence. In particular, visual systems that can infer dynamic physical properties from raw video could enhance robotic manipulation, embodied agents, and video understanding tasks in ways that go beyond the traditional perception tasks of recognition, detection, and segmentation.

Recent progress in video foundation models, including generative models [Xing et al. 2024; Liu et al. 2024a], self-supervised models [Bardes et al. 2023; Assran et al. 2025] and multi-modal large language models (MLLMs) [Hui et al. 2024; Comanici et al. 2025; Hurst et al. 2024], have shown impressive capability in synthesizing realistic dynamics, learning general-purpose video representations, and tackling semantic understanding tasks, for example, video question answering. However, a question that remains underexplored is: **do these models acquire an understanding of *dynamic* physical properties from videos?**

In this paper, we address this question by focusing on several representative physical properties that are not directly observable in static frames but instead emerge through temporal dynamics: the elasticity of a bouncing object, the viscosity of a flowing liquid, and the dynamic friction between a surface and a sliding object. These properties are especially compelling because their inference requires temporal reasoning and sensitivity to subtle visual cues—such as deformation, deceleration, spreading, or oscillation. By examining how well current video foundation models capture these dynamic attributes, we aim to assess their physical understanding beyond static appearance.

To support this investigation, we introduce a new dataset, *PhysVid*, specifically designed to evaluate the dynamic physical properties from video. Existing datasets lack ground-truth annotations for such properties, so we construct *PhysVid* using a

combination of synthetic videos—rendered via a physics simulator—and real-world videos sourced from the internet or captured in-house. Each video is annotated with physical property values, either derived from simulation parameters or estimated manually. The dataset is designed to facilitate the study of out-of-domain generalization, both within the synthetic domain and from synthetic to real-world data. To establish an upper bound on what is inferable from visual input alone, we implement an oracle method for each property. These oracles leverage privileged access to the visual cues that directly reflect the corresponding property.

We evaluate three categories of video foundation models: generative models, self-supervised models, and multi-modal large language models (MLLMs). For the generative and self-supervised models, we propose a simple yet effective read-out mechanism that extracts dynamic physical properties from pre-trained, frozen representations. Our method introduces a learnable query vector that attends to internal representation tokens via cross-attention, enabling the selective extraction of relevant information. This approach is both lightweight and training-efficient. More specifically, we have studied DynamiCrafter [Xing et al. 2024] for a video generative model, and V-JEPA-2 [Assran et al. 2025] for a video self-supervised model. For MLLMs, we explore various prompting strategies to elicit predictions of dynamic physical properties directly from video input. These strategies include few-shot prompting to provide task context, as well as procedural prompting that guides the model through the oracle estimation steps—helping it focus on the intrinsic visual cues that reveal the target properties. The MLLMs we have studied include QwenVL[Hui et al. 2024], GPT [Hurst et al. 2024], and Gemini [Comanici et al. 2025].

## 5.2 Related Work

**Physics Prediction from Images and Videos.** Inferring physical properties from visual observations remains a core challenge in computer vision. Early methods estimate latent physical parameters (e.g., mass, friction, stiffness) via differentiable physics engines or learning-based simulators [Wu et al. 2015; Ding et al. 2021; Jatavallabhula et al. 2021; Li et al. 2020; Wang et al. 2020c; Wang et al. 2018a], while later works infer salient attributes like viscosity or elasticity

from task-specific visual cues [Kawabe et al. 2014; Paulun et al. 2015; Assen et al. 2018; Norman et al. 2007; Kawabe and Nishida 2016; Paulun et al. 2017; Paulun and Fleming 2020], yet both rely heavily on simulation supervision, domain priors, or handcrafted heuristics. More recently, unsupervised learning of intuitive physics has emerged via next-frame prediction from large-scale everyday physical scenes [Voleti et al. 2022; Lu et al. 2023; Agrawal et al. 2016; Finn and Levine 2017; Babaeizadeh et al. 2021; Hafner et al. 2019; Fragkiadaki et al. 2016; Garcia et al. 2025], capturing latent dynamics without explicit physical supervision. However, the resulting representations are usually implicit and lack interpretability in terms of concrete physical quantities. In contrast, we infer physical properties by directly prompting pre-trained video foundation models, enabling explicit estimation without reliance on task-specific heuristics, or end-to-end prediction pipelines from scratch.

**Physics Datasets and Benchmarks.** An increasing number of physics-related datasets have been collected in recent years to provide ground truth annotations for different physical properties, including material [Sharma et al. 2023; Gao et al. 2024], shadow [Wang et al. 2020d; Wang et al. 2021], support relations [Silberman et al. 2012], occlusion [Zhan et al. 2022; Zhan et al. 2024b], mass and volume [Wu et al. 2016]. Another line of work [Chow et al. 2025; Shen et al. 2025; Riochet et al. 2022; Bordes et al. 2025; Tung et al. 2023; Bear et al. 2021] proposes broad benchmarks with video-image-text QA tasks to assess physical understanding in vision-language models, but the questions are typically qualitative and categorical. More recently, [Zhang et al. 2025b] introduces a benchmark consisting of 130 real-world videos capturing physical phenomena guided by conservation laws, to evaluate the physics plausibility of video generative models by assessing the trajectory of objects in their generated videos. In contrast, our datasets consist of both *synthetic* and *real-world* videos annotated with the *quantitative value* for the associated physical parameter of the coefficient of friction, elasticity, and viscosity.

### 5.3 Problem Scenario and The *Phys Vid* Datasets

In this paper, we address the problem of estimating physical properties from videos. Specifically, we focus on three properties: **elasticity** of a bouncing object, **viscos-**

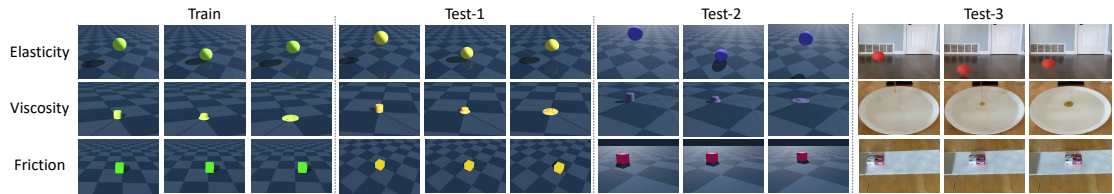
ity of a flowing liquid, and the **dynamic friction coefficient** between a surface and a sliding object. Given a video  $v \in \mathbb{R}^{T \times H \times W \times 3}$ , we consider two formulations, the first is **absolute value prediction**, where the input is a single video and the model is tasked with predicting the numerical value of the physical property, *i.e.*,  $y_{\text{abs}} = \Phi(v; \theta_1)$ . The second is **relative value comparison**, where the input is a pair of videos captured from the same viewpoint, and the model must determine whether the first video exhibits a higher physical property value than the second, *i.e.*,  $y_{\text{rel}} = \Phi(v_1, v_2; \theta_2)$ , and  $y_{\text{rel}}$  is binary.

Each scenario is parameterized by a set of variables, including the value of the target *physical property* (*e.g.*, elasticity, viscosity, or friction), and a set of *nuisance parameters* (including camera viewpoint, object appearance, lighting, *etc.*). While the model must be sensitive to changes in the physical property, it should be robust (ideally invariant) to variations in nuisance parameters.

To assess generalization, we define two domains of nuisance parameters, denoted as  $\mathcal{A}_1$  and  $\mathcal{A}_2$ , which differ in their distributions. For instance,  $\mathcal{A}_2$  may have different camera viewpoints or different lighting conditions to  $\mathcal{A}_1$  (full details of these differences are given in Appendix). We generate a dataset using a physics-based simulator, consisting of one training split and two test splits. The models are only trained on the training split from the simulator for all the evaluations. The training and **test-1** splits are sampled from  $\mathcal{A}_1$ , sharing the same nuisance distribution; **test-2** is drawn from  $\mathcal{A}_2$ , introducing a distribution shift. The target property values are sampled from a shared range across all splits to ensure consistency. Finally, **test-3** consists of real-world videos, used to evaluate generalization beyond simulation.

### 5.3.1 The *Phys Vid* Datasets

To study the dynamic physical properties of elasticity, viscosity, and friction, we construct a dataset for each, containing both synthetic and real-world videos. Synthetic ones are generated with the Genesis simulator [Zhou et al. 2024], and real ones are captured with an iPhone in slow-motion mode or downloaded from the Internet. For each property we have: 10,000 videos for **train**; 1000 videos for each of **test-1** and **test-2**; and 100 videos for **test-3**. Sample frames are shown in Figure 5.1. In the following we describe how each property is realized in the



**Figure 5.1. Examples of the *PhysVid* dataset.** Each row shows a different property, and each column shows three frames from video samples in the synthetic sets (**train**, **test-1**, and **test-2**) and the real **test-3** set. The **train** and **test-1** sets are from the same distribution. In **test-2** parameters, such as lighting, viewpoint and color, differ from those in **test-1**.

video. Please refer to Appendix for more details of the datasets.

## Elasticity

We study an object’s elasticity by analyzing the motion of a ball dropped onto the ground and its subsequent bounces. In physics, elasticity  $e$  is quantified as the ratio of the rebound velocity  $v_{\text{after impact}}$  to the impact velocity  $v_{\text{before impact}}$ , and also equals  $\sqrt{h_{\text{bounce}}/h_{\text{drop}}}$ , where  $h_{\text{drop}}$  is the dropping height and  $h_{\text{bounce}}$  is the bouncing height. Here and for the following properties, please refer to Appendix for the detailed derivations. These expressions are used for the oracle estimation in Section 5.4.1.

**Synthetic Dataset.** All synthetic videos are generated using Genesis [Zhou et al. 2024], with object’s elasticity as the target property. Nuisance factors include drop height, camera viewpoint, object appearance, and lighting conditions. The object is of the same size in all videos. Note, here and for the following properties, the ground truth property value is obtained directly from the simulator.

**Real-World Dataset.** The real-world videos are collected from YouTube using the search term “ball bouncing experiments”. Each clip is manually trimmed to include the drop-and-bounce sequence of a single ball. The dataset includes a wide range of materials (*e.g.*, rubber balls, tennis balls, basketballs, balloons, *etc*), resulting in diverse elasticity values. The ground truth elasticity values for the real sequences are estimated by computing  $\sqrt{h_{\text{bounce}}/h_{\text{drop}}}$ : the videos are chosen such that the balls bounce in a fronto-parallel plane, which means that ratios of image heights (differences in  $y$ -coordinates) are approximately equal to the ratio of heights in 3D. These image differences are obtained by manual annotation.

## Viscosity

We study the viscosity by observing a liquid column dropping and spreading on the ground. The viscosity can be reflected by the growth rate of the liquid area on the ground. The viscosity  $\mu$  is negatively correlated to the liquid area growth rate  $\frac{d(A(t))}{dt}$ , given the controlled liquid density  $\rho$ , controlled liquid column diameter  $D$ , and controlled dropping velocity  $v$  of the liquid column when it reaches the ground.

**Synthetic Dataset.** The synthetic videos are generated using Genesis [Zhou et al. 2024], where the target property is the viscosity of liquid. Nuisance factors include camera viewpoint, object appearance, and lighting conditions. The liquid column is of the same size in all videos.

**Real-World Dataset.** Since it is challenging to find real-world videos online that provide ground-truth viscosity values while controlling for other relevant physical parameters—such as  $\rho$ ,  $D$  and  $v$ , we collected real videos under controlled conditions. We use a funnel with a fixed nozzle diameter to produce a consistent liquid column. A funnel holder allows us to fix the height from which the liquid is poured, thereby controlling the initial velocity  $v$ . Ground-truth viscosity values for each liquid are obtained from standard physics reference tables. The selected liquids span a wide range of viscosities, from 1.2 (*e.g.*, coffee) to 225 (*e.g.*, maple syrup), allowing for a diverse and comprehensive evaluation.

## Friction

We study friction between an object and a surface by observing how the object slows down as it slides with an initial velocity. The dynamic friction coefficient  $\mu_k$  is proportional to the (negative) acceleration of the object  $a$ .

**Synthetic Dataset.** The synthetic videos are generated using Genesis [Zhou et al. 2024], where the target property is the dynamic friction coefficient at the contacting surface of the object and the ground. Nuisance factors include initial location and initial velocity of the object, camera viewpoint, object appearance, and lighting conditions. The object is of the same size in all videos.

**Real-World Dataset.** While many online videos depict objects sliding on surfaces, they lack ground-truth annotations for friction coefficients. We therefore collect a real video dataset featuring 5 different objects and 6 surface materials,

spanning a wide range of dynamic friction values. Each object is given an initial velocity by sliding it down from a slope and it then slides on a horizontal plane. To obtain ground-truth friction coefficients, we use a spring dynamometer to measure the friction force  $F$  for each object-surface pair (by dragging the object at constant speed), and record the object’s weight  $G$ . The dynamic friction coefficient is then computed as:  $\mu_k = F/G$ .

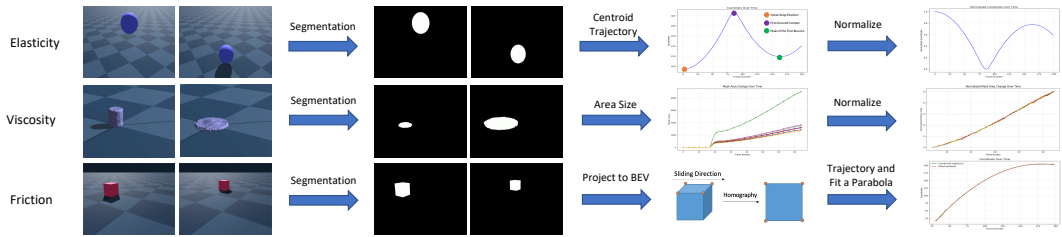
## 5.4 Inferring Physical Properties

This section presents the three different ways for inferring dynamic physical properties: an oracle method via classical computer vision techniques (Section 5.4.1); a visual prompt mechanism for video generative and self-supervised models (Section 5.4.2); and prompts for MLLMs (Section 5.4.3).

### 5.4.1 Oracle Estimation

#### Elasticity

We aim to estimate elasticity from both synthetic and real-world videos. The key visual cue is the relative height of the ball during its drop and subsequent bounce, observed in 3D. As noted earlier, the ratio in 3D can be approximated from their corresponding image-space measurements. This approximation is exact when the motion occurs in a fronto-parallel plane, and remains reasonably accurate otherwise—since the ratio of lengths between parallel line segments is invariant under affine transformations [Hartley and Zisserman 2004]. Given that perspective effects are minimal in our videos, the affine approximation provides a reliable estimate for elasticity. To automate this process, we extract the ball’s trajectory  $y(t)$  from the video and input the sequence of ratios into a GRU network to regress the elasticity. In detail, we segment the ball in each frame and use their centroids as the  $y$ -coordinate. From this trajectory, we identify key points: the initial drop position, the first ground contact, and the peak of the first bounce. The resulting trajectory is normalized to the range  $[0, 1]$ , by subtracting the  $y$ -coordinate of the first ground contact and dividing by the initial drop height. This normalization not only ensures invariance to viewpoint and scale, but also simplifies learning for the GRU by standardizing the input distribution. We train a GRU, as it is noisy to



**Figure 5.2. Oracle methods for physical properties.** The objective in each case is to extract a measurement from the sequence that can directly be used to predict the property. For elasticity, we extract the centroid trajectory from segmentation masks, and then normalize the  $y$ -coordinates into 0-1; the ratio of bouncing to dropping height over the sequence indicates the elasticity. For viscosity, we calculate the area size in the image via segmentation masks, and then normalize the area sizes by the area in the frame when the liquid first touches the ground; the slope of the normalized area size sequence reflects the viscosity. For friction, we transform to a bird’s eye view (using a homography transformation based on 4 corner points of the top surface of the sliding object), and fit a parabola  $x = at^2 + \beta t + c$  to the transformed trajectory; the parabola coefficient  $\alpha$  predicts the friction coefficient. For each video, we show the segmentation for two frames (left  $\rightarrow$  right).

directly obtain  $h_{\text{drop}}$  and  $h_{\text{bounce}}$  using heuristics (*e.g.*, determining the maximum and minimum points), and in practice a GRU provides a good estimate. The full pipeline is illustrated in Figure 5.2 (top row). For the **absolute prediction**, the normalized trajectory is fed into a GRU network, which directly regresses the elasticity value. For the **relative comparison**, the binary decision score between two videos  $v_1$  and  $v_2$  is calculated as:

$$\text{score} = \sigma\left(\log\left(\frac{e_1}{e_2}\right)\right), \quad (5.1)$$

where  $e_1$  and  $e_2$  are the estimated elasticities based on height ratios, and  $\sigma(\cdot)$  denotes the sigmoid function.

## Viscosity

The key visual cue for estimating viscosity is the rate at which the liquid spreads on the ground-plane, measured as an area ratio normalized by the initial area of the liquid column. As with elasticity, we approximate perspective using an affine transformation – here of the ground-plane. Since area ratios are invariant under affine transformations [Hartley and Zisserman 2004], the liquid’s normalized image-space area growth approximates its true normalized ground-plane expansion (in our setup the liquid spreads only within a limited area around the release point, and the camera is distant; consequently an affine viewing approximation

is adequate). Specifically, we extract segmentation masks for each frame and compute the liquid’s area over time. This area sequence is normalized by the area in the first frame where the liquid contacts the surface, ensuring invariance to viewpoint and scale. The process is illustrated in Figure 5.2 (middle row). For **absolute prediction**, we calculate the slope  $k$  of  $A(t)$  and use  $1/k$  to represent the viscosity value; For **relative comparison**, the binary decision score between two videos  $v_1$  and  $v_2$  is calculated as in Equation 5.1, where  $e_1$  and  $e_2$  are the estimated viscosities based on area growth rate.

### Friction

The key visual cue for estimating dynamic friction is the acceleration of the sliding object—*i.e.*, how quickly its velocity decreases due to friction—which can be inferred from its position over time. Since the object moves significantly in the video, we do not use an affine approximation, but instead take account of the projective geometry by mapping the object’s motion to a bird’s-eye view, allowing for consistent trajectory analysis. This is achieved by estimating a homography between the image and bird’s eye view (normal to the plane) from the four corners of the object’s top surface (see Figure 5.2, bottom row). We fit a parabola  $x = \alpha t^2 + \beta t + c$  to the transformed top surface trajectory to estimate the acceleration  $a$  from the coefficient  $\alpha$ , and the coefficient of friction  $\mu_k = 2\alpha/g$ . For **absolute prediction**, we use the estimated  $\mu_k$  to represent the friction coefficient value; For **relative comparison**, the binary decision score between two videos  $v_1$  and  $v_2$  is calculated as in Equation 5.1, where  $e_1$  and  $e_2$  are the estimated friction coefficients based on the transformed object trajectory.

## 5.4.2 Video Generative and Self-Supervised Models

### Video Feature Extraction

Given a video  $v \in \mathbb{R}^{T \times H \times W \times 3}$ , we extract features with a pre-trained video backbone, that can either be generative or self-supervised, resulting into spatiotemporal feature representations, *i.e.*,  $r = \psi(v) \in \mathbb{R}^{t \times h \times w \times c}$ , which can be detailed as follows.

**Generative Model as Backbone.** We adopt a pre-trained video diffusion

model (Figure 5.3, left), namely DynamiCrafter [Xing et al. 2024], to compute the visual features. Specifically, given an input video, we add noise to the latent representations after the pre-trained VAE encoder, and replace the text prompt with a learnable embedding. We extract multi-scale features from all U-Net layers at diffusion time step 50, which was shown to be effective for capturing 3D physics in prior work [Tang et al. 2023; Zhan et al. 2024a]. To aggregate the features, we introduce a learnable query vector  $q$ , which is first mapped to the different dimensions of the multi-scale features (see Appendix for details), and then attends to the diffusion tokens ( $r_i$ ) via cross-attention:

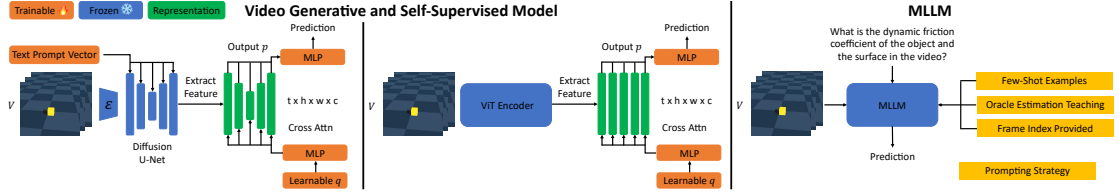
$$p = \sum_{i=1}^{t \times h \times w} \text{softmax}(q \cdot r_i) \cdot r_i \quad (5.2)$$

The resulting vectors  $p$  from different layers are then mapped by another MLP network to a common dimension and average pooled to generate the final video feature representation  $P$ . To predict the physical properties, we train the text token of the generative model, together with the ‘visual prompt’ architecture that includes the query  $q$  and the MLPs.

**Self-Supervised Model as Backbone.** Here, we adopt a pre-trained self-supervised model (Figure 5.3, middle), namely V-JEPA-2 [Assran et al. 2025], as the visual backbone. The input video is passed through the model, and we extract feature tokens from all layers of the ViT encoder. Similar to the generative setting, we introduce a learnable query vector  $q$  to extract the video feature representation  $P$  from the ViT tokens via attentive pooling. Although the feature dimension at each ViT layer is the same, we still use an MLP network to map  $q$  to generate the query vector of each layer (keeping it similar to the generative setting in terms of MLP network architecture), and use another MLP network to map the output vectors  $p$  to a same dimension as the generative setting before average pooling them to get  $P$ . Please see Appendix for more details.

### Physical Property Prediction

Given the computed feature  $P$  from video foundation models, we train an MLP network to predict the physical properties using the synthetic video dataset training split. The network for each property is trained separately.



**Figure 5.3. Architectures for dynamic physical property prediction.** **Left:** video generative model as backbone; **Middle:** video self-supervised model as backbone; **Right:** multimodal large language model (MLLM). For the pre-trained video diffusion model (U-Net, left) and the pre-trained self-supervised model (ViT, middle), the representations are kept frozen, and a ‘visual prompt’ learns to infer the physical properties. For the MLLMs, the physical properties are inferred using a language prompt (right).

**Absolute Value Prediction.** Given the resulting video feature ( $P$ ), we pass it through an MLP network  $\gamma$  to predict the absolute value  $\chi$  of the physical property:

$$\chi = \gamma(P) \quad (5.3)$$

For elasticity and friction, the absolute value prediction is supervised with L1 loss with the ground truth value; For viscosity, as the ground truth values may have very different scales, *i.e.*, from  $1e^{-5}$  to  $1e^{-2}$ , the absolute value prediction is trained with Log L1 loss, which calculates L1 loss between the log of the predicted value and the log of the ground truth value.

**Relative Value Prediction.** Given the resulting features for a pair of videos,  $P_1$  and  $P_2$ , we concatenate them and formulate a binary classification problem, indicating which video has a larger physical property value via an MLP network  $\gamma$ :

$$\xi = \gamma([P_1, P_2]) \quad (5.4)$$

The binary prediction for all three tasks is trained with binary cross entropy loss with the binary ground truth.

### 5.4.3 Multimodal Large Language Models

This section studies off-the-shelf multimodal large language models (MLLMs) for understanding dynamic physical properties from video. We explore various prompting strategies on state-of-the-art MLLMs, including Qwen2.5-VL-Max [Hui et al. 2024], GPT-4o [Hurst et al. 2024], and Gemini 2.5 Pro [Comanici et al. 2025], as illustrated in Figure 5.3 (right). Examples of the prompting strategies are provided in Appendix.

## Preliminary

The MLLM receives video frames as visual input. The text prompt includes (1) a brief description of the target property—for example: “we are studying the viscosity of the liquid, where water is 1.0 and honey is 5000.0.” This is followed by (2) a query, such as: “what is the viscosity value of the liquid in the video?” (absolute) or “which video shows a liquid with higher viscosity? please output a decision score between 0 and 1, indicating the likelihood that the first video exhibits a higher property value.” (relative). All the following prompt strategies provide (1) and (2) by default.

## Baseline Prompt

For *relative* tasks, we specify that the first  $n$  frames belong to the first video and the last  $n$  to the second. For *absolute* tasks, the default prompt is used. Appendix Figure 8 and Figure 13 provide an example of *baseline prompt* for the absolute formulation and the relative formulation, respectively.

## Oracle Estimation Teaching

For both *relative* and *absolute* settings, we provide the key cue to concentrate on from the Section 5.4.1 description to teach the MLLM how to estimate the properties step by step. Appendix Figure 9 and Figure 14 provide an example of *oracle estimation teaching* for the absolute formulation and the relative formulation, respectively.

## Few-Shot Examples

For both *relative* and *absolute* settings, we provide several examples, including the video input and desired ground truth. For fair comparison with visual prompting, we use examples in the synthetic training split. Appendix Figure 10 and Figure 15 provide an example of *few-shot examples* for the absolute formulation and the relative formulation, respectively.

## Frame Index Provided

For both *relative* and *absolute* settings, we input the text of the index of each frame along with the frames. In this way the MLLMs may have a better understanding

about the temporal relations between the input video frames. Appendix Figure 11 and Figure 16 provide an example of *frame index provided* for the absolute formulation and the relative formulation, respectively.

### **Black Frames in Between**

This strategy is only used for the *relative* setting. We insert black frames between the two video segments to clearly separate them. In the prompt, we refer to the videos as the frames before and after the black frames, rather than as the first and last  $n$  frames. Appendix Figure 17 provides an example of *black frames in between* for the relative formulation.

## **5.5 Experiments**

**Implementation Details.** During oracle estimation, we train the GRU network with a learning rate of  $1e^{-3}$  and the batch size is 128. For the generative and self-supervised video models, the backbones are frozen, the trainable parameters are optimised with a learning rate of  $1e^{-5}$  and the batch size 16. For MLLMs, we perform prompt selection, and use the best strategy that we find for each of the absolute and relative settings for the experiments. *Few-shot examples* and *oracle estimation teaching* work best for the absolute and relative settings, respectively, as they directly provide the model with more context information about the properties. Please refer to Appendix for the comparison results and analysis. All models are trained on H100/A6000/A40 GPUs. Please refer to Appendix for more implementation details.

**Evaluation Metrics.** For *relative value comparison*, we report the ROC AUC score; for *absolute value prediction*, we use the Pearson Correlation Coefficient between the prediction and ground truth as this automatically calibrates the predictions to the scale of the ground truth. Please refer to Appendix for more details and motivations on the evaluation metrics.

### **5.5.1 Results for Relative Value Comparison**

Table 5.1 (left) shows relative value comparison results across physical properties and model types. The oracle estimator performs nearly perfectly on `test-1`

and `test-2`, and strongly on `test-3`, indicating that the task is largely solvable using visual cues, geometry, and physics. Both the evaluated generative (Dynam-iCrafter) and self-supervised (V-JEPA-2) video models achieve strong results on synthetic splits (`test-1` and `test-2`). Notably, they can also generalize well to the real-world split (`test-3`) for most scenarios of viscosity and elasticity, which rely on simple height ratios and expansion. However, friction proves more challenging for V-JEPA-2. Trained on synthetic data, it struggles to generalize, likely due to the fact that reliance on visual references (*e.g.*, ground plane grids) is absent in real videos, and due to friction’s inherent complexity involving higher-order motion and projective geometry of the viewpoint. To further confirm, we introduce an additional real-world training split for friction videos with disjoint objects and surfaces from the test set (see Appendix for more details). Fine-tuning the visual prompting architecture on this data improves performance on the real test split, as shown by the \* values in Table 5.1. Multimodal large language models (MLLMs), though not working very well with *Baseline Prompt* (see Appendix), when prompted properly, also perform well, especially on real videos, which are more *in-distribution* for them – while on synthetic splits, their performance drops significantly. This is likely due to the fact that the models tend to leverage semantic cues, *e.g.*, the type of liquid or the category of object and surface, rather than visual motion.

### 5.5.2 Results for Absolute Value Prediction

Table 5.1 (right) shows results for absolute value prediction across physical properties and methods. This task is more challenging than relative comparison, as models must regress quantitative physical values rather than compare video pairs from the same viewpoint. Similar to the relative setting, the oracle estimator achieves near-perfect performance on `test-1` and `test-2`, and strong performance on `test-3`, confirming that the task is largely solvable through visual cues, multi-view geometry, and physical laws. We highlight several key observations: (i) **comparable performance across backbones**. Despite being trained for generative tasks, Dynam-iCrafter performs on par with V-JEPA-2 when predicting dynamic physical properties. (ii) **friction remains challenging**. Similar to the relative setting, both Dynam-iCrafter and V-JEPA-2 struggle with friction estimation. Per-

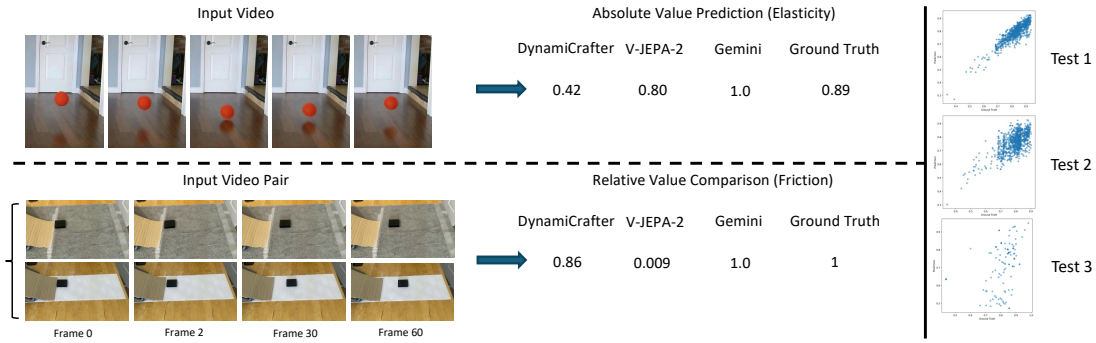
**Table 5.1. Results for relative value comparison and absolute value prediction.** Left: ROC AUC scores for relative comparisons (range  $[0, 1]$ ). Right: Pearson correlation coefficients for absolute predictions (range  $[-1, 1]$ ). \* indicates results after domain adaptation using a disjoint real training set. **test-1** is the synthetic in-distribution test split; **test-2** is the synthetic out-of-distribution test split; **test-3** is the real-world test split.

Property	Method	Relative – ROC AUC			Absolute – Pearson Corr.		
		Test-1	Test-2	Test-3	Test-1	Test-2	Test-3
Elasticity	<b>Oracle</b>	1.00	0.99	1.00	0.99	0.98	0.87
	DynamiCrafter	1.00	0.99	0.61	0.92	0.78	0.10
	V-JEPA-2	1.00	1.00	0.99	0.89	0.65	0.37
	Qwen2.5VL-max	0.59	0.50	0.54	-0.05	0.11	0.16
	GPT-4o	0.51	0.66	0.62	0.19	0.11	0.30
	Gemini-2.5-pro	0.64	0.80	0.47	0.04	0.15	0.24
Viscosity	<b>Oracle</b>	0.99	0.99	1.00	0.99	0.98	0.80
	DynamiCrafter	1.00	1.00	0.93	0.99	0.95	0.44
	V-JEPA-2	1.00	0.94	0.99	0.98	0.69	0.61
	Qwen2.5VL-max	0.64	0.61	0.86	0.16	0.06	0.02
	GPT-4o	0.63	0.59	0.99	0.18	0.08	0.55
	Gemini-2.5-pro	0.48	0.69	0.95	-0.06	-0.05	0.60
Friction	<b>Oracle</b>	1.00	1.00	0.87	0.99	1.00	0.83
	DynamiCrafter	0.96	0.90	0.97	0.95	0.78	0.25
	+ Domain Adaptation	–	–	–	–	–	0.80*
	V-JEPA-2	0.97	0.90	0.48	0.87	0.56	0.21
	+ Domain Adaptation	–	–	0.87*	–	–	0.73*
	Qwen2.5VL-max	0.52	0.48	0.80	-0.03	0.03	0.06
GPT-4o	0.49	0.40	0.67	0.07	0.10	0.38	
Gemini-2.5-pro	0.56	0.52	0.97	0.01	0.10	0.12	

formance again improves with domain adaptation. (iii) **MLLMs better on real test split than synthetic**. MLLMs continue to perform better on the real test split than synthetic test splits, benefiting from their familiarity with real-world visual semantics. (iv) **greater gap from oracle**. The performance gap between video foundation models and the oracle is more pronounced here than in the relative setting, indicating that accurate physical value regression remains a significant challenge for current video models.

### 5.5.3 Qualitative Results

Figure 5.4 (left) shows qualitative examples comparing model predictions across different tasks. In the **first row**, we illustrate an example from the elasticity absolute value prediction task. The video generative model (DynamiCrafter), self-supervised model (V-JEPA-2), and MLLM (Gemini) predict values of 0.42, 0.80, and 1.0, respectively—all reasonably close to the ground-truth value of 0.89. In the **second row**, we present a friction relative value comparison task. The input consists of two videos, where the first exhibits a higher dynamic friction coefficient than the second. Both the video generative model and the MLLM correctly assign high likelihoods to this relationship (0.86 and 1.0, respectively), aligning with



**Figure 5.4. Qualitative results.** **Top Left:** An example for elasticity absolute value prediction; **Bottom Left:** An example for friction relative value comparison. For each example, the original input video is shown on the left. Model predictions are shown on the right, including results from the Video Generative Model (DynamiCrafter), Video Self-Supervised Model (V-JEPA-2), and a MLLM (Gemini). For the relative formulation, the ground truth value of ‘1’ indicates that the first (top) video has larger dynamic friction coefficient than the second video. In this example, the initial velocity of the lego brick in the two videos is similar (note the same displacement from frame 0 to 2), but the velocity reduces to 0 at frame 30 in the first video, while the object is still moving in frame 30 to 60 in the second video. **Right:** Scatter plots of prediction vs ground truth for the elasticity property from the V-JEPA-2 model.

the ground truth. In contrast, the self-supervised model incorrectly predicts the reverse and does so with high confidence. Figure 5.4 (right) shows examples of the scatter plots for the absolute value prediction. More specifically, we show the scatter plots of video self-supervised model on the three test splits. It can be observed that the performance degrades from `test-1` to `test-3`, as `test-1` is of the same distribution as the synthetic training split, while `test-2` is out-of-distribution synthetic test and `test-3` is for real evaluation. We provide more scatter plots in Appendix.

## 5.6 Conclusion

We investigate the task of inferring dynamic physical properties—elasticity, viscosity, and friction—from videos. To support this, we introduce a benchmark dataset with ground-truth annotations and evaluate representative video foundation models under both absolute prediction and relative comparison settings. We adopt a simple architecture to extract physical cues from off-the-shelf generative and self-supervised video models, and explore prompting strategies to elicit predictions from MLLMs. Experiments show the evaluated generative and self-supervised models have generally similar and reasonable performance, though this evaluation should be extended to further models in the future. MLLMs perform worse overall

but improve with more informative prompting, especially on real-world data. The worse performance of MLLMs is consistent with previous work [Fu et al. 2025], where it is observed that visual information is not properly fused in the language model. However, all evaluated models fall short of the oracle, particularly in absolute value prediction. These results highlight the need to enhance physical reasoning in video models—a key direction for future research.

## Appendices

Appendices and future updates for this chapter can be found in the online version of the paper.<sup>1</sup>

## Statement of Authorship

A statement of authorship for this paper is provided in Appendix [A](#).

---

<sup>1</sup><https://arxiv.org/pdf/2510.02311>

## Part III

# Improving Visual-Language Foundation Models

## Chapter 6

# **ELIP: Enhanced Visual-Language Foundation Models for Image Retrieval**

The paper was published at the IEEE International Conference on Content-Based  
Multimedia Indexing (CBMI), 2025.

# ELIP: Enhanced Visual-Language Foundation Models for Image Retrieval

Guanqi Zhan<sup>1\*</sup>, Yuanpei Liu<sup>2\*</sup>, Kai Han<sup>2</sup>,

Weidi Xie<sup>1,3</sup>, Andrew Zisserman<sup>1</sup>

<sup>1</sup>VGG, University of Oxford

<sup>2</sup>The University of Hong Kong

<sup>3</sup>Shanghai Jiao Tong University

## Abstract

The objective in this paper is to improve the performance of text-to-image retrieval. To this end, we introduce a new framework that can boost the performance of large-scale pre-trained vision-language models, so that they can be used for text-to-image re-ranking. The approach, **Enhanced Language-Image Pre-training (ELIP)**, uses the text query, via a simple MLP mapping network, to predict a set of visual prompts to condition the ViT image encoding. ELIP can easily be applied to the commonly used CLIP, SigLIP and BLIP-2 networks. On the evaluation side, we set up two new out-of-distribution (OOD) benchmarks, *Occluded COCO* and *ImageNet-R*, to assess the zero-shot generalisation of the models to different domains. The results demonstrate that ELIP *significantly* boosts CLIP/SigLIP/SigLIP-2 text-to-image retrieval performance and outperforms BLIP-2 on several benchmarks, as well as providing an easy means to adapt to OOD datasets.

---

\*Equal contribution.

## 6.1 Introduction

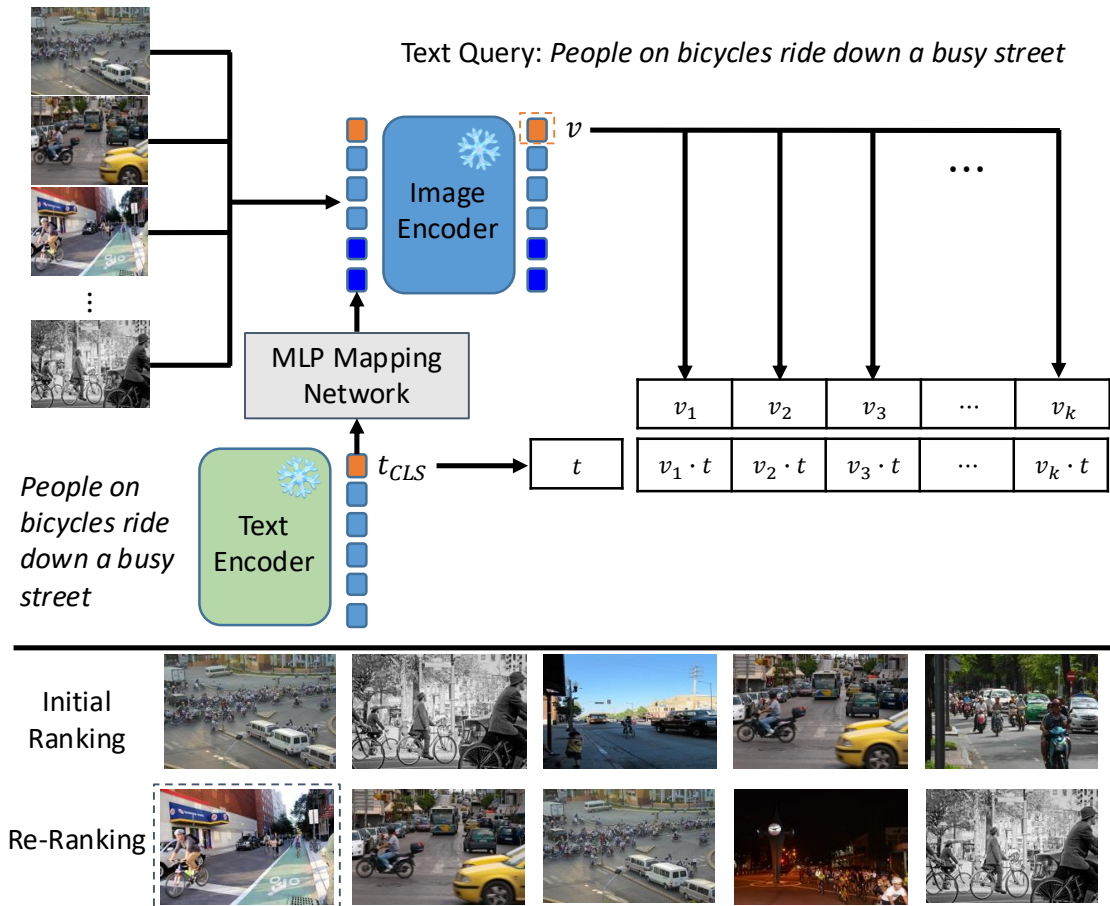
This paper considers the problem of text-to-image retrieval, that aims to rank image instances based on their relevance to a text query. Effective retrieval generally includes two stages: the first stage provides an initial ranking in a fast and efficient manner, while the second *re-ranking* stage refines this ranking by re-computing the relevance scores between the text query and each of the top-ranked candidates with a more expensive model.

Recent advances in text-to-image retrieval have primarily focused on the first stage. Notable models, such as CLIP [Radford et al. 2021] and ALIGN [Jia et al. 2021], leverage contrastive learning [Oord et al. 2018] on large-scale image-text pairs to learn joint representations, demonstrating impressive generalization capabilities for cross-modal retrieval tasks.

Our primary contribution here focuses on the second stage of the retrieval pipeline, namely, the re-ranking. Specifically, our goal is to enhance the performance of off-the-shelf vision-language foundation models, so that they can be re-purposed for re-ranking the top- $k$  candidates from the fast retrieval process. The approach we develop, termed *Enhanced Language-Image Pre-training (ELIP)*, requires only a few trainable parameters, and the training can be conducted efficiently with ‘student-friendly’ resources and data. We demonstrate that ELIP can boost the performance of the pre-trained CLIP [Radford et al. 2021], SigLIP [Zhai et al. 2023], SigLIP-2 [Tschannen et al. 2025], and BLIP-2 [Li et al. 2023a] for cross-modal retrieval.

To achieve this goal, we first introduce a lightweight, text-guided visual prompting module. As illustrated in Figure 6.1, a query text is mapped to a set of visual prompt vectors [Jia et al. 2022], that are then concatenated with the [CLS] and patch embeddings of the image encoder. These augmented embeddings are then passed into the frozen vision encoder to recompute the image representation. The resulting image embedding is aware of the text conditioning and this enhances its performance in re-ranking.

To assess the re-ranking performance of our proposed ELIP models, we experiment on the standard COCO [Lin et al. 2014] and Flickr30k [Plummer et al. 2015] text-to-image retrieval benchmarks. As a further challenge, we also evaluate the gener-



**Figure 6.1. The ELIP architecture.** *Top:* We propose a novel architecture that can be applied to pre-trained and frozen vision-language foundation models, such as CLIP, SigLIP, SigLIP-2 and BLIP-2, to enhance their text-to-image retrieval performance. The *key idea* is to use the text query to define a set of visual prompt vectors that are incorporated into the image encoder to make it aware of the query when generating the embedding. An MLP maps from the text space to the visual space of the input to the ViT encoder. The architecture is lightweight, and our data curation strategies enable efficient and effective training with limited resources. *Bottom:* In this retrieval example from the COCO benchmark, the top- $k$  ( $k=100$ ) images are re-ranked by the ELIP model for the text query: ‘People on bicycles ride down a busy street’. The ground truth image matching the query is not in the top-5 ranked images in the initial CLIP ranking, but is ranked top-1 (highlighted in the dashed box) by the re-ranking.

alisation of the ELIP-boosted models on out-of-distribution domains. To do so, we repurpose the Occluded COCO [Lee and Park 2022] and ImageNet-R [Hendrycks et al. 2021] datasets to be used for text-to-image retrieval benchmarks.

In summary, we make three contributions: *First*, we propose a novel architecture to improve text-based image retrieval on large pre-trained vision-language models, including the most popular CLIP/SigLIP architectures and the state-of-the-art BLIP-2 architecture. *Second*, to evaluate the generalisation capability of text-to-image retrieval models to different out-of-distribution domains, we set up two new benchmarks of text-to-image retrieval, *Occluded COCO* and *ImageNet-R*. *Third*, and most significantly, we demonstrate that ELIP *substantially* improves the image retrieval performance of CLIP and SigLIP architectures, and outperforms the state-of-the-art BLIP-2 architecture. Furthermore, it provides an efficient method to adapt these architectures to OOD datasets, again giving a tremendous boost with CLIP, SigLIP, SigLIP-2 and BLIP-2. As an additional contribution, we show that the model can be trained efficiently with limited computing resources, and develop a ‘student-friendly’ *best practice*, involving global hard sample mining and batch selection. This training is detailed in the arXiv version of this paper.

## 6.2 Related Work

**Text-to-Image Retrieval** is a fundamental and much researched task in cross-modal learning [Lee et al. 2018; Chen et al. 2020a; Zhang et al. 2022b; Radford et al. 2021; Li et al. 2021; Chen et al. 2020c; Wang et al. 2023a; Yu et al. 2022; Chen et al. 2024; Chen et al. 2021; Chen et al. 2023; Chun et al. 2021; Diao et al. 2021; Engilberge et al. 2018; Gu et al. 2018; Huang et al. 2017; Ji et al. 2019; Karpathy et al. 2014; Kim et al. 2023; Li et al. 2019; Liu et al. 2019b; Liu et al. 2020; Song and Soleymani 2019; Thomas and Kovashka 2020; Wang et al. 2020b; Wang et al. 2018b; Wang et al. 2019; Wang et al. 2023b; Wei et al. 2020a; Wei et al. 2020b; Yan et al. 2021; Zeng et al. 2022; Zhang et al. 2022a; Zhang et al. 2020; Zhang and Lu 2018; Zheng et al. 2019; Vendrow et al. 2024; Kordopatis-Zilos et al. 2025]. Large vision language models, such as CLIP [Radford et al. 2021; Ilharco et al. 2021], ALIGN [Jia et al. 2021], BLIP-2 [Li et al. 2023a], SigLIP [Zhai et al. 2023] and SigLIP-2 [Tschannen et al. 2025] that have powerful zero-shot capabilities

have now become the de facto method for open-set text-based image retrieval. The most recent work [Sogi et al. 2024] gives a slight improvement over BLIP-2 by incorporating the output of an object detector or annotations of detection bounding boxes. This succeeds in overcoming the failure cases where small but semantically important objects in an image are not properly understood by the model. We compare to this model and show superior performance.

**CIR and Universal Retrieval.** In composed image retrieval (CIR) [Liu et al. 2023c; Gu et al. 2023; Baldrati et al. 2023; Ventura et al. 2024], the query is specified by a composition of an image and text, with the text specifying how the image should be changed. For example, the query image may be of a dog lying down, and the query text may be ‘playing with a ball’. This composed query defines the target image to be retrieved from the gallery. This differs from our task, where the query is specified only by text, and the text alone defines the target image to be retrieved from the gallery. A more general setting is ‘universal retrieval’ [Wei et al. 2024; Liu et al. 2025a] where the query can be a combination of image, text, and instruction; and the target can be image alone, text alone, or image and text.

**Post-Retrieval Re-ranking.** For single modality image retrieval, where the query is an image, there has been a series of works that have re-ranked the top- $k$  images from an initial ranking via classical computer vision algorithms, such as ‘query expansion’, ‘geometric verification’, or a combination of the two [Jegou et al. 2008; Philbin et al. 2007; Chum et al. 2011; Chum et al. 2007; Tolias and Jégou 2014; Arandjelović and Zisserman 2012], as well as via learning-based algorithms [Cao et al. 2020; Hausler et al. 2021; Tan et al. 2021; Bhalgat et al. 2023]. Re-ranking algorithms have been relatively less explored in text-to-image retrieval [Yanagi et al. 2019; Qu et al. 2023; Long et al. 2024]. [Miech et al. 2021] introduced a method for computing the similarity score between an image and a text query by estimating the log-likelihood of the text conditioned on the image. While this approach has demonstrated strong performance, it remains computationally expensive both during training and inference, making it a *slow* process. Our paper also focuses on the re-ranking stage – developing a more powerful version of visual-language foundation models to give a better ranking of images that are hard to distinguish by the original retrieval model.

**Multi-Modal Datasets.** To obtain multi-modal foundation models with a strong capability of generalisation, it is important to train them on large-scale multi-modal datasets. Therefore, in recent years, there has been a significant increase in the number and scale of multimodal vision-language datasets that provide image-text pairs, such as COCO [Lin et al. 2014], SBU [Ordonez et al. 2011], Conceptual Captions [Sharma et al. 2018], LAION [Schuhmann et al. 2022], DataComp [Gadre et al. 2024]. The increase in the size of multi-modal datasets enables the training of more powerful visual-language foundation models. More recently, DataCompDR [Vasu et al. 2024] utilises prior knowledge from large-scale pre-trained image captioning models to generate synthetic captions for DataComp images, resulting in less noisy captions than the datasets collected from the web, such as the original DataComp dataset. In this paper, we have experimented with training the model using Conceptual Captions [Sharma et al. 2018] and DataCompDR [Vasu et al. 2024].

## 6.3 Preliminaries

**Re-Ranking in Image Retrieval.** Given an input query, the goal of a retrieval system is to rank all instances in a dataset  $\Omega = \{I_1, \dots, I_n\}$ , based on their relevance to the query. In the case of text-to-image retrieval, the query is specified by text ( $T$ ), and the ideal outcome is a set ( $\hat{\Omega}$ ), with the relevant images being ranked higher than those that are not. In general, an effective retrieval system proceeds in two stages: the first stage provides an initial ranking in a fast and efficient manner, while the second stage—referred to as re-ranking—refines this ranking by recomputing the relevance scores between the text query and each of the top- $k$  ranked candidates with a more powerful (and usually more expensive) ranking model. The  $k$  is selected such that in general there is a high recall for all the relevant images. In this paper, our novelty lies in the second stage, that aims to re-rank the top- $k$  candidates from the first stage results.

**Visual Prompt Tuning (VPT)** [Jia et al. 2022] is a method of enhancing the ViT image encoder by inserting additional learnable prompts into the transformer layers. It enables efficient adaptation of ViT, requiring only the few parameters of the learnable prompts to be trained. VPT has two different variants – *VPT-*

*Shallow* and *VPT-Deep*. *VPT-Shallow* only inserts the additional visual prompts into the first Transformer layer, whereas for *VPT-Deep*, prompts are introduced at every transformer layer’s input space. We insert our generated set of visual prompt vectors into the first transformer layer of ViT, which is similar to *VPT-Shallow*.

## 6.4 The ELIP Architecture

In this section, we describe the ELIP text-to-visual prompt mapping network, that can be efficiently applied to adapt the commonly used CLIP/SigLIP architectures as well as the more sophisticated BLIP-2 architectures for re-ranking. We first introduce the architecture of the network in Section 6.4.1, and the training/inference strategy in Sections 6.4.2 and 6.4.3 respectively. We refer to the network applied to CLIP as *ELIP-C*, applied to SigLIP/SigLIP-2 as *ELIP-S/ELIP-S-2*, and applied to BLIP-2 as *ELIP-B*.

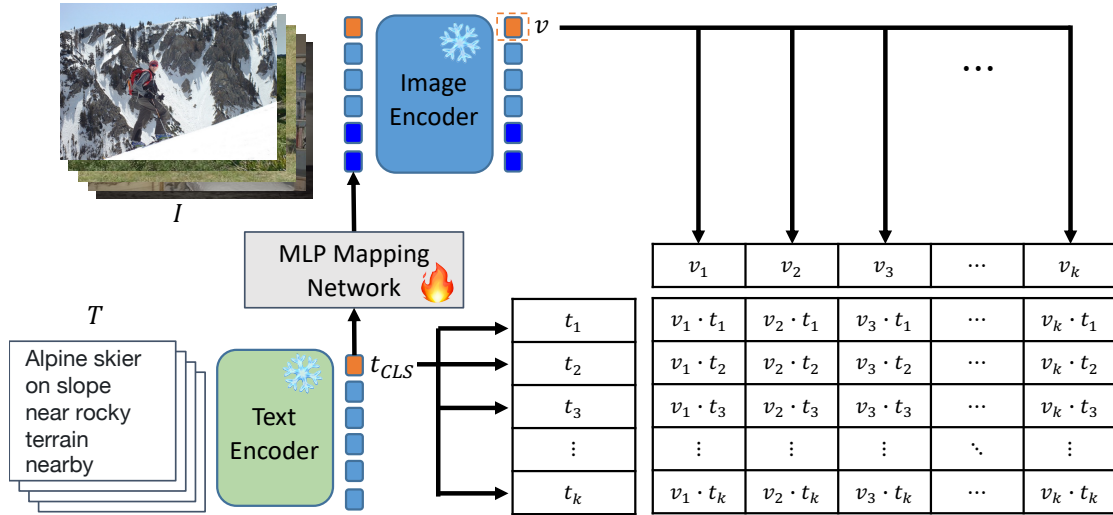
### 6.4.1 Text-Guided MLP Mapping Network

Here, we propose a mapping network that projects the embedding of the text query into a set of prompt vectors within the visual embedding space. This set of prompt vectors is then incorporated as additional tokens into the first layer of the Vision Transformer (ViT) image encoder, used to re-compute the visual embeddings:

$$[t_p^1, \dots, t_p^m, t_{\text{CLS}}] = \Phi_t(T)$$

$$v = \Phi_v([x_p^1, \dots, x_p^n, x_{\text{CLS}}; \psi_{\text{map}}(t_{\text{CLS}})])$$

where  $T$  denotes the query text, which is first encoded with a pre-trained, frozen text encoder ( $\Phi_t(\cdot)$ ) into  $m + 1$  embeddings. The [CLS] token is further fed into a **trainable** mapping network to generate the prompt vectors, which are concatenated with the  $n + 1$  image embeddings ( $[x_p^1, \dots, x_p^n, x_{\text{CLS}}]$ ), and passed into the pre-trained, frozen visual encoder ( $\Phi_v(\cdot)$ ). The MLP Mapping Network consists of 3 layers of linear layers with a GELU between every two linear layers. We expand the output dimension to be  $n$  times when we generate  $n$  tokens and then divide the generated vector into  $n$  tokens. The ELIP architecture is shown in Figure 6.2 and Figure 6.3.

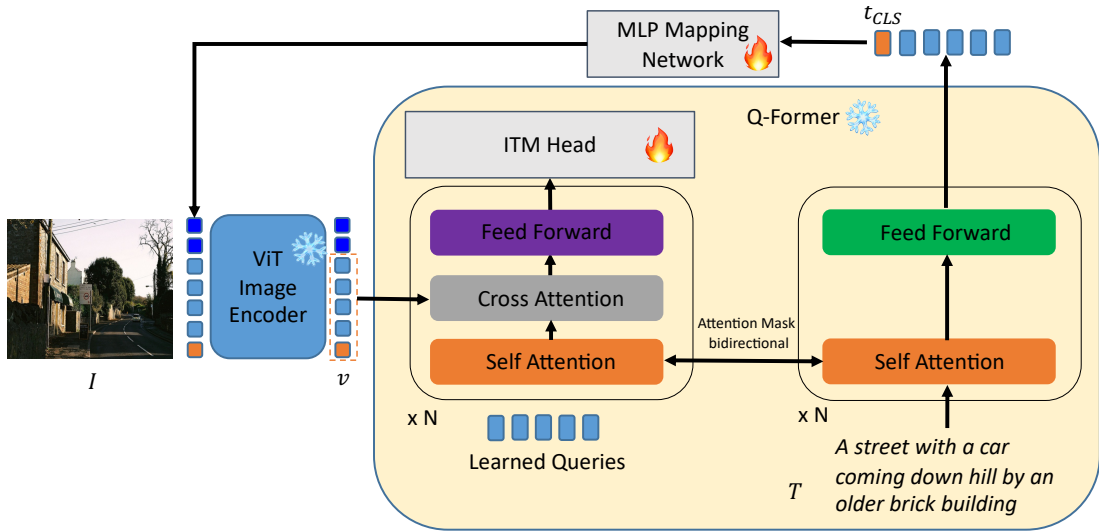


**Figure 6.2.** Architecture of ELIP-C / ELIP-S. At training time, a batch of text-image pairs is fed into the architecture. The text feature is mapped to the visual embedding space as a set of prompt vectors via the MLP mapping network and then guides the encoding of the image feature. We use color coding for the [CLS] token, patch tokens, and generated visual tokens from text. The architecture is trained with InfoNCE loss (for ELIP-C) and Sigmoid loss (for ELIP-S/ELIP-S-2), to align the text feature with the corresponding re-computed image feature.

## 6.4.2 Training and Testing ELIP-C/ELIP-S

**Text-Guided Contrastive Training.** At training time, we compute the dot product between the [CLS] token embedding of the text query ( $t_{CLS}$ ) and the re-computed image features guided by the query text, *i.e.*,  $\{v_1, \dots, v_b\}$  ( $b$  denotes the batch size). For ELIP-C, we train with the standard InfoNCE loss on the batches; For ELIP-S/ELIP-S-2, we train with pairwise Sigmoid loss. In the arXiv version of this paper, we provide more details on the batch selection scheme via global hard sample mining.

**Re-Ranking at Inference Time.** At inference time, for each text query, we first compute the similarity scores between the visual-language embedding, computed by the original CLIP/SigLIP model, to obtain an initial ranking of all images. We then select the top- $k$  candidates for further re-ranking, where the visual features are re-computed by incorporating the prompted vectors from the mapping network. The final ranking is obtained via the dot product of the re-computed image features and the text feature.



**Figure 6.3. Architecture of ELIP-B.** Similar to the architecture on CLIP/SigLIP, the *MLP Mapping Network* maps the text feature to the visual embedding space. The only difference is that the text-guided image features are further fed into the Q-Former to cross-attend the input text and then passed through the Image-Text Matching (ITM) Head to predict whether the image and text match or not. As the input image features to the ITM head have been changed, we also fine-tune the ITM head, which is a lightweight MLP network. The network is fed pairs of text and positive/negative image features at training time and is trained with binary cross entropy loss.

### 6.4.3 Training and Testing ELIP-B

Figure 6.3 illustrates the application of our architecture on BLIP-2. The only difference with that described for CLIP-type models is that BLIP-2 re-ranking does not use a dual encoder; rather, the image and text encoders attend to each other. However, the purpose of our mapping network and its training are essentially unchanged.

**Text-Guided Image-Text Matching Loss.** At training time, we feed the text query ( $T$ ) and the re-computed image features with the query text as prompts, *i.e.*,  $\{v_+, v_-\}$  ( $v_+$  denotes the positive image and  $v_-$  denotes the negative image), into the Q-Former, and then to an Image-Text Matching (ITM) Head to predict a score indicating whether the text and image match or not. The output of the ITM head is trained with binary cross entropy loss.

**Inference Time Re-Ranking.** For each text query, we first compute the similarity scores between the visual-language embedding, computed by the original BLIP-2 image and text encoders, to obtain an initial ranking of all images. We then select the top- $k$  candidates for further re-ranking, where the visual features are re-computed by incorporating the prompted vectors from the mapping network.

The final ranking is obtained via the sum of the initially computed similarity score and the score predicted by the ITM head based on the re-computed image features and text query.

## 6.5 Evaluation Datasets

Here we describe the details of the datasets we use for evaluation.

### 6.5.1 Standard Benchmarks

**COCO** [Lin et al. 2014], is a large-scale dataset for studying object detection, segmentation, and captioning. In terms of captioning, each image is annotated with 5 different captions. Previous works use the test split of 5,000 images and 25,010 captions for the evaluation of text-to-image retrieval.

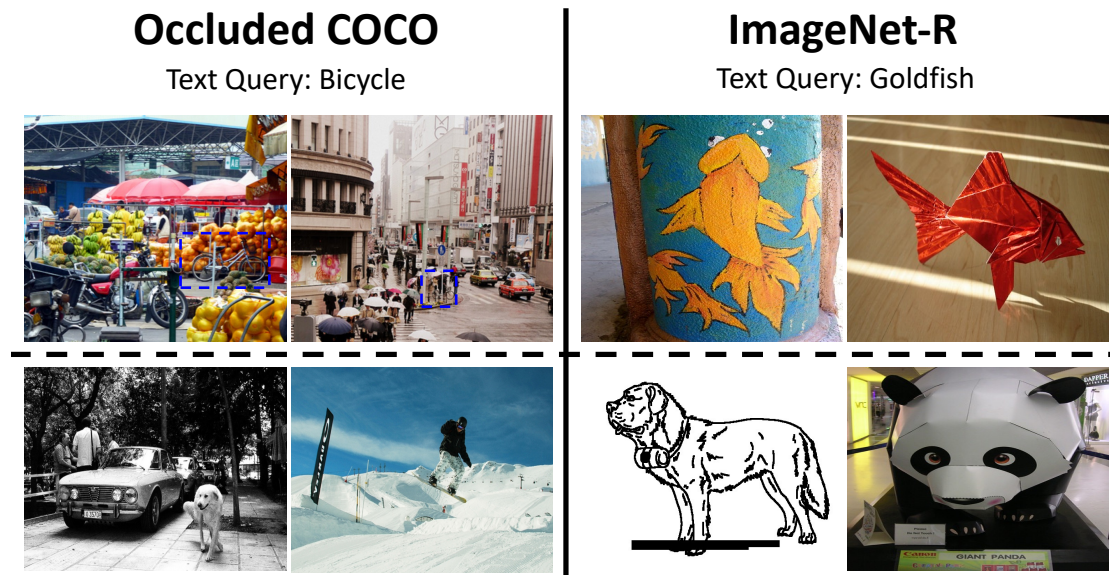
**The Flickr30k Dataset** [Plummer et al. 2015] contains images collected from Flickr, together with 5 reference sentences provided by human annotators. The test set for text-to-image retrieval consists of 1,000 images and 5,000 captions.

**Evaluation Metrics.** We adopt the standard metrics for assessing retrieval performance, namely, Recall@1, Recall@5 and Recall@10. Recall@k denotes the proportion of relevant images that are successfully retrieved within the top- $k$  results for each text query.

### 6.5.2 Out-of-Distribution Benchmarks

To evaluate a model’s capability for text-to-image retrieval in out-of-distribution (OOD) scenarios, we set up two new benchmarks for text-based image retrieval. Figure 6.4 shows examples from the Occluded COCO and ImageNet-R benchmarks.

**Occluded COCO** is curated with annotations from [Lee and Park 2022], with the method as described in [Zhan et al. 2022], where the occlusion relationship is utilised to collect images containing occluded objects. This dataset aims to evaluate the model’s performance on retrieving images with occluded target objects against images that do not contain the target object. It has 80 text queries and 5,000 images.



**Figure 6.4.** Examples of the out-of-distribution benchmarks. Occluded COCO is on the left, and ImageNet-R is on the right. For both benchmarks, the positive images contain the object described by the text query while the negative images do not contain the object. We display positive images in the first row and negative images in the second row. For Occluded COCO, the target object in the image is occluded, making it more difficult to be retrieved. For example, for the text query *Bicycle* in Occluded COCO, positive images have an occluded bicycle (highlighted in dashed box) while negative images do not have a bicycle in it; for the text query *Goldfish* in ImageNet-R, positive images have goldfish while negative images do not have goldfish.

**ImageNet-R** is generated using annotations from [Hendrycks et al. 2021] and aims to examine the model’s performance for retrieval across various domains, for example, art, cartoons, deviantart, graffiti, embroidery, graphics, origami, paintings, patterns, plastic objects, plush objects, sculptures, sketches, tattoos, toys, and video games. It has 200 text queries and 30,000 images.

**Evaluation Metrics.** Here, we use mAP as the evaluation metric. This is because there might be multiple positive images for each text query.

## 6.6 Experiment

**Training the Model.** The recent visual-language foundation models are often trained on massive numbers (billions) of paired image-caption samples, with considerable computing resources. Here, we explore a ‘resource efficient training’ *best practice* for data curation that enables improving large-scale visual-language models with limited resources. Specifically, there are two major challenges to be addressed: (i) training with a large batch size is challenging, due to limitation on GPU memory; (ii) training on billions of samples is prohibitively expensive on

**Table 6.1. Ablation study on ELIP-C** for choice of training dataset, hard sample mining, and number of prompt vectors generated.

Setting	Architecture	Training Dataset	Hard Sample Mining	Multiple Prompts	COCO			Flickr				
					R@1	R@5	R@10	Avg.	R@1	R@5	R@10	Avg.
A	CLIP	-			40.2	66.0	75.6	60.6	67.6	88.3	93.0	83.0
B	ELIP-C	CC3M [Sharma et al. 2018]			40.7	66.2	76.1	61.0	68.8	88.9	93.8	83.8
C	ELIP-C	CC3M [Sharma et al. 2018]	✓		41.8	67.5	77.5	62.3	69.5	89.7	94.1	84.4
D	ELIP-C	DataCompDR [Vasu et al. 2024]	✓		44.2	70.0	79.5	64.6	71.3	90.6	94.4	85.4
E	ELIP-C	DataCompDR [Vasu et al. 2024]	✓	✓	<b>45.6</b>	<b>71.1</b>	<b>80.4</b>	<b>65.7</b>	<b>72.3</b>	<b>90.6</b>	<b>94.7</b>	<b>85.9</b>

computation cost. In the arXiv version of this paper, we describe a strategy for global hard sample mining to make the training more effective with a small batch size, and a procedure for selecting and curating an image-text training dataset with maximum information.

**Implementation Details.** Due to computational resource constraints, we train the ELIP-C model with a batch size of 40, the ELIP-S model with a batch size of 10, and the ELIP-B model with a batch size of 12. The initial learning rate is set to  $1 \times 10^{-3}$  for ELIP-C, ELIP-S, and ELIP-S-2, and  $1 \times 10^{-5}$  for ELIP-B. All models are trained on the DataCompDR dataset by default, with additional experiments conducted on the smaller CC3M dataset for ablation studies. Training is performed on two A6000 or A40 GPUs. For re-ranking, we select the top- $k$  samples based on the dataset and model: for ELIP-C,  $k$  is set to 100 for COCO and Flickr, 500 for Occluded COCO, and 1000 for ImageNet-R; for ELIP-S and ELIP-S-2,  $k$  is set to 100 for COCO and Flickr, 500 for Occluded COCO, and 200 for ImageNet-R; for ELIP-B,  $k$  is set to 20 for COCO and Flickr, 100 for Occluded COCO, and 200 for ImageNet-R. The value of  $k$  is chosen to ensure high recall in the original ranking while maintaining fast inference. Compared to the original pre-training approach of CLIP, SigLIP and BLIP-2, our method significantly improves training efficiency in terms of reduced training time, GPU requirements, and batch size, with only a marginal increase in FLOPS introduced by the trainable MLP mapping network. Further details are provided in the arXiv version of this paper.

### 6.6.1 Results on COCO and Flickr Benchmarks

**Ablation Study.** In Table 6.1, we evaluate the contributions of different components of the ELIP framework for CLIP. A comparison between Settings A and B highlights the effectiveness of the ELIP-C boost over the original CLIP. The comparison between Settings B and C demonstrates the importance of hard sample

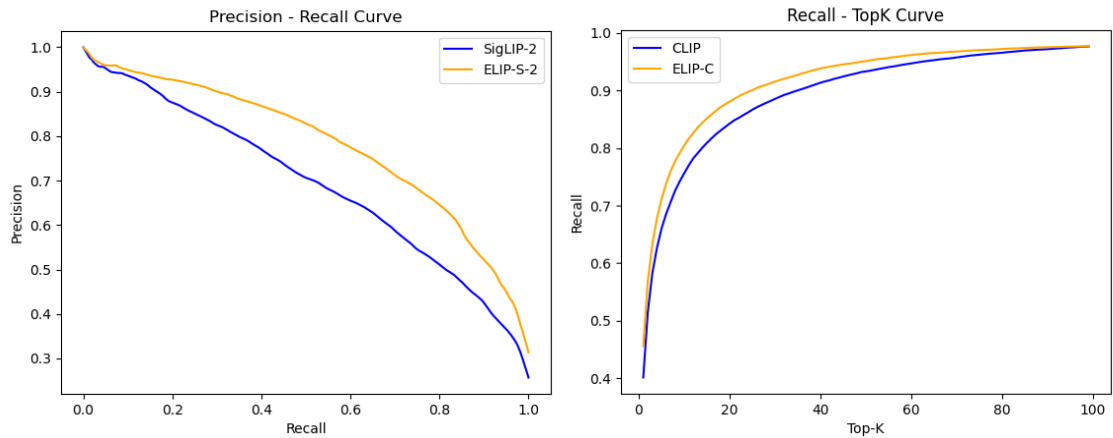
mining when training with a small batch size. Settings  $\mathbb{C}$  and  $\mathbb{D}$  show the benefit of training on larger datasets with less noisy captions. Finally, the comparison between Settings  $\mathbb{D}$  and  $\mathbb{E}$  reveals that generating multiple visual prompts (*e.g.*, 10 prompts in this study) is more beneficial than generating a single prompt. Further ablation studies on the number of generated prompts are detailed in the arXiv version of this paper.

**Table 6.2. Comparison with recent state-of-the-art methods.** Top: CLIP-based models; Middle: SigLIP-based models; Bottom: BLIP-2-based models. ELIP-C/ELIP-S brings a significant **zero-shot** performance boost of CLIP/SigLIP architectures, and ELIP-B outperforms the state-of-the-art BLIP-2 model. Results for models without \* are zero-shot, whereas results for models with \* are only zero-shot on Flickr, as the BLIP-2 model has been fine-tuned on COCO, and the \* models are based on BLIP-2. However, our method brings an improvement over BLIP-2 on both benchmarks when trained on DataCompDR.

Model	Year	COCO			Average	Flickr			Average
		Recall@1	Recall@5	Recall@10		Recall@1	Recall@5	Recall@10	
<i>CLIP</i> [Radford et al. 2021; Ilharco et al. 2021]	2021	40.16	65.95	75.62	60.58	67.56	88.34	93.00	82.97
<i>ELIP-C(Ours)</i>	-	<b>45.61</b>	<b>71.08</b>	<b>80.43</b>	<b>65.71</b>	<b>72.30</b>	<b>90.62</b>	<b>94.68</b>	<b>85.87</b>
<i>SigLIP</i> [Zhai et al. 2023]	2023	54.21	76.78	84.24	71.74	82.96	96.10	98.04	92.37
<i>ELIP-S(Ours)</i>	-	<b>61.03</b>	<b>82.62</b>	<b>88.70</b>	<b>77.45</b>	<b>87.62</b>	<b>98.16</b>	<b>99.16</b>	<b>94.98</b>
<i>SigLIP-2</i> [Tschannen et al. 2025]	2025	56.87	78.79	85.49	73.72	83.94	96.62	98.20	92.92
<i>ELIP-S-2(Ours)</i>	-	<b>62.91</b>	<b>83.86</b>	<b>89.70</b>	<b>78.82</b>	<b>87.74</b>	<b>97.96</b>	<b>98.94</b>	<b>94.88</b>
<i>BLIP-2*</i> [Li et al. 2023a]	2023	68.25	87.72	92.63	82.87	89.74	98.18	98.94	95.62
Q-Pert.(E)* [Sogi et al. 2024]	2024	68.34	87.76	92.63	82.91	89.82	98.20	99.04	95.69
Q-Pert.(D)* [Sogi et al. 2024]	2024	68.35	87.72	92.65	82.91	89.86	98.20	99.06	95.71
<i>ELIP-B(Ours)*</i>	-	<b>68.41</b>	<b>87.88</b>	<b>92.78</b>	<b>83.02</b>	<b>90.08</b>	<b>98.34</b>	<b>99.22</b>	<b>95.88</b>

**Comparison with State-of-the-Art.** As shown in Table 6.2, we compare our models (ELIP-C, ELIP-S, ELIP-S-2, and ELIP-B) with prior state-of-the-art methods. When trained on DataCompDR12M, our method demonstrates zero-shot performance improvements for CLIP, SigLIP, SigLIP-2, and BLIP-2 on the COCO and Flickr benchmarks. Notably, ELIP-B outperforms the most recent work [Sogi et al. 2024], establishing a new state-of-the-art for text-to-image retrieval on the BLIP-2 backbone. Furthermore, our ELIP-S, when applied to SigLIP and SigLIP-2, achieves performance comparable to BLIP-2. We have also compared ELIP with several baseline methods for re-ranking in the arXiv version of this paper.

**Recall Top- $k$  Curves.** Figure 6.5 (right) presents the Recall@Top- $k$  curves for the original CLIP model and our ELIP-C on the COCO benchmark. The curves are generated by plotting the Recall values across various Top- $k$  thresholds. Notably, there is a significant performance gap between the two models, demonstrating that ELIP-C re-ranking consistently improves text-to-image retrieval performance across different  $k$  values.



**Figure 6.5. Before/after comparisons.** Left: Precision-Recall curves for Occluded COCO retrieval, comparing SigLIP-2 initial rankings to the re-rankings given by ELIP-S-2. Right: Recall Top- $k$  curves for COCO retrieval, comparing CLIP initial rankings to the re-rankings given by ELIP-C.

**Qualitative Results.** Figure 6.6 provides a qualitative comparison between the initial rankings produced by the CLIP model and the re-ranked results obtained with ELIP-C on the COCO (left) and Flickr (right) benchmarks. In both cases, ELIP-C significantly improves the rankings by elevating the ground truth image (highlighted with a dashed box) to rank 1. Additional qualitative results are provided in the arXiv version of this paper.

**Visualisation of Attention Map.** Figure 6.7 visualises the cross-attention maps of the [CLS] token on patch tokens for both CLIP and ELIP-C on COCO. When the image matches the text query, our generated visual prompt vectors effectively enhance the selection of image features relevant to the query. This improvement can be attributed to ELIP-C’s early fusion approach, which integrates text features at the beginning of the image encoder, enabling the model to produce image embeddings more closely aligned with the query text. The visualisations provide strong evidence supporting this hypothesis.

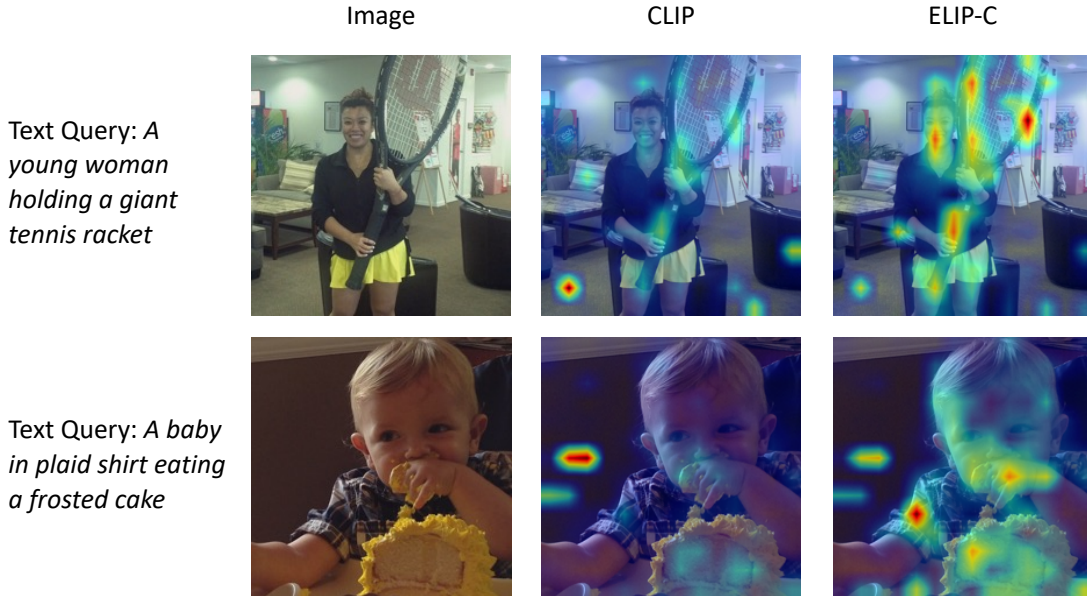
### 6.6.2 Results on OOD Benchmarks

The results on the out-of-distribution (OOD) benchmarks are presented in Table 6.3. ELIP achieves notable *zero-shot* improvements across all models on the OOD benchmarks, Occluded COCO and ImageNet-R, highlighting the strong generalization capabilities of the ELIP models. The performance can be improved further by fine-tuning the mapping network on suitable datasets (the image and text



**Figure 6.6. Qualitative comparison between CLIP initial ranking and ELIP-C re-ranking.** COCO: Columns 1–2; Flickr: Columns 3–4. The ground truth image for each query is highlighted with a dashed box, with the top-3 retrieved images shown. For the COCO query “A large wooden pole with a green street sign hanging from it”, CLIP ranks a non-wooden pole as top-1, while ELIP-C correctly re-ranks the large wooden pole to top-1. For the Flickr query “A man wearing bathing trunks is parasailing in the water”, CLIP ranks a wakeboarding person as top-1, whereas ELIP-C accurately re-ranks the parasailing man wearing bathing trunks to top-1.

encoders are frozen). Since it is not feasible to fine-tune on Occluded COCO (very few data samples) and ImageNet-R (evaluation only), for Occluded COCO retrieval, we fine-tune on the original COCO dataset, and for ImageNet-R retrieval, we fine-tune on ImageNet. As can be seen in Table 6.3 by this fine-tuning the performance of all the models is significantly boosted further. This demonstrates that fine-tuning ELIP enables efficient adaptation of the models to new datasets. The significant difference ELIP makes is also illustrated in Figure 6.5 (left). Please refer to the arXiv version of this paper on the fine-tuning.



**Figure 6.7. Visualisation of attention maps** comparing the cross-attention maps of the [CLS] token on patch tokens for CLIP and ELIP-C. For matched queries, ELIP-C enhances attention on image features relevant to the text. For example: (Row 1) ELIP-C focuses more on the giant tennis racket and the young woman for the query “A young woman holding a giant tennis racket”; (Row 2) ELIP-C highlights the cake, baby, and shirt for the query “A baby in plaid shirt eating a frosted cake”. Differences are minimal when the image does not match the query (examples provided in the arXiv version of this paper).

**Table 6.3. mAP Results on OOD datasets.** ELIP prompting achieves notable zero-shot improvements for CLIP, SigLIP series, and BLIP-2. These gains are further amplified through fine-tuning on relevant datasets. For example, to adapt to the Occluded COCO, the ELIP model is fine-tuned on COCO. Similarly, fine-tuning on ImageNet adapts ELIP to ImageNet-R. These results demonstrate ELIP’s capability for efficiently adapting the models to new datasets.

Model	Occluded COCO	ImageNet-R	Average
CLIP	47.47	76.01	61.74
ELIP-C (zero-shot)	48.89	76.81	62.85
ELIP-C (fine-tuned)	<b>59.88</b>	<b>81.44</b>	<b>70.66</b>
SigLIP	61.74	92.11	76.93
ELIP-S (zero-shot)	64.58	92.42	78.50
ELIP-S (fine-tuned)	<b>71.99</b>	<b>92.86</b>	<b>82.43</b>
SigLIP-2	66.40	92.66	79.53
ELIP-S-2 (zero-shot)	67.42	92.74	80.08
ELIP-S-2 (fine-tuned)	<b>76.10</b>	<b>94.00</b>	<b>85.05</b>
BLIP-2	62.73	82.31	72.52
ELIP-B (zero-shot)	63.40	82.99	73.20
ELIP-B (fine-tuned)	<b>70.49</b>	<b>83.68</b>	<b>77.09</b>

## 6.7 Conclusion

In this paper, we introduced *Enhance Language-Image Pre-training (ELIP)*, a method to improve visual-language foundation models for text-to-image retrieval. ELIP is a simple plug-and-play modification to pre-trained visual-language foundation models that significantly improves their zero-shot performance. Furthermore, the mapping network can be fine-tuned to efficiently adapt these models to OOD datasets, leading to still further improvements. We have also demonstrated, by visualising the attention maps, that ELIP enables the image encoder to attend to more relevant details. Future work could apply ideas similar to ELIP to enhance generative Multimodal Large Language Models by introducing more effective text-guided visual attention and encoding for both decoder-only [Liu et al. 2023a] and cross-attention-based [Alayrac et al. 2022] architectures. Please refer to the arXiv version of this paper for more details and future updates.

## Appendices

Appendices and future updates for this chapter can be found in the online version of the paper.<sup>1</sup>

## Statement of Authorship

A statement of authorship for this paper is provided in Appendix [A](#).

---

<sup>1</sup><https://arxiv.org/pdf/2502.15682>

## Chapter 7

# EGM: Efficient Visual Grounding Language Models

The paper is on arXiv, 2026.

# EGM: Efficient Visual Grounding Language Models

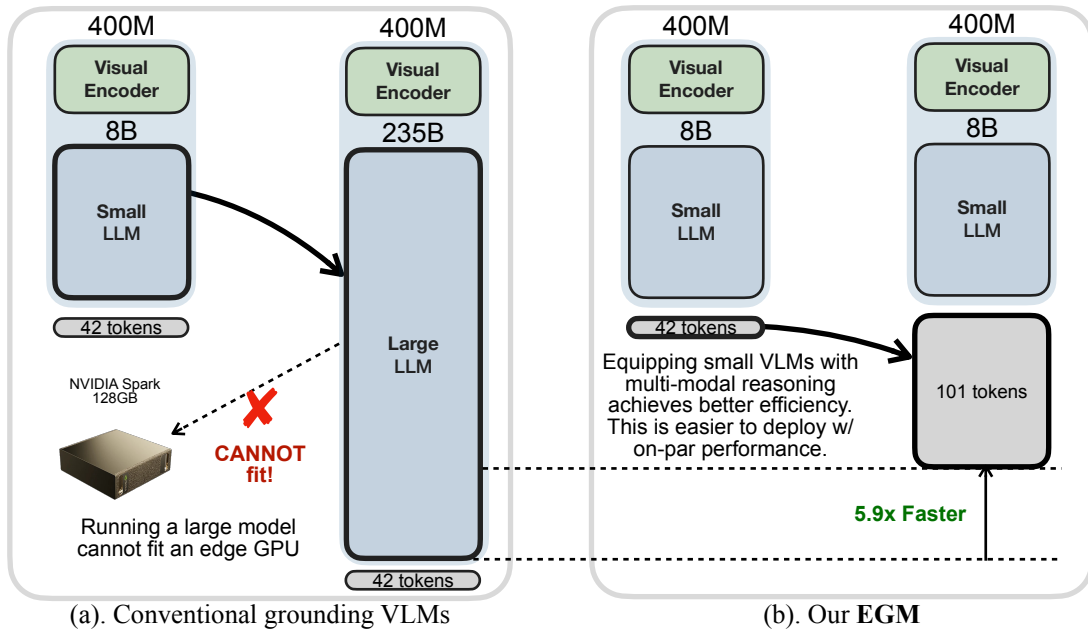
Guanqi Zhan<sup>1,3\*</sup>, Changye Li<sup>4\*</sup>, Zhijian Liu<sup>1</sup>, Yao Lu<sup>1</sup>,  
Yi Wu<sup>4</sup>, Song Han<sup>1,2</sup>, Ligeng Zhu<sup>1</sup>

<sup>1</sup>NVIDIA      <sup>2</sup>MIT

<sup>3</sup>University of Oxford      <sup>4</sup>Tsinghua University

## Abstract

Visual grounding is an essential capability of Visual Language Models (VLMs) to understand the real physical world. Previous state-of-the-art grounding visual language models usually have large model sizes, making them heavy for deployment and slow for inference. However, we notice that the sizes of visual encoders are nearly the same for small and large VLMs and the major difference is the sizes of the language models. Small VLMs fall behind larger VLMs in grounding because of the difference in language understanding capability rather than visual information handling. To mitigate the gap, we introduce ‘*Efficient visual Grounding language Models*’ (EGM): generate many mid-quality tokens (from small models) to match the performance of large VLMs with few high-quality but expensive tokens. This method is deployment-friendly, and yields better end-to-end latency: On the RefCOCO benchmark, our **EGM-Qwen3-VL-8B** demonstrates **91.4 IoU** with an average of 737ms (**5.9× faster**) latency while **Qwen3-VL-235B** demands 4,320ms to reach **90.5 IoU**. To validate our approach’s generality, we further set up a new amodal grounding setting that requires the model to predict both the visible and occluded parts of the objects. Experiments show our method consistently improves both vanilla and amodal grounding capabilities of small models to match or outperform larger models, thereby improving efficiency for visual grounding.



**Figure 7.1.** Overview of Efficient Visual Grounding Language Models. *Left.* Existing state-of-the-art grounding VLMs usually have large model sizes. *Right.* Our EGM enhances text understanding capabilities of small VLMs by equipping them with multi-modal reasoning capability, achieving better efficiency.

## 7.1 Introduction

Visual grounding is an essential capability of Visual Language Models (VLMs) to understand the physical world, especially for autonomous driving or robotics systems, where it is important to locate the target objects accurately according to the natural language instructions. Previous state-of-the-art visual grounding VLMs usually have large model sizes to achieve the SOTA performance, *e.g.*, InternVL-3.5 [Wang et al. 2025b] and Qwen3-VL [Bai et al. 2025]. This brings challenges to deployment and is unfriendly to edge systems (*e.g.*, Jetson Thor, Spark) while their small versions suffer from inferior performance.

A natural question then arises here: **What makes small VLMs fall behind bigger ones for visual grounding?** Our investigation begins with the scaling trends of the Qwen family [Bai et al. 2025; Alibaba 2025]. The visual encoder is the same across different sized VLMs, and the main upgrade is that larger models are built with a bigger Large Language Model (LLM). We hypothesize that the gap between different sized VLMs mainly comes from their text understanding ability (LLM) rather than visual information handling (ViT) as shown in Figure 7.1. We thoroughly verify it by visualizing the failure cases of a small VLM and find that a prominent failure pattern is “prompts too complicated”, where the text prompt

is semantically complicated and there are multiple similar candidates in the image (Figure 7.2). This failure pattern is gradually improved as the model size increases.

This insight suggests a pathway towards overcoming the issue: extending the test-time compute to enhance the text understanding ability and mitigate the performance gap with large models. More specifically, we encourage the VLM to generate the detailed reasoning process before yielding the grounding results. To achieve this, as in Figure 7.3, we use a proprietary VLM to generate detailed reasoning paths toward the target box, creating rich Supervised Fine-tuning (SFT) data for training. After the model learns the reasoning process, it is further trained with Reinforcement Learning (RL) to boost grounding capability. We find that this paradigm not only boosts the grounding performance of small VLMs but also brings better efficiency and deployment capability: many mid-quality tokens are actually cheaper than a few expensive ones (Figure 7.5). To further validate the effectiveness of the paradigm, we introduce a new grounding setting – amodal grounding. This task requires the model to predict both the visible and occluded parts of an object, and we apply our method to this new challenge and also observe solid improvements.

We conduct experiments on state-of-the-art open-source visual language models including QwenVL and InternVL in Section 7.5. Results show our method can consistently improve the visual grounding capability of small models and significantly mitigate the gap between small models and bigger models. We name our models ‘*Efficient visual Grounding language Models*’(EGM).

In summary, we make the following contributions:

- We identify the complicated prompt as the main reason why small VLMs lag behind bigger ones. To address this, we introduce EGM, which equips small VLMs with multi-modal reasoning abilities through extended test-time computation to enhance their visual grounding capabilities.
- We curated SFT and RL data that enables reasoning grounding skills for small VLMs. We also set up amodal grounding with fresh data to further validate our method’s generality, challenging the models to predict both visible and occluded parts.
- Our experiments show that EGM consistently boosts the grounding perfor-

mance of small VLMs across sizes and model families, in both vanilla and amodal grounding. For example, on the RefCOCO [Kazemzadeh et al. 2014; Mao et al. 2016] benchmark, our **8B model** achieves a **91.4 IoU**, beating the **235B model’s 90.5 IoU** while running **5.9× faster**.

## 7.2 Related Work

**Visual-Language Models (VLMs)** have advanced rapidly recently. Proprietary ones such as GPT [Achiam et al. 2023; Hurst et al. 2024], Gemini [Team et al. 2023; Team et al. 2024; Comanici et al. 2025], Claude [Anthropic 2025], and Grok [xAI 2025] have greatly improved performance on specific tasks and capability of multiple modalities. On the other hand, open-source VLMs have also developed continuously, closing the gap between open-source and proprietary models. Representative examples include QwenVL [Wang et al. 2024a; Bai et al. 2025], InternVL [Chen et al. 2024; Zhu et al. 2025a; Wang et al. 2025b], LLaVA [Li et al. 2024a], Llama [Touvron et al. 2023a; Touvron et al. 2023b; Dubey et al. 2024], VILA [Lin et al. 2024; Liu et al. 2025c] and Molmo [Deitke et al. 2024]. The main architecture of these models is LLaVA-like, *i.e.*, the image is firstly fed into a ViT encoder, and then projected to the LLM via an MLP. The application of VLMs has extended far beyond standard visual question answering. They are now increasingly adopted in complex real-world scenarios, encompassing autonomous driving, robotic manipulation, medical image analysis, and automated web-browsing agents.

**Grounding VLMs.** Visual grounding lets VLMs predict a bounding box for a target object given a text prompt. This is a fundamental capability of VLMs to understand the visual world. Leading open-source models, such as InternVL [Wang et al. 2025b] and QwenVL [Bai et al. 2025], perform best on the RefCOCO [Kazemzadeh et al. 2014; Mao et al. 2016] benchmark, even surpassing proprietary models such as GPT [Hurst et al. 2024] and Gemini [Comanici et al. 2025].

Beyond the standard grounding task, *amodal grounding*, predicting both the visible and hidden parts of the target described in the prompt, is also gaining increasing attention. This ‘amodal’ ability reflects how humans perceive objects as complete, even when occluded [Briscoe 2011; Kaup et al. 2024]. Although amodal completion

is gaining attention in computer vision [Li and Malik 2016; Zhu et al. 2017; Zhan et al. 2020; Li et al. 2022; Li et al. 2023c; Li et al. 2023b; Zhan et al. 2024b; Xu et al. 2024; Chen et al. 2025; Liu et al. 2025b; Wu et al. 2025; Li et al. 2025c; Lu et al. 2025], VLMs have not mastered this skill, even for the most advanced proprietary models. In this paper, we are the first to study *amodal grounding* in VLMs, aiming to boost their amodal grounding ability as well as the efficiency.

**Reasoning.** Reasoning has become a critical capability of modern LLMs, supporting key applications such as mathematical problem solving and code generation. [Guo et al. 2025] introduced an effective reasoning-training pipeline that combined a SFT cold start with Group Relative Policy Optimization (GRPO) [Shao et al. 2024], showing that structured reasoning traces together with reward optimization can substantially improve a model’s reasoning ability. Following this line of work, a series of GRPO-style algorithms have been proposed. [Yu et al. 2025] developed Dynamic Sampling Policy Optimization (DAPO), which used a token-level mean loss to enable importance sampling and advantage estimation at the token granularity. [Tan et al. 2025] further introduced Group Token Policy Optimization (GTPO), which incorporated entropy regularization into the GRPO framework to encourage more diverse model responses.



**Figure 7.2. Failure cases of small VLMs.** We find small VLMs, *e.g.*, InternVL-3-8B, tend to fail when the text prompt is *semantically complicated* and there are *multiple candidates* in the image that can confuse the model. We term this failure pattern ‘COMPLEX\_PROMPT’ and label the ground truth bbox in blue, and the 8B model prediction in orange in examples.

### 7.3 Why Small VLMs Fall Behind?

We start with the InternVL series of models. On RefCOCO benchmarks, we randomly select a subset and visualize the predictions from the 8B VLM alongside the ground truth bounding boxes. Following previous work [Zhu et al. 2025a; Wang et al. 2025b; Bai et al. 2025], we define the model  $M_\theta$  *succeeds* on a sample  $d_i$  if the Intersection-over-Union (IoU) between the bounding box predicted by the model  $M_\theta$  and the ground truth is higher than 0.5. The *grounding success* is defined as

$$S_{\theta,i} = \begin{cases} 1, & \text{if } \text{IoU}_{\theta,i} > 0.5, \\ 0, & \text{otherwise.} \end{cases} \quad (7.1)$$

We observe that a large proportion of the failure cases of the small VLM are due to ‘COMPLEX\_PROMPT’: the text prompt is semantically complex and there are multiple similar candidates in the image, so the model mis-locates to another candidate rather than the target object. For example, in the top left example of

**Table 7.1. Analysis of failure cases by proprietary models.** Reasons identified by different commercial models based on failure cases generated from small VLMs (IoU < 0.5). A large proportion of failure cases are due to ‘COMPLEX\_PROMPT’.

Failure Reason	GPT-4	GPT-5	Gemini-3-Pro
COMPLEX-PROMPT	62.8%	48.8%	46.8%

**Table 7.2. Performance of different models on the ‘COMPLEX\_PROMPT’ subset.** We report the performance on the subset of RefCOCO benchmarks where InternVL-3-8B fails due to ‘COMPLEX\_PROMPT’. ‘Acc’ denotes the grounding success ratio.

Model	InternVL-3		Gemini			Ours
	32B	78B	2.5-Flash	2.5-Pro	3-Pro	EGM-InternVL-3-8B
Acc	29.9%	42.6%	20.9%	22.1%	57.3%	42.7%

Figure 7.2, the model correctly understands the semantics of “chair”, but ignores the instructions “empty” and “next to guy”.

To further verify, we collect a set of failure cases and analyze how these cases improve with model size. Specifically, we collect the failure cases from InternVL-3-8B models then send these images together with ground truth to a proprietary model, and ask it to analyze the reason for all failure cases where the IoU between the prediction and ground truth boxes is lower than 0.5. The full prompt to the proprietary model is provided in Appendix. We categorize the failure cases into five categories: ‘COMPLEX-PROMPT’, ‘AMBIGUOUS-IMAGE’, ‘SMALL-OBJECT’, ‘GT-ERROR’, ‘OTHER’, and judge with different models to provide a fair judgement. Table 7.1 shows the key analysis using different proprietary models: Among all failure cases, all the proprietary models agree that a large proportion of failure cases (52.8% on average) are due to ‘COMPLEX\_PROMPT’, which confirms our observation. We attach the full failure reasons percentages in Appendix.

We further compare with the predictions of bigger models and find that the ‘COMPLEX\_PROMPT’ failure cases of the 8B model can be gradually resolved as the model size grows, as shown in Columns 2 and 3 of Table 7.2. This motivates us to introduce reasoning to small models to mitigate the gap in understanding ‘COMPLEX\_PROMPT’ compared to bigger models.

**Table 7.3. Statistics of curated training datasets** for vanilla grounding and amodal grounding. The vanilla grounding is curated from RefCOCO training dataset [Liu et al. 2025c] and amodal grounding dataset is curated from COCO-Amodal [Zhu et al. 2017] and TAO-Amodal [Hsieh et al. 2023].

Category	#Samples	#Images	Avg. #BBoxes	Width	Height
Vanilla Grounding	575,208	24,407	2.3 $\pm$ 1.3	585.7 $\pm$ 87.0	480.3 $\pm$ 93.1
Amodal Grounding	23,698	15,798	1.5 $\pm$ 1.5	1186.2 $\pm$ 332.0	717.2 $\pm$ 175.2

## 7.4 Method

This section presents our method to equip small models with multi-modal reasoning capability through extended test-time computation. More specifically, we train the model with SFT to obtain the thinking pattern, followed by RL training [Guo et al. 2025]. We describe the data curation for SFT in Section 7.4.1, the data curation for RL in Section 7.4.2, and the training pipeline in Section 7.4.3. Beyond the standard vanilla grounding task, we extend our method to the amodal grounding. Note that for **vanilla** grounding and **amodal** grounding, we curate different training datasets, and train the models **separately** for these two different tasks on the two different training datasets. Statistics of the curated training datasets are summarized in Table 7.3. We use proprietary VLMs in the data curation pipeline, and the corresponding prompts are provided in the Appendix.

### 7.4.1 SFT Data Curation

The objective of the SFT training stage is to teach the model to learn the reasoning pattern in grounding tasks. As analyzed in Section 7.3, addressing the key failure pattern requires reasoning about how to locate the target object accurately.

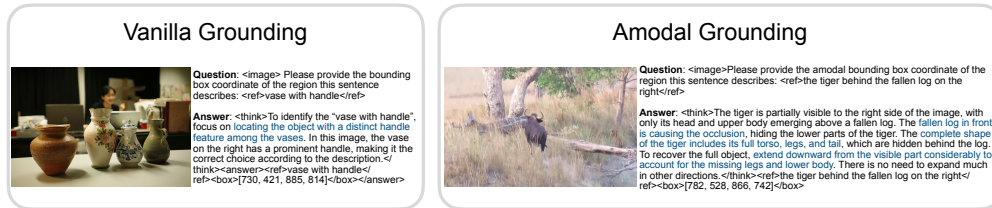
**Vanilla Grounding.** Given an off-the-shelf vanilla grounding training dataset  $\mathcal{D}_G = \{d_i\}_{i=1}^N$ , each sample  $d_i$  contains an image, a textual query, and the corresponding ground-truth bounding box. We present each  $d_i$  to a proprietary VLM  $\Phi$  to obtain its reasoning trace:

$$z_i = \Phi(d_i). \quad (7.2)$$

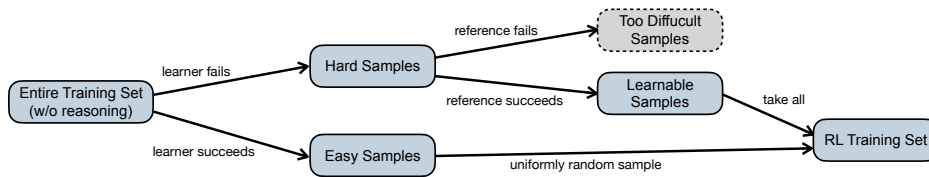
We then pair each original sample  $d_i$  with its reasoning trace  $z_i$  to obtain the reasoning training dataset  $\mathcal{D}_C = \{(d_i, z_i)\}_{i=1}^N$ . The process of SFT data curation is illustrated on the top of Figure 7.3. Dataset statistics for  $\mathcal{D}_G$  and  $\mathcal{D}_C$  are reported in Table 7.3 as the vanilla grounding category.



(a). Our SFT data curation pipeline



(b). Visualized examples of curated SFT data



(c). RL data curation

**Figure 7.3. Overview of our method.** *Top (a): Data curation pipeline of SFT training data with reasoning.* We feed the image, text prompt and ground truth bounding box of the target object into a proprietary VLM to generate the detailed reasoning process of how to locate the object correctly given the image and text prompt. The generated reasoning process is incorporated as part of the training data. *Middle (b): Examples of generated reasoning training data for vanilla grounding and amodal grounding.* The reasoning process of vanilla grounding analyzes the feature that distinguishes the target object from others, and the reasoning process of amodal grounding further involves what object causes the occlusion and in which directions the visible parts should be extended to recover the complete object. *Bottom (c): RL data curation for vanilla grounding.* The RL data is curated by collecting the instances with learnability  $> 0$  (i.e., the learner model fails while the reference model succeeds) and merging with *easy samples* where the learner model succeeds.

An example of vanilla grounding is shown on the middle left of Figure 7.3. For the text prompt ‘vase with handle’, the answer involves a reasoning process that analyzes the distinct feature of the target vase compared with others.

**Amodal Grounding.** Unlike vanilla grounding, there is no off-the-shelf data for training that provides both the text prompt and the corresponding amodal bounding box. To bridge the gap, we use a proprietary VLM to generate a more detailed description of the target object that can uniquely specify it in the existing amodal segmentation datasets, such as COCO-Amodal [Zhu et al. 2017] and TAO-Amodal [Hsieh et al. 2023]. We further conduct a verification step on the generated prompts to remove the noisy ones. We denote this generated dataset as  $\mathcal{D}_T$ .

Once we obtain the text prompt associated with the amodal bounding box, we next generate the reasoning process via the same pipeline. The only difference is

that, in addition to locating the target object, the reasoning should also involve: 1) what object(s) are causing the occlusion; 2) what this object’s complete unoccluded shape looks like; 3) in which direction(s) and how much the visible parts should be extended to recover the complete object. An example of generated reasoning data for amodal grounding is shown on the middle right of Figure 7.3. The task is to predict the amodal bounding box of the tiger behind the fallen log on the right. The reasoning process thus involves ‘the fallen log in front is causing the occlusion’, ‘the complete shape of the tiger includes its full torso, legs, and tail’, and ‘to recover the full object, extend downward from the visible part considerably’. This process yields the final amodal grounding dataset with reasoning  $\mathcal{D}_A$ . Additional details are provided in Appendix.

## 7.4.2 RL Data Curation

**Vanilla Grounding.** For RL training, we want to select harder but learnable samples to enable the model to learn the grounding capability. More specifically, we select a harder subset of the off-the-shelf vanilla grounding training dataset  $\mathcal{D}_G = \{d_i\}_{i=1}^N$  in Table 7.3 for vanilla grounding RL training by learnability [Evans et al. 2024; Zhan et al. 2025a].

More specifically, given the learner model  $M_\theta$  and the reference model  $M_{\text{ref}}$ , we first compute their IoU values on the entire training dataset  $\mathcal{D}_G$ , then we know whether a model *succeeds* on each sample as in Equation 7.1. The learnability of each sample  $d_i$  is calculated as  $l_i = S_{\text{ref},i} - S_{\theta,i}$ . The learner model is a small VLM that we train, and the reference model is a state-of-the-art visual grounding model. For each learner model, we then construct a learnable sample set  $\mathcal{D}_{\text{learnable}} = \{d_i \mid l_i > 0, d_i \in \mathcal{D}_G\}$ , where the learnability is greater than 0. This means that the learner model fails on these samples while the reference model succeeds on these samples. Therefore, these samples have not yet been learned by the learner model, while they are learnable as the reference model can succeed on them. The rest samples where the reference model also fails are too difficult or may be due to ground truth annotation errors and we thus do not use them.

If we only use samples where the learner model fails for RL training, the model may overfit on harder samples and the performance for general cases can degrade. Therefore, we also uniformly sample instances where the learner model succeeds

from  $\mathcal{D}_G$  to form an easy sample set  $\mathcal{D}_{\text{easy}}$  of equal size to  $\mathcal{D}_{\text{learnable}}$ . The final training dataset is the union of the two subsets:  $\mathcal{D} = \mathcal{D}_{\text{learnable}} \cup \mathcal{D}_{\text{easy}}$ .

**Amodal Grounding.** We use the entire amodal grounding dataset  $\mathcal{D}_T$  in Table 7.3 to make the most of the limited data and to strengthen the model’s performance in this new task.

### 7.4.3 Training Pipeline and Reward

**Training Pipeline.** We first perform SFT on the base model to teach it the desired reasoning pattern. Then, we train the model using reinforcement learning to obtain the final EGM models. We adopt GRPO [Shao et al. 2024] with a token-level mean loss [Yu et al. 2025]. We introduce KL divergence penalty and entropy regularization to ensure the model explores sufficiently during reasoning while avoiding pattern collapse.

Specifically, for a given input  $q$  and its  $i$ -th generated sequence  $o_i$  sampled from the behavior policy  $\pi_{\theta_{\text{old}}}$ , we denote its length by  $|o_i|$  and its token sequence as  $\{o_{i,t}\}_{t=1}^{|o_i|}$ . The overall GRPO objective is defined as:

$$\mathcal{J}_{\text{GRPO}}(\pi_{\theta}) = \mathbb{E}_{q \sim P(Q), \{o_i\}_{i=1}^G \sim \pi_{\theta_{\text{old}}}(O|q)} \quad (7.3)$$

$$\left[ \frac{1}{G} \sum_{i=1}^G \sum_{t=1}^{|o_i|} \min \left( \frac{\pi_{\theta}(o_{i,t} | q, o_{i,<t})}{\pi_{\theta_{\text{old}}}(o_{i,t} | q, o_{i,<t})} \hat{A}_{i,t}, \right. \right. \\ \left. \left. \text{clip} \left( \frac{\pi_{\theta}(o_{i,t} | q, o_{i,<t})}{\pi_{\theta_{\text{old}}}(o_{i,t} | q, o_{i,<t})}, 1 - \epsilon, 1 + \epsilon \right) \hat{A}_{i,t} \right) \right. \\ \left. - \beta \mathbb{D}_{\text{KL}}(\pi_{\theta} \| \pi_{\text{ref}}) + \gamma \mathbb{H}(\pi_{\theta}) \right], \quad (7.4)$$

$$\mathbb{D}_{\text{KL}}(\pi_{\theta} \| \pi_{\text{ref}}) = \frac{\pi_{\text{ref}}(o_i | q)}{\pi_{\theta}(o_i | q)} - \log \frac{\pi_{\text{ref}}(o_i | q)}{\pi_{\theta}(o_i | q)} - 1, \quad (7.5)$$

$$\mathbb{H}(\pi_{\theta}) = -\log \pi_{\theta}(o_i | q), \quad (7.6)$$

where  $\beta$  and  $\gamma$  are hyper-parameters;  $\epsilon$  is PPO clip ratio;  $\pi_{\text{ref}}$  is the reference model; and  $\hat{A}_{i,t}$  is the token level advantage, derived from the rewards  $\{r_1, r_2, \dots, r_G\}$

corresponding to the outputs within each group:

$$\hat{A}_i = \frac{r_i - \text{mean}(\{r_j\}_{j=1}^G)}{\text{std}(\{r_j\}_{j=1}^G)}, \hat{A}_{i,t} = \hat{A}_i / |o_i|. \quad (7.7)$$

**Reward.** We define the reward as a weighted combination of the IOU between the predicted bounding box and ground truth, and *grounding success* (as in Equation 7.1) with hyper-parameter  $\alpha$  :

$$r_i = \alpha IoU_{\theta,i} + (1 - \alpha) S_{\theta,i}. \quad (7.8)$$

The  $\alpha$  is set to 0.5 by default during experiments unless specifically clarified.

## 7.5 Experiments

### 7.5.1 Implementation Details

**Models.** We apply our method to the state-of-the-art visual grounding open-source VLM families InternVL and QwenVL, and add a prefix ‘EGM-’ to models to which our method has been applied. Our experiments are conducted on InternVL-3-1B, InternVL-3-2B and InternVL-3-8B for the InternVL family, and on Qwen3-VL-2B-Thinking, Qwen3-VL-4B-Thinking and Qwen3-VL-8B-Thinking for the QwenVL family.

**SFT.** For the SFT training stage, we use the official GitHub scripts of InternVL-3 and Qwen3-VL. The training dataset statistics are presented in Table 7.3. The learning rate is set to be  $1e^{-5}$ , the epoch is set to be 1 and the training batch size is 128. We train all models on 8 A100 GPUs.

**RL.** For the reinforcement learning stage, we employ the VeRL framework [Sheng et al. 2024]. All rollouts and inference are performed using the vLLM engine [Kwon et al. 2023]. We train with a learning rate of  $3e^{-6}$  for 5 epochs and a batch size of 256. The reward weight  $\alpha$  in Equation 7.8 is fixed to 0.5. For the QwenVL model series, we set the KL coefficient to  $\beta = 0.005$  and the entropy coefficient to  $\gamma = 0.0$ . For the InternVL model series, we use  $\beta = 0.0$  and  $\gamma = 0.01$ . We construct the RL training dataset following the procedure described in Section 7.4.2, using InternVL-

3.5-241B as the reference model  $M_{\text{ref}}$  and the corresponding fine-tuned model as the learner model  $M_{\theta}$ . The final dataset size varies across models. The full details about our RL training and dataset samples are attached in Appendix.

## 7.5.2 Evaluation Benchmarks and Metrics

**Vanilla Grounding Benchmark.** For vanilla grounding, we use the standard RefCOCO [Kazemzadeh et al. 2014; Mao et al. 2016] benchmarks for evaluation, which contain eight test splits. We adopt ‘accuracy’ as the evaluation metric, *i.e.*, the proportion of *grounding success*  $S_{\theta,i}$  as defined in Equation 7.1. We report the score on individual splits as well as the average score over the eight splits.

**Amodal Grounding Benchmark.** As there is no off-the-shelf benchmark for amodal grounding, we start from the standard amodal benchmark in the computer vision community – the val and test splits of COCO-Amodal [Zhu et al. 2017] dataset, and use the pipeline as introduced in Section 7.4.1 to generate text prompts. We therefore obtain an evaluation benchmark consisting of 11,261 samples and 2,474 images. We adopt the same evaluation metric as vanilla grounding.

**Efficiency Metric.** To measure efficiency, besides the number of model parameters, we also calculate the average latency per sample on a random 10% subset of the standard RefCOCO benchmarks for evaluation. Latency is measured as the time (in milliseconds) from sending the text prompt to receiving the entire answer from the model. For fair comparison, we report the total *GPU latency* on L20 GPUs.

We adopt identical inference hyper-parameters for all models during evaluation. The only differences in the prompts follow the model-family-specific formats prescribed in the official implementation. Further details of the evaluation setup are provided in Appendix.

## 7.5.3 Vanilla Grounding

Table 7.4 shows the results of applying our method to models of different sizes from the InternVL-3 family and the Qwen3-VL family. We show the ablation of SFT and RL in Appendix.

**Table 7.4. Accuracy results for vanilla grounding.** Our method (+EGM) consistently improves the performance of models of different sizes and from different families. The ablation study of reasoning process (+EGM w/o R) validates the effectiveness of our method. Entries marked with \* are from the official report [Wang et al. 2025b], and the others are measured by ourselves.

Model	RefCOCO			RefCOCO+			RefCOCOg		Avg. Acc
	val	test-A	test-B	val	test-A	test-B	val	test	
Qwen3-VL-2B-Instruct	88.7	90.8	84.7	80.1	85.7	73.1	84.8	85.7	84.2
Qwen3-VL-2B-Thinking	87.5	91.4	82.0	80.2	86.7	72.1	84.4	84.5	83.6
+ EGM	93.0	94.0	89.4	87.8	91.7	82.7	88.6	89.3	89.6 <sub>+6.0↑</sub>
Qwen3-VL-4B-Instruct	91.0	92.7	87.6	84.4	88.7	78.2	88.3	87.8	87.3
Qwen3-VL-4B-Thinking	90.0	92.7	85.6	85.2	89.5	79.3	87.8	87.7	87.2
+ EGM	93.5	95.1	90.9	89.7	93.1	84.9	90.4	90.8	91.0 <sub>+3.8↑</sub>
Qwen3-VL-8B-Instruct	91.6	93.3	87.8	85.8	90.3	79.9	88.7	88.7	88.3
Qwen3-VL-8B-Thinking	91.0	92.9	86.9	86.2	89.3	80.2	87.6	88.6	87.8
+ EGM w/o R	92.2	93.6	89.2	85.7	90.2	80.0	88.6	89.3	88.6 <sub>+0.8↑</sub>
+ EGM	93.9	95.0	91.2	90.1	93.3	85.9	90.4	91.2	91.4 <sub>+3.6↑</sub>
Qwen3-VL-235B-A22B-Instruct	92.9	94.6	90.2	88.6	92.4	84.2	90.3	90.8	90.5
Qwen3-VL-235B-A22B-Thinking	92.9	94.1	90.6	89.5	92.5	85.5	90.4	90.5	90.7
InternVL-3-1B	85.8	90.1	81.7	76.6	84.1	69.2	82.8	82.6	81.6*
+ EGM	90.2	93.2	87.0	83.8	88.8	77.5	86.4	87.5	86.8 <sub>+5.2↑</sub>
InternVL-3-2B	89.8	92.6	86.4	84.0	89.2	76.5	87.6	87.2	86.7*
+ EGM	92.2	94.0	87.4	85.6	91.2	79.2	88.5	88.7	88.4 <sub>+1.7↑</sub>
InternVL-3-8B	92.5	94.6	88.0	88.2	92.5	81.8	89.6	90.0	89.6*
+ EGM w/o R	91.3	93.9	87.0	85.7	90.6	78.1	87.3	88.2	87.7 <sub>-1.9↓</sub>
+ EGM	93.6	95.2	90.1	89.3	93.6	83.1	89.7	90.7	90.7 <sub>+1.1↑</sub>
InternVL-3-78B	93.4	95.4	90.3	90.1	93.8	85.3	91.5	91.5	91.4*
Gemini-2.5-Pro	66.8	66.8	69.2	58.2	58.2	60.6	65.0	65.7	63.8
Gemini-3-Pro	87.3	86.4	86.4	80.3	79.7	81.8	89.9	89.6	85.2
GPT-5	42.6	46.0	39.3	37.4	40.1	35.2	39.1	41.0	40.1

**EGM consistently improves the grounding performance.** EGM boosts the performance of Qwen3-VL-2B/4B/8B-Thinking by +6.0/+3.8/+3.6 and InternVL-3-1B/2B/8B by +5.2/+1.7/+1.1 for the average accuracy. This shows the generality of our EGM that can improve small VLMs of *different sizes* and from *different families* on RefCOCO benchmarks.

**EGM is mainly boosted by reasoning.** (1) The ablation study that disables reasoning (+EGM w/o R) in Table 7.4, with full results reported in Appendix, verifies the necessity of the reasoning process, as EGM performance drops notably without it. (2) The commercial models actually perform worse (last section of Table 7.4) compared with all EGM models; it is difficult to distill from an inferior model to boost performance.

**EGM mitigates the gap between small models and larger models.** For example, the performance of Qwen3-VL-2B/4B/8B-Thinking is 83.6/87.2/87.8. Our 2B and 4B models outperform Qwen3-VL-8B-Thinking by +1.8/+3.2. Our EGM shows that extending the test-time computation can boost the performance by improving text understanding. Furthermore, the fine-grained analysis in Appendix confirms that our method is particularly effective on hard samples, resolving nearly half of the cases with baseline IoU < 0.5.

**EGM is faster and more accurate than Qwen3-VL-Thinking.** For the Qwen3-VL models, 2B/4B/8B-Thinking does not outperform 2B/4B/8B-Instruct (-0.6/-0.1/-0.5), but our 2B/4B/8B reasoning models significantly outperform 2B/4B/8B-Instruct (+5.4/+3.7/+3.1), while being faster than 2B/4B/8B-Thinking, demonstrating the effectiveness and efficiency of our multi-modal reasoning.

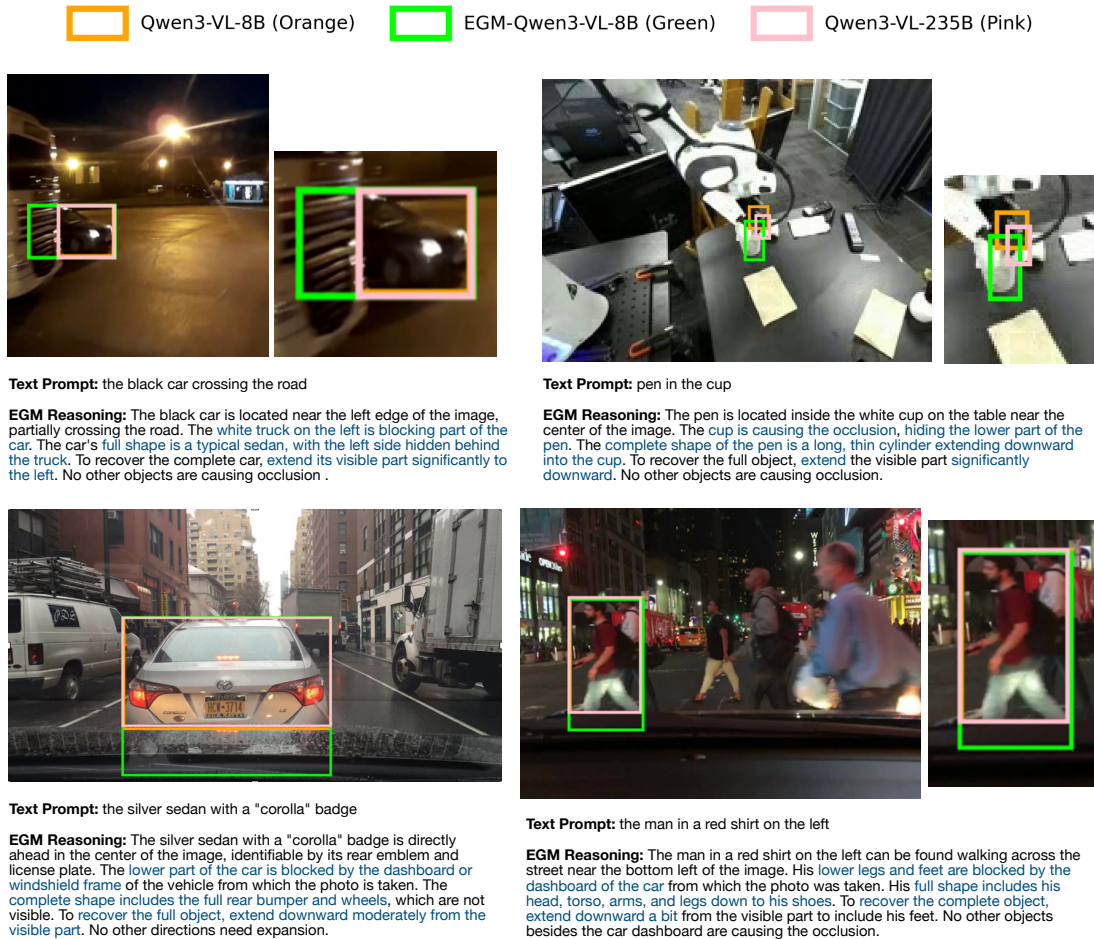
## 7.5.4 Amodal Grounding

**Table 7.5. Accuracy results for amodal grounding.** Similar to vanilla grounding, our method consistently boosts the performance of amodal grounding.

Setting	InternVL-3				Qwen3-VL				Others		
	1B	2B	8B	78B	2B	4B	8B	235B	Gemini-2.5-Pro	Gemini-3-Pro	GPT-5
Original	56.9	63.8	62.0	51.0	66.5	71.6	71.4	74.1	52.3	71.9	24.7
+ EGM	64.3 <sub>+7.4</sub> ↑	70.2 <sub>+6.4</sub> ↑	73.5 <sub>+11.5</sub> ↑	-	73.4 <sub>+6.9</sub> ↑	73.9 <sub>+2.3</sub> ↑	73.9 <sub>+2.5</sub> ↑	-	-	-	-

To further verify the effectiveness of our method, we show the accuracy results on the new task of amodal grounding in Table 7.5. Similar to vanilla grounding, **our method consistently boosts the performance of models of different sizes and from different families** by +7.4/+6.4/+11.5 (InternVL-3) and +6.9/+2.3/+2.5 (Qwen3-VL), demonstrating the wide applicability of our method for different tasks.

Moreover, in Figure 7.4 we show demos of our EGM amodal grounding models in autonomous driving and robotics scenarios. The top left demo is a situation called ‘ghost probe’, where a car crossing the road suddenly appears. In this situation, it is important to notice and figure out the complete shape of the occluded car when it has not fully appeared. The top right demo for robotics is collected from the DROID dataset [Khazatsky et al. 2024]. In this situation, the robot arm is going to manipulate the objects on the table and it is important to know the complete shape

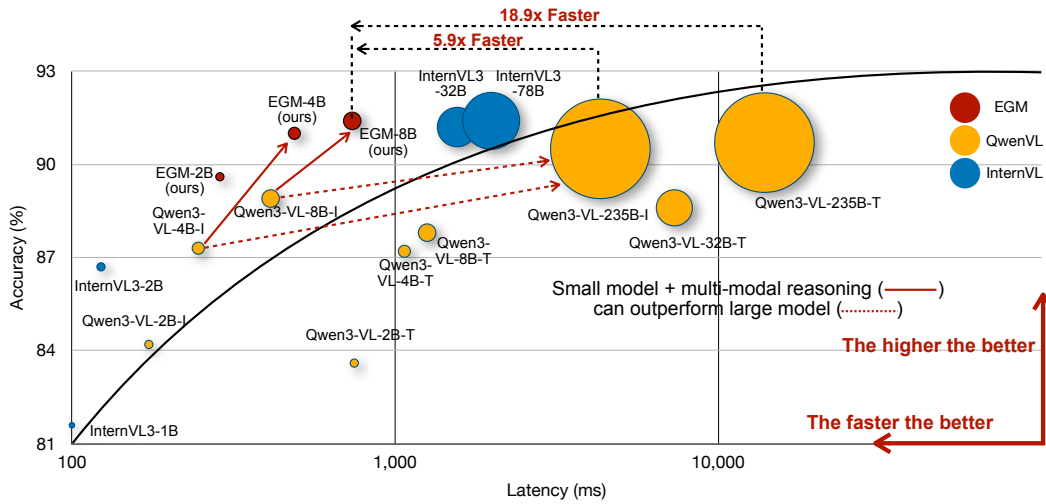


**Figure 7.4.** Demo of Qwen3-VL-8B, Qwen3-VL-235B, and our EGM-Qwen3-VL-8B for amodal grounding in autonomous driving and robotics scenarios.

of occluded objects, *e.g.*, the pen, for robust manipulation, as discussed in [Xia et al. 2025]. The bottom demos are two additional demos for driving scenario. For all cases, it can be observed that compared with Qwen3-VL-8B and Qwen3-VL-235B, our EGM-Qwen3-VL-8B can reason about the occlusion situations and correctly predict the complete shape of the occluded pedestrian, the occluded car and the occluded pen. This suggests the potential application of our EGM models in autonomous driving and robotic manipulation tasks.

### 7.5.5 Efficiency Comparison

Figure 7.5 compares the accuracy and efficiency of models in the QwenVL family. **Our EGM models significantly improve the efficiency of visual grounding:** Both our 4B and 8B models surpass the 235B models in accuracy and are over 5× faster in inference. Our 2B model also outperforms several larger models—32B-Thinking, 8B-Instruct, and 8B-Thinking—while achieving better GPU latency.



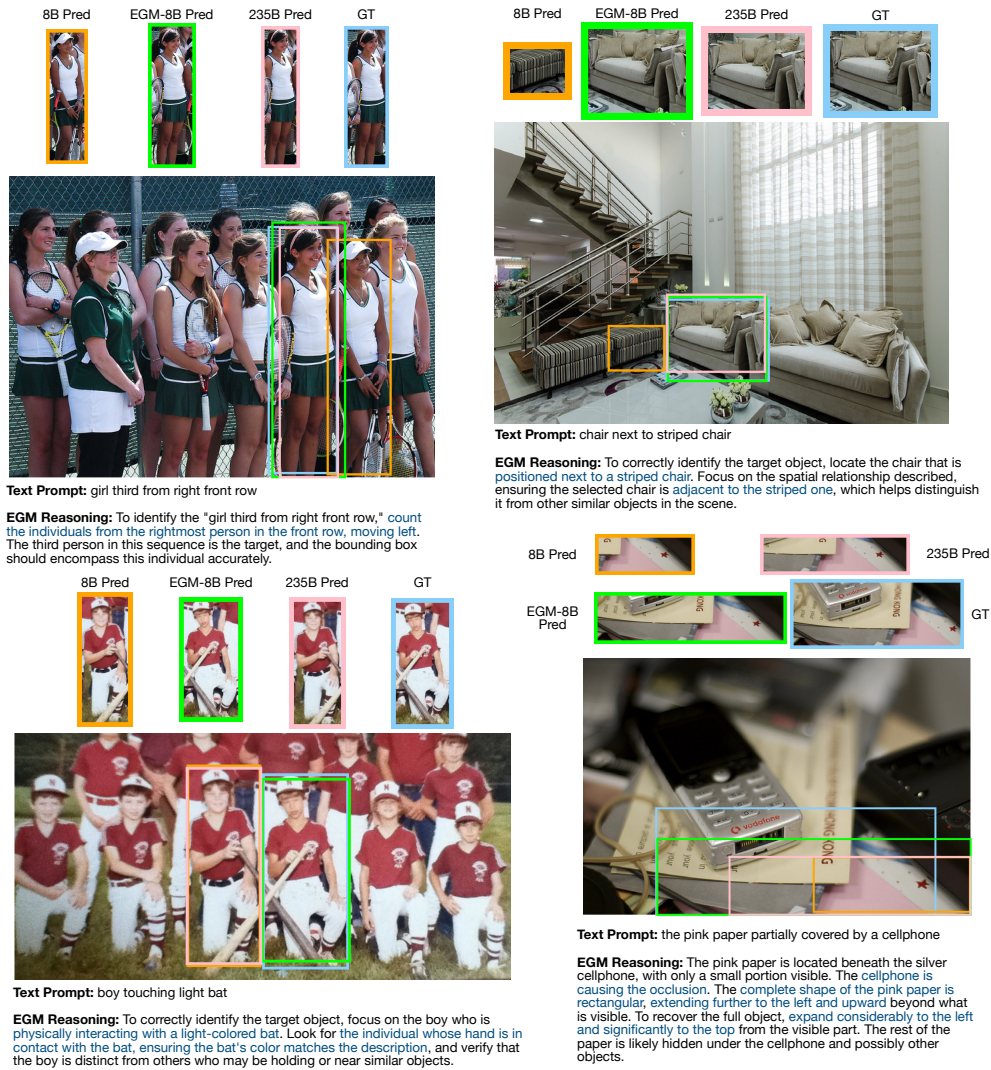
**Figure 7.5. Accuracy vs. Efficiency.** Our models, such as EGM-Qwen3-VL-4B and EGM-Qwen3-VL-8B, have greatly improved the efficiency of visual grounding. For example, EGM-Qwen3-VL-8B outperforms both the state-of-the-art Qwen3-VL-235B-Instruct and Qwen3-VL-235B-Thinking models for accuracy, while speeding up  $5.9\times/18.9\times$  in terms of GPU latency. For Qwen models, ‘-T’ denotes ‘-Thinking’ and ‘-I’ denotes ‘-Instruct’.

The observation is consistent with the results in Table 7.4, demonstrating the effectiveness of extending test-time computation for visual grounding.

### 7.5.6 Qualitative Comparison

Figure 7.6 shows the qualitative comparison of Qwen3-VL-8B, Qwen3-VL-235B, and our EGM-Qwen3-VL-8B for vanilla grounding and amodal grounding. Please refer to Appendix for more qualitative results.

The top left, top right and bottom left samples are vanilla grounding on RefCOCO benchmark. For the *top left* example, the text prompt is ‘girl third from right front row’, and the Qwen3-VL-8B model mis-locates to the ‘second’ girl. Our EGM-Qwen3-VL-8B correctly reasons the key feature ‘count the individuals from the rightmost person in the front row, moving left’ and generates a correct prediction. For the *top right* example, the text prompt is ‘chair next to striped chair’, and the Qwen3-VL-8B model confuses it with the striped chair itself. Our EGM-Qwen3-VL-8B, however, correctly reasons about the key feature ‘the chair that is positioned next to a striped chair’ and outputs the correct grounding box. For the *bottom left* example, the text prompt is ‘boy touching light bat’, and the Qwen3-VL-8B model fails to understand ‘touching light bat’ and mis-predicts the boy touching the darker bat. Our EGM-Qwen3-VL-8B predicts the correct boy, with



**Figure 7.6.** Qualitative comparison of Qwen3-VL-8B, Qwen3-VL-235B, and our EGM-Qwen3-VL-8B for vanilla grounding and amodal grounding. Vanilla grounding: top left, top right and bottom left; Amodal grounding: bottom right.

a reasoning that it should be ‘the individual whose hand is in contact with the bat, ensuring the bat’s color matches the description’. For the vanilla grounding cases, generally Qwen3-VL-235B performs better than Qwen3-VL-8B, as can be observed in the examples.

The *bottom right* example is amodal grounding on the COCO-Amodal benchmark. It can be observed that Qwen3-VL-8B and Qwen3-VL-235B fail to predict the amodal grounding box of ‘the pink paper partially covered by a cellphone’, while our EGM-Qwen3-VL-8B prediction is close to the ground truth, with reasoning process about locating the pink paper, analyzing it is occluded by the cellphone and other objects and should extend to the left and upward to recover the full unoccluded shape. This reasoning helps EGM-Qwen3-VL-8B to better locate the

amodal bounding box covering both the visible and occluded parts of the pink paper.

## 7.6 Conclusion

In this paper, we introduced ‘*Efficient visual Grounding language Models*’ (EGM), a method to improve the efficiency of visual grounding language models. Our method equips small models with the multi-modal reasoning capability via a two-stage SFT-RL training paradigm. Experiments show our method can be applied to different model families, different model sizes, and different tasks, to boost the performance consistently and significantly. Therefore, our EGM achieves a 91.4 IoU with an 8B model, outperforming the conventional 235B model with a 90.5 IoU, while speeding up 5.9 $\times$ . We hope systems developed in this way can help scenarios such as autonomous driving and robotics, where an efficient grounding language model with amodal capability is important.

## Appendices

Appendices and future updates for this chapter can be found in the online version of the paper.<sup>1</sup>

## Statement of Authorship

A statement of authorship for this paper is provided in Appendix [A](#).

---

<sup>1</sup><https://arxiv.org/pdf/2601.13633>

# Chapter 8

## Summary

In this chapter, we first summarise the achievements and the impact of the presented works in this thesis (Section 8.1), and then briefly suggest directions for future works (Section 8.2).

### 8.1 Achievement and Impact

#### 8.1.1 Part I: Occlusion Handling

**Chapter 2.** In Chapter 2, we formulate the occlusion problem for object detection and instance segmentation, by setting up two benchmarks of detecting / segmenting occluded objects. In addition to the benchmark contributions, we have also developed a tri-layer plugin to improve the performance of object detection and instance segmentation under occlusion for pre-trained models.

The benchmarks we proposed have been incorporated into the popular object detection library MMDetection [Chen et al. 2019b] as an additional evaluation for the model’s capability to detect or segment occluded objects. The benchmarks have also been used by subsequent works to evaluate the capability of object recognition under occlusion [Ozguroglu et al. 2024], the performance of instance segmentation [Lee et al. 2024], as well as the understanding of occlusion in Chapter 4 and text-to-image retrieval in Chapter 6.

Chapter 2 is mentioned by subsequent works to study the occlusion problem in different tasks, such as vehicle detection [He et al. 2024], pedestrian detection [Li

and Li 2026], product detection [Liu et al. 2024c], panoptic segmentation [Wei et al. 2025], instance segmentation [Lee et al. 2024], image classification [Kotwal et al. 2024], image generation [Zhan and Liu 2025; Walton et al. 2025], video action detection [Modi et al. 2023], video object tracking [Van Hoorick et al. 2023], and video object segmentation [Ding et al. 2025]. Our paper is also mentioned by another series of works on robotics [Liu et al. 2023b; Wu et al. 2023a; Xia et al. 2025; Qi et al. 2023], as occlusion frequently happens in robotic tasks and it is important to handle occlusion for robotics.

**Chapter 3.** Chapter 3 is a pioneering work in the amodal community. In this paper, we are the first to utilise the prior from pre-trained Stable Diffusion model for amodal completion. We have therefore developed a state-of-the-art amodal completion model. We have also introduced a new amodal benchmark via using 3D data to determine authentic ground truth amodal masks for partially occluded objects in real images.

Chapter 3 is mentioned by subsequent works to utilise the prior of diffusion models for amodal segmentation in images [Lee et al. 2025; Tran et al. 2024; Liu et al. 2024b; Fan et al. 2025; Chi et al. 2025; Ao et al. 2025; Noh and Chang 2025; Zhu et al. 2025b; Ozguroglu et al. 2024; Xu et al. 2024], and another series of works to utilise the prior of diffusion models for amodal segmentation in videos [Lu et al. 2025; Chen et al. 2025; Lee et al. 2025; Chi et al. 2025].

The model we developed has been used as a strong baseline for follow-up amodal papers [Lee et al. 2025; Zhang et al. 2025a; Li et al. 2025b; Noh and Chang 2025; Zhan et al. 2025b; Chen et al. 2025; Li et al. 2025e; Li et al. 2025d; Li et al. 2025f] for amodal segmentation in both images and videos. The new amodal benchmark we proposed, *i.e.*, MP3D-Amodal, has been used for evaluation of amodal segmentation in subsequent work [Zhang et al. 2025a]. Our work is noticed by Adobe research, and they have scaled-up the idea to develop an amodal foundation model [Liu et al. 2024b].

Although the area of amodal segmentation had not been paid a lot of attention on by the community in 2023, Chapter 3 has attracted more attention of the community on the amodal perception research [Saleh et al. 2024; Zhang et al. 2025c; Yang et al. 2025; Tsai et al. 2025] since 2024. Besides the impact on the amodal commu-

nity, our paper also provides a way of occlusion handling and thus is mentioned by works of other computer vision tasks, such as video instance segmentation [Tran et al. 2025], image recognition [Wang et al. 2025a], image editing [Canberk et al. 2024; Zhan and Liu 2025], robotics [Selvaratnam and Bazazian 2025; Xia et al. 2025; Li et al. 2024c], depth estimation [Li et al. 2025c], 3D reconstruction [Zhou and Tai 2025; Wu et al. 2025; Qu et al. 2025] and 3D decomposition [Damaraju et al. 2025].

### 8.1.2 Part II: 3D Physical Understanding

**Chapter 4.** In Chapter 4, we introduce a general and lightweight protocol to evaluate whether features of an off-the-shelf large vision model encode a number of physical ‘properties’ of the 3D scene. This work provides insights on understanding generative models and probing visual foundation models for 3D physical understanding. In the paper, we observe that there are representations in large-scale pre-trained networks, such as Stable Diffusion, DINO and CLIP, that encode a number of 3D physical properties, unlocking potential applications of understanding the 3D physical world using the priors from these large-scale pre-trained foundation models.

Chapter 4 is mentioned by a series of works [Sarkar et al. 2024; Du et al. 2023; Yuan et al. 2026; Beckmann 2025; Thozhiyoor et al. 2025; Xu et al. 2025] on understanding what is built in generative models, and another series of works [El Banani et al. 2024; Man et al. 2024; Danier et al. 2025; Espinosa et al. 2024; Azad et al. 2025; Eppel 2024; Stary et al. 2025; Lüddecke and Ecker 2025] to probe visual foundation models for visual understanding of the 3D physical world. Notably, Chapter 4 deeply inspires and is closely followed by [Danier et al. 2025] to evaluate monocular depth perception in large vision models.

Subsequent works therefore utilise priors from large-scale pre-trained networks for different visual understanding tasks, such as object pose estimation [Wang et al. 2024b; Von Gimborn et al. 2024; Huang and Xu 2025], camera pose estimation, visual similarity calculation [Song et al. 2025], point tracking [Aydemir et al. 2024; Aydemir 2025], amodal segmentation [Chen et al. 2025], BEV segmentation [Barn et al. 2024], video understanding [Khurana and Ramanan 2024], and 3D scene understanding [Burgess et al. 2024], as well as amodal completion in Chapter 3.

Chapter 4 is also mentioned by a series of follow-up works that further explore the topic of 3D physical understanding [Bhattad et al. 2025; Yuan et al. 2026; Li et al. 2025a], as well as Chapter 5. Understanding 3D and physical properties are very important for robotic tasks, so Chapter 4 is also mentioned by a series of works on 3D physical understanding for robotics and embodied AI [Li et al. 2024c; Murray et al. 2024; Murray 2025]. Additionally, Chapter 4 is also mentioned by subsequent works to develop better generative models [Kocsis et al. 2024; Xu et al. 2024; Li et al. 2024b; Seyfioglu et al. 2024; Parihar et al. 2025; Huang et al. 2025; Antsfeld and Chidlovskii 2024]. Chapter 4 is also extended by [Stern et al. 2025] to develop a task-aware evaluation framework for novel view synthesis.

**Chapter 5.** In Chapter 5, we predict dynamic physical properties in videos using different types of large-scale pre-trained video foundation models. More specifically, for video generative models and video self-supervised models, the paper has explored to extract features via cross-attention with pre-trained representations; for multi-modal large language models, the paper has explored different ways of prompt engineering to extract such knowledge. Therefore, as an initial exploration, this work opens up a new research direction and can potentially inspire future works to predict more different dynamic physical properties from video foundation models via different methods. The properties the paper studies include elasticity, viscosity and friction, and inferring these properties is important for enabling computer graphics applications, *e.g.*, simulate or render a realistic video. Besides, the properties of friction and viscosity should be important for embodied AI, as robots need to understand these properties to properly interact with the physical world.

### 8.1.3 Part III: Improving Visual-Language Foundation Models

**Chapter 6.** In Chapter 6, we improve CLIP-like large-scale pre-trained visual-language foundation models for text-to-image retrieval. More specifically, we focus on the re-ranking stage of text-to-image retrieval. To this end, we propose a novel architecture with a ‘student friendly’ best practice for data curation, and manage to enhance the pre-training of CLIP-like visual-language foundation models

for text-to-image retrieval with limited resources. For different models, consistent and significant improvement is observed on both standard text-to-image retrieval benchmarks and the out-of-distribution benchmarks that we newly set up. Although the paper has just been published recently, it is mentioned by subsequent work to improve cross-modal retrieval with a more expensive re-ranking method [Khaertdinov et al. 2025], and the proposed ELIP method is mentioned in a work on applications in aquaculture [Zhang et al. 2026]. We believe that in the future, this paper will bring more attention to research of re-ranking method for text-to-image retrieval, which has been relatively less explored before, compared with conventional image-to-image retrieval. On the other hand, the paper sheds light on enhancing the pre-training of large-scale visual-language foundation models with academic resources.

**Chapter 7.** In Chapter 7, we improve ChatGPT-like large-scale pre-trained visual-language foundation models for language instructed visual grounding. Previous state-of-the-art grounding visual language models usually have large model sizes, making them heavy for deployment and slow for inference. In this paper, we therefore explore enabling small VLMs to achieve state-of-the-art performance. More specifically, this paper investigates why small models fall behind large models for visual grounding. The answer is the small models have a smaller LLM so cannot understand text prompts that are semantically complicated. Therefore, to mitigate the gap, we equip small models with multi-modal reasoning capability to compete with large models, via multi-modal data curation, and a SFT-RL two-stage training. Eventually, we boost the performance of a 4B and 8B model to be higher than the state-of-the-art 235B model, while speeding up more than 5x. Although the paper has not been published, we believe that once published, it can inspire future works on efficiency, reasoning and visual capability of visual-language models. We believe our efficient model with state-of-the-art visual grounding performance can help edge deployment and downstream applications such as autonomous driving and robotics.

## 8.2 Future Work

**Occlusion Handling.** As mentioned in the ‘Conclusion and Future Work’ Section of Chapter 2, a limitation of the work is that it has only studied object detection in images. Therefore, a direct extension is to improve detection performance of occluded objects in videos using similar ideas, where multiple views of the objects and temporal cues are potentially available to help disambiguate the occlusions. For Chapter 3, a limitation is that for some types of objects, *e.g.*, articulated objects, there may be multiple possible amodal segmentations, and it is better to formulate the problem in a probabilistic way rather than the deterministic way in the future studies.

More broadly, occlusion naturally and frequently happens in images and videos, and computer vision tasks can suffer from occlusion, *i.e.*, the performance of models for these tasks degrade in occlusion scenarios. These tasks include object detection [Zhan et al. 2022; He et al. 2024; Li and Li 2026; Liu et al. 2024c], panoptic segmentation [Wei et al. 2025], instance segmentation [Lee et al. 2024; Kirillov et al. 2023], image classification [Kotwal et al. 2024], video action detection [Modi et al. 2023], video object tracking [Van Hoorick et al. 2023], and video object segmentation [Ding et al. 2025]. Although Chapter 2 and some subsequent works as cited above have made efforts in handling the occlusion problem in different tasks, the performance gap between challenging occlusion scenarios and normal situations still remains. Therefore, future work can (i) investigate how state-of-the-art models perform under occlusion; (ii) if there is still a significant gap, try to further mitigate the gap; (iii) if the gap is not very significant, try to improve the smaller models so that they can achieve on-par performance with state-of-the-art models efficiently; (iv) study the occlusion handling in more different tasks.

On amodal perception, as mentioned in Section 8.1, following Chapter 3 there has been increasing interest in studying recovering the occluded objects in the wild in 2D. However, estimating the complete shape of occluded objects in 3D has been relatively less explored and remains an open challenge. This new ‘amodal 3D’ setting is important and worth future exploration, as it can provide a better 3D reconstruction of the scene compared with conventional methods such as NeRF [Mildenhall et al. 2021] and Gaussian Splatting [Kerbl et al. 2023], where

occluded parts of the objects are not reconstructed. A possible future avenue would be (i) developing a model for occluded objects 3D reconstruction from a single image; (ii) developing a model to reconstruct all objects in the 3D scene from a single image or a video; (iii) making the model more general for different scenarios.

**3D Physical Understanding.** Understanding 3D and physical properties in the wild is an important topic towards artificial general intelligence, especially for future embodied AI systems. However, it is an open challenge due to the limited availability of 3D and physics data, despite the advancement of the works cited in Chapter 8.1.2, which mainly focuses on specific situations or specific properties. Chapter 4 and Chapter 5 suggest that (i) the knowledge from pre-trained foundation models can be utilised for 3D physical understanding. Besides, efforts on (ii) collecting more real data, or (iii) advancing 3D physics simulators for better and more synthetic data, can contribute to the progress of making AI understand 3D and physics. This is also important for future generation of generative models – the models can generate realistic videos that observe 3D and physics laws only if these models can properly understand 3D and physics.

More specifically, for (i), other probing and feature extraction methods can also be tried in addition to the SVM-based method in Chapter 4 and the cross-attention feature extraction method in Chapter 5. Besides the 3D physical properties studied in Chapter 4 and Chapter 5, other 3D physical properties such as object orientation, contact relations, and gravity can also be explored.

For (ii), similar to Chapter 5, we can further collect more images and videos with the ground truth annotations of the 3D and physical properties. The ground truth can be manually measured or obtained from a scientific table.

For (iii), over the years, the physics simulators have been continuously evolved, but there is still a gap between the simulation of a state-of-the-art physics engine and the real world. To bridge this gap, we can possibly learn a *neural residual model* to compensate for the simulation errors.

**Improving Visual-Language Foundation Models.** As mentioned in the ‘Conclusion’ Section of Chapter 6, a limitation of the work is that it has only been ap-

plied to CLIP-like visual-language foundation models. A possible direct extension is to apply ideas similar to ELIP to efficiently enhance ChatGPT-like generative visual-language foundation models by introducing more effective text-guided visual attention and encoding.

More broadly, visual-language foundation models play an important role in lots of applications these days, and it is important to research how to improve them. Although training a large-scale visual-language foundation model with massive data and computing resources has proven to be effective, it still remains a challenge to train the model efficiently and deploy the model efficiently.

Chapter 6 suggests efficient data curation and efficient architecture can make it possible to improve visual-language foundation models with limited resources, and this encourages academia labs to join this topic. In the future, for different application scenarios where the CLIP-like visual-language foundation models are usually used, such as open-world object detection or semantic segmentation, similar ideas of re-computing the confidence scores at the decision boundary with an enhanced model can also be tried. The model can thus be trained efficiently with similar ideas as in Chapter 6.

Chapter 7 suggests the paradigm of ‘small model + reasoning > large model’, which can be further explored for other visual-language tasks to achieve good performance efficiently, such as visual question answering and temporal visual grounding in videos. If this can be achieved, we will have a visual-expert small language model with fast inference speed. While the paradigm of ‘small model + reasoning > large model’ has been more explored for coding and mathematic tasks, it is also worth exploring the benefit of efficient multi-modal reasoning for efficient visual-expert language models.

# References

- J. Achiam, S. Adler, S. Agarwal, L. Ahmad, I. Akkaya, F. L. Aleman, D. Almeida, J. Altenschmidt, S. Altman, S. Anadkat, et al. (2023). “Gpt-4 technical report”. In: *arXiv preprint arXiv:2303.08774*.
- E. H. Adelson (2001). “On seeing stuff: the perception of materials by humans and machines”. In: *Human vision and electronic imaging VI*.
- P. Agrawal, A. V. Nair, P. Abbeel, J. Malik, and S. Levine (2016). “Learning to poke by poking: Experiential learning of intuitive physics”. In: *Advances on Neural Information Processing Systems (NeurIPS)*.
- J.-B. Alayrac, J. Donahue, P. Luc, A. Miech, I. Barr, Y. Hasson, K. Lenc, A. Mensch, K. Millican, M. Reynolds, et al. (2022). “Flamingo: a visual language model for few-shot learning”. In: *Advances in neural information processing systems (NeurIPS)*.
- Alibaba (2025). *Qwen3-VL*.
- Anthropic (2025). *Claude-4.5*.
- L. Antsfeld and B. Chidlovskii (2024). “3D-Consistent Image Inpainting with Diffusion Models”. In: *arXiv preprint arXiv:2412.05881*.
- J. Ao, Y. Jiang, Q. Ke, and K. A. Ehinger (2025). “Open-world amodal appearance completion”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- R. Arandjelović and A. Zisserman (2012). “Three things everyone should know to improve object retrieval”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- J. Assen, P. Barla, and R. Fleming (2018). “Visual Features in the Perception of Liquids”. In: *Current Biology*.
- M. Assran, A. Bardes, D. Fan, Q. Garrido, R. Howes, M. Muckley, A. Rizvi, C. Roberts, K. Sinha, A. Zholus, et al. (2025). “V-JEPA 2: Self-Supervised Video

- Models Enable Understanding, Prediction and Planning”. In: *arXiv preprint arXiv:2506.09985*.
- G. Aydemir (2025). “Online Long-term Point Tracking in the Foundation Model Era”. In: *arXiv preprint arXiv:2507.09217*.
- G. Aydemir, W. Xie, and F. Güney (2024). “Can Visual Foundation Models Achieve Long-term Point Tracking?” In: *arXiv preprint arXiv:2408.13575*.
- S. Azad, Y. Jain, R. Garg, V. Vineet, and Y. Rawat (2025). “Understanding Depth and Height Perception in Large Visual-Language Models”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- M. Babaeizadeh, M. T. Saffar, S. Nair, S. Levine, C. Finn, and D. Erhan (2021). “Fitvid: Overfitting in pixel-level video prediction”. In: *arXiv preprint arXiv:2106.13195*.
- S. Bai, K. Chen, X. Liu, J. Wang, W. Ge, S. Song, K. Dang, P. Wang, S. Wang, J. Tang, et al. (2025). “Qwen2. 5-vl technical report”. In: *arXiv preprint arXiv:2502.13923*.
- R. Baillargeon, A. Needham, and J. DeVos (1992). “The development of young infants’ intuitions about support”. In: *Early development and parenting*.
- A. Baldrati, L. Agnolucci, M. Bertini, and A. Del Bimbo (2023). “Zero-shot composed image retrieval with textual inversion”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*.
- M. E. Banani, A. Raj, K.-K. Maninis, A. Kar, Y. Li, M. Rubinstein, D. Sun, L. Guibas, J. Johnson, and V. Jampani (2024). “Probing the 3D Awareness of Visual Foundation Models”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
- D. Baranchuk, A. Voynov, I. Rubachev, V. Khrulkov, and A. Babenko (2022). “Label-Efficient Semantic Segmentation with Diffusion Models”. In: *International Conference on Learning Representations (ICLR)*.
- A. Bardes, Q. Garrido, J. Ponce, X. Chen, M. Rabbat, Y. LeCun, M. Assran, and N. Ballas (2023). “V-jepa: Latent video prediction for visual representation learning”. In: *OpenReview*.
- M. R. Barın, G. Aydemir, and F. Güney (2024). “Robust Bird’s Eye View Segmentation by Adapting DINOv2”. In: *arXiv preprint arXiv:2409.10228*.
- P. W. Battaglia, J. B. Hamrick, and J. B. Tenenbaum (2013). “Simulation as an engine of physical scene understanding”. In: *Proceedings of the national academy of sciences*.

- D. Bau, J.-Y. Zhu, H. Strobelt, B. Zhou, J. B. Tenenbaum, W. T. Freeman, and A. Torralba (2019). “GAN Dissection: Visualizing and Understanding Generative Adversarial Networks”. In: *Proceedings of the International Conference on Learning Representations (ICLR)*.
- D. Bear, E. Wang, D. Mrowca, et al. (2021). “Physion: Evaluating Physical Prediction from Vision in Humans and Machines”. In: *Advances in Neural Information Processing Systems (NeurIPS), Datasets and Benchmarks Track*.
- P. Beckmann (2025). “New horizons in machine understanding: explanatory and objectual understanding in deep learning video generation models”. In: *Synthese*.
- Y. Bhargat, J. F. Henriques, and A. Zisserman (2023). “A Light Touch Approach to Teaching Transformers Multi-view Geometry”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- A. Bhattad, D. McKee, D. Hoiem, and D. Forsyth (2023). “StyleGAN knows Normal, Depth, Albedo, and More”. In: *Advances in Neural Information Processing Systems (NeurIPS)*.
- A. Bhattad, K. Preechakul, and A. A. Efros (2025). “Visual Jenga: Discovering Object Dependencies via Counterfactual Inpainting”. In: *Advances in neural information processing systems (NeurIPS)*.
- F. Bordes, Q. Garrido, J. T. Kao, A. Williams, M. Rabbat, and E. Dupoux (2025). “IntPhys 2: Benchmarking Intuitive Physics Understanding In Complex Synthetic Environments”. In: *arXiv preprint arXiv:2506.09849*.
- R. E. Briscoe (2011). “Mental imagery and the varieties of amodal perception”. In: *Pacific Philosophical Quarterly*.
- G. J. Brostow and I. A. Essa (1999). “Motion based decompositing of video”. In: *Proceedings of the Seventh IEEE International Conference on Computer Vision (ICCV)*.
- J. Burgess, K.-C. Wang, and S. Yeung-Levy (2024). “Viewpoint textual inversion: Discovering scene representations and 3d view control in 2d diffusion models”. In: *European Conference on Computer Vision (ECCV)*.
- Z. Cai and N. Vasconcelos (2019). “Cascade R-CNN: high quality object detection and instance segmentation”. In: *IEEE transactions on pattern analysis and machine intelligence (TPAMI)*.
- Z. Cai, J. Zhang, D. Ren, C. Yu, H. Zhao, S. Yi, C. K. Yeo, and C. Change Loy (2020). “Messytable: Instance association in multiple camera views”. In: *Proceedings of the European Conference on Computer Vision (ECCV)*.

- A. Canberk, M. Bondarenko, E. Ozguroglu, R. Liu, and C. Vondrick (2024). “Erasedraw: Learning to insert objects by erasing them from images”. In: *European Conference on Computer Vision(ECCV)*.
- B. Cao, A. Araujo, and J. Sim (2020). “Unifying deep local and global features for image search”. In: *Proceedings of the European Conference on Computer Vision (ECCV)*.
- N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko (2020). “End-to-end object detection with transformers”. In: *Proceedings of the European Conference on Computer Vision (ECCV)*.
- M. Caron, H. Touvron, I. Misra, H. Jégou, J. Mairal, P. Bojanowski, and A. Joulin (2021). “Emerging Properties in Self-Supervised Vision Transformers”. In: *Proceedings of the International Conference on Computer Vision (ICCV)*.
- A. Chang, A. Dai, T. Funkhouser, M. Halber, M. Niessner, M. Savva, S. Song, A. Zeng, and Y. Zhang (2017). “Matterport3D: Learning from RGB-D Data in Indoor Environments”. In: *International Conference on 3D Vision (3DV)*.
- C.-C. Chang and C.-J. Lin (2011). “LIBSVM: a library for support vector machines”. In: *ACM Transactions on Intelligent Systems and Technology (TIST)*.
- H. Chefer, S. Gur, and L. Wolf (2021). “Transformer interpretability beyond attention visualization”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
- H. Chen, G. Ding, X. Liu, Z. Lin, J. Liu, and J. Han (2020a). “Imram: Iterative matching with recurrent attention memory for cross-modal image-text retrieval”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- J. Chen, H. Hu, H. Wu, Y. Jiang, and C. Wang (2021). “Learning the best pooling strategy for visual semantic embedding”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- K. Chen, J. Pang, J. Wang, Y. Xiong, X. Li, S. Sun, W. Feng, Z. Liu, J. Shi, W. Ouyang, et al. (2019a). “Hybrid task cascade for instance segmentation”. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR)*.
- K. Chen, J. Wang, J. Pang, Y. Cao, Y. Xiong, X. Li, S. Sun, W. Feng, Z. Liu, J. Xu, et al. (2019b). “MMDetection: Open mmlab detection toolbox and benchmark”. In: *arXiv preprint arXiv:1906.07155*.

- K. Chen, D. Ramanan, and T. Khurana (2025). “Using Diffusion Priors for Video Amodal Segmentation”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- S. Chen, H. J. Müller, and M. Conci (2016). “Amodal completion in visual working memory.” In: *Journal of Experimental Psychology: Human Perception and Performance*.
- X. Chen, Z. Ye, J. Sun, Y. Fan, F. Hu, C. Wang, and C. Lu (2020b). “Transferable active grasping and real embodied dataset”. In: *2020 IEEE International Conference on Robotics and Automation (ICRA)*.
- Y.-C. Chen, L. Li, L. Yu, A. El Kholy, F. Ahmed, Z. Gan, Y. Cheng, and J. Liu (2020c). “Uniter: Universal image-text representation learning”. In: *Proceedings of the European Conference on Computer Vision (ECCV)*.
- Y. Chen, Z. Ma, Z. Zhang, Z. Qi, C. Yuan, Y. Shan, B. Li, W. Hu, X. Qie, and J. Wu (2023). “ViLEM: Visual-Language Error Modeling for Image-Text Retrieval”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- Z. Chen, J. Wu, W. Wang, W. Su, G. Chen, S. Xing, M. Zhong, Q. Zhang, X. Zhu, L. Lu, et al. (2024). “Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- B. Cheng, I. Misra, A. G. Schwing, A. Kirillov, and R. Girdhar (2022). “Masked-attention mask transformer for universal image segmentation”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- S. Chi, E. Sachdeva, P.-H. Huang, and K. Lee (2025). “Contact-Aware Amodal Completion for Human-Object Interaction via Multi-Regional Inpainting”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*.
- W. Chow, J. Mao, B. Li, D. Seita, V. Guizilini, and Y. Wang (2025). “PhysBench: Benchmarking and Enhancing Vision-Language Models for Physical World Understanding”. In: *International Conference on Learning Representation (ICLR)*.
- O. Chum, J. Philbin, J. Sivic, M. Isard, and A. Zisserman (2007). “Total recall: Automatic query expansion with a generative feature model for object retrieval”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*.

- O. Chum, A. Mikulik, M. Perdoch, and J. Matas (2011). “Total recall II: Query expansion revisited”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- S. Chun, S. J. Oh, R. S. De Rezende, Y. Kalantidis, and D. Larlus (2021). “Probabilistic embeddings for cross-modal retrieval”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- M. B. Clowes (1971). “On seeing things”. In: *Artificial intelligence*.
- G. Comanici, E. Bieber, M. Schaekermann, I. Pasupat, N. Sachdeva, I. Dhillon, M. Blistein, O. Ram, D. Zhang, E. Rosen, et al. (2025). “Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities”. In: *arXiv preprint arXiv:2507.06261*.
- A. Dai, A. X. Chang, M. Savva, M. Halber, T. Funkhouser, and M. Nießner (2017). “Scannet: Richly-annotated 3d reconstructions of indoor scenes”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
- A. Damaraju, D. Hazineh, and T. Zickler (2025). “CObL: Toward Zero-Shot Ordinal Layering without User Prompting”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*.
- D. Danier, M. Aygün, C. Li, H. Bilen, and O. Mac Aodha (2025). “DepthCues: Evaluating monocular depth perception in large vision models”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- M. Deitke, C. Clark, S. Lee, R. Tripathi, Y. Yang, J. S. Park, M. Salehi, N. Muennighoff, K. Lo, L. Soldaini, et al. (2024). “Molmo and pixmo: Open weights and open data for state-of-the-art multimodal models”. In: *arXiv preprint arXiv:2409.17146*.
- J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei (2009). “Imagenet: A large-scale hierarchical image database”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
- K. Desai and J. Johnson (2021). “Virtex: Learning visual representations from textual annotations”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- H. Dhamo, N. Navab, and F. Tombari (2019). “Object-Driven Multi-Layer Scene Decomposition From a Single Image”. In: *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*.
- H. Diao, Y. Zhang, L. Ma, and H. Lu (2021). “Similarity reasoning and filtration for image-text matching”. In: *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*.

- H. Ding, K. Ying, C. Liu, S. He, X. Jiang, Y.-G. Jiang, P. H. Torr, and S. Bai (2025). “MOSEv2: A more challenging dataset for video object segmentation in complex scenes”. In: *arXiv preprint arXiv:2508.05630*.
- M. Ding, Z. Chen, T. Du, P. Luo, J. Tenenbaum, and C. Gan (2021). “Dynamic visual reasoning by learning differentiable physics models from video and language”. In: *Advances In Neural Information Processing Systems (NeurIPS)*.
- X. Du, N. Kolkin, G. Shakhnarovich, and A. Bhattad (2023). “Generative models: What do they know? do they know things? let’s find out!” In: *arXiv preprint arXiv:2311.17137*.
- A. Dubey, A. Jauhri, A. Pandey, A. Kadian, A. Al-Dahle, A. Letman, A. Mathur, A. Schelten, A. Yang, A. Fan, et al. (2024). “The llama 3 herd of models”. In: *arXiv preprint arXiv:2407.21783*.
- K. Ehsani, R. Mottaghi, and A. Farhadi (2018). “SeGAN: Segmenting and generating the invisible”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
- M. El Banani, A. Raj, K.-K. Maninis, A. Kar, Y. Li, M. Rubinstein, D. Sun, L. Guibas, J. Johnson, and V. Jampani (2024). “Probing the 3d awareness of visual foundation models”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- M. Engilberge, L. Chevallier, P. Pérez, and M. Cord (2018). “Deep semantic-visual embedding with localization”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- S. Eppel (2024). “Do large language vision models understand 3D shapes?” In: *arXiv preprint arXiv:2412.10908*.
- D. Erhan, Y. Bengio, A. Courville, and P. Vincent (2009). “Visualizing higher-layer features of a deep network”. In: *University of Montreal*.
- M. Espinosa, C. Yang, L. Ericsson, S. McDonagh, and E. J. Crowley (2024). “There is no samantics! exploring sam as a backbone for visual understanding tasks”. In: *arXiv preprint arXiv:2411.15288*.
- P. Esser, R. Rombach, and B. Ommer (2021). “Taming transformers for high-resolution image synthesis”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
- T. Evans, N. Parthasarathy, H. Merzic, and O. J. Henaff (2024). “Data curation via joint example selection further accelerates multimodal learning”. In: *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*.

- M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman (2010). “The pascal visual object classes (voc) challenge”. In: *International Journal of Computer Vision*.
- H. Fan, L. Wang, H. Chen, Z. Huang, J. Wu, and L. Sheng (2025). “Multi-agent amodal completion: Direct synthesis with fine-grained semantic guidance”. In: *Proceedings of the 33rd ACM International Conference on Multimedia*.
- C. Finn and S. Levine (2017). “Deep visual foresight for planning robot motion”. In: *2017 IEEE International Conference on Robotics and Automation (ICRA)*.
- R. W. Fleming (2014). “Visual perception of materials and their properties”. In: *Vision research*.
- P. Follmann, R. König, P. Härtinger, M. Klostermann, and T. Böttger (2019). “Learning to see the invisible: End-to-end trainable amodal instance segmentation”. In: *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*.
- R. Fong, M. Patrick, and A. Vedaldi (2019). “Understanding deep networks via extremal perturbations and smooth masks”. In: *Proceedings of the International Conference on Computer Vision (ICCV)*.
- R. Fong and A. Vedaldi (2017). “Interpretable explanations of black boxes by meaningful perturbation”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
- D. F. Fouhey, A. Gupta, and A. Zisserman (2016). “3D shape attributes”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
- K. Fragkiadaki, P. Agrawal, S. Levine, and J. Malik (2016). “Learning Visual Predictive Models of Physics for Playing Billiards”. In: *International Conference on Learning Representations (ICLR)*.
- A. Frome, G. S. Corrado, J. Shlens, S. Bengio, J. Dean, M. Ranzato, and T. Mikolov (2013). “Devise: A deep visual-semantic embedding model”. In: *Advances in neural information processing systems (NeurIPS)*.
- S. Fu, T. Bonnen, D. Guillory, and T. Darrell (2025). “Hidden in plain sight: VLMs overlook their visual representations”. In: *arXiv preprint arXiv:2506.08008*.
- S. Y. Gadre, G. Ilharco, A. Fang, J. Hayase, G. Smyrnis, T. Nguyen, R. Marten, M. Wortsman, D. Ghosh, J. Zhang, et al. (2024). “Datacomp: In search of the next generation of multimodal datasets”. In: *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*.

- J. Gao, B. Sarkar, F. Xia, T. Xiao, J. Wu, B. Ichter, A. Majumdar, and D. Sadigh (2024). “Physically grounded vision-language models for robotic manipulation”. In: *International Conference on Robotics and Automation (ICRA)*.
- J. Gao, X. Qian, Y. Wang, T. Xiao, T. He, Z. Zhang, and Y. Fu (2023). “Coarse-to-Fine Amodal Segmentation with Shape Prior”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*.
- A. C. Garcia, J. Warchocki, J. van Gemert, D. Brinks, and N. Tomen (2025). “Learning Physics From Video: Unsupervised Physical Parameter Estimation for Continuous Dynamical Systems”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- A. Geiger, P. Lenz, and R. Urtasun (2012). “Are we ready for autonomous driving? the kitti vision benchmark suite”. In: *2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
- J. J. Gibson (1979). *The ecological approach to visual perception*.
- W. Goodwin, S. Vaze, I. Havoutis, and I. Posner (2022). “Zero-shot category-level object pose estimation”. In: *European Conference on Computer Vision (ECCV)*.
- G. Gu, S. Chun, W. Kim, H. Jun, Y. Kang, and S. Yun (2023). “Compodiff: Versatile composed image retrieval with latent diffusion”. In: *arXiv preprint arXiv:2303.11916*.
- J. Gu, J. Cai, S. R. Joty, L. Niu, and G. Wang (2018). “Look, imagine and match: Improving textual-visual cross-modal retrieval with generative models”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- D. Guo, D. Yang, H. Zhang, J. Song, R. Zhang, R. Xu, Q. Zhu, S. Ma, P. Wang, X. Bi, et al. (2025). “Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning”. In: *Nature*.
- A. Gupta, P. Dollar, and R. Girshick (2019). “Lvis: A dataset for large vocabulary instance segmentation”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- D. Hafner, T. Lillicrap, I. Fischer, R. Villegas, D. Ha, H. Lee, and J. Davidson (2019). “Learning latent dynamics for planning from pixels”. In: *International conference on machine learning (ICML)*.
- R. I. Hartley and A. Zisserman (2004). “Multiple View Geometry in Computer Vision”. In: *Cambridge university press*.
- S. Hausler, S. Garg, M. Xu, M. Milford, and T. Fischer (2021). “Patch-netvlad: Multi-scale fusion of locally-global descriptors for place recognition”. In:

- J. He, H. Chen, B. Liu, S. Luo, and J. Liu (2024). “Enhancing YOLO for occluded vehicle detection with grouped orthogonal attention and dense object repulsion”. In: *Scientific Reports*.
- K. He, G. Gkioxari, P. Dollár, and R. Girshick (2017). “Mask r-cnn”. In: *Proceedings of the IEEE international conference on computer vision (ICCV)*.
- R. He, S. Sun, X. Yu, C. Xue, W. Zhang, P. Torr, S. Bai, and X. Qi (2023). “Is synthetic data from generative models ready for image recognition?” In: *The Eleventh International Conference on Learning Representations (ICLR)*.
- E. Hedlin, G. Sharma, S. Mahajan, H. Isack, A. Kar, A. Tagliasacchi, and K. M. Yi (2023). “Unsupervised semantic correspondence using stable diffusion”. In: *Advances in Neural Information Processing Systems (NeurIPS)*.
- D. Hendrycks, S. Basart, N. Mu, S. Kadavath, F. Wang, E. Dorundo, R. Desai, T. Zhu, S. Parajuli, M. Guo, D. Song, J. Steinhardt, and J. Gilmer (2021). “The Many Faces of Robustness: A Critical Analysis of Out-of-Distribution Generalization”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*.
- C.-Y. Hsieh, T. Khurana, A. Dave, and D. Ramanan (2023). “Tracking Any Object Amodally”. In: *arXiv preprint arXiv:2312.12433*.
- C.-H. P. Huang, N. Mitra, H. Jeong, J. S. Yoon, and D. Ceylan (2025). “JOG3R: Towards 3D-Consistent Video Generators”. In: *arXiv preprint arXiv:2501.01409*.
- N. Huang and C. Xu (2025). “Multi-Level Features Fusion for Zero-Shot Object Pose Estimation”. In: *2025 IEEE International Conference on Industrial Technology (ICIT)*.
- Y. Huang, W. Wang, and L. Wang (2017). “Instance-aware image and sentence matching with selective multimodal lstm”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- D. A. Huffman (1971). “Impossible objects as nonsense sentences”. In: *Machine intelligence*.
- B. Hui, J. Yang, Z. Cui, J. Yang, D. Liu, L. Zhang, T. Liu, J. Zhang, B. Yu, K. Lu, et al. (2024). “Qwen2. 5-coder technical report”. In: *arXiv preprint arXiv:2409.12186*.
- A. Hurst, A. Lerer, A. P. Goucher, A. Perelman, A. Ramesh, A. Clark, A. Ostrow, A. Welihinda, A. Hayes, A. Radford, et al. (2024). “Gpt-4o system card”. In: *arXiv preprint arXiv:2410.21276*.

- G. Ilharco, M. Wortsman, R. Wightman, C. Gordon, N. Carlini, R. Taori, A. Dave, V. Shankar, H. Namkoong, J. Miller, H. Hajishirzi, A. Farhadi, and L. Schmidt (July 2021). *OpenCLIP*. Version 0.1. URL: <https://doi.org/10.5281/zenodo.5143773>.
- Y. Inagaki, R. Araki, T. Yamashita, and H. Fujiyoshi (2019). “Detecting layered structures of partially occluded objects for bin picking”. In: *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*.
- A. Jahanian, X. Puig, Y. Tian, and P. Isola (2022). “Generative Models as a Data Source for Multiview Representation Learning”. In: *International Conference on Learning Representations (ICLR)*.
- K. Jatavallabhula, M. Macklin, F. Golemo, V. Voleti, L. Petrini, M. Weiss, B. Considine, J. Parent-Levesque, K. Xie, K. Erleben, L. Paull, F. Shkurti, D. Nowrouzezahrai, and S. Fidler (2021). “gradSim: Differentiable simulation for system identification and visuomotor control”. In: *International Conference on Learning Representations (ICLR)*.
- H. Jegou, M. Douze, and C. Schmid (2008). “Hamming embedding and weak geometric consistency for large scale image search”. In: *Proceedings of the European Conference on Computer Vision (ECCV)*.
- Z. Ji, H. Wang, J. Han, and Y. Pang (2019). “Saliency-guided attention network for image-sentence matching”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*.
- C. Jia, Y. Yang, Y. Xia, Y.-T. Chen, Z. Parekh, H. Pham, Q. Le, Y.-H. Sung, Z. Li, and T. Duerig (2021). “Scaling up visual and vision-language representation learning with noisy text supervision”. In: *Proceedings of the International Conference on Machine Learning (ICML)*.
- M. Jia, L. Tang, B.-C. Chen, C. Cardie, S. Belongie, B. Hariharan, and S.-N. Lim (2022). “Visual prompt tuning”. In: *Proceedings of the European Conference on Computer Vision (ECCV)*.
- N. Jojic and B. J. Frey (2001). “Learning flexible sprites in video layers”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- A. Kanazawa, S. Tulsiani, A. A. Efros, and J. Malik (2018). “Learning category-specific mesh reconstruction from image collections”. In: *Proceedings of the European Conference on Computer Vision (ECCV)*.

- A. Kar, S. Tulsiani, J. Carreira, and J. Malik (2015). “Category-specific object reconstruction from a single image”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*.
- A. Karpathy, A. Joulin, and L. F. Fei-Fei (2014). “Deep fragment embeddings for bidirectional image sentence mapping”. In: *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*.
- T. Karras, S. Laine, and T. Aila (2019). “A style-based generator architecture for generative adversarial networks”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
- B. Kaup, R. Ulrich, K. M. Bausenhardt, D. Bryce, M. V. Butz, D. Dignath, C. Dudschig, V. H. Franz, C. Friedrich, C. Gawrilow, et al. (2024). “Modal and amodal cognition: an overarching principle in various domains of psychology”. In: *Psychological Research*.
- T. Kawabe, K. Maruya, R. Fleming, and S. Nishida (2014). “Seeing liquids from visual motion”. In: *Vision Research*.
- T. Kawabe and S. Nishida (2016). “Seeing Jelly: Judging Elasticity of a Transparent Object”. In: *Proceedings of the ACM Symposium on Applied Perception*.
- S. Kazemzadeh, V. Ordonez, M. Matten, and T. Berg (2014). “Referitgame: Referring to objects in photographs of natural scenes”. In: *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*.
- B. Ke, A. Obukhov, S. Huang, N. Metzger, R. C. Daudt, and K. Schindler (2024). “Repurposing diffusion-based image generators for monocular depth estimation”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- L. Ke, Y.-W. Tai, and C.-K. Tang (2021). “Deep occlusion-aware instance segmentation with overlapping bilayers”. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR)*.
- P. J. Kellman and C. M. Massey (2013). “Perceptual learning, cognition, and expertise”. In: *Psychology of learning and motivation*.
- B. Kerbl, G. Kopanas, T. Leimkühler, and G. Drettakis (2023). “3D Gaussian Splatting for Real-Time Radiance Field Rendering”. In: *ACM Transactions on Graphics (TOG)*.
- B. Khaertdinov, A. Ganesh, M. Popa, and N. Tintarev (2025). “Beyond Similarity: Two-Stage Retrieval for News Image Search”. In: *Multimedia Evaluation Workshop (MediaEval)*.

- A. Khazatsky et al. (2024). “DROID: A Large-Scale In-The-Wild Robot Manipulation Dataset”. In.
- T. Khurana and D. Ramanan (2024). “Predicting Long-horizon Futures by Conditioning on Geometry and Time”. In: *arXiv preprint arXiv:2404.11554*.
- D. Kim, N. Kim, and S. Kwak (2023). “Improving cross-modal retrieval with set of diverse embeddings”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- S.-K. Kim and M. Likhachev (2016). “Planning for grasp selection of partially occluded objects”. In: *2016 IEEE International Conference on Robotics and Automation (ICRA)*.
- D. P. Kingma and M. Welling (2014). “Auto-encoding variational Bayes”. In: *Proceedings of the International Conference on Learning Representations (ICLR)*.
- A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, T. Xiao, S. Whitehead, A. C. Berg, W.-Y. Lo, et al. (2023). “Segment anything”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*.
- P. Kocsis, J. Philip, K. Sunkavalli, M. Nießner, and Y. Hold-Geoffroy (2024). “Lightit: Illumination modeling and control for diffusion models”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- G. Kordopatis-Zilos, V. Stojnić, A. Manko, P. Suma, N.-A. Ypsilantis, N. Efthymiadis, Z. Laskar, J. Matas, O. Chum, and G. Toliás (2025). “Ilias: Instance-level image retrieval at scale”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- K. Kotwal, T. Deshmukh, and P. Gopal (2024). “Latent Enhancing Autoencoder for Occluded Image Classification”. In: *2024 IEEE International Conference on Image Processing (ICIP)*.
- P. Kumar, P. H. S. Torr, and A. Zisserman (2008). “Learning layered motion segmentations of video”. In: *International Journal of Computer Vision (IJCV)*.
- A. Kuznetsova, H. Rom, N. Alldrin, J. Uijlings, I. Krasin, J. Pont-Tuset, S. Kamali, S. Popov, M. Mallocci, A. Kolesnikov, et al. (2020). “The open images dataset v4”. In: *International Journal of Computer Vision (IJCV)*.
- W. Kwon, Z. Li, S. Zhuang, Y. Sheng, L. Zheng, C. H. Yu, J. E. Gonzalez, H. Zhang, and I. Stoica (2023). “Efficient Memory Management for Large Language Model Serving with PagedAttention”. In: *Proceedings of the ACM SIGOPS 29th Symposium on Operating Systems Principles*.

- H. Lee, S. Hwang, and S. Kwak (2024). “Extreme point supervised instance segmentation”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- H. Lee and S. P. Vecera (2005). “Visual cognition influences early vision: the role of visual short-term memory in amodal completion”. In: *Psychological Science*.
- H. Lee and J. Park (2022). “Instance-wise occlusion and depth orders in natural scenes”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- J. J. Lee, B. Benes, and R. A. Yeh (2025). “Tuning-Free Amodal Segmentation via the Occlusion-Free Bias of Inpainting Models”. In: *arXiv preprint arXiv:2503.18947*.
- K.-H. Lee, X. Chen, G. Hua, H. Hu, and X. He (2018). “Stacked cross attention for image-text matching”. In: *Proceedings of the European conference on computer vision (ECCV)*.
- B. Li, Y. Zhang, D. Guo, R. Zhang, F. Li, H. Zhang, K. Zhang, P. Zhang, Y. Li, Z. Liu, et al. (2024a). “Llava-onevision: Easy visual task transfer”. In: *arXiv preprint arXiv:2408.03326*.
- J. Li, D. Li, S. Savarese, and S. Hoi (2023a). “Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models”. In: *Proceedings of the International Conference on Machine Learning (ICML)*.
- J. Li, R. Selvaraju, A. Gotmare, S. Joty, C. Xiong, and S. C. H. Hoi (2021). “Align before fuse: Vision and language representation learning with momentum distillation”. In: *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*.
- K. Li and J. Malik (2016). “Amodal instance segmentation”. In: *Proceedings of the European Conference on Computer Vision (ECCV)*.
- K. Li, Y. Zhang, K. Li, Y. Li, and Y. Fu (2019). “Visual semantic reasoning for image-text matching”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*.
- R. Li, C. Zheng, C. Rupprecht, and A. Vedaldi (2025a). “DSO: Aligning 3D Generators with Simulation Feedback for Physical Soundness”. In: *arXiv preprint arXiv:2503.22677*.
- S. Li, T. Hu, J. van de Weijer, F. S. Khan, T. Liu, L. Li, S. Yang, Y. Wang, M.-M. Cheng, and J. Yang (2024b). “Faster diffusion: Rethinking the role of the encoder for diffusion model inference”. In: *Advances in Neural Information Processing Systems (NeurIPS)*.

- X. Li, C. Yi, J. Lai, M. Lin, Y. Qu, S. Zhang, and L. Cao (2025b). “Synergyamodal: Deocclude anything with text control”. In: *Proceedings of the 33rd ACM International Conference on Multimedia (ACM MM)*.
- X. Li and X. Li (2026). “Insight any invisible: An occlusion perception method for robust pedestrian detection in crowded scenes”. In: *Expert Systems with Applications*.
- Y. Li, R. Wu, H. Lu, C. Ning, Y. Shen, G. Zhan, and H. Dong (2024c). “Broadcasting support relations recursively from local dynamics for object retrieval in clutters”. In: *arXiv preprint arXiv:2406.02283*.
- Y. Li, T. Lin, K. Yi, D. Bear, D. L. Yamins, J. Wu, J. B. Tenenbaum, and A. Torralba (2020). “Visual Grounding of Learned Physical Models”. In: *International Conference on Machine Learning (ICML)*.
- Z. Li, M. Lavreniuk, J. Shi, S. F. Bhat, and P. Wonka (2025c). “Amodal depth anything: Amodal depth estimation in the wild”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*.
- Z. Li, Y. Liu, C. Hui, J. Lee, S. Lee, and W. Lin (2025d). “Shape Distribution Matters: Shape-specific Mixture-of-Experts for Amodal Segmentation under Diverse Occlusions”. In: *arXiv preprint arXiv:2508.01664*.
- Z. Li, Y. Liu, C. Hui, and W. Lin (2025e). “Single Point, Full Mask: Velocity-Guided Level Set Evolution for End-to-End Amodal Segmentation”. In: *arXiv preprint arXiv:2508.01661*.
- Z. Li, W. Ye, T. Jiang, and T. Huang (2022). “2D Amodal Instance Segmentation Guided by 3D Shape Prior”. In: *Proceedings of the European Conference on Computer Vision (ECCV)*.
- Z. Li, W. Ye, T. Jiang, and T. Huang (2023b). “GIN: Generative INvariant Shape Prior for Amodal Instance Segmentation”. In: *IEEE Transactions on Multimedia*.
- Z. Li, W. Ye, J. Terven, Z. Bennett, Y. Zheng, T. Jiang, and T. Huang (2023c). “MUVA: A New Large-Scale Benchmark for Multi-View Amodal Instance Segmentation in the Shopping Scenario”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*.
- Z. Li, H. Yoon, S. Lee, and W. Lin (2025f). “Unveiling the Invisible: Reasoning Complex Occlusions Amodally with AURA”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*.
- Z. Li, Q. Zhou, X. Zhang, Y. Zhang, Y. Wang, and W. Xie (2023d). “Open-vocabulary Object Segmentation with Diffusion Models”. In: *Proceedings of the International Conference on Computer Vision (ICCV)*.

- J. Lin, H. Yin, W. Ping, P. Molchanov, M. Shoeybi, and S. Han (2024). “Vila: On pre-training for visual language models”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick (2014). “Microsoft coco: Common objects in context”. In: *Proceedings of the European Conference on Computer Vision (ECCV)*.
- H. Ling, D. Acuna, K. Kreis, S. W. Kim, and S. Fidler (2020). “Variational amodal object completion”. In: *Advances in Neural Information Processing Systems (NeurIPS)*.
- C. Liu, K. Kim, J. Gu, Y. Furukawa, and J. Kautz (2019a). “Planercnn: 3d plane detection and reconstruction from a single image”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- C. Liu, Z. Mao, A.-A. Liu, T. Zhang, B. Wang, and Y. Zhang (2019b). “Focus your attention: A bidirectional focal attention network for image-text matching”. In: *Proceedings of the 27th ACM International Conference on Multimedia (ACMMM)*.
- C. Liu, Z. Mao, T. Zhang, H. Xie, B. Wang, and Y. Zhang (2020). “Graph structured network for image-text matching”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- H. Liu, C. Li, Q. Wu, and Y. J. Lee (2023a). “Visual instruction tuning”. In: *Advances in neural information processing systems (NeurIPS)*.
- R. Liu, X. Wang, W. Wang, and Y. Yang (2023b). “Bird’s-eye-view scene graph for vision-language navigation”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*.
- Y. Liu, P. Chen, J. Cai, X. Jiang, Y. Hu, J. Yao, Y. Wang, and W. Xie (2025a). “LamRA: Large Multimodal Model as Your Advanced Retrieval Assistant”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- Y. Liu, J. Yao, Y. Wang, and W. Xie (2023c). “Zero-shot Composed Text-Image Retrieval”. In: *British Machine Vision Conference (BMVC)*.
- Y. Liu, K. Zhang, Y. Li, Z. Yan, C. Gao, R. Chen, Z. Yuan, Y. Huang, H. Sun, J. Gao, et al. (2024a). “Sora: A review on background, technology, limitations, and opportunities of large vision models”. In: *arXiv preprint arXiv:2402.17177*.
- Z. Liu, H. Hu, Y. Lin, Z. Yao, Z. Xie, Y. Wei, J. Ning, Y. Cao, Z. Zhang, L. Dong, F. Wei, and B. Guo (2022). “Swin Transformer V2: Scaling Up Capacity and Resolution”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

- Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo (2021). “Swin Transformer: Hierarchical Vision Transformer using Shifted Windows”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*.
- Z. Liu, L. Qiao, X. Chu, L. Ma, and T. Jiang (2025b). “Towards Efficient Foundation Model for Zero-shot Amodal Segmentation”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- Z. Liu, Q. Liu, C. Chang, J. Zhang, D. Pakhomov, H. Zheng, Z. Lin, D. Cohen-Or, and C.-W. Fu (2024b). “Object-level scene deocclusion”. In: *ACM SIGGRAPH 2024 Conference Papers*.
- Z. Liu, L. Zhu, B. Shi, Z. Zhang, Y. Lou, S. Yang, H. Xi, S. Cao, Y. Gu, D. Li, et al. (2025c). “Nvila: Efficient frontier visual language models”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- Z. Liu, Z. Shi, R. Liu, L. Liu, T. Yamamoto, and D. Uchida (2024c). “Self-Checkout Product Detection with Occlusion Layer Prediction and Intersection Weighting”. In: *2024 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*.
- Z. Long, X. Ge, R. McCreadie, and J. M. Jose (2024). “CFIR: Fast and Effective Long-Text To Image Retrieval for Large Corpora”. In: *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*.
- H. Lu, G. Yang, N. Fei, Y. Huo, Z. Lu, P. Luo, and M. Ding (2023). “VDT: An Empirical Study on Video Diffusion with Transformers”. In: *Proceedings of the International Conference on Learning Representations (ICLR)*.
- R. Lu, Y. Chen, Y. Liu, J. Tang, J. Ni, D. Wan, G. Zeng, and S. Huang (2025). “Taco: Taming diffusion for in-the-wild video amodal completion”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*.
- T. Lüddecke and A. S. Ecker (2025). “Characterizing Vision Backbones for Dense Prediction with Dense Attentive Probing”. In: *Transactions on Machine Learning Research*.
- G. Luo, L. Dunlap, D. H. Park, A. Holynski, and T. Darrell (2023). “Diffusion Hyperfeatures: Searching Through Time and Space for Semantic Correspondence”. In: *Advances in Neural Information Processing Systems (NeurIPS)*.
- G. Lupyan (2012). “What do words do? Toward a theory of language-augmented thought”. In: *Psychology of learning and motivation*.

- A. Mahendran and A. Vedaldi (2015). “Understanding deep image representations by inverting them”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
- Y. Man, S. Zheng, Z. Bao, M. Hebert, L. Gui, and Y.-X. Wang (2024). “Lexicon3d: Probing visual foundation models for complex 3d scene understanding”. In: *Advances in Neural Information Processing Systems(NeurIPS)*.
- J. Mao, J. Huang, A. Toshev, O. Camburu, A. L. Yuille, and K. Murphy (2016). “Generation and comprehension of unambiguous object descriptions”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- A. Miech, J.-B. Alayrac, I. Laptev, J. Sivic, and A. Zisserman (2021). “Thinking fast and slow: Efficient text-to-visual retrieval with transformers”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition(CVPR)*.
- B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng (2021). “Nerf: Representing scenes as neural radiance fields for view synthesis”. In: *Communications of the ACM*.
- R. Modi, V. Vineet, and Y. Rawat (2023). “On occlusions in video action detection: Benchmark datasets and training recipes”. In: *Advances in Neural Information Processing Systems(NeurIPS)*.
- R. Mohan and A. Valada (2022). “Amodal panoptic segmentation”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- M. Murray (2025). “Scaling Human Supervision for Robotic Manipulation”. PhD thesis. University of Washington.
- M. Murray, E. Su, and M. Cakmak (2024). “Diffusion-PbD: Generalizable Robot Programming by Demonstration with Diffusion Features”. In: *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*.
- K. Nguyen and S. Todorovic (2021). “A weakly supervised amodal segmenter with boundary uncertainty estimation”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*.
- S. Y. Noh and J. Y. Chang (2025). “Stable Diffusion-Based Approach for Human De-Occlusion”. In: *Proceedings of the 33rd ACM International Conference on Multimedia (ACM MM)*.
- J. Norman, E. Wiesemann, H. Norman, M. Taylor, and W. Craft (2007). “The Visual Discrimination of Bending”. In: *Perception*.
- P. O. O Pinheiro, R. Collobert, and P. Dollár (2015). “Learning to segment object candidates”. In: *Advances in neural information processing systems (NeurIPS)*.

- A. v. d. Oord, Y. Li, and O. Vinyals (2018). “Representation learning with contrastive predictive coding”. In: *arXiv preprint arXiv:1807.03748*.
- M. Oquab, T. Darcet, T. Moutakanni, H. Vo, M. Szafraniec, V. Khalidov, P. Fernandez, D. Haziza, F. Massa, A. El-Nouby, et al. (2023). “Dinov2: Learning robust visual features without supervision”. In: *arXiv preprint arXiv:2304.07193*.
- V. Ordonez, G. Kulkarni, and T. Berg (2011). “Im2text: Describing images using 1 million captioned photographs”. In: *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*.
- E. Ozguroglu, R. Liu, D. Surís, D. Chen, A. Dave, P. Tokmakov, and C. Vondrick (2024). “pix2gestalt: Amodal segmentation by synthesizing wholes”. In: *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- R. Parihar, V. Agrawal, S. VS, and V. B. Radhakrishnan (2025). “Compass Control: Multi Object Orientation Control for Text-to-Image Generation”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- S. Patni, A. Agarwal, and C. Arora (2024). “ECoDepth: Effective Conditioning of Diffusion Models for Monocular Depth Estimation”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- V. Paulun, T. Kawabe, S. Nishida, and R. Fleming (2015). “Seeing liquids from static snapshots”. In: *Vision research*.
- V. Paulun, F. Schmidt, J. Assen, and R. Fleming (2017). “Shape, motion, and optical cues to stiffness of elastic objects”. In: *Journal of Vision*.
- V. C. Paulun and R. W. Fleming (2020). “Visually inferring elasticity from the motion trajectory of bouncing cubes”. In: *Journal of Vision*.
- J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman (2007). “Object retrieval with large vocabularies and fast spatial matching”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- B. A. Plummer, L. Wang, C. M. Cervantes, J. C. Caicedo, J. Hockenmaier, and S. Lazebnik (2015). “Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*.
- J. Qi, Y. Gao, Y. Hu, X. Wang, X. Liu, X. Bai, S. Belongie, A. Yuille, P. Torr, and S. Bai (2022). “Occluded Video Instance Segmentation: A Benchmark”. In: *International Journal of Computer Vision*.
- L. Qi, L. Jiang, S. Liu, X. Shen, and J. Jia (2019). “Amodal Instance Segmentation With KINS Dataset”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

- S. Qi, J. Xie, H. Yan, and S. Guo (2023). “DenseXFormer: An Effective Occluded Human Instance Segmentation Network based on Density Map for Nursing Robot”. In: *2023 IEEE International Conference on Robotics and Biomimetics (ROBIO)*.
- L. Qu, M. Liu, W. Wang, Z. Zheng, L. Nie, and T.-S. Chua (2023). “Learnable Pillar-based Re-ranking for Image-Text Retrieval”. In: *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*.
- Y. Qu, S. Dai, X. Li, Y. Wang, Y. Shen, L. Cao, and R. Ji (2025). “DeOcc-1-to-3: 3D De-Occlusion from a Single Image via Self-Supervised Multi-View Diffusion”. In: *arXiv preprint arXiv:2506.21544*.
- A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, et al. (2021). “Learning transferable visual models from natural language supervision”. In: *Proceedings of the International Conference on Machine Learning (ICML)*.
- R. Ranftl, A. Bochkovskiy, and V. Koltun (2021). “Vision transformers for dense prediction”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*.
- R. Ranftl, K. Lasinger, D. Hafner, K. Schindler, and V. Koltun (2020). “Towards Robust Monocular Depth Estimation: Mixing Datasets for Zero-shot Cross-dataset Transfer”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*.
- N. D. Reddy, R. Tamburo, and S. G. Narasimhan (2022). “WALT: Watch And Learn 2D amodal representation from Time-lapse imagery”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- S. Ren, K. He, R. Girshick, and J. Sun (2015). “Faster r-cnn: Towards real-time object detection with region proposal networks”. In: *Advances in neural information processing systems (NeurIPS)*.
- R. Riochet, M. Y. Castro, M. Bernard, A. Lerer, R. Fergus, V. Izard, and E. Dupoux (2022). “IntPhys 2019: A Benchmark for Visual Intuitive Physics Understanding”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*.
- R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer (2022). “High-resolution image synthesis with latent diffusion models”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
- K. Saleh, S. Szénási, and Z. Vámosy (2021). “Occlusion Handling in Generic Object Detection: A Review”. In: *2021 IEEE 19th World Symposium on Applied Machine Intelligence and Informatics (SAMi)*.

- K. Saleh, S. Szénási, and Z. Vámosy (2024). “Mask guided gated convolution for amodal content completion”. In: *2024 IEEE 22nd Jubilee International Symposium on Intelligent Systems and Informatics (SISY)*.
- A. Sarkar, H. Mai, A. Mahapatra, S. Lazebnik, D. A. Forsyth, and A. Bhattad (2024). “Shadows Don’t Lie and Lines Can’t Bend! Generative Models don’t know Projective Geometry... for now”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- C. Schuhmann, R. Beaumont, R. Vencu, C. Gordon, R. Wightman, M. Cherti, T. Coombes, A. Katta, C. Mullis, M. Wortsman, et al. (2022). “Laion-5b: An open large-scale dataset for training next generation image-text models”. In: *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*.
- R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra (2017). “Grad-cam: Visual explanations from deep networks via gradient-based localization”. In: *Proceedings of the International Conference on Computer Vision (ICCV)*.
- D. Selvaratnam and D. Bazazian (2025). “3D reconstruction in robotics: A comprehensive review”. In: *Computers & Graphics*.
- M. S. Seyfioglu, K. Bouyarmane, S. Kumar, A. Tavanaei, and I. B. Tutar (2024). “Diffuse to choose: Enriching image conditioned inpainting in latent diffusion models for virtual try-all”. In: *arXiv preprint arXiv:2401.13795*.
- Z. Shao, P. Wang, Q. Zhu, R. Xu, J. Song, X. Bi, H. Zhang, M. Zhang, Y. K. Li, Y. Wu, and D. Guo (2024). “DeepSeekMath: Pushing the Limits of Mathematical Reasoning in Open Language Models”. In: *arXiv preprint arXiv:2402.03300*.
- P. Sharma, N. Ding, S. Goodman, and R. Soiccut (2018). “Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning”. In: *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (ACL)*.
- P. Sharma, J. Philip, M. Gharbi, B. Freeman, F. Durand, and V. Deschaintre (2023). “Materialistic: Selecting Similar Materials in Images”. In: *ACM Transactions on Graphics (TOG)*.
- H. Shen, T. Wu, Q. Han, Y. Hsieh, J. Wang, Y. Zhang, Y. Cheng, Z. Hao, Y. Ni, X. Wang, et al. (2025). “PhyX: Does Your Model Have the " Wits" for Physical Reasoning?” In: *arXiv preprint arXiv:2505.15929*.
- G. Sheng, C. Zhang, Z. Ye, X. Wu, W. Zhang, R. Zhang, Y. Peng, H. Lin, and C. Wu (2024). “HybridFlow: A Flexible and Efficient RLHF Framework”. In: *European Conference on Computer Systems*.

- Z. Shi, Y. Shen, J. Zhu, D.-Y. Yeung, and Q. Chen (2022). “3d-aware indoor scene synthesis with depth priors”. In: *European Conference on Computer Vision (ECCV)*.
- N. Silberman, D. Hoiem, P. Kohli, and R. Fergus (2012). “Indoor segmentation and support inference from rgb-d images.” In: *Proceedings of the European Conference on Computer Vision (ECCV)*.
- K. Simonyan, A. Vedaldi, and A. Zisserman (2014). “Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps”. In: *Proceedings of the International Conference on Learning Representations (ICLR)*.
- N. Sogi, T. Shibata, and M. Terao (2024). “Object-Aware Query Perturbation for Cross-Modal Image-Text Retrieval”. In: *Proceedings of the European Conference on Computer Vision (ECCV)*.
- Y. Song and M. Soleymani (2019). “Polysemous visual-semantic embedding for cross-modal retrieval”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- Y. Song, X. Liu, and M. Z. Shou (2025). “DiffSim: Taming diffusion models for evaluating visual similarity”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*.
- M. Stary, J. Gaubil, A. Tewari, and V. Sitzmann (2025). “Understanding Multi-View Transformers”. In: *arXiv preprint arXiv:2510.24907*.
- S. Stern, I. Sobol, and O. Litany (2025). “Appreciate the View: A Task-Aware Evaluation Framework for Novel View Synthesis”. In: *arXiv preprint arXiv:2511.12675*.
- D. Sun, E. B. Sudderth, and M. J. Black (2012). “Layered segmentation and optical flow estimation over time”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- D. Sun, J. Wulff, E. B. Sudderth, H. Pfister, and M. J. Black (2013). “A fully-connected layered model of foreground and background flow”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- Y. Sun, A. Kortylewski, and A. Yuille (2022). “Amodal Segmentation Through Out-of-Task and Out-of-Distribution Generalization With a Bayesian Model”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- F. Tan, J. Yuan, and V. Ordonez (2021). “Instance-level image retrieval using reranking transformers”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*.

- H. Tan, J. Pan, J. Lin, T. Chen, Z. Zheng, Z. Tang, and H. Yang (2025). “GTPO and GRPO-S: Token and Sequence-Level Reward Shaping with Policy Entropy”. In: *arXiv preprint arXiv:2508.04349*.
- M. K. Tanenhaus, M. J. Spivey-Knowlton, K. M. Eberhard, and J. C. Sedivy (1995). “Integration of visual and linguistic information in spoken language comprehension”. In: *Science*.
- L. Tang, M. Jia, Q. Wang, C. P. Phoo, and B. Hariharan (2023). “Emergent Correspondence from Image Diffusion”. In: *Advances in Neural Information Processing Systems (NeurIPS)*.
- G. Team, R. Anil, S. Borgeaud, J.-B. Alayrac, J. Yu, R. Soricut, J. Schalkwyk, A. M. Dai, A. Hauth, K. Millican, et al. (2023). “Gemini: a family of highly capable multimodal models”. In: *arXiv preprint arXiv:2312.11805*.
- G. Team, P. Georgiev, V. I. Lei, R. Burnell, L. Bai, A. Gulati, G. Tanzer, D. Vincent, Z. Pan, S. Wang, et al. (2024). “Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context”. In: *arXiv preprint arXiv:2403.05530*.
- C. Thomas and A. Kovashka (2020). “Preserving semantic neighborhoods for robust cross-modal retrieval”. In: *Proceedings of the European Conference on Computer Vision (ECCV)*.
- V. V. Thozhiyoor, S. Tripathi, V. B. Radhakrishnan, and A. Bhattad (2025). “Objects in Generated Videos Are Slower Than They Appear: Models Suffer Sub-Earth Gravity and Don’t Know Galileo’s Principle... for now”. In: *arXiv preprint arXiv:2512.02016*.
- G. Toliás and H. Jégou (2014). “Visual query expansion with or without geometry: refining local descriptors by feature aggregation”. In: *Pattern recognition*.
- H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar, et al. (2023a). “Llama: Open and efficient foundation language models”. In: *arXiv preprint arXiv:2302.13971*.
- H. Touvron, L. Martin, K. Stone, P. Albert, A. Almahairi, Y. Babaei, N. Bashlykov, S. Batra, P. Bhargava, S. Bhosale, et al. (2023b). “Llama 2: Open foundation and fine-tuned chat models”. In: *arXiv preprint arXiv:2307.09288*.
- M. Tran, T. Pham, W. Bounsavy, T. Nguyen, and N. Le (2025). “A2VIS: Amodal-Aware Approach to Video Instance Segmentation”. In: *Image and Vision Computing*.
- M. Tran, K. Vo, T. Nguyen, and N. Le (2024). “Amodal instance segmentation with diffusion shape prior estimation”. In: *Proceedings of the Asian Conference on Computer Vision (ACCV)*.

- M. Tran, K. Vo, K. Yamazaki, A. Fernandes, M. Kidd, and N. Le (2022). “AISFormer: Amodal Instance Segmentation with Transformer”. In: *British Machine Vision Conference (BMVC)*.
- Y.-J. Tsai, B. Price, Q. Liu, L. Figueroa, D. Pakhomov, Z. Ding, S. Cohen, and M.-H. Yang (2025). “CompleteMe: Reference-based Human Image Completion”. In: *arXiv preprint arXiv:2504.20042*.
- M. Tschannen, A. Gritsenko, X. Wang, M. F. Naeem, I. Alabdulmohsin, N. Parthasarathy, T. Evans, L. Beyrer, Y. Xia, B. Mustafa, et al. (2025). “SigLIP 2: Multilingual Vision-Language Encoders with Improved Semantic Understanding, Localization, and Dense Features”. In: *arXiv preprint arXiv:2502.14786*.
- H.-Y. Tung, M. Ding, Z. Chen, D. Bear, C. Gan, J. Tenenbaum, D. Yamins, J. Fan, and K. Smith (2023). “Physion++: Evaluating physical scene understanding that requires online inference of different physical properties”. In: *Advances in Neural Information Processing Systems (NeurIPS)*.
- P. Upchurch and R. Niu (2022). “A Dense Material Segmentation Dataset for Indoor and Outdoor Scene Parsing”. In: *European Conference on Computer Vision (ECCV)*.
- B. Van Hoorick, P. Tokmakov, S. Stent, J. Li, and C. Vondrick (2023). “Tracking through containers and occluders in the wild”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- J. Varley, C. DeChant, A. Richardson, J. Ruales, and P. Allen (2017). “Shape completion enabled robotic grasping”. In: *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*.
- P. K. A. Vasu, H. Pouransari, F. Faghri, R. Vemulapalli, and O. Tuzel (2024). “Mobileclip: Fast image-text models through multi-modal reinforced training”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- E. Vendrow, O. Pantazis, A. Shepard, G. Brostow, K. Jones, O. Mac Aodha, S. Beery, and G. Van Horn (2024). “INQUIRE: A natural world text-to-image retrieval benchmark”. In: *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*.
- L. Ventura, A. Yang, C. Schmid, and G. Varol (2024). “CoVR: Learning composed video retrieval from web video captions”. In: *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*.

- O. Vinyals, A. Toshev, S. Bengio, and D. Erhan (2015). “Show and tell: A neural image caption generator”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- V. Voleti, A. Jolicoeur-Martineau, and C. Pal (2022). “Masked conditional video diffusion for prediction, generation, and interpolation”. In: *arXiv preprint arXiv:2205.09853*.
- B. Von Gimborn, P. Ausserlechner, M. Vincze, and S. Thalhammer (2024). “Diffusion Features for Zero-Shot 6DoF Object Pose Estimation”. In: *arXiv preprint arXiv:2411.16668*.
- S. Walton, A. Hassani, X. Xu, Z. Wang, and H. Shi (2025). “Efficient Image Generation with Variadic Attention Heads”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- A. Wang, Y. Sun, A. Kortylewski, and A. L. Yuille (2020a). “Robust Object Detection Under Occlusion With Context-Aware CompositionalNets”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- B. Wang, P. Kry, Y. Deng, U. Ascher, H. Huang, and B. Chen (2018a). “Neural material: Learning elastic constitutive material and damping models from sparse data”. In: *arXiv preprint arXiv:1808.04931*.
- H. Wang, Y. Zhang, Z. Ji, Y. Pang, and L. Ma (2020b). “Consensus-aware visual-semantic embedding for image-text matching”. In: *Proceedings of the European Conference on Computer Vision (ECCV)*.
- J. Y. A. Wang and E. H. Adelson (1994). “Representing Moving Images with Layers”. In: *The IEEE Transactions on Image Processing Special Issue: Image Sequence Compression*.
- K. Wang, M. Aanjaneya, and K. Bekris (2020c). “A first principles approach for data-efficient system identification of spring-rod systems via differentiable physics engines”. In: *Learning for Dynamics and Control*.
- L. Wang, N. Wu, H. Wu, W. Yu, F. Zhang, and S. Chen (2025a). “GCVPN: A Graph Convolutional Visual Prior-Transform Network for Actual Occluded Image Recognition”. In: *Proceedings of the 34th ACM International Conference on Information and Knowledge Management*.
- L. Wang, Y. Li, J. Huang, and S. Lazebnik (2018b). “Learning two-branch neural networks for image-text matching tasks”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*.

- P. Wang, S. Bai, S. Tan, S. Wang, Z. Fan, J. Bai, K. Chen, X. Liu, J. Wang, W. Ge, et al. (2024a). “Qwen2-vl: Enhancing vision-language model’s perception of the world at any resolution”. In: *arXiv preprint arXiv:2409.12191*.
- T. Wang, G. Hu, and H. Wang (2024b). “Object pose estimation via the aggregation of diffusion features”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- T. Wang, X. Hu, C.-W. Fu, and P.-A. Heng (2021). “Single-Stage Instance Shadow Detection With Bidirectional Relation Learning”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
- T. Wang, X. Hu, Q. Wang, P.-A. Heng, and C.-W. Fu (2020d). “Instance Shadow Detection”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
- W. Wang, Z. Gao, L. Gu, H. Pu, L. Cui, X. Wei, Z. Liu, L. Jing, S. Ye, J. Shao, et al. (2025b). “Internvl3. 5: Advancing open-source multimodal models in versatility, reasoning, and efficiency”. In: *arXiv preprint arXiv:2508.18265*.
- W. Wang, H. Bao, L. Dong, J. Bjorck, Z. Peng, Q. Liu, K. Aggarwal, O. K. Mohammed, S. Singhal, S. Som, et al. (2023a). “Image as a foreign language: Beit pretraining for vision and vision-language tasks”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- Z. Wang, Z. Gao, K. Guo, Y. Yang, X. Wang, and H. T. Shen (2023b). “Multilateral semantic relations modeling for image text retrieval”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- Z. Wang, X. Liu, H. Li, L. Sheng, J. Yan, X. Wang, and J. Shao (2019). “Camp: Cross-modal adaptive message passing for text-image retrieval”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*.
- W. H. Warren Jr, E. E. Kim, and R. Husney (1987). “The way the ball bounces: visual and auditory perception of elasticity and control of the bounce pass”. In: *Perception*.
- C. Wei, Y. Chen, H. Chen, H. Hu, G. Zhang, J. Fu, A. Ritter, and W. Chen (2024). “Uniir: Training and benchmarking universal multimodal information retrievers”. In: *Proceedings of the European Conference on Computer Vision (ECCV)*.
- J. Wei, X. Xu, Y. Yang, Y. Ji, Z. Wang, and H. T. Shen (2020a). “Universal weighting metric learning for cross-modal matching”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

- W. Wei, J. Wang, and A. Bhalerao (2025). “COCO-OLAC: A Benchmark for Occluded Panoptic Segmentation and Image Understanding”. In: *ICASSP 2025-2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*.
- X. Wei, T. Zhang, Y. Li, Y. Zhang, and F. Wu (2020b). “Multi-modality cross attention network for image and sentence matching”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- L. Wen, D. Du, Z. Cai, Z. Lei, M.-C. Chang, H. Qi, J. Lim, M.-H. Yang, and S. Lyu (2020). “UA-DETRAC: A new benchmark and protocol for multi-object detection and tracking”. In: *Computer Vision and Image Understanding*.
- Wikipedia contributors (2025). *Amodal completion* — *Wikipedia, The Free Encyclopedia*. URL: [https://en.wikipedia.org/wiki/Amodal\\_completion](https://en.wikipedia.org/wiki/Amodal_completion).
- J. Winn and J. Shotton (2006). “The layout consistent random field for recognizing and segmenting partially occluded objects”. In: *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*.
- J. Wu, J. J. Lim, H. Zhang, J. B. Tenenbaum, and W. T. Freeman (2016). “Physics 101: Learning Physical Object Properties from Unlabeled Videos.” In: *British Machine Vision Conference (BMVC)*.
- J. Wu, I. Yildirim, J. J. Lim, B. Freeman, and J. Tenenbaum (2015). “Galileo: Perceiving physical object properties by integrating a physics engine with deep learning”. In: *Advances in neural information processing systems (NeurIPS)*.
- R. Wu, K. Cheng, Y. Zhao, C. Ning, G. Zhan, and H. Dong (2023a). “Learning environment-aware affordance for 3d articulated object manipulation under occlusions”. In: *Advances in Neural Information Processing Systems (NeurIPS)*.
- S. Wu, R. Li, T. Jakab, C. Rupprecht, and A. Vedaldi (2023b). “MagicPony: Learning Articulated 3D Animals in the Wild”. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR)*.
- T. Wu, C. Zheng, F. Guan, A. Vedaldi, and T.-J. Cham (2025). “Amodal3r: Amodal 3d reconstruction from occluded 2d images”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*.
- W. Wu, Y. Zhao, H. Chen, Y. Gu, R. Zhao, Y. He, H. Zhou, M. Z. Shou, and C. Shen (2023c). “Datasetdm: Synthesizing data with perception annotations using diffusion models”. In: *Advances in Neural Information Processing Systems (NeurIPS)*.
- W. Wu, Y. Zhao, M. Z. Shou, H. Zhou, and C. Shen (2023d). “Diffumask: Synthesizing images with pixel-level annotations for semantic segmentation using diffusion models”. In: *Proceedings of the International Conference on Computer Vision (ICCV)*.

- Z. Wu, D. Lischinski, and E. Shechtman (2021). “StyleSpace analysis: Disentangled controls for stylegan image generation”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- J. Wulff and M. J. Black (2015). “Efficient sparse-to-dense optical flow estimation using a learned basis and layers”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- J. Wulff and M. J. Black (2014). “Modeling blurred video with layers”. In: *Proceedings of the European Conference on Computer Vision (ECCV)*.
- xAI (2025). *Grok-4*.
- Y. Xia, R. Ding, Z. Qin, G. Zhan, K. Zhou, L. Yang, H. Dong, and D. Cremers (2025). “TARGO: benchmarking target-driven object grasping under occlusions”. In: *International Journal of Computer Vision (IJCV)*.
- Y. Xiang, T. Schmidt, V. Narayanan, and D. Fox (2018). “PoseCNN: A Convolutional Neural Network for 6D Object Pose Estimation in Cluttered Scenes”. In: *Robotics: Science and Systems (RSS)*.
- Y. Xiao, Y. Xu, Z. Zhong, W. Luo, J. Li, and S. Gao (2021). “Amodal segmentation based on visible region segmentation and shape prior”. In: *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*.
- J. Xie, W. Xie, and A. Zisserman (2022). “Segmenting moving objects via an object-centric layered representation”. In: *Advances in Neural Information Processing Systems (NeurIPS)*.
- J. Xing, M. Xia, Y. Zhang, H. Chen, W. Yu, H. Liu, G. Liu, X. Wang, Y. Shan, and T.-T. Wong (2024). “Dynamicrafter: Animating open-domain images with video diffusion priors”. In: *European Conference on Computer Vision (ECCV)*.
- J. Xu, S. Liu, A. Vahdat, W. Byeon, X. Wang, and S. De Mello (2023). “Open-vocabulary panoptic segmentation with text-to-image diffusion models”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- K. Xu, L. Zhang, and J. Shi (2024). “Amodal completion via progressive mixed context diffusion”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- K. Xu, L. Zhang, and J. Shi (2025). “Detecting Origin Attribution for Text-to-Image Diffusion Models”. In: *2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*.

- S. Yan, L. Yu, and Y. Xie (2021). “Discrete-continuous action space policy gradient-based attention for image-text matching”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- R. Yanagi, R. Togo, T. Ogawa, and M. Haseyama (2019). “Text-to-Image GAN-Based Scene Retrieval and Re-Ranking Considering Word Importance”. In: *IEEE Access*.
- L. Yang, B. Kang, Z. Huang, X. Xu, J. Feng, and H. Zhao (2024a). “Depth anything: Unleashing the power of large-scale unlabeled data”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- L. Yang, B. Kang, Z. Huang, Z. Zhao, X. Xu, J. Feng, and H. Zhao (2024b). “Depth Anything V2”. In: *Advances in Neural Information Processing Systems (NeurIPS)*.
- Y. Yang, S. Hallman, D. Ramanan, and C. Fowlkes (2010). “Layered object detection for multi-class segmentation”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- Y. Yang, S. Hallman, D. Ramanan, and C. C. Fowlkes (2011). “Layered object models for image segmentation”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence*.
- Y. Yang, H. Deng, and Y. Wu (2025). “MFC-Net: Amodal instance segmentation with multi-path fusion and context-awareness”. In: *Image and Vision Computing*.
- J. Yu, Z. Wang, V. Vasudevan, L. Yeung, M. Seyedhosseini, and Y. Wu (2022). “Coca: Contrastive captioners are image-text foundation models”. In: *Transactions on Machine Learning Research (TMLR)*.
- Q. Yu, Z. Zhang, R. Zhu, Y. Yuan, X. Zuo, Y. Yue, W. Dai, T. Fan, G. Liu, L. Liu, X. Liu, H. Lin, Z. Lin, B. Ma, G. Sheng, Y. Tong, C. Zhang, M. Zhang, W. Zhang, H. Zhu, J. Zhu, J. Chen, J. Chen, C. Wang, H. Yu, Y. Song, X. Wei, H. Zhou, J. Liu, W.-Y. Ma, Y.-Q. Zhang, L. Yan, M. Qiao, Y. Wu, and M. Wang (2025). “DAPO: An Open-Source LLM Reinforcement Learning System at Scale”. In: *arXiv preprint arXiv:2503.14476*.
- J. Yuan, F. Pizzati, F. Pinto, L. Kunze, I. Laptev, P. Newman, P. Torr, and D. De Martini (2026). “LikePhys: Evaluating Intuitive Physics Understanding in Video Diffusion Models via Likelihood Preference”. In: *Proceedings of the International Conference on Learning Representations (ICLR)*.
- X. Yuan, A. Kortylewski, Y. Sun, and A. Yuille (2021). “Robust instance segmentation through reasoning about multi-object occlusion”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- M. D. Zeiler and R. Fergus (2014). “Visualizing and understanding convolutional networks”. In: *13th European Conference on Computer Vision (ECCV)*.

- S. Zeng, C. Liu, J. Zhou, Y. Chen, A. Jiang, and H. Li (2022). “Learning hierarchical semantic correspondences for cross-modal image-text retrieval”. In: *Proceedings of the International Conference on Multimedia Retrieval (ICMR)*.
- X. Zhai, B. Mustafa, A. Kolesnikov, and L. Beyer (2023). “Sigmoid loss for language image pre-training”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*.
- G. Zhan, Y. Liu, K. Han, W. Xie, and A. Zisserman (2025a). “ELIP: Enhanced Visual-Language Foundation Models for Image Retrieval”. In: *Proceedings of the IEEE International Conference on Content-Based Multimedia Indexing (CBMI)*.
- G. Zhan, W. Xie, and A. Zisserman (2022). “A Tri-Layer Plugin to Improve Occluded Detection”. In: *British Machine Vision Conference (BMVC)*.
- G. Zhan, C. Zheng, W. Xie, and A. Zisserman (2023). “What Does Stable Diffusion Know about the 3D Scene?” In: *arXiv preprint arXiv:2310.06836*.
- G. Zhan, C. Zheng, W. Xie, and A. Zisserman (2024a). “A general protocol to probe large vision models for 3d physical understanding”. In: *Advances in Neural Information Processing Systems (NeurIPS)*.
- G. Zhan, C. Zheng, W. Xie, and A. Zisserman (2024b). “Amodal ground truth and completion in the wild”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- J. Zhan, Y. Luo, C. Guo, Y. Wu, B. Yang, J. Wang, and J. Liu (2025b). “Amodal instance segmentation with dual guidance from contextual and shape priors”. In: *Applied Soft Computing*.
- X. Zhan and D. Liu (2025). “LaRender: Training-Free Occlusion Control in Image Generation via Latent Rendering”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*.
- X. Zhan, X. Pan, B. Dai, Z. Liu, D. Lin, and C. C. Loy (2020). “Self-Supervised Scene De-occlusion”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*.
- B. Zhang, Z. Tian, X. Tao, S. Tang, G. Lu, J. Yu, and W. Pei (2025a). “Amodal SAM: Open-World Amodal Segmentation”. In: *OpenReview*.
- C. Zhang, D. Cherniavskii, A. Tragoudaras, A. Vozikis, T. Nijdam, D. W. Prunzhorn, M. Bodracska, N. Sebe, A. Zadaianchuk, and E. Gavves (2025b). “Morpheus: Benchmarking physical reasoning of video generative models with real physical experiments”. In: *arXiv preprint arXiv:2504.02918*.

- F. Zhang, S. You, Y. Li, and Y. Fu (2023a). “Atlantis: Enabling Underwater Depth Estimation with Stable Diffusion”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- H. Zhang, Z. Mao, K. Zhang, and Y. Zhang (2022a). “Show your faith: Cross-modal confidence-aware network for image-text matching”. In: *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*.
- J. Zhang, C. Herrmann, J. Hur, L. P. Cabrera, V. Jampani, D. Sun, and M.-H. Yang (2023b). “A Tale of Two Features: Stable Diffusion Complements DINO for Zero-Shot Semantic Correspondence”. In: *Advances in Neural Information Processing Systems (NeurIPS)*.
- K. Zhang, Y. Fu, S. Borse, H. Cai, F. Porikli, and X. Wang (2023c). “Self-supervised geometric correspondence for category-level 6d object pose estimation in the wild”. In: *Proceedings of the International Conference on Learning Representations (ICLR)*.
- K. Zhang, Z. Mao, Q. Wang, and Y. Zhang (2022b). “Negative-aware attention framework for image-text matching”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- Q. Zhang, Z. Lei, Z. Zhang, and S. Z. Li (2020). “Context-aware attention network for image-text retrieval”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- Q. Zhang, S. Wang, T. Zhang, G. Ren, L. Zhang, Y. Wang, B. Zheng, J. Li, and H. Zheng (2026). “Toward smart aquaculture: A review of multimodal methods, datasets, and applications from the modality perspective”. In: *Computers and Electronics in Agriculture*.
- T. Zhang, H.-X. Yu, R. Wu, B. Y. Feng, C. Zheng, N. Snavely, J. Wu, and W. T. Freeman (2024). “Physdreamer: Physics-based interaction with 3d objects via video generation”. In: *European Conference on Computer Vision (ECCV)*.
- X. Zhang, M. Z. Irshad, A. Yezzi, Y.-C. Tsai, and Z. Kira (2025c). “EscherNet++: Simultaneous Amodal Completion and Scalable View Synthesis through Masked Fine-Tuning and Enhanced Feed-Forward 3D Reconstruction”. In: *arXiv preprint arXiv:2507.07410*.
- Y. Zhang and H. Lu (2018). “Deep cross-modal projection learning for image-text matching”. In: *Proceedings of the European Conference on Computer Vision (ECCV)*.

- W. Zhao, Y. Rao, Z. Liu, B. Liu, J. Zhou, and J. Lu (2023). “Unleashing text-to-image diffusion models for visual perception”. In: *Proceedings of the International Conference on Computer Vision (ICCV)*.
- C. Zheng, D.-S. Dao, G. Song, T.-J. Cham, and J. Cai (2021). “Visiting the Invisible: Layer-by-Layer Completed Scene Decomposition”. In: *International Journal of Computer Vision (IJCV)*.
- X. Zheng, R. Ji, X. Sun, B. Zhang, Y. Wu, and F. Huang (2019). “Towards optimal fine grained retrieval via decorrelated centralized loss with normalize-scale layer”. In: *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*.
- B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba (2016). “Learning deep features for discriminative localization”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
- J. Zhou and Y.-W. Tai (2025). “AmodalGen3D: Generative Amodal 3D Object Reconstruction from Sparse Unposed Views”. In: *arXiv preprint arXiv:2511.21945*.
- X. Zhou, Y. Qiao, Z. Xu, T.-H. Wang, Z. Chen, J. Zheng, Z. Xiong, Y. Wang, M. Zhang, P. Ma, Y. Wang, Z. Dou, B. Kim, Y. Tian, Y. Chen, X. Qiu, C. Lin, T. He, Z. Si, Y. Zhang, Z. Yang, T. Liu, T. Li, K. Yamazaki, H. Zhang, H. Ha, Y. Zhang, M. Liu, S. Zheng, Z. Fu, Q. Wu, Y. Geng, F. Chen, Milky, Y. Hu, G. Shi, L. Liu, T. Komura, Z. Erickson, D. Held, M. Li, L. ". Fan, Y. Zhu, W. Matusik, D. Gutfreund, S. Song, D. Rus, M. Lin, B. Zhu, K. Fragkiadaki, and C. Gan (2024). *Genesis: A Universal and Generative Physics Engine for Robotics and Beyond*. URL: <https://github.com/Genesis-Embodied-AI/Genesis>.
- J. Zhu, W. Wang, Z. Chen, Z. Liu, S. Ye, L. Gu, H. Tian, Y. Duan, W. Su, J. Shao, et al. (2025a). “Internvl3: Exploring advanced training and test-time recipes for open-source multimodal models”. In: *arXiv preprint arXiv:2504.10479*.
- X. Zhu, W. Su, L. Lu, B. Li, X. Wang, and J. Dai (2021). “Deformable detr: Deformable transformers for end-to-end object detection”. In: *International Conference on Learning Representations (ICLR)*.
- Y. Zhu, Y. Tian, D. Metaxas, and P. Dollár (2017). “Semantic amodal segmentation”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
- Y. Zhu, Q. Zhang, Y. Wang, Y. Nie, and W.-S. Zheng (2025b). “EntityErasure: Erasing Entity Cleanly via Amodal Entity Segmentation and Completion”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

- W. Zhuo, M. Salzmann, X. He, and M. Liu (2017). “Indoor Scene Parsing with Instance Segmentation, Semantic Labeling and Support Relationship Inference”. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
- C. L. Zitnick, S. B. Kang, M. Uyttendaele, S. Winder, and R. Szeliski (2004). “High-quality video view interpolation using a layered representation”. In: *ACM transactions on graphics (TOG)*.
- C. Zou and D. Hoiem (2020). “Silhouette Guided Point Cloud Reconstruction beyond Occlusion”. In: *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*.

# Appendix A

## Statement of Authorship

A statement of authorship is provided for each multi-authored paper included in this thesis. The statements describe the candidate's and co-authors' independent research contributions in the thesis publications. For each publication, there exists a complete statement that is filled out and signed by the candidate and supervisor.

## Statement of Authorship for the paper “A Tri-Layer Plugin to Improve Occluded Detection” in Chapter 2.

Paper title	A Tri-Layer Plugin to Improve Occluded Detection
Authors	Guanqi Zhan, Weidi Xie, Andrew Zisserman
Publication status	Published
Publication details	British Machine Vision Conference (BMVC), 2022.

### Student Confirmation

Student name	Guanqi Zhan	
Contribution to the paper	First-author contribution: <ul style="list-style-type: none"><li>• conception of research ideas</li><li>• design and implementation of models</li><li>• running of large-scale experiments</li><li>• writing and presentation of the paper</li></ul>	
Signature and Date		December 15th 2025

### Supervisor Confirmation

By signing the Statement of Authorship, the supervisor is certifying that the candidate made a substantial contribution to the publication, and that the description above is accurate.

Supervisor name	Prof. Andrew Zisserman	
Supervisor comments		
Signature and Date		December 15th 2025

## Statement of Authorship for the paper “Amodal Ground Truth and Completion in the Wild” in Chapter 3.

Paper title	Amodal Ground Truth and Completion in the Wild
Authors	Guanqi Zhan, Chuanxia Zheng, Weidi Xie, Andrew Zisserman
Publication status	Published
Publication details	IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2024.

### Student Confirmation

Student name	Guanqi Zhan	
Contribution to the paper	First-author contribution: <ul style="list-style-type: none"><li>• conception of research ideas</li><li>• design and implementation of models</li><li>• running of large-scale experiments</li><li>• writing and presentation of the paper</li></ul>	
Signature and Date		December 15th 2025

### Supervisor Confirmation

By signing the Statement of Authorship, the supervisor is certifying that the candidate made a substantial contribution to the publication, and that the description above is accurate.

Supervisor name	Prof. Andrew Zisserman	
Supervisor comments		
Signature and Date		December 15th 2025

**Statement of Authorship for the paper “A General Protocol to Probe Large Vision Models for 3D Physical Understanding” in Chapter 4.**

Paper title	A General Protocol to Probe Large Vision Models for 3D Physical Understanding
Authors	Guanqi Zhan, Chuanxia Zheng, Weidi Xie, Andrew Zisserman
Publication status	Published
Publication details	Advances in Neural Information Processing Systems (NeurIPS), 2024.

**Student Confirmation**

Student name	Guanqi Zhan	
Contribution to the paper	First-author contribution: <ul style="list-style-type: none"><li>• conception of research ideas</li><li>• design and implementation of models</li><li>• running of large-scale experiments</li><li>• writing and presentation of the paper</li></ul>	
Signature and Date		December 15th 2025

**Supervisor Confirmation**

By signing the Statement of Authorship, the supervisor is certifying that the candidate made a substantial contribution to the publication, and that the description above is accurate.

Supervisor name	Prof. Andrew Zisserman	
Supervisor comments		
Signature and Date		December 15th 2025

**Statement of Authorship for the paper “Inferring Dynamic Physical Properties from Video Foundation Models” in Chapter 5.**

Paper title	Inferring Dynamic Physical Properties from Video Foundation Models
Authors	Guanqi Zhan, Xianzheng Ma, Weidi Xie, Andrew Zisserman
Publication status	Published
Publication details	The VGBE Workshop of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2026.

**Student Confirmation**

Student name	Guanqi Zhan	
Contribution to the paper	First-author contribution: <ul style="list-style-type: none"><li>• conception of research ideas</li><li>• design and joint implementation of models</li><li>• joint running of large-scale experiments</li><li>• writing and presentation of the paper</li></ul>	
Signature and Date		December 15th 2025

**Supervisor Confirmation**

By signing the Statement of Authorship, the supervisor is certifying that the candidate made a substantial contribution to the publication, and that the description above is accurate.

Supervisor name	Prof. Andrew Zisserman	
Supervisor comments		
Signature and Date		December 15th 2025

**Statement of Authorship for the paper “ELIP: Enhanced Visual-Language Foundation Models for Image Retrieval” in Chapter 6.**

Paper title	ELIP: Enhanced Visual-Language Foundation Models for Image Retrieval
Authors	Guanqi Zhan, Yuanpei Liu, Kai Han, Weidi Xie, Andrew Zisserman
Publication status	Published
Publication details	[Best Paper Nominee & Long Oral] IEEE International Conference on Content-Based Multimedia Indexing (CBMI), 2025.

**Student Confirmation**

Student name	Guanqi Zhan	
Contribution to the paper	First-author contribution: <ul style="list-style-type: none"><li>• conception of research ideas</li><li>• design and joint implementation of models</li><li>• running of large-scale experiments</li><li>• writing and presentation of the paper</li></ul>	
Signature and Date		December 15th 2025

**Supervisor Confirmation**

By signing the Statement of Authorship, the supervisor is certifying that the candidate made a substantial contribution to the publication, and that the description above is accurate.

Supervisor name	Prof. Andrew Zisserman	
Supervisor comments		
Signature and Date		December 15th 2025

## Statement of Authorship for the paper “EGM: Efficient Visual Grounding Language Models” in Chapter 7.

Paper title	EGM: Efficient Visual Grounding Language Models
Authors	Guanqi Zhan, Changye Li, Zhijian Liu, Yao Lu, Yi Wu, Song Han, Ligeng Zhu
Publication status	Not published
Publication details	N/A

### Student Confirmation

Student name	Guanqi Zhan	
Contribution to the paper	First-author contribution: <ul style="list-style-type: none"><li>• conception of research ideas</li><li>• design and joint implementation of models</li><li>• joint running of large-scale experiments</li><li>• writing and presentation of the paper</li></ul>	
Signature and Date		December 15th 2025

### Supervisor Confirmation

By signing the Statement of Authorship, the supervisor is certifying that the candidate made a substantial contribution to the publication, and that the description above is accurate.

Supervisor name	Prof. Andrew Zisserman	
Supervisor comments		
Signature and Date		December 15th 2025