THE UNIVERSITY *of* EDINBURGH

Learning Shape, Structure, and Semantics: Self-Supervised Learning with 3D Priors

Mehmet Aygün

Doctor of Philosophy

Institute for Adaptive and Neural Computation
School of Informatics
University of Edinburgh

2025

Abstract

The world exists in three dimensions, yet when 3D objects are projected onto a 2D image plane, vital spatial information is inevitably lost. Despite this limitation, humans possess a remarkable ability to infer 3D structure from 2D images, enabling us to navigate and interact seamlessly with our surroundings. In contrast, modern computer vision algorithms primarily interpret the world as a collection of 2D patterns (e.g. bag of 2D visual words), leading to several shortcomings: poor generalization to novel environments, difficulty in learning object categories from limited training samples, and vulnerability to adversarial attacks, where minor texture modifications can drastically degrade performance.

This thesis aims to reduce the gap between human and machine perception by improving the extraction of 3D object shape information from 2D images and leveraging 3D understanding to enhance high-level vision tasks such as semantic correspondence estimation. To do so, we take inspiration from developmental psychology which suggests that human vision is strongly driven by shape cues, particularly in early cognitive development. However, with the rise of deep learning, classical approaches that explicitly encode shape, such as pictorial structure models and deformable part-based models, have largely been abandoned in favor of end-to-end learning paradigms.

In this thesis, we first assess the capabilities of unsupervised computer vision models on semantic correspondence tasks using a novel evaluation protocol that jointly captures semantic and geometric understanding. Our findings reveal that current models fall short on this task, and we proposed a new method that improved the state-of-the-art performance at the time, demonstrating significant advancements over existing approaches. Next, we introduce a method for extracting the 3D shape of articulated objects, such as animals, from single-view images without requiring manual supervision. Finally, we present a novel approach to integrate 3D priors into self-supervised learning frameworks, improving robustness for semantic tasks such as image recognition while maintaining accuracy. By emphasizing the role of 3D shape in visual learning, this work introduces new methods that enhance the robustness of machine perception, advancing it toward human-level competence.

Lay Summary

We live in a 3D world, but when we take a photo, it becomes flat and loses important details about depth and shape. People are naturally good at understanding what objects look like in 3D, even from a flat image. However, computers struggle with this. Most computer vision systems focus on patterns and textures instead of shapes, which can cause problems. They often fail in new situations, require many examples to learn from, and can be tricked by small changes to images.

This thesis introduces new methods to help computers see and understand shapes more like people do. Studies show that humans use shapes to recognize objects, especially as children. But modern computer models tend to ignore shape information in favor of quicker, less effective methods. First, we tested how well current computer vision models understand object shapes and meaning using a new method for matching parts on different object instances. The results showed that these systems had trouble with the task. To fix this, we created a better method that made significant improvements. Next, we developed a tool that can estimate the 3D shape of animals and other flexible objects from just one photo, without needing extra labeled data (e.g., other images or different sensors). Finally, we added 3D shape knowledge to computer learning systems, making them more accurate and harder to trick.

By teaching computers to understand shapes more like humans, this thesis aims to develop methods that will result in future smarter and safer artificial intelligence systems that perform better in the real world.

Acknowledgements

First and foremost, I would like to express my heartfelt gratitude to my amazing advisor, Oisin Mac Aodha. When I was searching for a PhD advisor, I faced two options: a senior faculty member with vast experience and deep insights, but limited availability, or a junior faculty member with less experience, but full of time and energy to provide daily support. Throughout my career, I've been fortunate in my choice of advisors, and once again, I feel like I hit the jackpot. I found someone who combines the experience and insights of a senior faculty member with the dedication and energy of a junior faculty member. This thesis would not have been possible without his enthusiasm, commitment, and hard work. *Go raibh maith agat Oisin!*

I would also like to thank my friends from all around the world—both the old ones from Turkey and Germany, with whom I may only meet once a year but always pick up right where we left off, and the new friends I've made during my PhD journey here in Edinburgh, who made this sometimes existential and dreadful journey a bit more fun.

A long-overdue thank you also goes to my family, especially my Dad and Mom, who won't understand a single word of this document since they don't speak a word of English. Despite this, they have always supported me and believed in me every step of the way, even when they didn't fully understand the journey I was on.

Lastly, I would like to thank Prof. Hazim Kemal Ekenel, who accepted me into his lab when I was just a first-year bachelor's student with an empty CV. His lab introduced me to the world of computer vision and research, which has shaped my career and, in one way or another, become the reason why I am writing this document.

Declaration

I declare that this thesis was composed by myself, that the work contained herein is my own except where explicitly stated otherwise in the text, and that this work has not been submitted for any other degree or professional qualification except as specified.

Chapter 1: I was responsible for the ideation and writing the original draft. Oisin Mac Aodha was responsible for reviewing and editing text, and supervision.
Chapter 2 and Chapter 3: I was responsible for problem formulation, designing and implementation of experiments, analysis, and writing the original draft. Oisin Mac Aodha was responsible for reviewing and editing text, and supervision.
Chapter 4: I was responsible for problem formulation, designing and implementation of experiments, analysis, and writing the original draft. Oisin Mac Aodha, Prithviraj Dhar, Zhicheng Yan, and Rakesh Ranjan were responsible for reviewing and editing text, and supervision.
Chapter 5: I was responsible for the ideation and writing the original draft. Oisin Mac Aodha was responsible for reviewing and editing text, and supervision.

(*Mehmet Aygün*)

Table of Contents

1 Introduction **1**

1.1 Motivation and Objectives 1

1.2 Key Contributions 7

1.3 List of Publications 9

2 Demystifying Unsupervised Semantic Correspondence Estimation **11**

2.1 Introduction 11

2.2 Related Work 12

2.3 Semantic Correspondence Estimation 15

2.3.1 Problem Setup 15

2.3.2 Unsupervised Semantic Correspondence Learning 17

2.3.3 Unsupervised Asymmetric Correspondence Loss 18

2.4 Evaluation Protocol 19

2.4.1 Evaluation Metrics 19

2.4.2 Evaluation Datasets 21

2.4.3 Implementation Details 22

2.5 Experiments 23

2.5.1 Impact of Unsupervised Correspondence Objective 23

2.5.2 Impact of Backbone Model and Pre-training Objective 24

2.5.3 Impact of Pre-training Dataset 25

2.5.4 Impact of Finetuning Correspondence Dataset 25

2.5.5 Detailed Error Analysis 26

2.6 Additional Ablation Experiments and Analysis 29

2.6.1 Impact of the Temperature Value 29

2.6.2 Impact of Design Choices for ASYM 29

2.6.3 Impact of Encoder Feature Layer 33

2.6.4 Impact of Input Image Resolution 33

2.6.5 Example Images and Qualitative Results 34
2.6.6 Visualizing Learned Feature Embeddings 35
2.7 Discussion and Limitations 39
2.8 Conclusion 40

3 SAOR: Single-View Articulated 3D Object Reconstruction 41
3.1 Introduction 41
3.2 Related Work 43
3.3 Method 46
3.3.1 SAOR Model 47
3.3.2 Skeleton-Free Articulation 48
3.3.3 Swap Loss and Balanced Sampling 50
3.3.4 Optimization 52
3.3.5 Implementation Details 54
3.4 Experiments 54
3.4.1 Data and Pre-Processing 54
3.4.2 Quantitative Results 55
3.4.3 Ablation Experiments 60
3.4.4 Qualitative Results 61
3.5 Discussion and Limitations 66
3.6 Conclusion 66

4 Enhancing 2D Representation Learning with a 3D Prior 71
4.1 Introduction 71
4.2 Related Work 74
4.3 Method 76
4.3.1 Background 77
4.3.2 3D Aware Robust Representation Learning 78
4.3.3 Preventing Forgetting 80
4.3.4 Implementation Details 80
4.4 Experiments 82
4.4.1 Robustness 82
4.4.2 Downstream Tasks 86
4.4.3 Shape Bias 86
4.4.4 Ablations 87
4.4.5 Qualitative Results 90

4.5 Discussion and Limitations 91
4.6 Conclusion 93

5 Conclusion **95**
5.1 Summary 95
5.2 Key Takeaways 96
5.2.1 The Need for Improved Metrics and Better Datasets 96
5.2.2 The Importance of Data-Driven Approaches and Reducing Reliance on Explicit Priors 97
5.2.3 Rethinking Representation Learning with 3D 98
5.3 Limitations and Future Work 99

Bibliography **103**

A SAOR: Single-View Articulated 3D Object Reconstruction **123**
A.1 Architecture 123
A.2 Training 125

List of Tables

2.1 Summary of the different datasets that we use for evaluating semantic correspondence performance. 22
2.2 Comparison of different unsupervised semantic correspondence methods. 23
2.3 Detailed error types for both unsupervised and supervised correspondence losses on Spair using two different distance thresholds. 27
2.4 Evaluation of error types across four different datasets. 29
2.5 Temperature ablation experiment for unsupervised losses on Spair-71K. 30
2.6 Loss and temperature ablation for ASYM and LEAD on Spair-71K. . 30
2.7 Evaluation of using pre-trained features from different layers for the Resnet50 trained with Imagenet. 33

3.1 Keypoint transfer results on CUB dataset using the PCK metric. . . . 57
3.2 Keypoint transfer results for quadruped animals. 57
3.3 3D evaluation on the Animal3D dataset. 58
3.4 Keypoint transfer ablation results for SAOR. 60
3.5 Keypoint transfer results on Pascal horses. 60

4.1 Robustness evaluation of DINOv2 with and without our 3D-Prior method. 84
4.2 Robustness evaluation on ImageNet-3DCC dataset. 84
4.3 Downstream task evaluation using DINOv2 with and without our 3D-Prior method. 87
4.4 Ablation experiments on various robustness benchmarks using a DinoV2 ViT-B/14 model. 89
4.5 Ablation experiments on various robustness benchmarks using DinoV2 ViT-B/14. 91
4.6 MoCov3 ablation experiment. 91
4.7 Triplane decoder architecture. 93

A.1 Architecture details of our Deformation Net f_d. 123
A.2 Architecture details of our Articulation Net f_a. K is the number of parts and N is the number of vertices, π is camera parameters. 124
A.3 Architecture details of our Texture Net f_t. 124
A.4 Architecture details of our Pose Net f_p. 124
A.5 Training hyperparameters. 126

List of Figures

1.1 Motivation figure. 3
1.2 Summary of contributions. 8

2.1 Unsupervised approaches for semantic correspondence estimation. . . 16
2.2 Error types in our new evaluation framework for semantic correspondence. 20
2.3 Impact of different pre-training datasets used to train a CNN feature encoder using self-supervised training. 26
2.4 Cross dataset evaluation results. 26
2.5 Feature matching scores for different methods for the keypoint on on the birds head. 31
2.6 Histograms for cosine similarity scores of embeddings for None, LEAD, and ASYM. 32
2.7 Semantic correspondence performance of CNNs and Transformers with different input sizes on Spair-71K with no projection. 34
2.8 Examples from each of the datasets with the keypoint annotations that we consider in our experiments. 35
2.9 Qualitative matching results. 36
2.10 More qualitative matching results. 37
2.11 More qualitative matching results. 38
2.12 t-SNE visualization the embeddings learned by different unsupervised losses. 39

3.1 Overview of the generation phase of our SAOR method. 47
3.2 Illustration of our articulated swap loss. 50
3.3 Visualization of clustering of object masks. 52
3.4 Sample images from our training dataset used to train SAOR-101. . . 55

3.5 Visualization of the cluster centers obtained from estimated silhouettes of various animal categories used in our balanced sampling. 56
3.6 Keypoint transfer results. 58
3.7 Capabilities of SAOR. 59
3.8 Comparison of our model to Unicorn and UMR on horses. 61
3.9 Comparison of our model to MagicPony. 62
3.10 Comparison with A-CSM on horses using example images from their paper. 63
3.11 Disentanglement of articulation and deformation. 64
3.12 SAOR estimates 3D shape and viewpoint from different domains. . . 64
3.13 Comparison of models trained with and without relative depth supervision. 65
3.14 Additional qualitative results for SAOR. 67
3.15 Additional qualitative results for SAOR. 68
3.16 Failure cases on cows. 69

4.1 Comparison of shape bias in humans and computer vision. 72
4.2 Overview of our self-supervised single-view 3D reconstruction approach. 77
4.3 Top-5 predictions from linear classifiers trained with and without our 3D prior. 85
4.4 Quantification of the shape bias of different DINOv2 representations with and without our 3D-Prior method. 88
4.5 Comparison of performance of our approach using different amounts of training data. 90
4.6 Visualization of 3D reconstruction by our model for a subset of ImageNet validation images. 92

Chapter 1

Introduction

1.1 Motivation and Objectives

The world is 3D, and there is a loss in information when we project objects and scenes from our 3D world to the 2D image plane. However, humans have a remarkable capability of perceiving useful 3D information such as depth, shape, size, and spatial relationships between objects (Palmer, 1999; Bruce et al., 2014) and are able to interact and navigate by acting on this extracted knowledge. In contrast, current computer vision algorithms have a hard time inferring necessary 3D knowledge from single 2D observations alone. As multiple different 3D scenes can create the same observation in 2D, capturing the 3D shape or properties of objects or scenes from single-view 2D images is a highly ill-posed problem due to this ambiguity.

The role of shape and structure in visual recognition has a long and distinguished history. In the context of computer vision, shape typically refers to the geometric outline or silhouette of an object, capturing its spatial extent and boundaries, while structure encompasses the spatial arrangement and relationships between an object's parts or components. Early approaches, such as pictorial structure models (Fischler and Elschlager, 1973), constellation models (Weber et al., 2000; Fergus et al., 2003), and, at their peak, the widely adopted deformable parts-based models (Felzenszwalb et al., 2010), all leveraged shape as a fundamental cue for object representation. Yet, with the advent of deep learning, these models have largely fallen out of favor. This shift is particularly striking given compelling evidence from developmental psychology that the human visual system is strongly attuned to shape, especially in the early stages of cognitive development (Landau et al., 1988; Gershkoff-Stowe and Smith, 2004).

In contrast, current state-of-the-art computational visual understanding systems

(i.e., deep neural networks) tend to rely more heavily on texture cues than on shape information, as demonstrated in Geirhos et al. (2019, 2021). These models effectively perceive the world as a collection of 2D image patches rather than as structured objects with inherent spatial relationships (Brendel and Bethge, 2019). However, this reliance on texture comes at a cost. Deep networks struggle to generalize to novel environments (Beery et al., 2018), perform poorly on object categories in the long tail of the distribution due to limited training samples (Van Horn and Perona, 2017), and are remarkably vulnerable to adversarial perturbations, where subtle texture modifications can drastically alter predictions (Goodfellow et al., 2015).

Inspired by Turing's vision of machine intelligence as a learning process akin to human development (Turing and Haugeland, 1950), we argue that enabling models to reason about the 3D shape of objects is a crucial step toward creating artificial visual agents capable of self-supervised learning in embodied settings. Self-supervised learning is a learning paradigm in which models generate supervisory signals from the data itself—typically by solving pretext tasks such as predicting missing parts of an image or aligning different views—without relying on manual labels. Unlike unsupervised learning, which focuses on discovering patterns or structures (e.g., clustering) without explicit targets, self-supervised learning defines specific objectives that guide representation learning, often leading to features that are more useful for downstream tasks.

Humans can effortlessly infer the approximate 3D structure of an object from a single view, can establish part-level and local correspondences across different instances of a category, and can leverage shape priors for semantic association across a wide range of tasks. For example, fine-grained visual recognition, such as distinguishing between closely related animal species, can potentially be enhanced by incorporating 3D shape information, as illustrated in Figure 1.1. Moreover, state-of-the-art methods for face recognition, which is one of the most fine-grained tasks, also leverage shape information, such as surface geometry and depth (Apple, 2024). In this context, depth refers to the distance of points on the face surface from the camera, often represented as a depth map. Unless stated otherwise, we use relative depth throughout this thesis, which captures the depth relationships between different regions of an object rather than their absolute distance from the camera.

In this thesis, we investigate the interplay between shape and semantics, develop methods for improving single-view 3D shape estimation, and leverage 3D representations to enhance high-level visual understanding. As we aim to explore the intri-

Figure 1.1. Can you identify the characteristic feature that differentiates these two Gazelle species? We can mentally 'align' these two animals in 3D and compare their differences. However, current general-purpose computer vision algorithms for visual recognition lack the ability to reason about the 3D shapes of objects and require large amounts of training data to distinguish fine-grained categories. In this thesis, we aim to enhance computer vision models with 3D geometric reasoning capabilities to improve their semantic understanding of the world. For interested readers, the Thomson's gazelle (left) has a black textured area in the middle of their body, compared to the Grant's gazelle (right) who do not.

cate relationship between 3D shape information and semantics in computer vision, we focus on the problem of semantic correspondence as a key proxy task. Semantic correspondence entails establishing dense or sparse correspondences between semantically similar regions across different images containing two different instances of the same object category, even when there are significant variations in appearance, pose, or background. In contrast to traditional geometric correspondence, which relies on low-level features such as edges and textures, semantic correspondence aligns regions based on their meaning and function, capturing higher-level relationships. This makes it a powerful tool for studying the interplay between shape and semantics in computer vision. By analyzing correspondences across a diverse range of objects and scenes, we can evaluate how effectively a model captures structural and functional similarities, even in the presence of intra-class variations. This is especially important for tasks like object recognition, scene understanding, and image synthesis, where an accurate representation of both shape and semantics is essential for reliable predictions and meaningful generalization.

Despite being a fundamental and intuitive task for humans, semantic correspondence has often been overlooked in the computer vision community compared to tasks like object detection, image segmentation, and visual categorization. Establishing

meaningful correspondences between semantically similar regions across images is crucial for understanding how models capture high-level structural and functional relationships. While some datasets have been created for semantic correspondence tasks (Ham et al., 2017; Min et al., 2019b), a benchmarking framework with a standardized protocol and diagnostic tools has been lacking, hindering our ability to carefully compare methods and assess progress in a systematic manner. Developing robust and detailed benchmarks is essential for advancing the field, as they provide a clear and consistent means of measuring a model's effectiveness in tackling a given task.

In a manner analogous to Winston Churchill's famous quote, "Democracy is the worst form of government, except for all the others," benchmarks may be the least optimal way to measure progress in machine learning and computer vision, except for all other alternatives (Everingham et al., 2015). Nevertheless, benchmarks like Pascal (Everingham et al., 2015) and ImageNet (Deng et al., 2009) have played a pivotal role in advancing computer vision. Standardized datasets and benchmarking protocols have enabled fair and transparent evaluation of competing approaches, providing definitive insights into what truly works. For example, the seminal study comparing various local visual descriptors (Mikolajczyk and Schmid, 2005) solidified SIFT (Lowe, 1999) as the dominant choice for visual recognition tasks for over a decade. Furthermore, the ImageNet (Deng et al., 2009) benchmark ushered in a new era in artificial intelligence, spurred by the remarkable success of AlexNet (Krizhevsky et al., 2012). The impact of these benchmarks stems not only from the datasets themselves, but also from their carefully designed evaluation protocols and metrics, which have provided a common ground for measuring progress and fostering innovation.

Many computer vision benchmarks, including the aforementioned, typically rely on a single summary metric to compare different methods. However, a deeper understanding of their limitations and the improvements introduced by newer approaches is paramount for future progress. To address this, several studies have proposed diagnostic tools and frameworks to assess methods across a broad range of challenges (Hoiem et al., 2012; Russakovsky et al., 2013; Everingham et al., 2015; Zhang et al., 2016b; Sigurdsson et al., 2017; Ruggero Ronchi and Perona, 2017; Alwassel et al., 2018). Inspired by these benchmarking protocol works that offer a more nuanced analysis of competing approaches, the first part of this thesis aims to develop a more comprehensive evaluation framework with a standardized evaluation protocol for comparing semantic correspondence methods.

In addition to advancing our understanding of the relationship between 3D shape

and semantics, we aim to enhance the capabilities of computer vision systems in extracting 3D information from single-view images. One of the earliest and most foundational works in this domain, Roberts' PhD thesis (Roberts, 1963), focused on the estimation of 3D shape from a single image. Despite considerable progress over the past six decades (Blanz and Vetter, 1999; Cashman and Fitzgibbon, 2012; Kar et al., 2015; Kanazawa et al., 2018b), this problem remains inherently challenging, primarily due to its ill-posed nature. In contrast, humans can effortlessly infer 3D shape from a single image (Bruce et al., 2014), relying on a combination of prior knowledge about the natural world and familiarity with the object's category. While some low-level priors, such as symmetry or smoothness, can be explicitly modeled, the task of manually encoding and effectively utilizing high-level priors such as 3D shape templates for various object categories remains a formidable challenge. Recent advances in deep learning and differentiable rendering have led to multiple methods for estimating 3D shape from 2D images (Loper and Black, 2014; Kato et al., 2018; Liu et al., 2019). These approaches have yielded impressive results for synthetic, man-made categories (Choy et al., 2016b; Kato et al., 2018; Wang et al., 2018) and human models (Loper et al., 2015; Güler et al., 2018), where full or partial 3D supervision is readily available.

However, when 3D supervision is unavailable, reconstructing objects remains challenging. To address this issue in the absence of 3D supervision, various approaches have relied on category-specific 3D templates (Kokkinos and Kokkinos, 2021a; Kulkarni et al., 2020; Zuffi et al., 2019) or leveraged multi-view training data, such as videos (Wu et al., 2023a; Kokkinos and Kokkinos, 2021a; Yang et al., 2021a). Yet, these approaches limit the generalizability of the methods, as template-based techniques cannot extend to a multi-category setting, and multi-view data is often not available for many object classes. While some methods have gone beyond this by using single-view image collections, they have constrained them to a single category and have not modeled articulation (Kanazawa et al., 2018b; Kulkarni et al., 2019a; Goel et al., 2020; Monnier et al., 2022). Not modeling articulation limits methods to work only with rigid categories, such as man-made objects, and fails to address intra-instance variations in image collections, which makes scaling the methods for multi-category settings very challenging.

In this thesis, we aim to develop a more general 3D reconstruction method specifically designed for animals. We choose animals due to the difficulty of collecting multi-view or 3D data for them, their highly articulated structures, and the significant shape variability across and within species. Our goal is to create a method that

does not rely on 3D templates, as we seek to estimate the 3D shape of multiple object categories within a single model. This model should be capable of handling articulation and should not require multi-view or 3D data during training. Instead, it should use only single-view image collections, which are often gathered in an automatic way from the web and might include noise such as low-quality or blurry images, irrelevant content, occlusions, and background clutter. Unlike some recent works (Wu et al., 2023a,b; Li et al., 2024) that rely on 3D skeleton priors, we aim to avoid this requirement, resulting in a model capable of estimating the 3D shape of animals with varying bone topologies, such as bipeds and quadrupeds in a unified manner.

In addition to improving the estimation of 3D structures in the world, utilizing these 3D cues for high-level semantic understanding is also crucial for developing agents capable of understanding and acting in the open world. When we observe humans, the visual stimuli processed by a binocular, actively moving, human observer provides direct information about the 3D world around them (Gibson, 1950). As a result, humans have a remarkable ability to perceive useful 3D shape cues, enabling them to interact and navigate adeptly in complex environments. Most impressively, the power of the human visual system is not understood to be a resulting property of supervised learning, i.e., it has developed thanks largely to 'self-supervision' (Smith and Gasser, 2005). Moreover, it is well established, especially in the early years of cognitive development, that infants more heavily rely on shape cues compared to other cues such as texture during early category learning (Landau et al., 1988; Spelke, 1990; Spelke and Kinzler, 2007).

These findings suggest two ways to enhance artificial vision systems: (i) by developing models that can learn from data in an unsupervised manner and (ii) by encouraging models to better utilize shape information. Fortunately, significant progress has been made in the first area, as we now have techniques that generate effective visual representations through self-supervision alone, such as e.g., (Wu et al., 2018; Chen et al., 2020b; Bao et al., 2022; He et al., 2022; Oquab et al., 2024). However, relatively less research has focused on improving models' ability to leverage shape information. While there are methods for extracting shape-related information, like depth, using self-supervision from image pairs (Godard et al., 2017) or video sequences (Zhou et al., 2017), these approaches often rely on strong assumptions about the scenes they are trained on (e.g., smooth camera motion, static scenes, limited visual diversity, etc.). As a result, the most effective depth prediction methods today still require explicit depth supervision during training (Ranftl et al., 2022; Bochkovskii et al., 2024).

Furthermore, even with depth supervision, leveraging it to improve performance on other tasks remains non-trivial (Zamir et al., 2018; Standley et al., 2020). Meanwhile, emerging evidence suggests that solving a geometric task could indeed aid semantic tasks (Lao et al., 2024).

Inspired by previous self-supervised learning methods (Zhang et al., 2016a; Pathak et al., 2016; Gidaris et al., 2018; Noroozi et al., 2017) that use proxy tasks to promote the learning of useful information, we posit that the single-view 3D reconstruction problem can be leveraged to encourage the use of shape information, thereby enriching the visual representations of already trained self-supervised networks. Estimating the 3D shape of an object from a single-view image requires the model to encode shape-related cues, which can enhance the shape-awareness of the underlying visual representation. However, understanding shape alone is insufficient for many vision-related tasks. For instance, while global shape information may help distinguish a dog from a cat, distinguishing between two dogs may require more local, texture-based details. To harmonize the learning of shape-aware features with the use of texture or 2D-related cues, we propose a new learning framework that combines the single-view 3D reconstruction proxy task with a knowledge distillation framework (Hinton et al., 2015). We hypothesize that with this framework we can achieve satisfactory performance across visual recognition tasks similar to baseline self-supervised models, while also improving their robustness by promoting shape-bias behavior, which is observed in humans.

Thesis Statement: This thesis aims to bridge the gap between human and machine visual understanding by investigating how to effectively extract and utilize 3D shape information from single-view 2D images to improve semantic tasks such as visual recognition and semantic correspondence. Specifically, it develops novel methods for (i) establishing semantic correspondence using unsupervised learning, (ii) reconstructing 3D shapes of articulated objects like animals without category-specific templates or multi-view supervision, and (iii) integrating 3D shape priors into self-supervised learning frameworks to enhance the robustness and shape-awareness of visual representations.

1.2 Key Contributions

The main aim of this thesis is to narrow the gap between humans' and machines' abilities to extract information from 2D images depicting 3D objects and to leverage

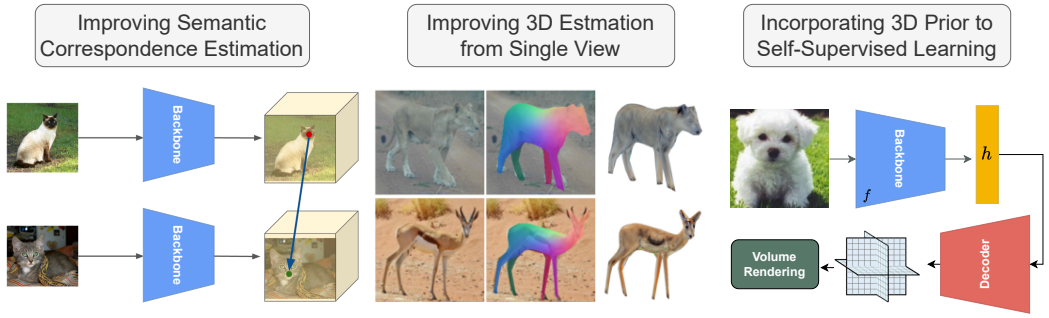


Figure 1.2. **Summary of Contributions**: First, we introduce a novel diagnostic framework to explore the interplay between shape and semantics in computer vision and an unsupervised approach to enhance semantic correspondence matching, resulting in improved performance. Next, we present SAOR, a skeleton-free 3D reconstruction model that enables the estimation of shape, texture, and viewpoint of over one hundred different articulated animal categories from a single image, without relying on category-specific templates. Finally, we incorporate a 3D prior into a self-supervised learning framework, leveraging 3D representations to enhance high-level visual understanding and improve the robustness of learned visual representations.

3D information to enhance performance on semantic tasks like visual recognition and semantic correspondence estimation.

In Chapter 2, we explore semantic correspondence estimation through the lens of unsupervised learning. We thoroughly evaluate several recently proposed unsupervised methods across multiple challenging datasets using a standardized evaluation protocol where we vary factors such as the backbone architecture, the pre-training strategy, and the pre-training and finetuning datasets. To better understand the failure modes of these methods, and in order to provide a clearer path for improvement, we provide a new diagnostic framework along with a new performance metric that is better suited to the semantic matching task. Finally, we introduce a new unsupervised correspondence approach which utilizes the strength of pre-trained features while encouraging better matches during training. This leads to significantly better matching performance than current state-of-the-art methods, with PCK scores improving by 5–20%.

In Chapter 3, we introduce SAOR, a novel approach for estimating the 3D shape, texture, and viewpoint of an articulated object from a single image captured in the wild. Unlike prior approaches that rely on pre-defined category-specific 3D templates or tailored 3D skeletons, SAOR learns to articulate shapes from single-view image collections with a skeleton-free part-based model without requiring any 3D object shape priors. To prevent ill-posed solutions, we propose a cross-instance consistency loss

that exploits disentangled object shape deformation and articulation. This is helped by a new silhouette-based sampling mechanism to enhance viewpoint diversity during training. Our method only requires estimated object silhouettes and relative depth maps from off-the-shelf pre-trained networks during training. At inference time, given a single-view image, it efficiently outputs an explicit mesh representation. We obtained improved qualitative and quantitative results on challenging quadruped animals compared to existing work at the time of writing.

In Chapter 4, we present a new way to incorporate a 3D prior into a self-supervised learning framework. Learning robust and effective representations of visual data is a fundamental task in computer vision. Traditionally, this is achieved by training models with labeled data which can be expensive to obtain. Self-supervised learning attempts to circumvent the requirement for labeled data by learning representations from raw unlabeled visual data alone. However, unlike humans who can extract rich 3D information from their binocular vision and through motion, the majority of current self-supervised methods are tasked with learning from monocular 2D image collections. This is noteworthy as it has been demonstrated that shape-centric visual processing is more robust compared to texture-biased automated methods. Inspired by this, we propose a new approach for strengthening existing self-supervised methods by explicitly enforcing a strong 3D structural prior directly into the model during training. Through experiments, across a range of datasets, we demonstrate that our resulting 3D aware representations are more robust compared to conventional self-supervised baselines.

Conclusion: This thesis demonstrates that incorporating 3D geometric reasoning significantly enhances computer vision models' capacity to understand complex visual scenes. By establishing the relationship between shape and semantics through a proxy task of semantic correspondence, developing novel methods to estimate the 3D shape of articulated objects such as animals from single-view images, and integrating 3D shape priors into a self-supervised learning framework, this work contributes to improving the robustness and accuracy of image recognition models.

A summary of contributions can be seen in Figure 1.2.

1.3 List of Publications

The following papers form the basis of Chapter 2, Chapter 3, and Chapter 4, respectively:

- **Mehmet Aygün** and Oisin Mac Aodha.“Demystifying Unsupervised Semantic Correspondence Estimation.” *European Conference on Computer Vision, (ECCV)*. 2022.
- **Mehmet Aygün** and Oisin Mac Aodha. “SAOR: Single-view Articulated Object Reconstruction.” *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, (CVPR)*. 2024.
- **Mehmet Aygün**, Prithviraj Dhar, Zhicheng Yan, Oisin Mac Aodha, and Rakesh Ranjan. “Enhancing 2D Representation Learning with a 3D Prior.” *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, (CVPR Workshops)*. 2024.

In addition, the following publication, which the author contributed to during his PhD, is not included in this thesis:

- Danier, Duolikun, **Mehmet Aygün**, Changjian Li, Hakan Bilen, and Oisin Mac Aodha. “DepthCues: Evaluating Monocular Depth Perception in Large Vision Models.” *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, (CVPR)*. 2025.

Chapter 2

Demystifying Unsupervised Semantic Correspondence Estimation

In this chapter, we investigate the relationship between 3D shape information and semantics in computer vision through the lens of semantic correspondence, with a particular focus on its unsupervised form. Unlike traditional geometric correspondence, which matches low-level features, semantic correspondence aligns regions based on their meaning and function, capturing higher-level relationships. Inspired by how humans learn largely without supervision, we explore how models can achieve similar capabilities. By analyzing correspondences across varied objects and scenes, we assess how well models capture structural and functional similarities, a crucial aspect for tasks like object recognition, scene understanding, and image synthesis.

2.1 Introduction

In metaphysics, the correspondence theory of truth posits that without the notion of correspondence, there cannot be truth (David, 2016). Analogously, correspondence estimation also holds a very important place as one of the core problems in computer vision. The ability to reliably obtain accurate pixel-level correspondence underpins a diverse range of tasks from stereo estimation, optical flow, structure-from-motion, through to visual tracking. Distinct from these lower-level objectives, semantic correspondence estimation, the task of matching different regions, parts, and landmarks across distinct object instances, is crucial to developing systems that can perform higher-level visual reasoning in diverse environments with objects that can vary significantly in both appearance and the configuration of their constituent parts.

Manually obtaining semantic correspondence supervision, for example in the form of annotated object landmarks, is an arduous and time consuming task. As a result, several works have instead attempted to understand to what extent semantic regions and parts emerge from conventionally trained supervised image classification networks (Long et al., 2014; Zeiler and Fergus, 2014; Zhou et al., 2016; Gonzalez-Garcia et al., 2018). These works have shown such semantic information is indeed present in the representations encoded by these networks, at least to some degree. Recently, a body of work has emerged that aims to learn semantic correspondence through self-supervision alone, i.e., without the need for ground truth supervision at training time (Thewlis et al., 2017b, 2019; Cheng et al., 2021; Karmali et al., 2022).

While we have observed progress on unsupervised semantic correspondence estimation, a number of questions are still underexplored and unanswered. For instance, it is not clear how well current approaches generalize beyond more simplified object categories such as human faces to more complex non-rigidly deforming categories that vary in terms of both pose and appearance. Recent works have also leveraged advances in self-supervised learning of general visual representations (Cheng et al., 2021; Karmali et al., 2022), making it difficult to properly assess how they compare to older methods that do not utilize such self-supervised pre-training. In this chapter, we attempt to shine light on the above questions in addition to exploring the role of other factors such as the impact of pre-training and finetuning data, backbone models, and the underlying evaluation criteria used to assess performance. Inspired by detailed benchmarking investigation in human pose estimation (Ruggero Ronchi and Perona, 2017), we provide a thorough evaluation of the success and failure modes of current methods to provide guidance for future progress.

We make the following three contributions: (i) A standardized evaluation of multiple existing approaches for unsupervised semantic correspondence estimation across five challenging datasets. (ii) A new, conceptually simple, unsupervised training objective that results in superior semantic matching performance. (iii) A detailed breakdown of the current failure cases for current best performing approaches and our proposed new unsupervised method.

2.2 Related Work

Supervised Semantic Correspondence. Pre-deep learning work tackled semantic correspondence estimation as a local region matching problem using hand-crafted fea-

tures (Liu et al., 2010; Kim et al., 2013; Bristow et al., 2015), or as offset matching using object proposals (Ham et al., 2017). In the deep learning era, several works investigated if object parts and regions emerge from image classification models (Zeiler and Fergus, 2014; Zhou et al., 2016; Gonzalez-Garcia et al., 2018), i.e., models trained only with image-level class supervision. (Long et al., 2014) showed that deep CNN features could actually be used for semantic matching. Subsequent work built on this by proposing new architectures specifically designed for semantic matching (Choy et al., 2016a; Han et al., 2017; Kim et al., 2017; Rocco et al., 2017; Huang et al., 2019; Lee et al., 2019; Kim et al., 2018). Some of these approaches focused on combining multilevel features (i.e., hypercolumn features) from deep networks (Ufer and Ommer, 2017; Min et al., 2019a, 2020; Zhao et al., 2021), aggregating information from features using 4D convolutions (Rocco et al., 2018, 2020; Li et al., 2020a; Lee et al., 2021), leveraging geometric relations via Hough transforms (Min and Cho, 2021), or using optimal transport (Sarlin et al., 2020; Liu et al., 2020). Some matching methods formulate the problem as one of flow estimation between images (Liu et al., 2010; Min et al., 2019a). However, unlike optical flow, semantic correspondence methods need to be able to handle intra and inter-class variations when matching points. Recently, the use of transformer-based models has also been explored (Cho et al., 2021; Jiang et al., 2021). In contrast to most of the above works, we focus on the unsupervised setting, whereby no supervised keypoint annotations are used to train our models.

Unsupervised Semantic Correspondence. Recent progress in self-supervised learning has resulted in a suite of methods that are capable of extracting discriminative whole image representations without requiring explicit supervision (Van den Oord et al., 2018; Wu et al., 2018; Chen et al., 2020b; Grill et al., 2020; He et al., 2020). While the majority of these methods optimize objectives to discriminate global image representations by using augmented image pairs, (Cheng et al., 2021; Karmali et al., 2022) showed that these approaches can also be utilized in correspondence estimation. Recently, several approaches proposed optimizing alternative objectives on a denser level (Roh et al., 2021; Wei et al., 2021; Araslanov et al., 2021; O Pinheiro et al., 2020; Wang et al., 2021a,b; Zhong et al., 2021). However, these methods have been applied to tasks such as object detection and segmentation, but not directly for semantic correspondence. Another line of work proposed methods to discover semantic keypoint locations in an unsupervised way (Jakab et al., 2018; Zhang et al., 2018b; Kulkarni et al., 2019b; Jakab et al., 2020; Ryou and Perona, 2021).

For the problem of correspondence estimation, images augmented with artificial

spatial deformations were used by (Kanazawa et al., 2016; Rocco et al., 2017) to learn transformations between image pairs without any external supervision. Instead of learning a function to match image pairs, (Thewlis et al., 2017b,a) framed the problem as one of learning a function that can extract local features which can be used for semantic matching across all instances of a category of interest. To introduce greater invariance for intra-category differences, DVE (Thewlis et al., 2019) extended EQ (Thewlis et al., 2017a) with the use of additional non-augmented auxiliary images during training.

More recent work has been able to make use of advances in self-supervised learning in order to learn more effective representations. CL (Cheng et al., 2021) proposed a two-stage approach, combining image-level instance-based discrimination (He et al., 2020) together with dense equivariant learning. They trained a linear projection head on top of frozen learned features computed via an image-level self-supervised pre-training task, where the goal of the projection step was to enforce the dense features to be spatially distinct within an image. LEAD (Karmali et al., 2022) also followed a similar two-stage approach, starting with instance-level discrimination using (Grill et al., 2020). In the second stage, instead of encouraging the features to be spatially distinct, their projection operation minimized the dissimilarity between feature correlation maps from the instance-level features and correlation maps from the projected features. This can be viewed as a form of dimensionality reduction as the projected features are smaller in size compared to the original features.

The above methods, while effective on some datasets, have limitations. EQ (Thewlis et al., 2017a) is only able to learn invariances that can be expressed via image augmentations. DVE (Thewlis et al., 2019) assumes that the images have the same visible keypoints, and can thus be negatively impacted by incorrect matches on background pixels. The projection step used by CL (Cheng et al., 2021) runs the risk of discarding invariances learned during the pre-training stage. While LEAD (Karmali et al., 2022) maintains learned invariances from the first stage, if the pre-trained features generate incorrect matches, their loss can end up optimizing possibly incorrect feature correlations. In this work, we thoroughly benchmark the performance of these approaches by evaluating them on several challenging datasets. We also propose a new semantic correspondence loss, which learns more effective dense features by both preserving the learned invariances while also making the features more distinct.

Performance Evaluation and Error Diagnosis. Benchmarking model performance with a single summary metric is one of the best tools that we have for objectively

measuring progress on a given task. However, accurately understanding the limitations and improvements provided by new methods is even more crucial for future progress. Several works have introduced different diagnostic tools and frameworks to analyze methods across a variety of problems (Hoiem et al., 2012; Russakovsky et al., 2013; Everingham et al., 2015; Zhang et al., 2016b; Sigurdsson et al., 2017; Alwassel et al., 2018). For the semantic correspondence problem, the vast majority of existing works only report performance via single summary metrics, e.g., the Percentage of Correct Keypoints (PCK) with a fixed distance threshold. This allows us to get an overall sense of performance but does not reveal *why* a given method performs better than others. Recent works (Musgrave et al., 2020; Choe et al., 2020) have emphasized the importance of detailed evaluation in order to better understand what components specific performance improvements can be attributed to. In this work, in the spirit of (Ruggero Ronchi and Perona, 2017), we introduce a more thorough evaluation for analyzing semantic correspondence methods. We also propose a new version of PCK which better captures correspondence errors and present standardized baseline results across multiple datasets to fairly compare semantic correspondence performance.

2.3 Semantic Correspondence Estimation

2.3.1 Problem Setup

Given a source-target image pair, $\mathbf{x}_s$ and $\mathbf{x}_t$, the goal of correspondence estimation is to find the locations of a set of points of interest from the source image in the target image. Unlike in optical flow or stereo estimation, where the task is to compute correspondence across time or viewpoint, in the case of semantic correspondence, the goal is to find matching locations across different depictions of the same object category. This is a challenging setting as the objects of interest can vary in terms of appearance, pose, and shape, in addition to difficulty arising from other nuisance factors such as the background, occlusion, and lighting.

We pose the correspondence problem as a nearest-neighbor matching task in a learned local feature embedding space. Formally, for a pixel location, $u \in \Omega = \{1,...,H\} \times \{1,...,W\}$, in a source image of size $H \times W$, we find the corresponding point $\hat{u}$ in the target image $\mathbf{x}_t$ as, $\hat{u} = \arg\max_{k \in \Omega} f(\Phi_u(\mathbf{x}_s), \Phi_k(\mathbf{x}_t))$, where $\Phi_u(\mathbf{x}_s)$ represents an embedding vector of the point u from image $\mathbf{x}_s$, and f is a similarity function. We use a deep neural network as our embedding function Φ, and the similarity is computed

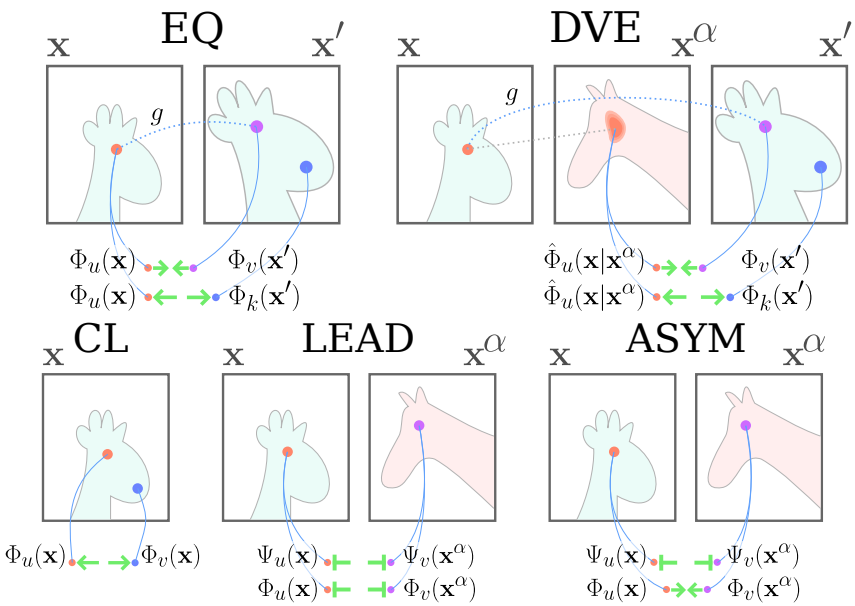


Figure 2.1. Unsupervised approaches for semantic correspondence estimation. $\mathbf{x}'$ is a synthetically augmented version of image $\mathbf{x}$, and $\mathbf{x}^\alpha$ is a different image of the same semantic category. EQ (Thewlis et al., 2017a) minimizes the distance between embeddings of point pairs with known geometric transformations g. DVE (Thewlis et al., 2019) builds on EQ by using an additional auxiliary image. CL (Cheng et al., 2021) maximizes the distance between embeddings of points within an image. LEAD (Karmali et al., 2022) enforces the same distance between pre-trained and projected embeddings. Our ASYM method extends LEAD by enforcing projected embeddings to be closer in the feature space.

via the dot product of the ℓ_2 normalized embedding vectors. In practice, we decompose the embedding function into a feature encoder, followed by a projection step, i.e., $\Phi(\mathbf{x}) = \rho(\Psi(\mathbf{x}))$, where the encoder is a deep network. The purpose of the projection is to reduce the dimensionality of the feature, and could be a linear operation (Cheng et al., 2021) or a network (Karmali et al., 2022).

In the next section, we review several existing unsupervised methods designed for learning dense representations with an emphasis on matching (see Figure 2.1 for an overview). While more sophisticated methods have been proposed for estimating semantic correspondence, e.g., using optimal transport (Sarlin et al., 2020; Liu et al., 2020), distance re-weighting with spatial regularizers (Min et al., 2019a), or restricting the search area with class activation maps (Zhou et al., 2016) as in (Liu et al., 2020), we focus on learning embedding functions as recent work has shown that combining self-supervised representation learning with correspondence specific finetuning produces state of the art results (Cheng et al., 2021; Karmali et al., 2022).

2.3.2 Unsupervised Semantic Correspondence Learning

EQ (Thewlis et al., 2017a) proposed an unsupervised method that utilizes the equivariance principle to learn dense matchable features. During training, their model takes an image $\mathbf{x}$ along with an augmented version of it $\mathbf{x}'$ and tries to minimize feature similarity of known corresponding pixel locations u and v. Here, $\mathbf{x}'$ is derived from $\mathbf{x}$ using artificial spatial and appearance-based augmentations and the pixel coordinates u and v are locations from the two images which are related by a known transformation g, such that $v = gu$. They minimize the following loss,

$$\mathcal{L}_{eq} = \frac{1}{|\Omega|^2} \sum_{u \in \Omega} \sum_{v \in \Omega} \|gu - v\| \; p(v|u; \Phi, \mathbf{x}, \mathbf{x}', \tau), \tag{2.1}$$

$$p(v|u; \Phi, \mathbf{x}, \mathbf{x}', \tau) = \frac{\exp(\langle \Phi_u(\mathbf{x}), \Phi_v(\mathbf{x}') \rangle / \tau)}{\sum_{k \in \Omega} \exp(\langle \Phi_u(\mathbf{x}), \Phi_k(\mathbf{x}') \rangle / \tau)}, \tag{2.2}$$

where τ is the temperature parameter for the softmax function and Ω is the set of possible pixel locations on the image grid. In essence, the model aims to embed corresponding points nearby in the learned embedding space, while also pushing other points further away.

EQ uses artificially augmented image pairs and can thus only learn invariances up to those expressible by these augmentations. Subsequently, DVE (Thewlis et al., 2019) extended EQ using an auxiliary image, $\mathbf{x}^\alpha$, to calculate correspondence from $\mathbf{x} \rightarrow \mathbf{x}^\alpha$ and then $\mathbf{x}^\alpha \rightarrow \mathbf{x}'$. This is achieved by replacing the $\Phi_u(\mathbf{x})$ term in Equation 2.2 with $\hat{\Phi}_u(\mathbf{x}|\mathbf{x}^\alpha) = \sum_w \Phi_u(\mathbf{x}^\alpha) p(w|u; \Phi, \mathbf{x}, \mathbf{x}^\alpha, \tau)$. Importantly, the ground truth correspondence to the auxiliary image does not need to be known as the mapping from $\mathbf{x} \rightarrow \mathbf{x}'$ is available.

Recently, two-stage methods for learning dense embeddings have been proposed (Cheng et al., 2021; Karmali et al., 2022). In these approaches, the first stage makes use of an image-level self-supervised training objective (e.g., (He et al., 2020; Grill et al., 2020)) in order to train the feature encoder. Then the projection head is tuned to refine the representation so that it is better for matching. Like EQ, CL (Cheng et al., 2021) also aims to make features distinct within the image. However, in contrast to EQ, the dense D dimensional feature vectors from $\Psi(\mathbf{x})$ are linearly projected to a lower dimension D' using a linear projection with weights $\mathbf{w} \in \mathbb{R}^{D \times D'}$. They use the same loss as Equation 2.1, but simply use $\mathbf{x}$ instead of $\mathbf{x}'$, i.e., they do *not* use a pair of augmented images.

LEAD (Karmali et al., 2022) also employs a two-stage approach but aims to maximize the similarity between feature correlation maps calculated using the original self-

supervised features $\Psi(\mathbf{x})$ and the projected features $\Phi(\mathbf{x})$. The first term in their loss represents the probability that point u from image $\mathbf{x}$ is matched with point v in image $\mathbf{x}^\alpha$ using embeddings from the feature encoder Ψ. In the second term, embeddings are projected to a lower dimensional space using the combined encoder and projection head,

$$\mathcal{L}_{lead} = \frac{1}{|\Omega|^2} \sum_{u\in\Omega} \sum_{v\in\Omega} -p(v|u;\Psi,\mathbf{x},\mathbf{x}^\alpha,\tau) \log p(v|u;\Phi,\mathbf{x},\mathbf{x}^\alpha,\tau). \tag{2.3}$$

LEAD uses 'real' image pairs, as opposed to augmented images, i.e., $\mathbf{x}^\alpha$ is not a synthetically augmented version of $\mathbf{x}$, but instead it is an auxiliary real image depicting the same object class. This is possible as their formulation does not require any ground truth correspondence during training. In essence, LEAD implements a form of learned dimensionality reduction, which can be effective if the pre-trained features already contain useful information for matching.

EQ and DVE were originally designed such that their embedding network Φ was trained in an end-to-end manner, while CL and LEAD separately trained the encoder network Ψ, followed by the learned projection function ρ. Existing methods often use different network architectures for the encoder and decoder which makes it challenging to compare the objective functions directly. To fairly evaluate these approaches, in our experiments, we use frozen pre-trained networks as the encoder Ψ, and train a separate linear projection head ρ, i.e., $\Phi(\mathbf{x}) = \rho(\Psi(\mathbf{x}))$, for each of the losses.

2.3.3 Unsupervised Asymmetric Correspondence Loss

The LEAD objective aims to preserve distances between features before and after they have been projected into a lower-dimensional feature space. Given two points, u and v, from different images, the loss term effectively tries to enforce $f(\Psi_u(\mathbf{x}), \Psi_v(\mathbf{x}^\alpha))$ and $f(\Phi_u(\mathbf{x}), \Phi_v(\mathbf{x}^\alpha))$ to be as close as possible. The projection tries to maintain both what is similar and not similar between point pairs by preserving their distance. However, the structure of the embedding space does *not* change after this projection step which means that performance is bounded by the quality of the features in the original feature space.

We make a conceptually simple change to the LEAD objective in order to provide the flexibility to allow the model to change distances in the projected feature space. Unlike LEAD, instead of using the same temperature value in the softmax function for both feature spaces, we utilize a different temperature when we calculate the similarity between point embeddings. Specially, we use a smaller temperature for the original

feature space and a larger one for the projected feature space, i.e., $\tau_1 < \tau_2$, resulting in the following loss,

$$\mathcal{L}_{asym} = \frac{1}{|\Omega|^2} \sum_{u \in \Omega} \sum_{v \in \Omega} || \; p(v|u; \Psi, \mathbf{x}, \mathbf{x}^{\alpha}, \tau_1) - p(v|u; \Phi, \mathbf{x}, \mathbf{x}^{\alpha}, \tau_2) ||. \tag{2.4}$$

A smaller temperature makes the distance between closer points smaller and far away points larger. To match these same distance scores, the projection needs to make embeddings of closer points closer and vice versa. Moreover, the objective also preserves the order of distances of point pairs, i.e., close points remain closer compared to further away ones. As a result, the projection needs to capture what is common between already matching point pairs in order to optimize the loss which leads to better embeddings for matching. While this is a relatively small change in the loss formulation, it results in a significant improvement in the performance. As we use different temperature parameters, we refer to our asymmetric projection loss as ASYM. The other difference between ASYM and LEAD is that we make use of Euclidean distance instead of cross entropy as we found this to be more effective. We compare the impact of these design choices via detailed ablation experiments.

2.4 Evaluation Protocol

2.4.1 Evaluation Metrics

There are two dominant approaches for benchmarking the performance of unsupervised correspondence estimation methods: (i) landmark regression and (ii) feature matching. For landmark regression, an additional supervised regression head is trained for each of the landmarks of interest (e.g., the keypoints of a human face) on top of the representation learned by the correspondence network. For matching, one simply computes the distance in feature space to all the points in the second image for a given point of interest in a source image and then selects the closest match as the corresponding point.

We argue that matching is a better task for evaluating the power of learned feature embeddings as regression requires ground truth supervision to train the additional parameters. As matching uses raw feature embeddings it cannot incorporate biases from datasets, e.g., exploiting the average locations of keypoints. While current literature tends to focus on regression evaluation, there are some exceptions to this. However, by and large, matching results are only presented for comparably easier datasets. For

(a) Source (b) Match (c) Miss (d) Jitter (e) Swap

Figure 2.2. For the keypoint denoted in red in the source image (a), we see the correct match in (b). If the point matches with the background it is a miss (c), if it is close to the correct location it is a jitter (d). If the match is in the correct vicinity but closer to another semantic part, it is a swap error (e).

example, (Thewlis et al., 2019; Cheng et al., 2021; Karmali et al., 2022) only present matching results on the MAFL dataset (Zhang et al., 2015). MAFL contains cropped and aligned images of human faces, and current methods perform very well on it, with matching errors close to two pixels on average.

2.4.1.1 Percentage of Correct Keypoints (PCK)

Traditionally, matching performance is measured using the PCK metric. Given a set of ground truth keypoints $\mathcal{P} = \{\mathbf{p}_m\}_{m=1}^{M}$ and predictions $\hat{\mathcal{P}} = \{\hat{\mathbf{p}}_m\}_{m=1}^{M}$, PCK is calculated as $PCK(\mathcal{P}, \hat{\mathcal{P}}) = \frac{1}{M}\sum_{m=1}^{M} \mathbb{1}[\|\hat{\mathbf{p}}_m - \mathbf{p}_m\| \leq d]$. Here, $d = \alpha \max(W^b, H^b)$ is a distance threshold, chosen as a proportion (e.g., $\alpha = 0.1$ of the maximum side length) of the object bounding box (with width W^b and height H^b) size. A prediction is counted as correct if it is inside of the target keypoint area.

2.4.1.2 Detailed Error Evaluation

Inspired by (Ruggero Ronchi and Perona, 2017), we define additional error metrics to analyze the performance of different methods in more detail. A visual overview is illustrated in Figure 2.2. If a point is matched with a point that not is close to any of the keypoints in the target image, we denote this error as a 'miss'. This error generally occurs when a point is matched with the image background: $E_{miss} = \mathbb{1}[d < \min\{\|\hat{\mathbf{p}}_m - \mathbf{p}\| \mid \mathbf{p} \in \mathcal{P}\}]$. If a prediction is in the correct vicinity, but outside of the defined distance threshold, we denote this a 'jitter', $E_{jitter} = \mathbb{1}[d < \|\hat{\mathbf{p}}_m - \mathbf{p}_m\| < 2d]$. The last error type is a 'swap' which occurs when a point matches in an area that is closer to a different keypoint, $E_{swap} = \mathbb{1}[\delta \neq \|\hat{\mathbf{p}}_m - \mathbf{p}_m\| \wedge\ d > \delta]$, where $\delta = \min\{\|\hat{\mathbf{p}}_m - \mathbf{p}\| \mid \mathbf{p} \in P\}$.

The miss and jitter errors are also counted as incorrect by the PCK metric, but swaps may still be counted as correct. For instance, a prediction which is in the middle

of a pair of eyes might still be counted as correct according to PCK even if it is closer to the wrong eye since it could be still within the distance threshold. As our goal is to estimate semantic correspondence, we should aim to match with the *correct* semantic part. As a result, we propose a new version of PCK which penalizes these swaps. Under this metric, to make a correct prediction, a point needs to both match close to the corresponding keypoint *and* the closest keypoint should be the same semantic keypoint,

$$PCK^{\dagger}(\mathcal{P},\hat{\mathcal{P}}) = \frac{1}{M}\sum_{m=1}^{M} \mathbb{1}[\|\hat{\mathbf{p}}_m - \mathbf{p}_m\| \leq d \ \wedge \ \delta = \|\hat{\mathbf{p}}_m - \mathbf{p}_m\|]. \tag{2.5}$$

2.4.2 Evaluation Datasets

In order to evaluate semantic correspondence performance we perform experiments on five different datasets: AFLW (Koestinger et al., 2011), Spair-71k (Min et al., 2019b), CUB-200-2011 (CUB) (Wah et al., 2011), Stanford Dogs Extra (SDog) (Khosla et al., 2011; Biggs et al., 2020), and Awa-Pose (Xian et al., 2018; Banik et al., 2021). These datasets were chosen as they span a range of object category types (e.g., from man-made to natural world classes) and exhibit different levels of difficulty (e.g., from topologically simple human faces to deformable animals). AFLW (Koestinger et al., 2011) contains images of human faces with various backgrounds from different viewpoints. However, due to the structured nature of faces, the visual difference between images are limited and thus the task is relatively easy compared to the other datasets. SDog (Khosla et al., 2011; Biggs et al., 2020) and CUB (Wah et al., 2011) contain images of fine-grained visual categories (dogs and birds, respectively) and include highly varying appearance, diverse backgrounds, and non-rigid poses which result in a challenging matching task. Awa-Pose (Xian et al., 2018; Banik et al., 2021) contains images from 35 different animal species and allows us to assess inter-class correspondence as the keypoints are shared across the species. SPair-71k (Min et al., 2019b) contains scenes featuring multiple man-made objects with complex backgrounds, varying object sizes, challenging illumination conditions, and some symmetric object classes such as bottles and plant pots, making the dataset particularly difficult. However, the image pairs are drawn from the same object class, and the overall dataset size is relatively small. An overview can be found in Table 2.1.

Only the annotations in SPair-71K were explicitly collected with a focus on semantic correspondence evaluation. For the other datasets, there are no pre-defined image pairs or standardized correspondence evaluation splits. In the existing litera-

ture random image pairs are selected that make direct comparisons between alternative methods challenging (Zhao et al., 2021; Li et al., 2020a; Choy et al., 2016a). As the keypoint annotations are semantically consistent across instances in these datasets, we create splits for each dataset, where random image pairs are selected from test splits of the datasets. We published these splits in order to aid future evaluation.

Dataset Name	# Images	# Pairs	# Classes	Annotations	Matching Diversity
SPair-71k Min et al. (2019b)	2k	70k	18	KP (3-30), Bbox	Med
Stanford Dogs (SDog) Biggs et al. (2020)	10k	10k	120	KP (24), Bbox	Med
CUB-200-2011 (CUB) Wah et al. (2011)	11k	10k	200	KP(15), Bbox	Med
AFLW Koestinger et al. (2011)	13k	10k	-	KP(5)	Low
Awa-Pose Banik et al. (2021)	10k	10k	36	KP (30-40), Bbox	High

Table 2.1. Summary of the different datasets that we use for evaluating semantic correspondence performance. We also report the metadata that is provided with each dataset: KP (keypoints/landmarks) and Bbox (bounding boxes). With the exception of Spair-71k, there are no pre-defined evaluation pairs for the datasets.

2.4.3 Implementation Details

We perform experiments with two different types of backbone models for our feature encoder Ψ. For the CNN, unless otherwise specified, we extract features from images resized to 384×384, and use the 1024 dimensional features from the conv3 layer of a ResNet-50 (He et al., 2016). We use a ResNet-50 trained on Imagenet (Russakovsky et al., 2015) as our supervised baseline, and MoCov3 (Chen et al., 2021) as our unsupervised CNN. For the Transformer, 8×8 patches from 224×224 images with stride 8 are used as input (similar to (Amir et al., 2022)) and we extract 736 dimensional features from the 9th layer. We also investigate supervised and self-supervised trained backbones. The supervised and self-supervised CNNs are from (He et al., 2016) and (Chen et al., 2021) and the Transformer models are from (Kolesnikov et al., 2021) and (Caron et al., 2021), respectively. During training, we upsample feature maps to 64×64 via bilinear interpolation.

For our projection head ρ, a single 1×1 2D convolution is trained and the dimension of the features is reduced to 256. During training, as in (Cheng et al., 2021), we freeze the feature encoder Ψ. The projection head is trained for 50 epochs using Adam (Kingma and Ba, 2014) optimizer with a learning rate of 0.001. Unless stated otherwise, we report results using the standard PCK metric with $\alpha = 0.1$ for direct comparison to other methods. For EQ, DVE, and LEAD, we set the temperature τ to

0.05 and 0.14 for CL as described in their papers, and set τ_1 to 0.2 and τ_2 to 0.4 for ASYM.

2.5 Experiments

In our experiments, we attempt to answer the following questions: i) how well do current unsupervised correspondence methods perform on challenging datasets, ii) how does the choice of backbone architecture and pre-training objective impact performance, iii) how does the pre-training data source impact performance, iv) how does the data source used for finetuning the correspondence model impact performance, and finally, v) what are the current source of errors, and thus what needs to be done to close the gap between current state-of-the-art supervised and unsupervised methods.

2.5.1 Impact of Unsupervised Correspondence Objective

Projection(ρ)	Spair-71K	SDogs	CUB	AFLW	Awa
None	31.8	34.9	51.3	57.4	28.8
NMF	27.4	33.9	49.6	53.6	28.0
PCA	32.2	35.5	53.1	57.8	29.7
Random	26.9	30.5	43.1	54.9	23.4
Supervised	38.7	53.2	72.7	80.8	46.1
EQThewlis et al. (2017a)	16.4	21.2	28.1	48.5	15.6
DVEThewlis et al. (2019)	16.3	20.5	27.7	58.7	15.4
CLCheng et al. (2021)	30.8	37.0	54.5	**67.3**	31.7
LEADKarmali et al. (2022)	31.7	35.1	51.5	58.0	29.1
ASYM (Ours)	**34.0**	**40.4**	**60.8**	63.6	**34.1**

(a) Ψ = Sup. pre-trained - CNN

Projection(ρ)	Spair-71K	SDogs	CUB	AFLW	Awa
None	30.7	34.3	47.5	64.3	27.6
NMF	20.6	19.9	44.0	40.8	15.6
PCA	27.4	29.8	50.7	51.0	24.1
Random	26.6	31.5	40.0	60.2	23.3
Supervised	39.5	54.0	73.4	83.8	48.2
EQThewlis et al. (2017a)	14.3	20.5	26.4	62.8	15.5
DVEThewlis et al. (2019)	15.0	19.4	28.7	60.6	14.7
CLCheng et al. (2021)	29.7	37.9	54.1	**77.1**	**33.4**
LEADKarmali et al. (2022)	30.5	34.4	48.3	64.9	28.1
ASYM (Ours)	**33.2**	**38.2**	**54.4**	69.7	32.1

(b) Ψ = Unsup. pre-trained - CNN

Projection(ρ)	Spair-71K	SDogs	CUB	AFLW	Awa
None	**33.5**	38.0	66.3	54.1	34.1
NMF	23.3	29.2	55.5	51.5	24.7
PCA	33.0	38.1	66.4	53.9	34.1
Random	31.9	36.9	63.3	52.9	31.8
Supervised	38.5	48.2	78.2	70.5	47.9
EQThewlis et al. (2017a)	15.5	15.9	24.0	60.2	11.7
DVEThewlis et al. (2019)	15.4	17.5	23.8	55.6	11.8
CLCheng et al. (2021)	30.5	35.8	67.1	**68.4**	31.0
LEADKarmali et al. (2022)	32.7	37.6	65.8	53.8	33.9
ASYM (Ours)	33.2	**41.7**	**72.2**	54.2	**38.5**

(c) Ψ = Sup. pre-trained - Transformer

Projection(ρ)	Spair-71K	SDogs	CUB	AFLW	Awa
None	**34.1**	42.7	61.0	64.2	36.1
NMF	26.3	39.0	51.9	61.0	32.9
PCA	34.0	42.7	61.0	64.2	36.1
Random	32.3	42.1	59.6	61.9	34.6
Supervised	38.1	52.7	72.9	92.0	47.4
EQThewlis et al. (2017a)	9.0	12.5	15.0	62.5	8.8
DVEThewlis et al. (2019)	8.5	13.1	14.1	60.6	9.0
CLCheng et al. (2021)	25.8	32.3	54.1	**81.8**	25.0
LEADKarmali et al. (2022)	33.6	42.5	60.8	64.2	35.8
ASYM (Ours)	32.9	**45.2**	**65.2**	65.9	**39.9**

(d) Ψ = Unsup. pre-trained - Transformer

Table 2.2. Comparison of different unsupervised semantic correspondence methods. Here we vary the backbone models and pre-training strategies. The unsupervised correspondence methods are trained on the respective evaluation datasets.

To evaluate the unsupervised correspondence methods outlined in Section 2.3, in Table 2.2, we train a linear projection head ρ on top of the embeddings from a frozen pre-trained backbone Ψ. Additional baselines are also presented, including: pre-trained features directly from the backbone models with no projection (None), Non-Negative Matrix Factorization (NMF), Principal Component Analysis (PCA), projection using a Random weight matrix, and Supervised projection where we optimize the objective in Equation 2.1 using ground truth keypoint pairs. We explore CNNs and Transformers as backbones that are pre-trained either in a supervised or self-supervised fashion.

Overall, our proposed ASYM approach obtains better scores than other unsupervised methods on all datasets, independent of the choice of backbone or pre-training method, with the exception of the AFLW face dataset. Compared to LEAD, our proposed adaptation improves performance on datasets where the visual diversity is high (i.e., non-face datasets). EQ and DVE perform poorly on datasets where the visual appearance is high across instances, but it is worth noting that these methods were originally designed for the end-to-end trained setting. CL obtains good performance in some cases and is the best on AFLW. However, our ASYM method is still consistently strong. Perhaps somewhat surprisingly, PCA based projection performs better than most of the baselines, while NMF did not perform well. PCA's performance can be partially explained by the strength of the original features (i.e., None). Although the performance of unsupervised methods differs across different backbones, the relative ordering stays the same – Sup, ASYM, CL, PCA, NONE, LEAD, NMF, EQ, and DVE.

2.5.2 Impact of Backbone Model and Pre-training Objective

While (Cho et al., 2021) claims that the choice of CNNs or Transformers as the backbone model does not affect the performance, recently (Amir et al., 2022) presented impressive correspondence results using a Transformer-based model. In order to explore further, we compared features from models pre-trained on Imagenet with either supervised (Sup.) or unsupervised (Unsup.) objectives.

When a projection layer is trained with keypoint supervision, the performance difference between architectures diminishes, as can be observed by comparing the supervised baseline to original embeddings (None) in Table 2.2. However, when the projection layer is trained using no supervision, the best results are obtained in the cases where the initial embeddings were the best on a given dataset. For instance, the unsupervised pre-trained Transformer obtains the best results with no projection on the

SDog and Awa datasets compared to other backbone models. Training the unsupervised methods from these embeddings also results in the best performance compared to other pre-trained backbones. In summary, if keypoint supervision is available, the choice of backbone does not significantly impact the end result. However, in the unsupervised case, starting with good performing embeddings is important. Furthermore, the pre-training strategy does not affect the performance of CNNs, while unsupervised Transformers generally perform better than supervised ones (see Table 2.2).

2.5.3 Impact of Pre-training Dataset

Here, we explore the impact of the pre-training data source used to train the feature encoder. We train correspondence losses using embeddings from a CNN trained via contrastive self-supervision on either Imagenet (Russakovsky et al., 2015) (various categories), iNat2021 (Van Horn et al., 2021) (natural world categories), or Celeb-A (Liu et al., 2015) (human faces). Specifically, we use MoCov3 from (Chen et al., 2021) for Imagenet, MoCov2 (Chen et al., 2020c) for iNat from (Van Horn et al., 2021), and MoCov2 from (Cheng et al., 2021) for CelebA. These results are presented in Figure 2.3.

It is clear that the choice of pre-training data has an impact on all unsupervised methods, with Imagenet outperforming other sources. The CelebA model performs poorly on all tasks with the exception of AFLW, as the features likely only contain information about faces. iNat2021 does not contain any man-made objects or dog categories, and as a result, models trained on it perform worse on SDog and Spair. While iNat2021 contains many bird images, it contains an order of magnitude less mammals making it less effective on Awa-Pose.

2.5.4 Impact of Finetuning Correspondence Dataset

Next, we explore how transferable the embeddings are trained on one dataset and evaluated on another. For instance, what happens if the linear projection is trained on dog images and then tested on birds, or in an extreme case, trained on human faces and tested on animal categories. The correspondence losses are trained on top of the sup. CNN from Table 2.2. The results are outlined in Figure 2.4.

The generalization performance across other datasets is poor for supervised losses compared to the unsupervised ones. The performance drop is largest for models trained on faces, but when training on other data and tested on faces, the performance does not

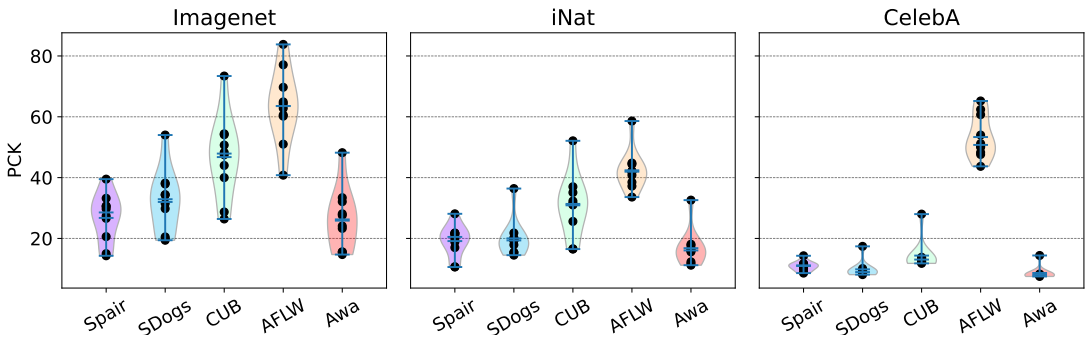


Figure 2.3. Impact of different pre-training datasets used to train a CNN feature encoder using self-supervised training. For each of the three datasets, we report the performance of different methods shown as individual dots. Models pretrained on ImageNet tend to perform better, as their learned features are more generalizable across diverse datasets. Since the three datasets are roughly the same size, differences in dataset size and measurement scale are not significant factors here.

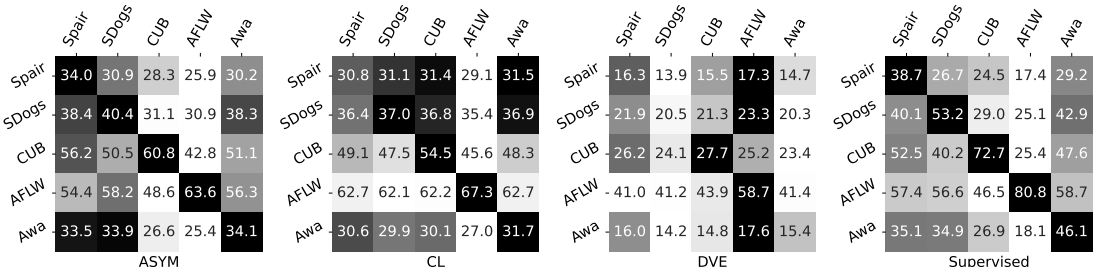


Figure 2.4. Cross dataset evaluation results. Each row represents the test source data, and each column is the dataset that a given correspondence loss is trained on. Note that the colormaps are row normalized. These results use the same initial encoder as the 'Sup. pre-trained - CNN' results in Table 2.2.

drop significantly. Models trained on Spair-71k generally perform reasonably well on other datasets.

2.5.5 Detailed Error Analysis

Here we break down the different error types in order to better understand where the different methods fail and thus require improvement. We compare unsupervised correspondence losses and supervised projection to the current best-performing methods CATs (Cho et al., 2021), CHM (Min and Cho, 2021), and MMNet (Zhao et al., 2021) on Spair-71K. The results are presented in Table 2.3.

For the supervised methods, MMNet has significantly lower miss errors compared to all other methods, although it results in a lot of swaps. As this method combines

	FT	Method	Miss↓	Jitter↓	Swap↓	PCK↑	PCK†↑
Unsup.		CL	51.5	13.7	24.3	30.8	24.2
		EQ	68.3	15.0	18.9	16.4	12.8
		DVE	67.9	14.9	19.7	16.3	12.4
		LEAD	47.1	13.6	27.4	31.7	25.4
		ASYM	44.1	13.2	28.6	34.0	27.2
Sup.		Supervised	40.2	14.9	29.4	38.7	30.4
		CATs Cho et al. (2021)	46.3	21.0	21.9	42.4	31.7
	✓	CATs Cho et al. (2021)	40.1	19.1	20.3	49.9	39.6
	✓	CHM Min and Cho (2021)	40.3	18.2	23.8	44.2	35.8
	✓	MMNet-FCNZhao et al. (2021)	28.5	14.7	28.8	52.2	42.6

(a) $\alpha = 0.1$

	FT	Method	Miss↓	Jitter↓	Swap↓	PCK↑	PCK†↑
Unsup.		CL	71.5	13.2	12.9	17.7	15.6
		EQ	85.1	8.8	8.0	7.6	6.9
		DVE	85.3	9.0	8.3	7.3	6.5
		LEAD	66.9	12.4	15.9	19.3	17.3
		ASYM	63.3	12.6	17.5	21.5	19.2
Sup.		Supervised	61.1	14.6	17.6	24.1	21.3
		CATs Cho et al. (2021)	71.0	20.7	10.8	21.6	18.1
	✓	CATs Cho et al. (2021)	64.8	22.2	10.7	27.7	24.4
	✓	CHM Min and Cho (2021)	64.5	18.7	12.4	25.6	23.1
	✓	MMNet-FCNZhao et al. (2021)	51.7	19.0	18.1	33.3	30.2

(b) $\alpha = 0.05$

Table 2.3. Detailed error types for both unsupervised and supervised correspondence losses on Spair using two different distance thresholds. FT indicates if the backbone was finetuned with keypoint supervision. Our baselines use the 'Sup. pre-trained - CNN' encoder from Table 2.2, in other cases we use the public models by the authors. All models use a ResNet backbone, except MMNet-FCNZhao et al. (2021).

correlation maps from different layers, it is able to capture more global context, which helps reduce misses. However, while CATs and CHM produce more misses compared to MMNet, swaps are reduced, as they use more sophisticated aggregation methods (6D convolution and attention) to resolve ambiguities during matching. Moreover, as these two lines of work complement each other in the error types, they could potentially be combined to obtain better results.

For the unsupervised methods, we see that the most common error type is miss across all methods. While ASYM reduces misses compared to other unsupervised methods, it is not as good as the supervised approaches. As swaps are instances where a match has occurred, but to the wrong keypoint, methods with a high number of misses will not have many swaps by definition. ASYM results in fewer misses, which is desirable, but this increases the chance that swaps can occur. The 'Supervised' baseline reduces misses, but compared to the more sophisticated supervised approaches, it generates more swaps. We argue that while more supervision might help to reduce misses, in order to reduce swaps, better matching mechanisms are needed, as in (Cho et al., 2021; Min and Cho, 2021).

Jitter occurs when a prediction falls within the correct vicinity but lies outside the defined distance threshold. Since this type of error requires the prediction to already be near the target, it cannot be meaningfully interpreted on its own and should be considered alongside the PCK scores. If two methods yield similar jitter scores but one achieves a higher PCK score, this indicates that the method with the higher PCK has fewer jitter errors. For instance, we observe that MMNet-FCN (Zhao et al., 2021) achieves a low jitter error along with strong PCK scores, indicating fewer jitter errors compared to the supervised baseline, which shows a similar jitter score but lower PCK performance.

Finally, we can see that our PCK† metric is reduced by $\sim 20\%$ compared to the original PCK metric in all cases. This indicates that in one in five cases, the source point matches an area closer to another keypoint instead of the correct corresponding point. For some applications, these errors might not affect the end performance drastically, while for others, this disparity could be significant.

We also present the detailed error analysis and report scores using our PCK† metric in Table 2.4 for the other datasets that were used in the previous experiments. Similar to the Spair-71k results from the previous experiments, the most common error type is 'miss' among all datasets. Our ASYM approach generally reduces misses compared to other unsupervised losses. With the exception of the AFLW dataset, there is a noticeable difference between PCK† and PCK scores. For AFLW, the keypoints that correspond to each other are well-defined and far apart from each other as the faces are large. As a result, there are far fewer swaps, and so PCK† scores are close to their PCK counterparts. In contrast, for CUB, most of the points are distributed close to the head region of the birds which leads to a lot of swaps and a drop in scores for our new proposed metric. This highlights the importance of using a proper metric for evaluating

the semantic correspondence task. Matching a keypoint from the beak of a bird to the eye of another bird is not a correct semantic match, but with the current PCK metric, it would be labeled as correct if it was within the distance threshold.

Method	Miss↓	Jitter↓	Swap↓	PCK↑	PCK† ↑
EQ	55.9	21.4	25.9	21.2	18.2
DVE	57.7	21.8	24.8	20.5	17.5
CL	40.9	17.9	27.3	37.0	31.9
LEAD	38.0	16.2	31.2	35.1	30.8
ASYM	33.1	16.3	31.4	40.4	35.5
Supervised	23.7	16.7	29.0	53.2	47.3

(a) SDogs

Method	Miss↓	Jitter↓	Swap↓	PCK↑	PCK† ↑
EQ	44.0	24.8	35.2	28.1	20.9
DVE	44.3	24.6	35.7	27.7	20.0
CL	24.8	20.1	34.6	54.5	40.7
LEAD	28.1	17.4	31.8	51.5	40.1
ASYM	21.7	16.9	29.8	60.8	48.5
Supervised	14.3	15.2	25.4	72.7	60.2

(b) CUB

Method	Miss↓	Jitter↓	Swap↓	PCK↑	PCK† ↑
EQ	38.0	26.0	14.2	48.5	47.8
DVE	24.9	21.2	17.3	58.7	57.8
CL	18.0	11.4	15.2	67.3	66.8
LEAD	13.6	10.7	28.8	58.0	57.5
ASYM	11.7	7.9	25.2	63.6	63.1
Supervised	7.0	4.7	12.7	80.8	80.4

(c) AFLW

Method	Miss↓	Jitter↓	Swap↓	PCK↑	PCK† ↑
EQ	52.0	19.6	38.7	15.6	10.3
DVE	52.1	19.2	37.8	15.4	10.1
CL	38.4	16.8	41.5	31.7	20.1
LEAD	37.1	16.3	44.0	29.1	18.9
ASYM	32.2	16.7	45.6	34.1	22.1
Supervised	23.4	18.3	46.3	46.1	30.3

(d) AWA

Table 2.4. Evaluation of error types across four different datasets. In addition to PCK, we also report scores for our PCK† metric.

2.6 Additional Ablation Experiments and Analysis

2.6.1 Impact of the Temperature Value

In Table 2.5, we explore the impact of the temperature for the different unsupervised losses. While the performance of LEAD, ASYM, and DVE do not change significantly with different temperature choices, the performance of CL is impacted drastically, i.e., when using the recommended value of 0.14 from their paper, we obtain a PCK of 30.8 for Spair-71K in Table 2.2. As noted in the main experiments, for EQ, DVE, and LEAD we set the temperate τ to 0.05 and used 0.14 for CL based on the recommendations in the original papers. We use the same temperature values for all datasets.

2.6.2 Impact of Design Choices for ASYM

As our new proposed ASYM loss is an adaptation of LEAD, here we present experiments ablating our design choices. ASYM differs from LEAD in two respects: (i)

Metric	τ_1	τ_2	DVE	CL	LEAD	ASYM
PCK	0.02	0.04	16.5	9.2	31.9	31.7
	0.05	0.1	16.3	8.2	31.7	32.1
	0.1	0.2	16.0	17.2	31.9	33.0
	0.2	0.4	15.7	26.6	31.4	34.0
	0.4	0.8	9.2	15.8	30.1	29.5
$PCK^{\dagger}$	0.02	0.04	12.9	7.5	25.5	25.4
	0.05	0.1	12.4	6.6	25.4	25.8
	0.1	0.2	12.4	13.8	25.4	26.6
	0.2	0.4	12.1	20.0	25.1	27.2
	0.4	0.8	6.9	11.2	23.8	23.1

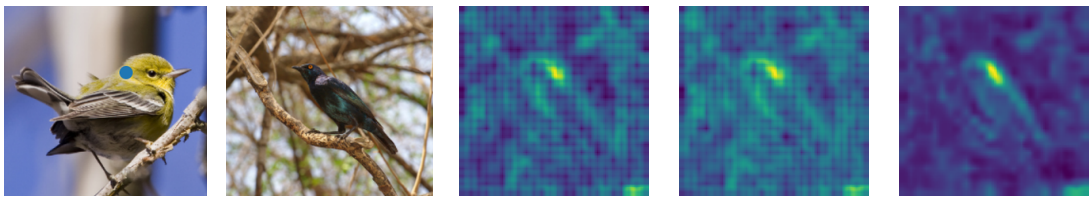
Table 2.5. Temperature ablation experiment for unsupervised losses on Spair-71K. Here we use the 'Sup. pre-trained - CNN' encoder from the previous experiments. With the exception of ASYM, all methods use τ_1 as their τ and do not use τ_2 at all.

ASYM uses different temperature values for the correlation maps for the original features and the projected features, and (ii) ASYM uses a mean square error (MSE), as opposed to cross entropy (CE) which is used in LEAD. As can be seen in Table 2.6, the MSE loss performs worse for LEAD while it improves the performance of ASYM. However, the main difference in overall performance is not a result of the choice of penalty function (i.e., MSE versus CE), but the usage of different temperature parameters. In Table 2.6, we can see that changing the temperature for LEAD has no significant impact on the final performance.

Method	τ_1	τ_2	MSE	CE
LEAD	0.05	-	31.5	31.7
	0.1	-	30.6	31.9
	0.2	-	29.9	31.4
	0.4	-	27.4	30.3
ASYM	0.05	0.1	32.1	32.0
	0.1	0.2	33.0	32.8
	0.2	0.4	34.0	32.0

Table 2.6. Loss and temperature ablation for ASYM and LEAD on Spair-71K. For both methods, Mean Square Error (MSE) and Cross-Entropy (CE) losses are used. ASYM using CE with the same temperature value for both τ_1 and τ_2 is equivalent to LEAD.

Due to changes in the formulation, the objectives that ASYM and LEAD optimize

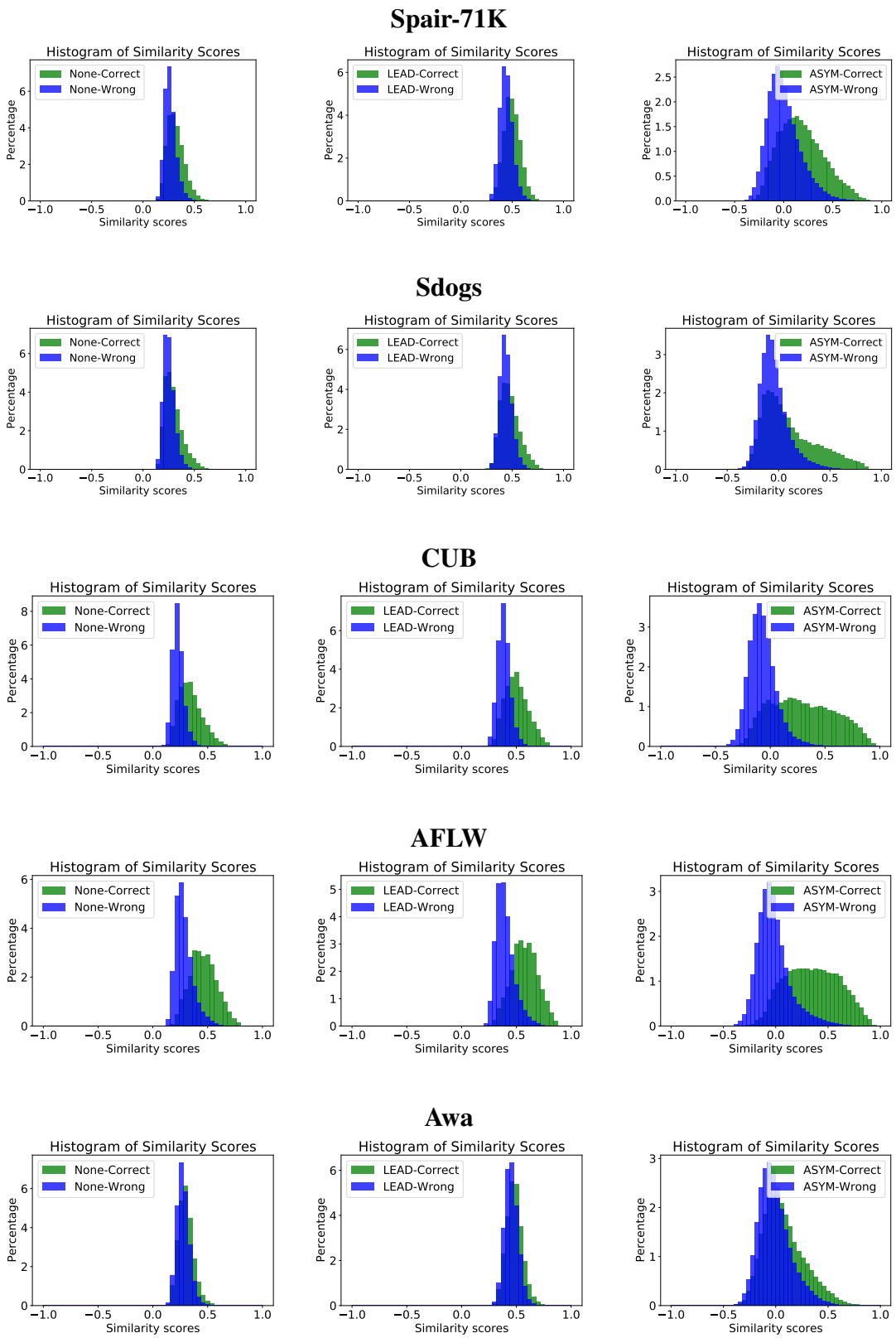


(a) Source (b) Target (c) None (d) LEAD (e) ASYM

Figure 2.5. Feature matching scores for different methods for the keypoint on on the birds head (indicated in blue) from the source images in (a) to the target in (b). By design, LEAD matches the distribution from the original feature space shown in (c). We can see that our ASYM method results in a much more sharper distribution around the correct location compared to LEAD.

also differ. For a given pair of points and their similarity score, LEAD reduces the dimensionality of the embeddings for these points while maintaining the same similarity scores as the input feature space. This is achieved by capturing both what is common and not common between the pair of points. Using higher or lower temperature values does not change the feature distances in the LEAD. However, in our ASYM objective, for a point pair which has a high similarity score, the projection needs to make these points even closer in order to match with the same similarity score from the input features as the projected embeddings use a higher temperature value. A visualization of the result of this can be observed in Figure 2.5. As expected, for a given keypoint and a target image LEAD produces a very similar similarity map compared to the one calculated with the original features. In contrast, ASYM produces a more 'peaked' similarity map, since matching points from original features become closer in the new embedding space.

We also compare how the similarity scores change after unsupervised projection. For a source keypoint, we calculate the cosine similarity scores for all pixel embeddings in the target image. If a point is within the threshold area of a target keypoint we refer to these points as 'correct' matches, otherwise they are classed as 'wrong' matches. We visualize the histogram of these scores for all datasets in Figure 2.6. As can be seen from the distributions, LEAD results in histograms that are very similar to original input features (i.e., None). However, ASYM reduces the overlap between the correct and wrong distributions. As expected, if the similarity scores for correct matches are not larger than wrong matches, ASYM cannot improve the embeddings significantly, as seen in the Awa dataset.



(a) None (b) LEAD (c) ASYM

Figure 2.6. Histograms for cosine similarity scores of embeddings for (a) None, (b) LEAD, and (c) ASYM. Each row is a different dataset.

2.6.3 Impact of Encoder Feature Layer

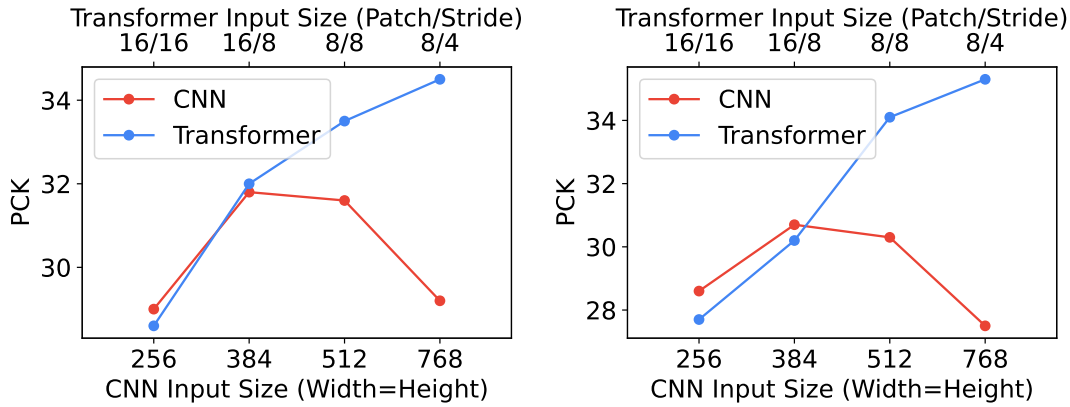
In Table 2.7 we experiment with using features from different feature layers from a CNN (Resnet50 (He et al., 2016)) trained using supervision on Imagenet. The third convolution layer performs best on all datasets, and so we use features from it in all of our experiments for CNNs. For Transformer backbones (Kolesnikov et al., 2021; Caron et al., 2021), we used the 9th layer as the initial features, as they were shown to perform best in (Amir et al., 2022).

Layer	Spair-71K	SDogs	CUB	AFLW	Awa
$conv_1$	7.3	5.1	7.9	11.6	5.6
$conv_2$	12.9	8.6	13.3	27.2	9.1
$conv_3$	31.8	34.9	51.3	57.4	28.8
$conv_4$	15.8	10.3	14.0	31.3	9.3

Table 2.7. Evaluation of using pre-trained features from different layers for the Resnet50 trained with Imagenet. The results here for $conv_3$ correspond to the no projection model (i.e., 'None) from Table 2.2 (a).

2.6.4 Impact of Input Image Resolution

In Figure 2.7, we explore the impact of different input image resolutions, using pre-trained embeddings without any projection (i.e., None), for CNN and Transformer backbones. We used CNNs are from (He et al., 2016) and (Chen et al., 2021) as the supervised and unsupervised CNN, (Kolesnikov et al., 2021) and (Caron et al., 2021) as the supervised and unsupervised Transformer. Transformers scale well as the number of tokens increases, while the performance of the CNNs saturates as the image resolution is increased. We argue that this is due to the non-adaptive nature of the receptive field sizes of CNNs which may overfit to the trained image resolution. As CNNs best performed using an input resolution of 384x384, we use that resolution for in our experiments. While 8x8 patches with stride 4 is the best-performing version for transformers, due to computational constraints, we used 8x8 patches with stride 8 as the transformer input in our experiments.



(a) Supervised Pre-training

(b) Unsupervised Pre-training

Figure 2.7. Semantic correspondence performance of CNNs and Transformers with different input sizes on Spair-71K with no projection. Pre-trained features from models trained on Imagenet with (a) supervised or (b) unsupervised losses are used. Image resolution is fixed to 224x224 for the Transformers. Note that the effective resolution of feature maps from CNNs and Transformers are not comparable for each vertical position in the plots.

2.6.5 Example Images and Qualitative Results

Random instance pairs from each dataset are depicted in Figure 2.8. Spair-71K contains examples of different classes, spanning man-made objects to animal classes. StanfordDogs (SDogs) contains different breeds of dogs in challenging poses with varying appearances. CUB contains bird species. AFLW contains human faces which occupy most of the frame. Unlike CUB and SDogs which only contain images from one species, Awa includes different vertebrate animal categories which enables us to assess inter-category correspondence performance.

We also present some qualitative results for the different unsupervised losses, for all datasets, in Figure 2.9 and Figure 2.10. While ASYM generally improves the predictions compared to other unsupervised losses, it still lags behind supervised projection which makes use of ground truth matches for training. AFLW generally contains easy examples with a small percentage of background pixels and only minor changes in pose which makes the task easier. While the PCK scores for AFLW and CUB are close to each other, as can be seen from qualitative results, this can be explained by how PCK evaluates matches which does not necessarily reflect the difficulty of the dataset in some cases.

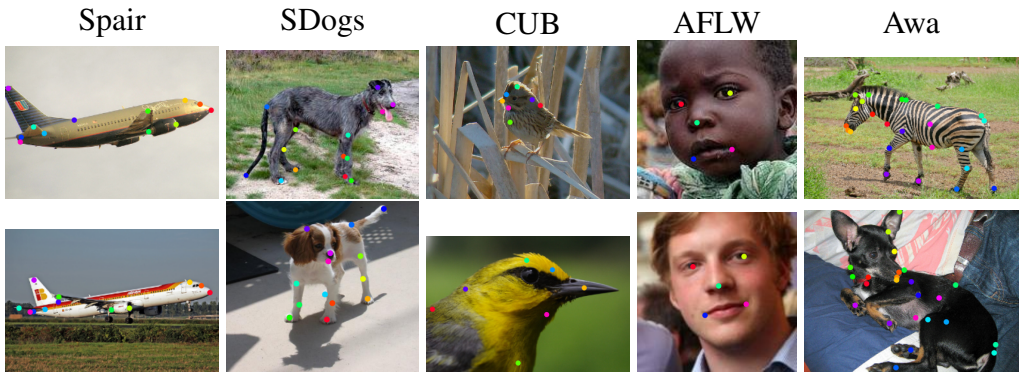
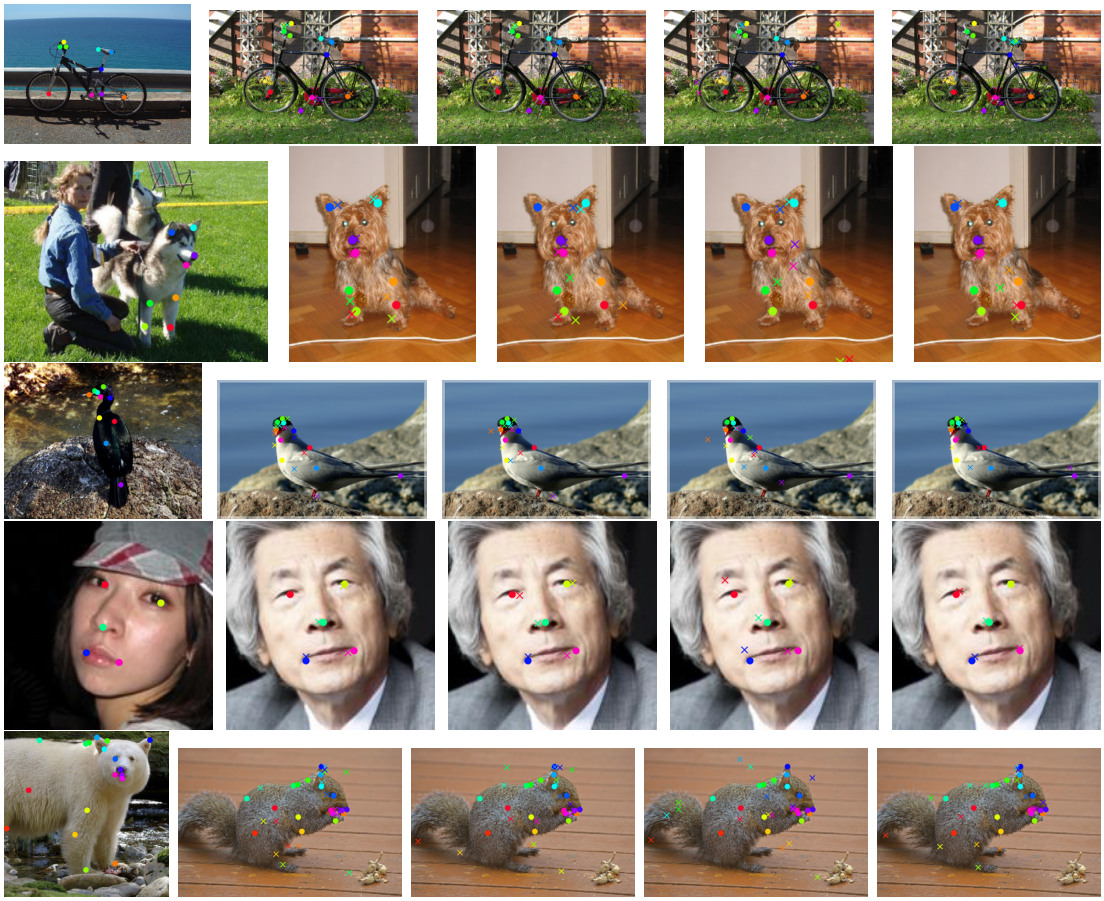


Figure 2.8. Examples from each of the datasets with the keypoint annotations that we consider in our experiments. The top row illustrates a source instance and the bottom a target instance.

2.6.6 Visualizing Learned Feature Embeddings

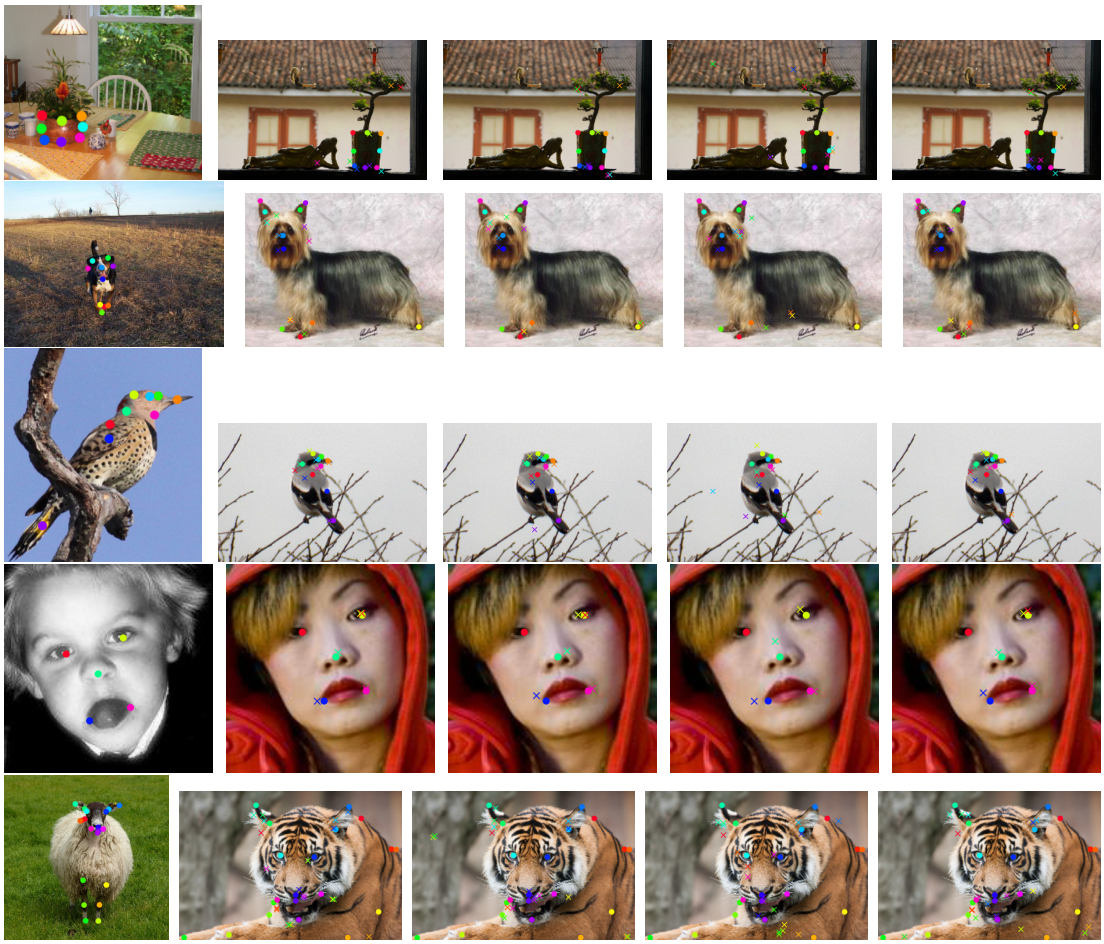
We present 2d t-SNE (Van der Maaten and Hinton, 2008) visualizations of the keypoint embeddings for the AFLW, CUB, and SDogs datasets in Figure 2.12. Since Spair contains different classes wherein the keypoints are not semantically consistent across classes, we did not present a t-SNE visualization of Spair. Moreover, the Awa dataset contains more than 30 keypoints which makes visualizing them difficult, thus we exclude that as well. To create these plots, we first extracted embeddings from only the keypoint locations. These are 1024 dimensional for the None projection and 256 for other unsupervised methods. We then project these embeddings to 2D using t-SNE, and finally plot them. Each color represents a different keypoint type, which is different depending on the dataset.

LEAD and ASYM look similar to the original feature space. One interesting thing is that, CL manages to separate overlapping embeddings when compared to the 'no projection' baseline on the AFLW dataset. This is reflected by their superior PCK scores for this dataset. However, for CUB there are cases where it splits clusters of keypoints which were a single prominent cluster in the original embeddings space. This perhaps indicates that applying CL can sometimes destroy invariances that were captured in the pre-trained features, thus leading to undesirable changes in the embedding space.



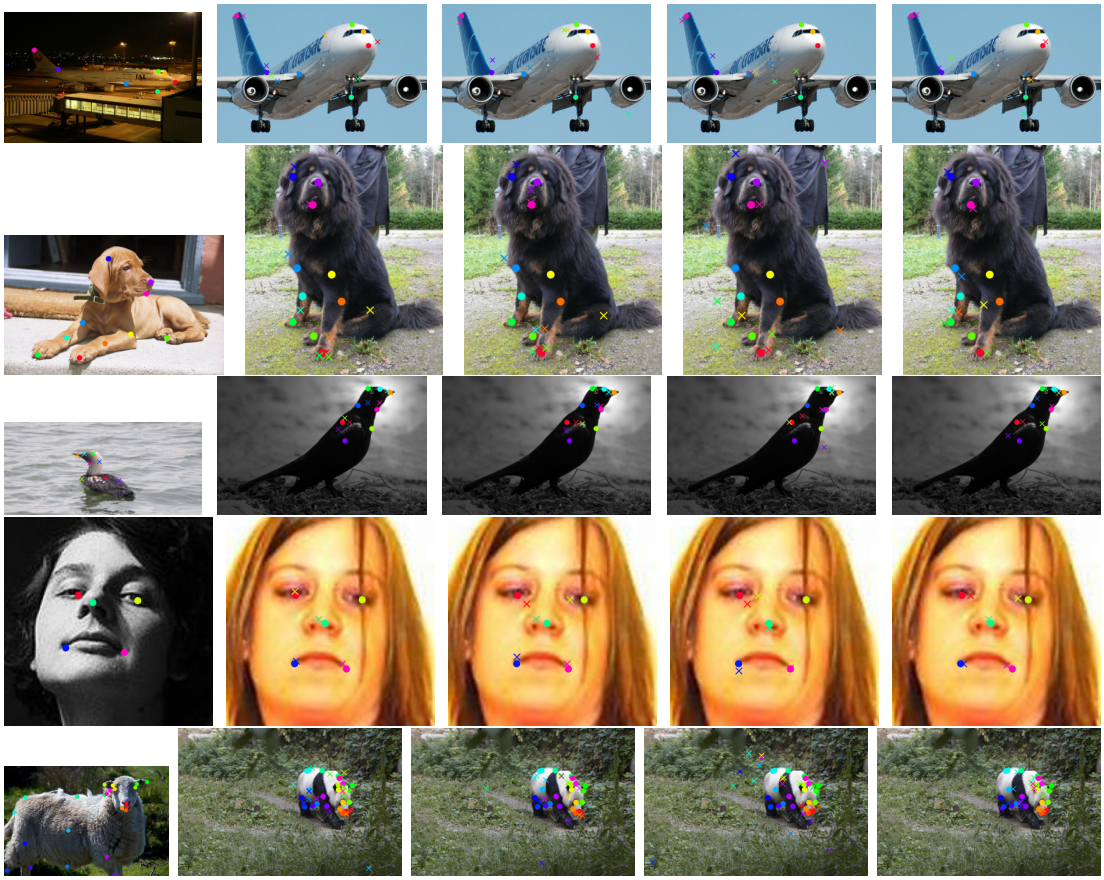
(a) Source (b) ASYM (c) CL (d) DVE (e) Sup

Figure 2.9. Qualitative matching results. Each row is a different dataset: Spair, SDogs, CUB, AFLW, and Awa, from top to bottom. The leftmost image for each row is a source example, and the remaining images visualize matches from different unsupervised methods, where 'o' indicates a ground truth location and 'x' indicates a prediction. Overall, while ASYM cannot match the performance of Supervised projection, it is better than other unsupervised methods. For instance, in the AFLW example, only our proposed ASYM and supervised baseline are able to precisely find correspondences for all keypoints.



(a) Source (b) ASYM (c) CL (d) DVE (e) Sup

Figure 2.10. More qualitative matching results. Each row is a different dataset: Spair, SDogs, CUB, AFLW, and Awa, from top to bottom. The leftmost image for each row is a source example, and the remaining images visualize matches from different unsupervised methods, where 'o' indicates a ground truth location and 'x' indicates a prediction. For the Awa-Pose dataset example in the bottom row, all of the methods struggle as visual diversity is high between instances and the target example is in a different pose.



(a) Source (b) ASYM (c) CL (d) DVE (e) Sup

Figure 2.11. More qualitative matching results. Each row is a different dataset: Spair, SDogs, CUB, AFLW, and Awa, from top to bottom. The leftmost image for each row is a source example, and the remaining images visualize matches from different unsupervised methods, where 'o' indicates a ground truth location and 'x' indicates a prediction. While most methods perform reasonably good on the AFLW dataset instance, the predictions for the highly articulated objects (e.g., animals), even the supervised baseline cannot obtain satisfactory results.

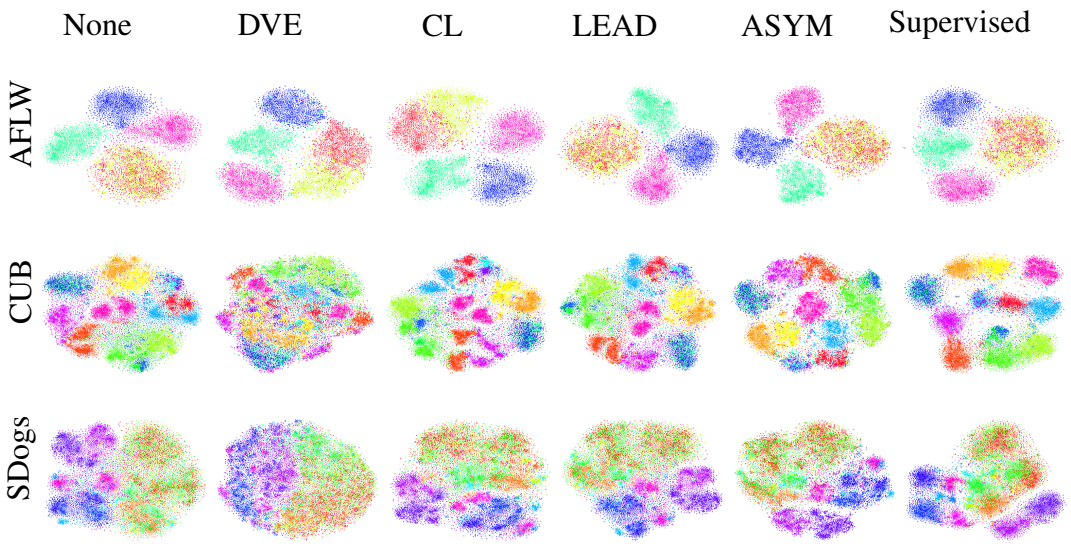


Figure 2.12. t-SNE visualization the embeddings learned by different unsupervised losses. Each row is a different dataset, and the colors indicate the ground truth identity of different keypoints.

2.7 Discussion and Limitations

Our exhaustive experiments show that evaluating with varied challenging datasets is crucial in order to see the benefits of current methods as human face data results (e.g., AFLW) alone can be misleading (Table 2.2). While unsupervised performance may not yet be at the level of fully supervised baselines, they are not far off but have the benefit of generalizing better across datasets (Figure 2.4). Current performance metrics (i.e., PCK) do not penalize all error types and thus result in overly optimistic performance (Table 2.3). The choice of pre-training can have a big impact, but in most instances, Imagenet pre-training is superior (Figure 2.3).

It is not feasible to control all hyper-parameter values as the space is too large. As a result, to ensure fair and controlled comparisons, we adopted a two-stage pipeline, with frozen backbone models, as advocated in recent start-of-the-art work (Cheng et al., 2021). While this two-stage approach simplifies training, it may limit adaptability and overall performance due to restricted feature learning.

We justified the important design choices and provided additional ablation experiments for the choices that we made. Finally, the keypoints used for evaluating correspondence are derived from object landmarks which are detectable and salient by design. In future work, it would be interesting to use additional annotations from other object parts which are not necessarily easily annotated but still have semantically

meaningful correspondences across instances.

2.8 Conclusion

We presented a thorough evaluation of existing unsupervised methods for semantic correspondence estimation and presented a new approach that consistently outperforms existing methods. We showed that while matching performance on human face data is strong, there is still a way to go on more challenging datasets. Our analysis sheds light on some of the reasons for failure as well as providing some further insight into the role of data, models, and losses which we hope will enable others to make further progress on this important task. Furthermore, the methods we studied do not incorporate explicit shape information to inform semantic correspondence, which could be a promising direction for future work.

Chapter 3

SAOR: Single-View Articulated 3D Object Reconstruction

From our investigation of semantic correspondence in the previous chapter, it is apparent that 3D shape information could provide valuable cues for improving correspondence. The challenge of finding correspondences across images containing significant variations in appearance, pose, and background is inherently tied to understanding object structure. Without an explicit notion of shape, this task becomes considerably harder.

To address this, we propose a general 3D reconstruction method that operates across multiple animal categories. Crucially, our approach dispenses with conventional 3D priors such as templates or skeletal models. Instead, it learns to infer shape directly from 2D image collections, relying solely on 2D-derived supervision. While this removes the need for 3D annotations, it also allows the model to generalize across diverse object classes. We believe that such a shape estimation framework can offer a more principled foundation for semantic correspondence and related vision tasks.

3.1 Introduction

Considered as one of the first PhD theses in computer vision, Roberts (Roberts, 1963) aimed to reconstruct 3D objects from single-view images. Despite significant progress in the preceding sixty years (Blanz and Vetter, 1999; Cashman and Fitzgibbon, 2012; Kar et al., 2015; Kanazawa et al., 2018b), the problem remains very challenging, especially for highly deformable categories photographed in the wild, e.g., animals. In contrast, humans can infer the 3D shape of an object from a single image by making

use of priors about the natural world and familiarity with the object category present. Some of these natural-world low-level priors can be explicitly defined (e.g., symmetry or smoothness), but manually encoding and utilizing high-level priors (e.g., 3D category shape templates) for all categories of interest is not a straightforward task.

Recently, multiple methods have attempted to learn 3D shape by making use of advances in deep learning and progress in differentiable rendering (Loper and Black, 2014; Kato et al., 2018; Liu et al., 2019). This has resulted in impressive results for synthetic man-made categories (Choy et al., 2016b; Kato et al., 2018; Wang et al., 2018) and humans (Loper et al., 2015; Güler et al., 2018), where full or partial 3D supervision is readily available. However, when 3D supervision is not available, the reconstruction of articulated object classes remains challenging. This is due to factors such as: (i) methods not modeling articulation (Kanazawa et al., 2018b; Kulkarni et al., 2019a; Goel et al., 2020; Monnier et al., 2022), (ii) the reliance on category-specific 3D template (Kokkinos and Kokkinos, 2021a; Kulkarni et al., 2020; Zuffi et al., 2019) or manually defined 3D skeleton supervision (Wu et al., 2023a,b; Li et al., 2024), or (iii) requiring multi-view training data such as video (Wu et al., 2023a; Kokkinos and Kokkinos, 2021a; Yang et al., 2021a).

In this chapter, we introduce **SAOR**, a novel self-supervised **S**ingle-view **A**rticulated **O**bject **R**econstruction method that can estimate the 3D shape of articulating object categories, e.g., animals. We forgo the need for explicit 3D object shape or skeleton supervision at training time by making use of the following assumption: *objects are made of parts, and these parts move together*. Given a single input image, our proposed method predicts the 3D shape of the object and partitions it into parts. It also predicts the transformation for each part and deforms the initially estimated shape, in a skeleton-free manner, using a linear skinning approach. We only require easy to obtain information derived from single-view images during training, e.g., estimated object silhouettes and predicted relative depth maps. SAOR is trained end-to-end, and outputs articulated 3D object shape, texture, 3D part assignments, and camera viewpoint. Example qualitative results can be seen in Figure 3.7.

We make the following contributions: (i) We demonstrate that articulation can be learned using image-based self-supervision alone via our new part-based SAOR approach which is trained on multiple categories simultaneously without requiring any 3D template or skeleton prior. (ii) As estimating the 3D shape of an articulated object from a single image is an under-constrained problem, we introduce a cross-instance swap consistency loss that leverages our disentanglement of shape deformation and

articulation, in addition to a new silhouette-based sampling mechanism, that enhances the diversity of object viewpoints sampled during training. (iii) We illustrate the effectiveness of our approach on a diverse set of over 100 challenging categories covering quadrupeds and bipeds, and present quantitative results where we outperform existing methods that do not use explicit 3D supervision.

3.2 Related Work

Here we discuss works that attempt to estimate the 3D shape of an object in a single image using image-based 2D supervision during training. We do not focus on works that require explicit 3D supervision (Choy et al., 2016b; Kato et al., 2018; Wang et al., 2018; Mescheder et al., 2019) or multi-view images for training (Yu et al., 2021; Jain et al., 2021; Vasudev et al., 2022; Liu et al., 2023). We also do not cover methods that only reconstruct single object instances (Mildenhall et al., 2021; Park et al., 2021; Poole et al., 2023) or models for multi-object scenes (Niemeyer and Geiger, 2021). For a recent overview of related topics, we refer readers to (Tretschk et al., 2022; Yunus et al., 2024).

Deformable 3D Models. The pioneering work of Blanz and Vetter (Blanz and Vetter, 1999) marked the introduction of deformable models to represent the 3D shape of an object category using vector spaces. By using 3D scans of human faces, they created a deformable model which captured inter-subject shape variation and demonstrated the ability to reconstruct 3D faces from unseen single-view images. This concept was later expanded to more complex shapes such as the human body (Loper et al., 2015; Anguelov et al., 2005), hands (Taylor et al., 2014; Khamis et al., 2015), and animals (Zuffi et al., 2017).

Recent work has combined deep learning with 3D deformable models (Loper et al., 2015; Zuffi et al., 2019; Biggs et al., 2020; Rueegg et al., 2022) to predict the shape of articulated objects from single-view input images. Given an input image, these methods estimate the parameters of a known deformable 3D model and render the object using the predicted camera viewpoint. Although this line of work has led to impressive results for the human body (Loper et al., 2015), the results for deformable animal categories are lacking (Zuffi et al., 2019; Biggs et al., 2020; Rueegg et al., 2022). This is because popular human deformable models, e.g., SMPL (Loper et al., 2015), are constructed using thousands of high-quality real human 3D scans. In contrast, animal focused 3D models, e.g., SMAL (Zuffi et al., 2019), are generated using 3D scans from

a small number of toy animals.

The above models are parameter-efficient due to their low dimensional shape parameterization, which facilitates easier optimization. However, beyond common categories, such as dogs (Rueegg et al., 2022), it can be prohibitively difficult to find 3D scans for each new object category of interest. In this work, we eliminate the need for prior 3D scans of objects by combining linear vertex deformation with a skeleton-free (Liao et al., 2022) linear blend skinning (Lewis et al., 2000) approach to model the 3D shape of articulated objects using only images at training time.

Unsupervised Learning of 3D Shape. To overcome the need for large collections of aligned 3D scans from an object category of interest, there has been a growing body of work that attempts to learn 3D shape using images from only minimal, if any, 3D supervision. The common theme of these methods is that they treat shape estimation as an image synthesis task during training while enforcing geometric constraints on the rendering process.

One of the first object-centric deep learning-based methods to not use dense 3D shape supervision for single-view reconstruction was CMR (Kanazawa et al., 2018b). CMR utilizes camera pose supervision estimated from structure from motion, along with human-provided 2D semantic keypoint supervision during training and a coarse template mesh initialized from the keypoints. Subsequently, U-CMR (Goel et al., 2020) removes the keypoint supervision by using a multi-camera hypothesis approach which assigns and optimizes multiple cameras for each instance during training. IMR (Tulsiani et al., 2020) starts from a category-level 3D template and learns to estimate shape and camera viewpoint from images and segmentation masks. UMR (Li et al., 2020c) enforces consistency between per-instance unsupervised 2D part segmentations and 3D shape. They do not assume access to a 3D shape template (or keypoints) but instead learn one via iterative training. SMR (Hu et al., 2021) also uses object part segmentation from a self-supervised network as weak supervision. Shelf-SS (Ye et al., 2021) uses a semi-implicit volumetric representation and obtains consistent multi-view reconstructions using generative models similar to (Henzler et al., 2019). Like us, all of these methods use object silhouettes (i.e., foreground masks) as supervision.

Recently, Unicorn (Monnier et al., 2022) combined curriculum learning with a cross-instance swap loss to help encourage approximate multi-view consistency across object instances when training a reconstruction network without silhouettes. Their swap loss makes use of an online memory bank to select pairs of images that contain similar shape or texture. The pairs are restricted to be observed from different esti-

mated viewpoints. Then a consistency loss is applied which explicitly forces pairs to share the same shape or texture. In essence, this is a form of weak multi-view supervision under the assumption that the shape of the object pair are the same. However, this assumption breaks down for articulated objects. Inspired by this, we propose a more efficient and effective swap loss designed for articulating objects.

There are also approaches that predict a mapping from image pixels to the surface of a 3D object template as in (Güler et al., 2018; Neverova et al., 2020). CSM (Kulkarni et al., 2019a) eliminates the need for large-scale 2D to 3D surface annotations via an unsupervised 2D to 3D cycle consistency loss. The goal of their loss is to minimize the discrepancy between a pixel location and a corresponding 3D surface point that is reprojection based on the estimated camera viewpoint. In contrast, we do not require any 3D templates or manually defined 2D annotations, such as keypoints, as they are arduous to collect and not scalable.

Learning Articulated 3D Shape. Most natural object categories are non-rigid and can thus exhibit some form of articulation. This natural shape variation between individual object instances violates the simplifying assumptions made by approaches that do not attempt to model articulation.

A-CSM (Kulkarni et al., 2020) extends CSM (Kulkarni et al., 2019a) by making the learned mapping articulation aware. Given a 3D template of the object category, they first manually define the parts of the object category and a hierarchy between the parts. Then, given an input image, they predict transformation parameters for each part so they can articulate the initial 3D template before calculating the mapping between the 3D template and the input pixels. Recently (Stathopoulos et al., 2023) show that A-CSM can be trained with noisy keypoint labels. Instead of manually defining parts, (Kokkinos and Kokkinos, 2021a) initialize sparse handling points, predict displacements for these points, and articulate the shape using differentiable Laplacian deformation. However, each of these methods requires a pre-defined 3D template of the object category.

DOVE (Wu et al., 2023a), LASSIE (Yao et al., 2022), and MagicPony (Wu et al., 2023b) are recent methods that are capable of predicting the 3D geometry of articulated objects without requiring a 3D category template shape. However, they require a predefined category-level 3D skeleton prior in order to model articulating object parts such as legs. While 3D skeletons are easier to define compared to full 3D shapes, they still need to be provided for each object category of interest and have to be tailored to the specifics of each category, e.g., the trunk of the elephant is not present in other

quadrupeds. In the case of MagicPony (Wu et al., 2023b), in addition to the skeleton and its connectivity, per-bone articulation constraints are also provided, which necessitates more manual labor. Additionally, a single skeleton may be insufficient if there are large shape changes exhibited across instances of the category.

MagicPony (Wu et al., 2023b) builds on DOVE (Wu et al., 2023a), by removing the need for explicit video data during training. Inspired by UMR (Li et al., 2020c), MagicPony makes use of weak correspondence supervision from a pre-trained self-supervised network to enforce pixel-level consistency between 2D images and learned 3D shape. Concurrent to our work 3D-Fauna (Li et al., 2024) extends MagicPony for quadrupeds in a multi-category setting. LASSIE (Yao et al., 2022) is another skeleton-based approach that uses correspondence information from self-supervised features and manually pre-defined part primitives. Like us, they model object parts, but their goal is not to learn a model that can directly predict shape from a single image. Instead, their approach learns instance shape from a set of images via test-time optimization. In recent work, (Yao et al., 2023) automatically extracts the skeleton from a user-defined canonical image, but still requires test-time optimization.

We train with single-view image collections, but there are also several works that use video as a data source for modeling articulating objects (Li et al., 2020b; Wu et al., 2023a; Yang et al., 2021a,b) and other methods that perform expensive test-time optimization for fitting or refinement (Zuffi et al., 2019; Kokkinos and Kokkinos, 2021b; Li et al., 2020b; Wu et al., 2023b; Yao et al., 2022). In contrast, we only require self-supervision derived from single-view images and our inference step is performed efficiently via a single forward pass through a deep network.

3.3 Method

Our objective is to estimate the shape S, texture T, and camera pose (i.e., viewpoint) P of an object from an input image I. To accomplish this, we employ a self-supervised analysis-by-synthesis framework (Grenander, 1978; Kulkarni et al., 2015) which reconstructs images using a differentiable rendering operation, denoted as $\hat{I} = \Pi(S, T, P)$. The model is optimized by minimizing the discrepancy between a real image I and the corresponding rendered one $\hat{I}$. In this section, we describe how the above quantities are estimated to ensure that the predicted 3D shape is plausible. An overview of the generation phase of our method can be seen in Figure 3.1

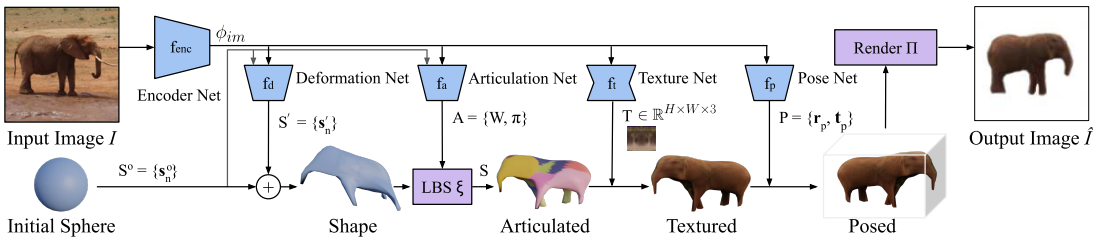


Figure 3.1. Overview of the generation phase of our SAOR method. Given a single image I as input, we extract a global feature vector ϕ_{im} which is decoded by four separate networks (f_d, f_a, f_t, and f_p) to generate a final output image $\hat{I}$. We start by deforming an initial sphere, articulate it using a part-based linear blend skinning (LBS) operation ξ, texture the mesh, and render it using a differential render Π so that it is depicted from the same viewpoint as the input image. The parameters for each of the networks presented are trained in an end-to-end manner using image reconstruction-based self-supervision from multiple different categories using the same model.

3.3.1 SAOR Model

Taking inspiration from previous works (Kanazawa et al., 2018b; Ye et al., 2021; Monnier et al., 2022), we initialize a sphere-shaped mesh with initial vertices S° with fixed connectivity. We then extract a global image representation $\phi_{im} = f_{enc}(I) \in \mathbb{R}^D$ using a neural network encoding function. From this, we utilize several modules, described below, to predict the shape deformation, articulation, camera viewpoint, and object texture necessary to generate the final target shape.

Shape. We predict the object shape by deforming and articulating an initial sphere mesh $S^\circ = \{\boldsymbol{s}^\circ_n\}^N_n$. Here, each of the N elements of S° are 3D coordinates. We estimate the vertices of the deformed shape using a deformation function $\boldsymbol{s}'_i = \boldsymbol{s}^\circ_i + f_d(\boldsymbol{s}^\circ_i, \phi_{im})$, which outputs the displacement vector for the initial points. The deformation function f_d is modeled as a functional field, which is a 3-layer MLP similar to (Niemeyer and Geiger, 2021; Monnier et al., 2022). As most natural objects exhibit bilateral symmetry, similar to (Kanazawa et al., 2018b), we only deform the vertices of the zero-centered initial shape that are located on the positive side of the xy-plane and reflect the deformation for the vertices on the negative side. We then articulate the deformed shape using linear skinning (Lewis et al., 2000) in a skeleton-free manner (Liao et al., 2022) to obtain the final shape $S = \xi(S', A)$, where A is the output of our articulation prediction function, which we describe in more detail later in Section 3.3.2.

Texture. To predict the texture of the object, we generate a UV image by transforming the global image feature, $T = f_t(\phi_{im})$. The function f_t is implemented as a convolu-

tional decoder, which maps a one-dimensional input representation to a texture map, $f_t : \mathbb{R}^D \mapsto \mathbb{R}^{H \times W \times 3}$. This approach is similar to previous works (Monnier et al., 2022; Niemeyer and Geiger, 2021). However, unlike existing work (Kanazawa et al., 2018b; Li et al., 2020c) that copy the pixel colors of the input image directly to create a texture image using a predicted flow field, we predict texture directly. In initial experiments, we found that estimating texture flow only gave minimal improvements, for an increase in complexity.

Camera Pose. We use Euler angles (azimuth, elevation, and roll) along with camera translation to predict the camera pose, similar to previous works (Goel et al., 2020; Monnier et al., 2022). Instead of using multiple camera hypotheses for each input instance (Monnier et al., 2022), for each forward pass, or optimizing them for each training instance (Goel et al., 2020), we use several camera pose predictors, but only select the one with the highest confidence score for each forward pass, as described in (Wu et al., 2023b). Specifically, we predict the camera pose as $P \in \mathbb{R}^6 = f_p(\phi_{im})$. Here, $P = \boldsymbol{r}_p, \boldsymbol{t}_p$ represents the predicted camera rotation and translation. This approach accelerates the training process and reduces memory requirements since we only need to compute the loss for one camera in each forward pass. We only incorporate priors about the ranges of elevation and roll predictions, instead of a strong uniformity constraint on the distribution of the camera poses as in (Monnier et al., 2022) or fixed elevation as in (Wu et al., 2023b).

We describe how the entire system is trained, including the loss functions in Section 3.3.4, and further implementation details in Section 3.3.5.

3.3.2 Skeleton-Free Articulation

Many natural world object categories exhibit some form of articulation, e.g., the legs of an animal. Existing work has attempted to model this via deformable 3D template models (Rueegg et al., 2022) or by using manually defined category-level skeleton priors (Wu et al., 2023a,b). However, this assumes one has access to category-level 3D supervision during training. This would be difficult to obtain in our setting as we train on over 100 categories simultaneously. We instead propose a skeleton-free approach by modeling articulation using a part-based model. Our approach is inspired by (Liao et al., 2022), who proposed a related skeleton-free representation for the task of pose transfer between 3D meshes for the human body. However, in our case, we train a model that can predict parts in an image from self-supervision alone.

Our core idea is to partition the 3D shape into parts and deform each part based on predicted transformations. These parts are modeled as mechanically rigid segments that move together, each governed by a common transformation. To achieve this, we predict a part assignment matrix $W \in \mathbb{R}^{N\times K}$, that represents how likely it is that a vertex belongs to a particular part, where $\sum_k^K W_{i,k} = 1$. Here, K is a hyperparameter that represents the number of parts and N is the number of vertices in the mesh. We also predict transformation parameters $\boldsymbol{\pi} = \{(\boldsymbol{z}_k, \boldsymbol{r}_k, \boldsymbol{t}_k)\}_k^K$ for each part which consists of scale $\boldsymbol{z}_k \in \mathbb{R}^3$, rotation $\boldsymbol{r}_k \in \mathbb{R}^{3\times 3}$, and translation $\boldsymbol{t}_k \in \mathbb{R}^3$. Each of these parameters are predicted using different MLPs that take the global image feature ϕ_{im} as input and output $f_a(S^\circ, \phi_{im}) = A = \{W, \boldsymbol{\pi}\}$.

Articulation can be applied to a shape using a set of deformations using the linear blend skinning equation (Jacobson et al., 2014). Here, each vertex needs to be associated with deformations by the skinning weights. In previous works (Wu et al., 2023a; Yao et al., 2022; Wu et al., 2023b), skinning weights are calculated using a skeleton prior (e.g., a set of bones and their connectivity). We instead estimate skinning weights using a part-based model that does not require a prior skeleton or any ground truth part segmentations. We first calculate the centers for each part from the vertices of the deformed shape $\boldsymbol{s}'_i \in S'$,

$$\boldsymbol{c}_k = \frac{\sum_i^N \boldsymbol{s}'_i * W_{i,k}}{\sum_i^N W_{i,k}}. \tag{3.1}$$

The final position of a vertex $\boldsymbol{s}_i$ for the final shape S is then calculated using the skinning weight of the vertex and estimated part transformations as

$$\boldsymbol{s}_i = \sum_k^K W_{i,k} \boldsymbol{z}_k \odot (\boldsymbol{r}_k(\boldsymbol{s}'_i - \boldsymbol{c}_k) + \boldsymbol{t}_k), \tag{3.2}$$

where $\boldsymbol{z}_k$, $\boldsymbol{r}_k$, and $\boldsymbol{t}_k$ are the predicted scale, rotation, and translation parameters corresponding to part k and $\odot$ is an element-wise multiplication. In addition to the reconstruction losses, we apply regularization on the part assignment matrix W that encourages the size of each part segment to be similar for each instance. As each of the above operations are differentiable, articulation is learned via self-supervised without requiring any 3D template shapes (Kulkarni et al., 2020), predefined skeletons (Wu et al., 2023b), or part segmentations (Li et al., 2020c).

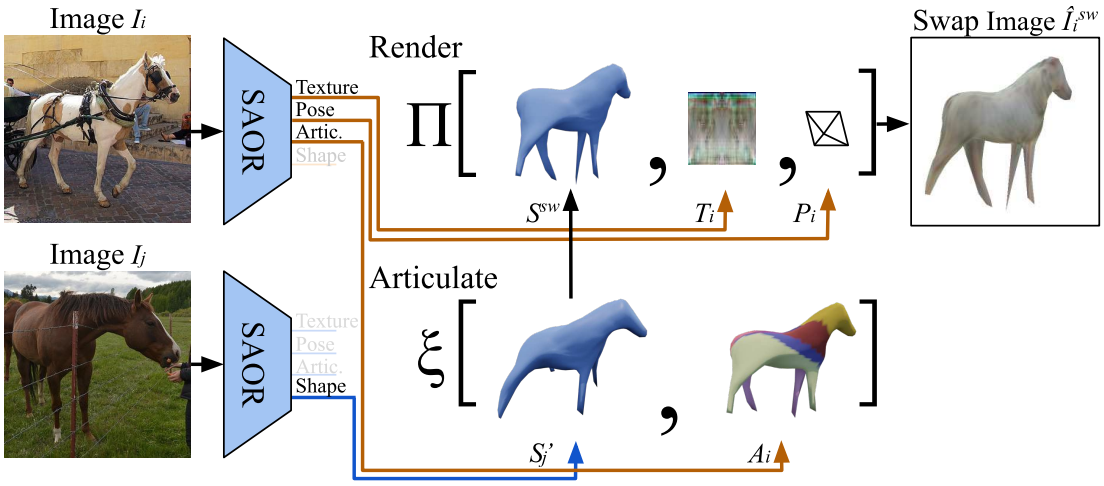


Figure 3.2. Illustration of our articulated swap loss. To calculate the loss, a swap image $\hat{I}_i^{sw}$ is rendered using a randomly chosen paired image's shape S_j', combined with estimated texture, viewpoint, and articulation (T_i, P_i, A_i) from the input image I_i. It ensures that 3D predictions are not degenerate and helps disentangle deformation and articulation.

3.3.3 Swap Loss and Balanced Sampling

One of the hardest challenges in single-view 3D reconstruction is the tendency to predict degenerate solutions as a result of the ill-posed nature of the task (i.e., an infinite number of 3D shapes can explain the same 2D input). Examples of such failure cases include models predicting flat 2D textured planes which are visually consistent when viewed from the same pose as the input image but lack full 3D shape (Monnier et al., 2022). To mitigate these issues, and to ensure multi-view consistency of our 3D reconstructions, we build on the swap loss idea recently introduced in (Monnier et al., 2022).

To estimate their swap loss, (Monnier et al., 2022) take a pair of images (I_i, I_j) that depict two different instances of the same object category, and estimate their respective shape, texture, and camera pose, $(\{S_i, T_i, P_i\}, \{S_j, T_j, P_j\})$. They then generate an image $\hat{I}_i^{sw} = \Pi(S_j, T_i, P_i)$ by swapping the shape encodings S_i and S_j, where Π is a differentiable renderer. Finally, they estimate the appearance loss between I_i and $\hat{I}_i^{sw}$ which aims to enforce cross-instance consistency. The intuition here is that the shape from I_j and texture from I_i should be sufficient to describe the appearance of I_i, even though I_j is potentially captured from a different viewpoint.

In (Monnier et al., 2022), the shapes S_i and S_j should be similar, while the predicted viewpoints P_i and P_j should be different to get a useful 'multi-view' training signal. To obtain similar shapes, they store latent shape codes in a memory bank which is

queried online via a nearest neighbor lookup. This memory bank is updated at each iteration for the selected shape codes using the current state of the network. Moreover, they limit the search neighborhood based on the predicted viewpoints to ensure that they obtain some viewpoint variation, i.e., in (Monnier et al., 2022) the viewpoints P_i and P_j should not be too similar, or too different. While this results in plausible predictions for mostly rigid categories such as birds and cars, for highly articulated animal categories it can lead to degenerate solutions due to more variety in terms of shape appearance, as can be seen in Figure 3.8.

Swap Loss. To address this issue, we introduced a straightforward but more effective swap loss that generalizes to articulated object classes. Our hypothesis is that given a set of images that contain a variety of viewpoints exhibiting disentangled deformation and articulation, we can use randomly chosen image pairs to calculate the swap loss. Since we model the articulation along with the deformation to obtain the final shape, articulation can be used to explain the difference between shapes. In our proposed loss, we swap random deformed shapes S'_i and S'_j from instances of the same object category, but use the original estimated articulation $S^{sw} = \xi(S'_j, A_i)$ and reconstruct the swap image $\hat{I}^{sw}_i = \Pi(S^{sw}, T_i, P_i)$ to calculate the swap loss $\mathcal{L}_{swap}(I_i, \hat{I}^{sw}_i)$. Our loss is illustrated in Figure 3.2. This loss also helps in cases of occlusion, as the model must reason about the occluded regions to minimize the loss with the swapped image, where the same area may not be occluded.

Balanced Sampling. For our swap loss to be successful, it requires the selected image pairs to ideally be from different viewpoints. To obtain informative image pairs, we propose an image sampling mechanism which makes use of the segmentation masks of the input images. Before training, we cluster predicted segmentation masks of the training images and then during training, we sample images from each cluster uniformly to form batches. This ensures that each batch includes the object of interest depicted from different viewpoints. In Figure 3.3 we can see that cluster centers mostly capture the rough distribution of viewpoints and thus help stabilize training. As our image pairs (I_i, I_j) are sampled from within the same batch during training, this results in varied images from different viewpoints for the swap loss. Combined, our swap and balanced sampling steps drastically simplifies the swap loss from (Monnier et al., 2022) and improves reconstruction quality and training stability on articulated classes.

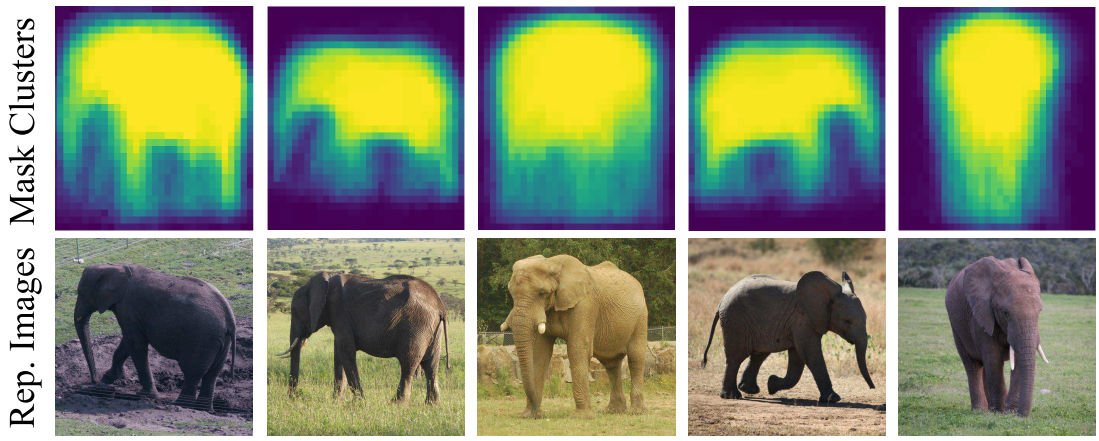


Figure 3.3. (Top) A subset of the resulting cluster centers that arise from clustering the object segmentation masks. (Bottom) Representative images from each of the clusters above. We can see that our simple clustering operation captures the main viewpoint variations present in the data, e.g., left-facing, frontal, right-facing, *etc*.

3.3.4 Optimization

Given an input image, I, we reconstruct it as $\hat{I}$ using estimated shape, texture, and viewpoint. In addition, we use the swapped shape to predict another image $\hat{I}^{sw}$ and calculate the swap loss, as discussed in Section 3.3.3. We also use differentiable rendering to obtain a predicted object segmentation mask and depth derived from the predicted 3D shape, $\hat{M}$ and $\hat{D}$ respectively. Our model is trained using a combination of the following losses,

$$\mathcal{L} = \mathcal{L}_{appr} + \mathcal{L}_{mask} + \mathcal{L}_{depth} + \mathcal{L}_{swap} + \mathcal{L}_{reg}. \tag{3.3}$$

The appearance loss, $\mathcal{L}_{appr}(I,\hat{I})$, is an RGB and perceptual loss (Zhang et al., 2018a), $\mathcal{L}_{depth}(D,\hat{D})$ is the translation and shift-invariant depth loss introduced in (Ranftl et al., 2021), and $\mathcal{L}_{mask}(M,\hat{M})$ estimates silhouette discrepancy. To avoid degenerate solutions, we use $\mathcal{L}_{swap}(I,\hat{I}^{sw})$ and regularize predictions using $\mathcal{L}_{reg}$, which encourages smoothness (Desbrun et al., 1999) and normal consistency on the predicted 3D shape along with a uniform distribution on the part assignment. While we use predicted segmentation masks and relative depth during training, at test time, our model only requires a single image. Below, we describe the training losses in detail.

The appearance loss is a combination of an RGB and perceptual loss (Zhang et al., 2018a). $\mathcal{L}_{appr} = \lambda_{rgb}\mathcal{L}_{rgb} + \lambda_{percp}\mathcal{L}_{percp}$. These terms are defined below,

$$\mathcal{L}_{rgb} = ||\sum_{i,j} I_{i,j} - \hat{I}_{i,j}||_2, \tag{3.4}$$

$$\mathcal{L}_{percp} = ||\phi_p(I_{i,j}) - \phi_p(I_{i,j})||_2, \tag{3.5}$$

where ϕ_p is a function that extracts features from different layers of the VGG-16 (Simonyan and Zisserman, 2015) network.

The mask loss is calculated based on the difference between the automatically generated ground truth segmentation mask M and the estimated mask $\hat{M}$ derived from our predicted 3D shape,

$$\mathcal{L}_{mask} = \lambda_{mask} \sum_{i,j} ||M_{i,j} - \hat{M}_{i,j}||_2. \tag{3.6}$$

Likewise, the depth loss is computed using the automatically generated relative depth D and the estimated depth $\hat{D}$ from the predicted shape,

$$\mathcal{L}_{depth} = \lambda_{depth} \sum_{i,j} ||D_{i,j} - \hat{D}_{i,j}||_2. \tag{3.7}$$

Our swap loss is a combination of the RGB and mask loss between the input image I and swapped image I^{sw},

$$\mathcal{L}_{swap} = \lambda_{swap} \left[\mathcal{L}_{mask}(I, I^{sw}) + \mathcal{L}_{rgb}(I, I^{sw}) \right]. \tag{3.8}$$

Finally, we also employ part regularization on the part assignment matrix W to encourage equal-sized parts,

$$\mathcal{L}_{part} = \lambda_{part} \sum_{k}^{K} \left((\sum_{i}^{N} W_{i,k}) - N/K \right)^2, \tag{3.9}$$

where N is the number of vertices in the mesh and K is the number of parts.

We also apply 3D regularization on the 3D shape,

$$\mathcal{L}_{smooth} = \lambda_{smooth} \sum LS \tag{3.10}$$

where L is the laplacian of shape S and $\mathcal{L}_{normal}$ which is defined below,

$$\mathcal{L}_{normal} = \lambda_{normal} \sum_{\mathbf{n_i}, \mathbf{n_j}} 1 - \frac{\mathbf{n_i}.\mathbf{n_j}}{||\mathbf{n_i}||.||\mathbf{n_j}||}. \tag{3.11}$$

Here, n_i, n_j are normals of neighbor faces, and the smoothness regularization is defined as $\lambda_{smooth}\mathcal{L}_{smooth} = ||LV||$, where L is the Laplacian operator on the vertices. The final regularization term is defined as,

$$\mathcal{L}_{reg} = \lambda_{part}\mathcal{L}_{part} + \lambda_{smooth}\mathcal{L}_{smooth} + \lambda_{normal}\mathcal{L}_{normal}. \tag{3.12}$$

3.3.5 Implementation Details

We employ a ResNet (He et al., 2016) as our global encoder, f_{enc}, and perform end-to-end training using Adam (Kingma and Ba, 2014). Object masks M and depths D are obtained for training by utilizing off-the-shelf pre-trained networks. To implement all 3D operations in our model we use the Pytorch3D framework (Ravi et al., 2020) using their default mesh rasterization (Liu et al., 2019) which is differentiable and enables end-to-end training. Prior to being passed to the model, images are resized to 128x128 pixels. We disable articulation for the first 100 epochs when training a model from scratch, and continue training models for another 100 epochs by enabling deformation and articulation jointly. The lightweight design of our proposed method enables the estimation of the final shape, articulation, texture, and viewpoint in approximately 15 ms per image. We provide more details regarding architecture, hyperparameters, and optimization in Appendix A.

3.4 Experiments

Here, we present results on multiple quadruped and biped animal categories, providing both quantitative and qualitative comparisons to previous work.

3.4.1 Data and Pre-Processing

For our experiments, we trained two models: **SAOR-Bird** and **SAOR-101**. The bird model is trained from scratch using the CUB (Wah et al., 2011) dataset following the original train/test split. SAOR-101, the general animal model, is trained on 101 animal categories that contain birds, quadrupeds, and bipeds. This model is first trained using only horse images from the LSUN (Yu et al., 2015) dataset with an additional 500 front-facing horse images from iNaturalist (iNaturalist, 2023), as LSUN mostly contains side-view images of horses. Then, as in (Wu et al., 2023b), we finetune the horse model on a new dataset that we collected from iNaturalist (iNaturalist, 2023) which contains 90k images from 101 different animal classes. Sample images from our newly collected dataset are shown in Figure 3.4

When constructing our training datasets, we run a general-purpose animal detector (Beery et al., 2019) and eliminate objects if any of the following criteria hold: i) the confidence of the detection is less than 0.8, ii) the minimum side of the bounding box is less than 32 pixels, iii) the maximum side of the bounding box is less than 128 pixels,

Figure 3.4. Sample images from our training dataset, showcasing a diverse set of animal categories under challenging conditions. The dataset includes variations in occlusion, articulation, viewpoint, and appearance, as well as differences in skeletal topology, such as bipeds and quadrupeds.

and iv) there is no margin greater than 10 pixels on all sides of the bounding box. We then automatically extract segmentation masks using the Segment Anything Model (Kirillov et al., 2023) with the detected bounding box. We automatically estimate the relative monocular depth using the transformer-based Midas (Ranftl et al., 2021, 2022), using their Large DPT model.

To obtain cluster centers for the balanced sampling step in Section 3.3.3, we resize the estimated segmentation masks to 32×32, and cluster the 1024-dimensional vectors into 10 clusters using a Gaussian mixture model in all of our experiments. Visualization of cluster centers of various animals can be found in Figure 3.5.

3.4.2 Quantitative Results

To compare to existing work, we quantitatively evaluate using the 2D keypoint transfer task, which reflects the quality of the estimated shape and viewpoint, and 3D evaluation which reflects how predicted and ground truth depth is aligned. We report results using the PCK metric with a 0.1 threshold for the keypoint transfer task, not the $PCK^{\dagger}$ and other detailed error metrics introduced in the previous chapter, as those metrics are related to semantic understanding whereas here the goal is geometric understanding. Furthermore, we use normalized L1 Chamfer distance for 3D evaluation.

Birds. Keypoint transfer results on CUB (Wah et al., 2011) are presented in Table 3.1 both for all bird classes and the non-aquatic subset as in (Wu et al., 2023b). Our method obtains the best results out of methods that do not use keypoint supervision, 3D object

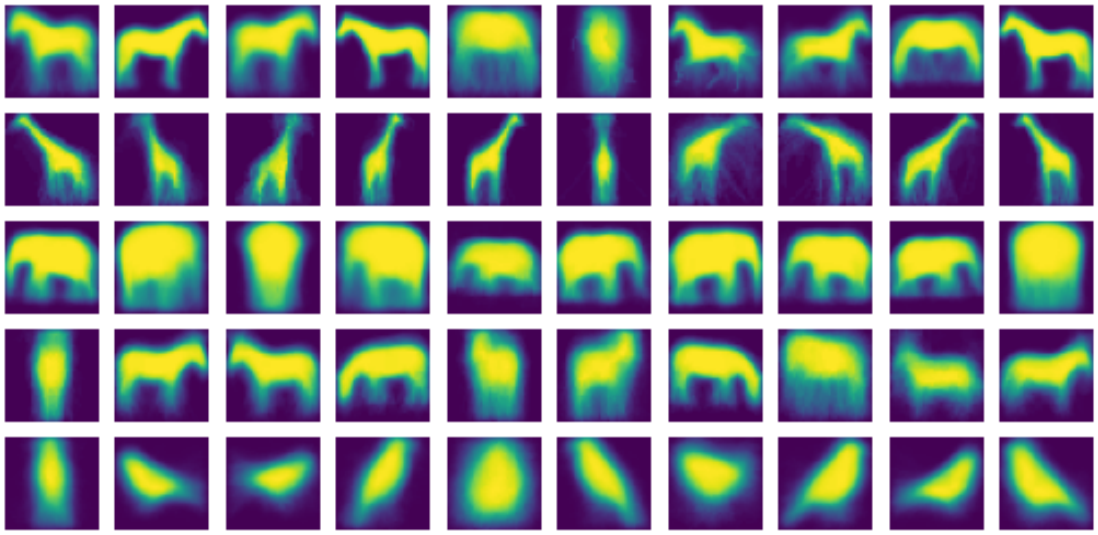


Figure 3.5. Visualization of the cluster centers obtained from estimated silhouettes of various animal categories used in our balanced sampling. We observe that these cluster centers broadly capture the dominant viewpoints of each object category. Top to bottom: horse, giraffe, elephant, zebra, and bird.

priors (e.g., 3D templates or skeletons (Wu et al., 2023a,b)), or additional data (e.g., (Wu et al., 2023a,b)).

Quadrupeds. Keypoint transfer results for quadruped animals from the Pascal dataset (Everingham et al., 2015) are presented in Table 3.2. As noted earlier, we trained the horse model from scratch, while the other models were finetuned using data from iNaturalist (iNaturalist, 2023). For the Unicorn (Monnier et al., 2022) baseline, we used their pre-trained model which was also trained on LSUN horses. For the remaining categories, we also finetuned their model in a similar fashion to ours. Our method outperforms CSM (Kulkarni et al., 2019a) and its articulated version A-CSM (Kulkarni et al., 2020), which use a 3D template of the object category and 3D part segmentation for the horse and cow category. Moreover, our method achieved significantly better scores than Unicorn (Monnier et al., 2022), which produces degenerate (i.e., flat) shape predictions for these classes (see Figure 3.8). We visualize some keypoint transfer results in Figure 3.6.

We also present 3D evaluation using results using Animal3D dataset (Xu et al., 2023) on a few quadruped categories in Table 3.3. The dataset includes pairs of input images with their corresponding 3D models, which are estimated via optimizing the SMAL (Zuffi et al., 2017) model. Moreover, the 3D models are manually verified to

Supervision	Method	all	w/o aqua
🧊*, ✂, 🐾, 📷	CMR (Kanazawa et al., 2018b)	54.6	59.1
🧊*, ✂	U-CMR (Goel et al., 2020)	35.9	41.2
🧍, ✂*, 📹, ⏩*	DOVE (Wu et al., 2023a)	44.7	51.0
🧍, ✂*, 🗜, †	MagicPony (Wu et al., 2023b)	55.5	63.5
✂	CMR (Kanazawa et al., 2018b)	25.5	27.7
✂, SCOPS*	UMR (Li et al., 2020c)	51.2	55.5
None	Unicorn (Monnier et al., 2022)	49.0	53.5
✂*, 📐*	SAOR-Bird	51.9	57.8

Table 3.1. Keypoint transfer results on CUB (Wah et al., 2011) using the PCK metric with 0.1 threshold (higher is better). 🧊 3D template shape, 🧍 3D skeleton, 📷 camera viewpoint, 🐾 2D keypoints, ✂ segmentation mask, ⏩ optical flow, 📹 video, 🗜 DINO features, SCOPS part segmentation, and 📐 monocular depth. † also uses additional video frames from (Wu et al., 2023a). The initial 3D template in (Kanazawa et al., 2018b; Goel et al., 2020) is derived from 2D keypoints. * indicates that the supervision is predicted, hence it is weak supervision. We obtain the best results for methods that do not use 3D templates (🧊), skeletons (🧍), or extra data during training in addition to CUB (e.g., (Wu et al., 2023a,b)).

Supervision	Method	Horse	Cow	Sheep
✂	Dense-Equi (Thewlis et al., 2017a)	23.3	20.9	19.6
✂, 🧊	CSM (Kulkarni et al., 2019a)	31.2	26.3	24.7
✂, 🧊	A-CSM (Kulkarni et al., 2020)	32.9	26.3	28.6
None	Unicorn (Monnier et al., 2022)	14.9	12.1	11.0
🧍, ✂*, 🗜, †	MagicPony (Wu et al., 2023b)	42.9	42.5	26.2
✂*, 📐*	SAOR-101	44.9	33.6	29.1

Table 3.2. Keypoint transfer results for quadruped animals.

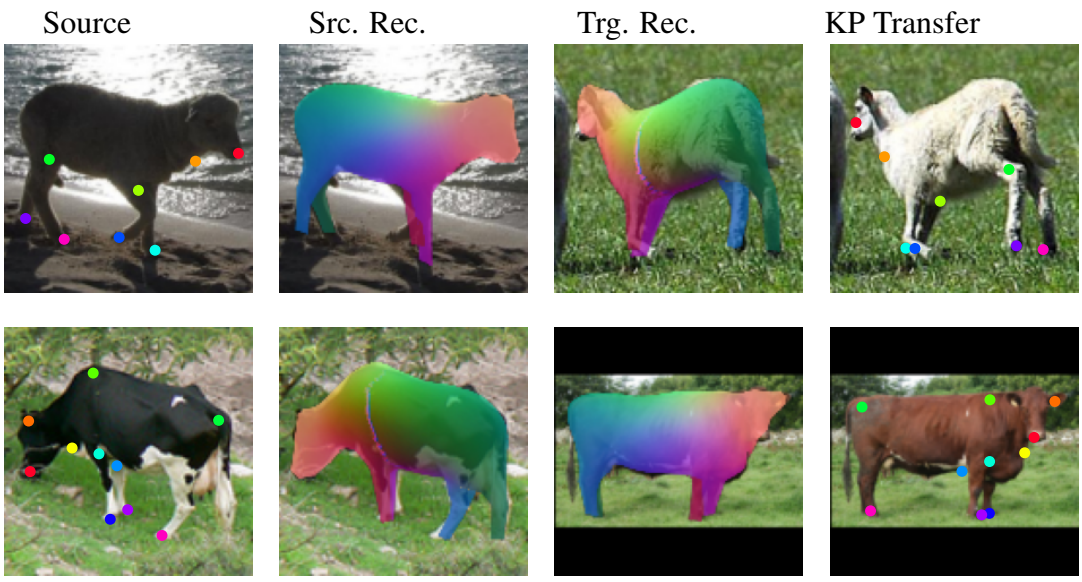


Figure 3.6. Keypoint transfer results. Our model captures articulation and viewpoint differences between images.

Supervision	Method	Horse	Cow	Sheep
None	Unicorn (Monnier et al., 2022)	0.091	0.118	0.134
🧍, ✂*,🗜, †	MagicPony (Wu et al., 2023b)	0.046	0.040	-
✂*, ◢*	SAOR-101	0.046	0.043	0.045

Table 3.3. 3D evaluation on the Animal3D dataset (Xu et al., 2023) using normalized L1 Chamfer error, where lower is better.

eliminate poorly estimated shapes.

The dataset includes pairs of input images with their corresponding 3D We used the test split of the dataset for the horse, cow, and sheep categories. As there is no global pose alignment between our predictions and the dataset, we run the ICP algorithm to align them. We optimize rotation, $R \in \mathcal{R}^3$, translation $T \in \mathcal{R}^3$, and global scale $s \in \mathcal{R}^1$ with the Adam optimizer (Kingma and Ba, 2014) using L1 norm as our alignment objective. We also follow the same alignment steps for the baselines. SAOR obtains better results than Unicorn (Monnier et al., 2022) and similar results to MagicPony (Wu et al., 2023b), while being category agnostic.

Figure 3.7. SAOR capable of predicting the 3D shape of an articulated object category from a single image. Our model is trained on multiple categories simultaneously using self-supervision on single-view image collections. It can efficiently predict object pose, 3D shape reconstruction, and unsupervised part-level assignment using only a single forward pass per image at test time in a category-agnostic way.

Method	Horse	Cow	Sheep	Bird
Ours	44.9	33.6	29.1	51.9
Ours w/o depth	42.4	30.1	26.8	49.9
Ours w/o swap	30.8	17.7	18.4	44.5
Ours w/o sampling	27.5	20.1	18.3	38.8
Ours w/o articulation	26.3	19.4	17.9	41.7

Table 3.4. Keypoint transfer ablation results for SAOR where we disable individual components to measure their impact.

3.4.3 Ablation Experiments

Components. To provide insight into the impact of our proposed model components, we provide ablation experiments on Pascal for quadrupeds and on CUB for birds in Table 3.4. While depth information helps to improve results, we can see that our articulation and swap modules are significantly more important. Our model trained without the swap loss obtains reasonable keypoint matching performance for birds but produces degenerate flat plane-like solutions and fails miserably for quadrupeds. The performance also drops if articulation is not utilized. This is because we choose random pairs for the swap loss (unlike (Monnier et al., 2022)'s more expensive pair selection), and thus only viewpoint changes can be used to explain the difference between images.

Part Ablations. We also conducted an additional ablation experiment on the number of parts used for horses. Results are provided in Table 3.5. Notably, the PCK scores do not significantly vary with different numbers of parts. Therefore, for all other experiments, we used 12 parts.

Number of Parts	6	12	24
PCK	43.8	44.9	44.1

Table 3.5. Keypoint transfer results on Pascal horses (Everingham et al., 2015) where the number of parts are varied.

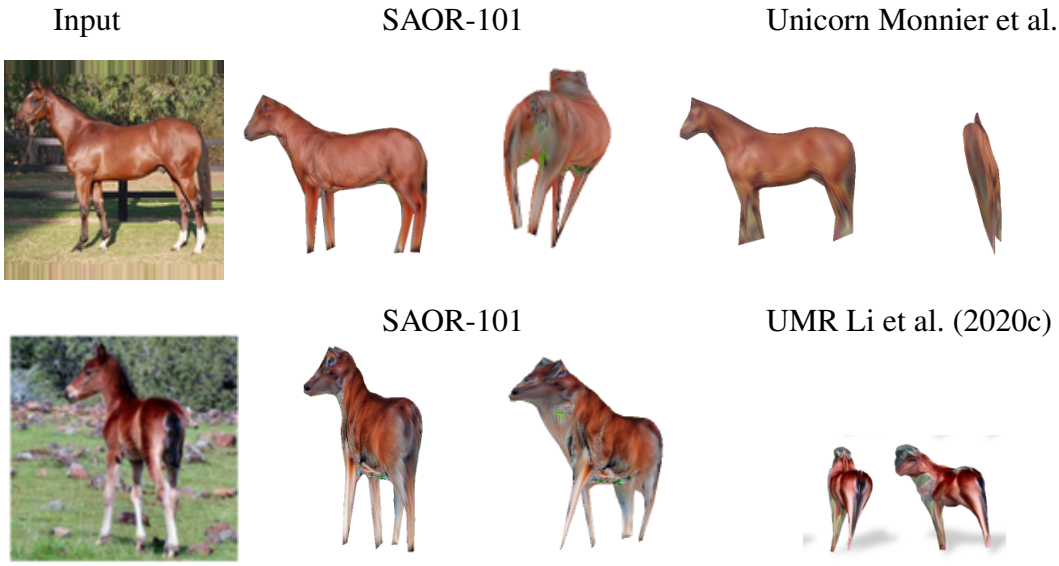


Figure 3.8. Comparison of our model to Unicorn Monnier et al. (2022) and UMR Li et al. (2020c) on horses. Compared to UMR which predicts thin shapes with two legs, we can reconstruct multi-view consistent results with four legs. Unicorn fails to produce 3D consistent shapes.

3.4.4 Qualitative Results

Comparison with Previous Work. We compare SAOR with methods that do not use any 3D shape priors (i.e., Unicorn (Monnier et al., 2022) and UMR (Li et al., 2020c)) and methods that use a 3D skeleton prior (i.e., MagicPony (Wu et al., 2023b)). A comparison of shape predictions for horses can be seen in Figure 3.8 and Figure 3.9. While Unicorn produces reasonable reconstructions from the input viewpoint, their predictions are flat from the side. UMR also predicts thin 3D shapes and does not generate four legs. Our method reconstructs multi-view consistent 3D shapes, with prominent four legs.

In general, our method produces similar results to MagicPony. However, MagicPony's hybrid volumetric-mesh representation requires an extra transformation from implicit to explicit representation using (Shen et al., 2021) and requires multiple rendering operations to estimate the final shape. Moreover, the texture predictions of our methods do not require test-time optimization.

We also compared SAOR's surface estimates with A-CSM (Kulkarni et al., 2020) in Figure 3.10. Unlike A-CSM, our method does not use any 3D parts or 3D shape priors but is still able to capture finer details like discriminating left and right legs. A-CSM groups left and right legs as a single leg while their reference 3D template has

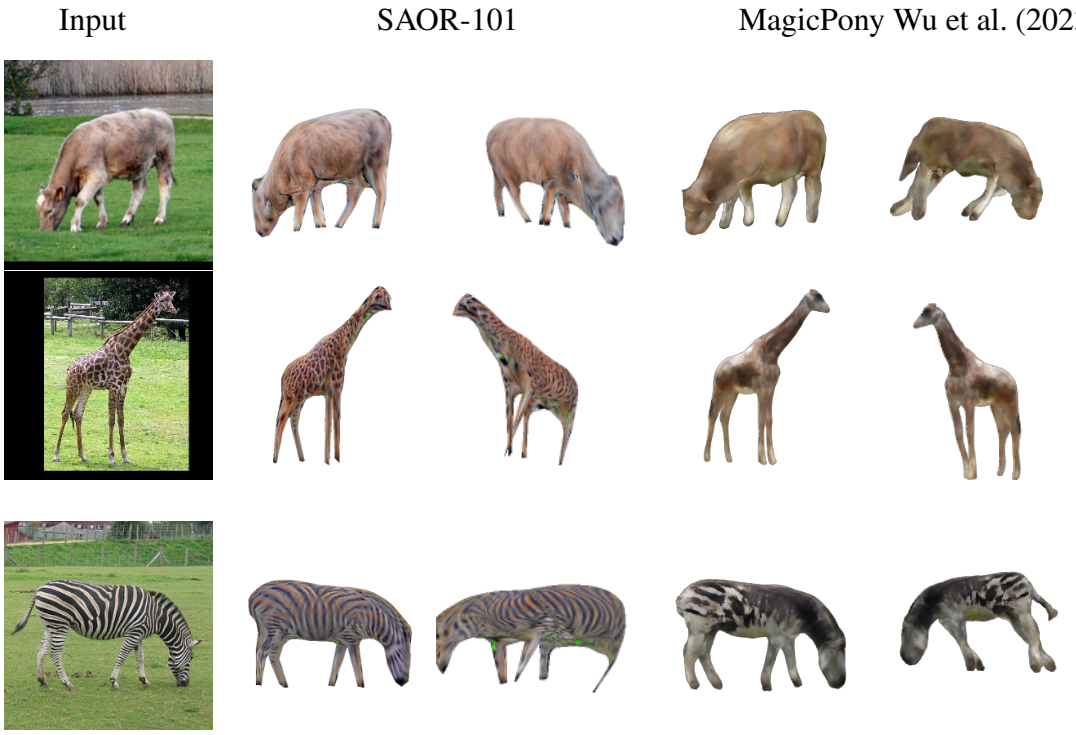


Figure 3.9. Comparison of our model to MagicPony Wu et al. (2023b) (without texture refinement) which uses a category-specific skeleton prior during training. We obtain on-par reconstructions compared to MagicPony without using any 3D prior on the articulation of the object class and with a simpler and more efficient architecture.

left and right legs as a separate entity. Moreover, it mixes left-right consistency if the viewpoint changes.

Deformation and Articulation Disentanglement. In Figure 3.11, we illustrate the disentanglement of articulation and deformation learned by our model. Given two images depicting differently articulating instances, we interpolate the deformation and articulation features between them to visualize reconstructions. While interpolating the articulation feature changes the result, changing the deformation feature does not, as the shape difference between both images can be explained via articulation changes.

Part Consistency. After finetuning the pre-trained horse model on different quadruped categories, we observe that the predicted part assignments stay consistent across categories, as can be seen in Figure 3.7. For instance, although the shapes of giraffes and elephants are significantly different, our method is able to assign similar parts to similarly articulated areas. Here, each color represents the part that is predicted with the highest probability from the part assignment matrix W by the articulation network f_a.

Out-of-Distribution Images. We illustrate the generalization capabilities of our model by predicting 3D shapes from non-photoreal images, e.g., drawings. Figure 3.12 shows

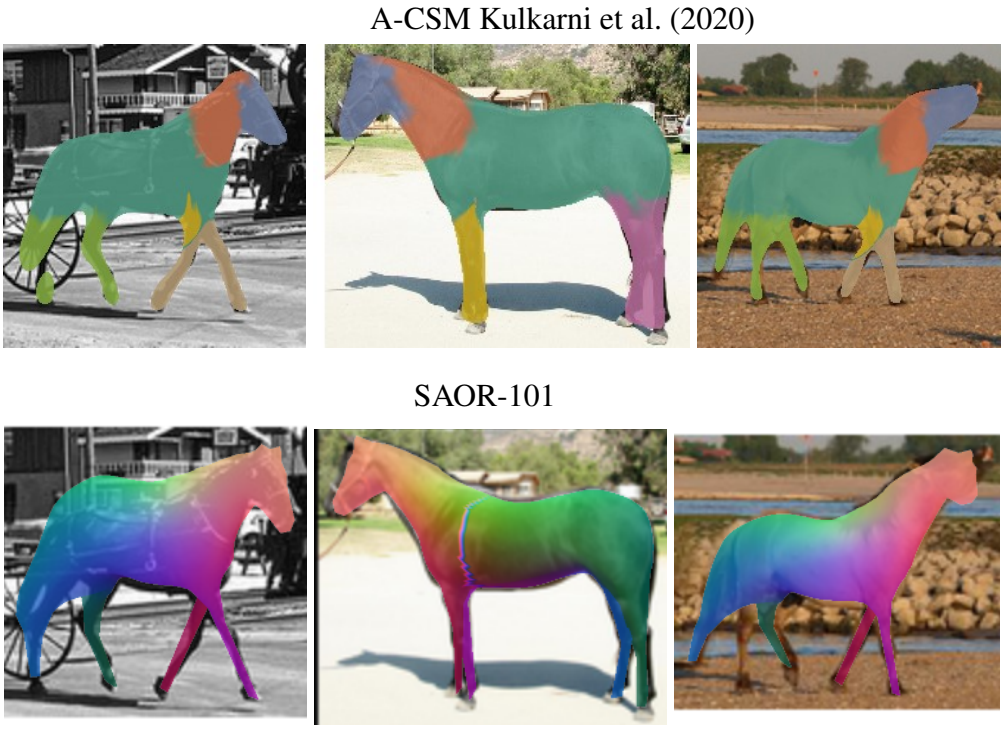


Figure 3.10. Comparison with A-CSM Kulkarni et al. (2020) on horses using example images from their paper. Even though A-CSM uses a 3D template with pre-defined fixed parts, it still maps left and right legs to the same leg in the template and the legs are not consistent across viewpoints (i.e., the part assignment is different in the top row depending on whether the horse is facing left or right. In contrast, despite not using any 3D object priors at training time, our method is much more consistent in its assignment. However, it does mistake one of the left legs for the horse's tail in the final column.

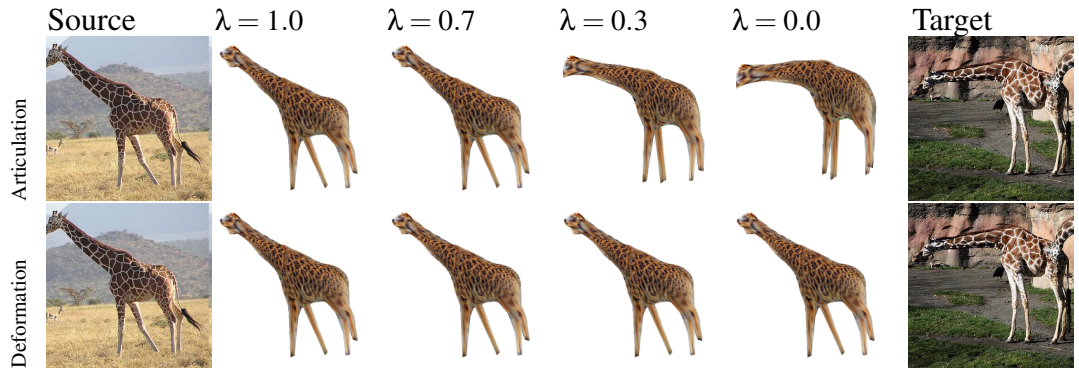


Figure 3.11. Disentanglement of articulation and deformation. On top, we interpolate articulation latent features between a source and target image, and on the bottom do the same for shape deformation features. $\lambda = 1$ indicates that original features are used for reconstruction, while $\lambda = 0$ indicates the target ones. We can see that the difference between the reconstructions is explained by articulation changes between the source and target image pairs.

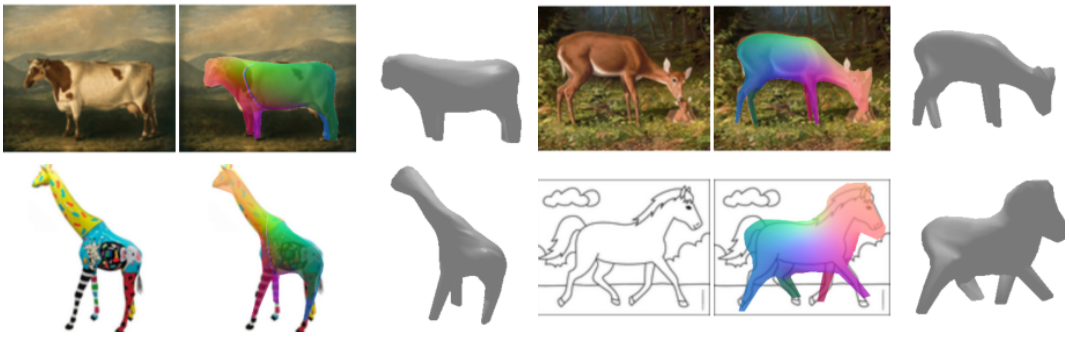


Figure 3.12. Our model, trained on real-world images, plausibly estimates 3D shape and viewpoint from different domains, e.g., cartoons, line drawings, and paintings.

that we can reconstruct plausible shapes and poses from input images that are very different from the training domain.

Without Depth. We also demonstrate examples from a variant of our model that was trained *without* using relative depth map supervision in Figure 3.13. We observe that this model is still capable of estimating detailed 3D shapes with accurate viewpoints and similar textures as the full model. However, the model trained without depth maps tends to produce wider shapes compared to the full model. This shows that, while not crucial, depth provides important spatial cues that help the model generate more accurate 3D shapes. Quantitative results for our model without relative depth are available in Table 3.4.

Additional Qualitative Results. In Figure 3.14 and Figure 3.15, we present additional qualitative results on various animal categories all generated using our SAOR

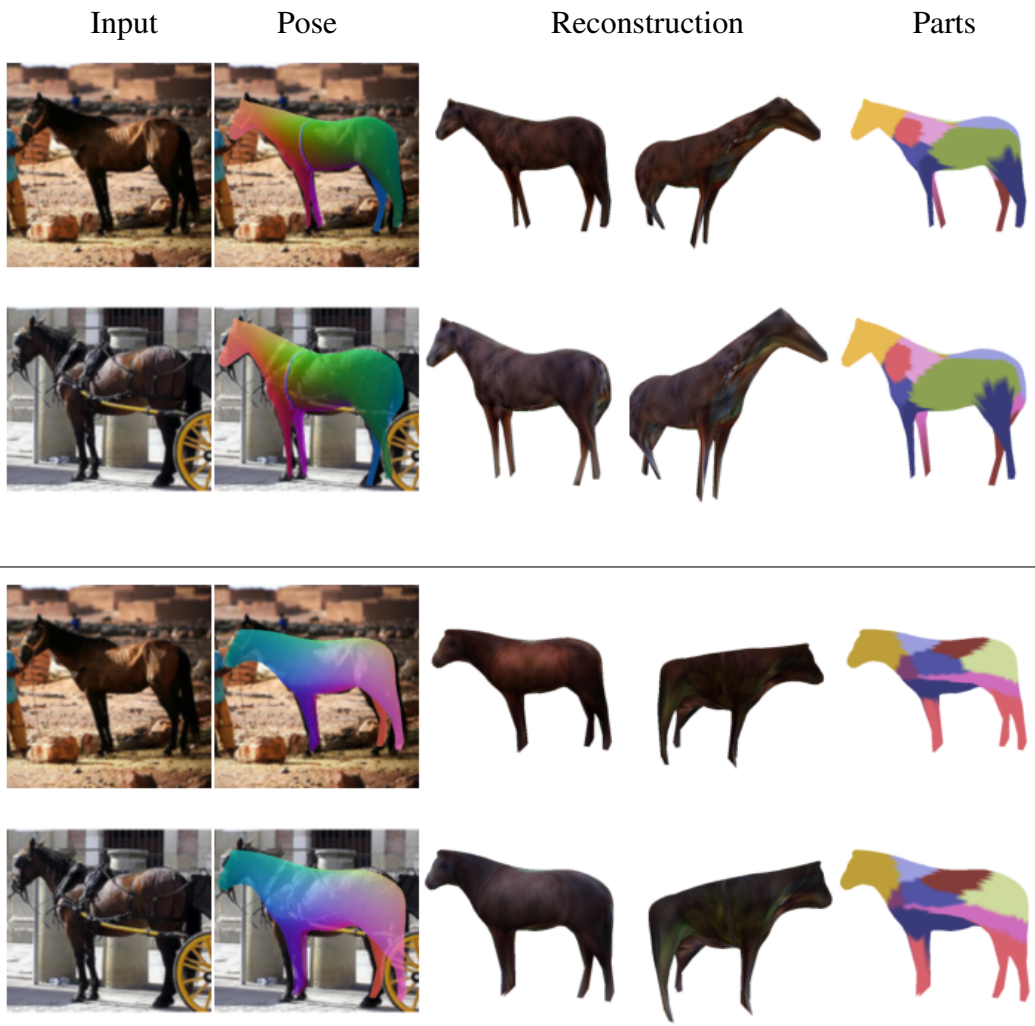


Figure 3.13. Comparison of models trained with relative depth supervision (top) and without (bottom). Our model trained without depth also estimates detailed 3D shapes with the correct viewpoint. However, the 3D predictions are marginally worse as the model without depth produces slightly wider 3D shapes. Please note that part assignment and pose orientation are different since the two models started from different random initializations.

models that are trained on multiple categories. We provide additional results showing full 360-degree predictions for multiple different categories on the project website: mehmetaygun.github.io/saor.

3.5 Discussion and Limitations

Although our proposed approach is able to estimate plausible 3D shapes, the texture predictions are still not fully realistic. This could be improved using test-time refinement similar to (Wu et al., 2023b) or alternative texture representations. During training, our method uses estimated silhouettes and relative depth maps as supervision. Both depth maps and silhouettes come from generic pre-trained models (Ranftl et al., 2022; Kirillov et al., 2023), hence are free to acquire.

Finally, our method fails to predict accurate shape if the input images contain unusual viewpoints that differ significantly from the training images or if the object is not fully visible. We showcase some failure cases of our method in Figure 3.16. Our method fails when the animal is captured from the back, as there is insufficient data available from that angle in the training sets. Note, methods such as (Wu et al., 2023b) partially address this by using alternative training data that includes image sequences from video. Furthermore, when there is also partial visibility (e.g., only the head is visible), our method produces less meaningful results as our architecture does not explicitly model occlusion.

3.6 Conclusion

We presented SAOR, a new approach for single-view articulated object reconstruction. SAOR is capable of predicting the 3D shape of articulated object categories without requiring any explicit object-specific 3D information, e.g., 3D templates or skeletons, at training time. To achieve this, we learn to segment objects into parts which move together and propose a new swap-based regularization loss that improves 3D shape consistency in addition to simplifying training compared to competing methods. These contributions enable us to simultaneously represent over 100 different categories, with diverse shapes, in one model.

Figure 3.14. Additional qualitative results for our SAOR approach on various different animal categories. Note that the part assignment displays the part with the highest probability for each vertex, but in practice, the articulation for each vertex can be explained by a linear combination of multiple parts.

Figure 3.15. Additional qualitative results for our SAOR approach on various different animal categories. Note that the part assignment displays the part with the highest probability for each vertex, but in practice, the articulation for each vertex can be explained by a linear combination of multiple parts.

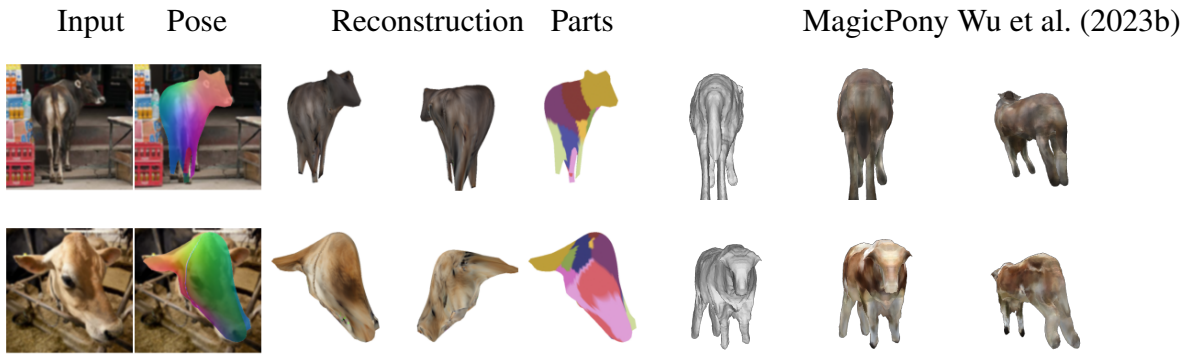


Figure 3.16. Failure cases on cows. On the left we see SAOR-101 predictions (estimated pose, original viewpoint reconstruction, different view, and estimated parts). On the right we display MagicPony Wu et al. (2023b) (original viewpoint reconstruction, textured reconstruction, different view). When the pose is very different than the typical ones present in the training set (top) or there is too much occlusion (bottom) our method fails to produce a sensible shape estimate. For the first example, MagicPony fails to capture the articulation of the head, and for the second occluded example it predicts an average template shape with the wrong pose.

Chapter 4

Enhancing 2D Representation Learning with a 3D Prior

Our exploration of semantic correspondence in Chapter 2 underscored the difficulty of aligning semantically similar regions without a clear understanding of 3D shape. This realization led to the development of a general 3D reconstruction method in Chapter 3, emphasizing the importance of learning shape directly from 2D images without relying on explicit 3D priors. Building on these insights, here we investigate how incorporating 3D priors into self-supervised learning can further improve visual representation learning. In our case, a 3D prior refers to a learned model or system that encodes an implicit or explicit understanding of object geometry, enabling the reconstruction of objects in 3D space.

We introduce a novel approach that integrates 3D shape information into self-supervised frameworks, encouraging models to learn more shape-biased representations. By emphasizing the role of structure in visual learning, this method enhances robustness for semantic tasks like image recognition. Our results suggest that incorporating 3D priors can improve the perceptual alignment of machine vision with human understanding, advancing machine perception toward human-level competence.

4.1 Introduction

The visual stimuli processed by a binocular, actively moving, human observer provides direct information about the 3D world around them (Gibson, 1950). As a result, humans have a remarkable ability to perceive useful 3D shape cues, enabling them to interact and navigate adeptly in complex environments. Most impressively, the power

(a) (b) (c)

Figure 4.1. Humans have no difficulty in recognizing the categories depicted in the above images, even though the texture of the objects has been perturbed. This is thought to be in large part due to our reliance on shape, as opposed to texture, cues (Landau et al., 1988; Spelke and Kinzler, 2007; Geirhos et al., 2019). However, an automated recognition system built on top of a state-of-the-art self-supervised representation learning approach (i.e., DINOv2 (Oquab et al., 2024)) classifies these examples as dog, chair, and knife respectively, as the texture of the images resembles those object classes. We introduce a new approach to improve the robustness of self-supervised methods using a proxy 3D reconstruction task which encourages representations that emphasize shape cues more. As a result, our approach correctly predicted these examples as (a) bear, (b) car, and (c) elephant.

of the human visual system is not understood to be a resulting property of supervised learning, i.e., it has developed thanks largely to 'self-supervision' (Smith and Gasser, 2005).

While great advances have been made in the past decade in developing computer vision systems, their success can be mostly attributed to large-scale *supervised* representation learning. Moreover, current artificial vision systems are not yet nearly as robust as the human equivalent (Geirhos et al., 2019). For example, existing commonly used architectures are known to rely heavily on texture cues, which results in sub-optimal generalization performance (Geirhos et al., 2019; Naseer et al., 2021).

Encouragingly, neural networks that also make use of more shape cues have also been observed to be more robust to different types of image distortions (Geirhos et al., 2019, 2021).

These observations point to two important questions that are potentially hindering our artificial vision systems: (i) how do we reduce the over-reliance on supervised labeled data and (ii) how do we encourage models to make greater use of shape information to improve their robustness? Thankfully, great progress has been made on the

first question as we now have methods for obtaining effective visual representations through self-supervision alone, e.g., (Wu et al., 2018; Chen et al., 2020b; Bao et al., 2022; He et al., 2022; Oquab et al., 2024). While methods exist for extracting shape-adjacent information in the form of depth using self-supervision from collections of image pairs (Godard et al., 2017) or video sequences (Zhou et al., 2017), these approaches tend to require strong assumptions about the scenes they are trained on (e.g., smooth camera motion, static scenes, limited visual diversity, *etc*.). As a result, the current most effective approaches for predicting depth require explicit depth supervision during training (Ranftl et al., 2022). Moreover, even when depth supervision is available, it is not trivial to use it to improve the performance on other tasks (Zamir et al., 2018; Standley et al., 2020).

In this work, we attempt to address these combined challenges by proposing a new method to improve existing self-supervised representation learning approaches by enforcing these models to reason about object/scene shape during training.

We build on recent advances in 3D generative modeling (Chan et al., 2022; Skorokhodov et al., 2023) to develop a self-supervised reconstruction method that generates a 3D representation of the input image. Our model is trained with a self-supervised reconstruction objective, starting from an already trained self-supervised network (e.g., (Oquab et al., 2024)). Given an input image, we first extract a global feature representation using a pre-trained backbone network and then predict a 3D representation of the scene depicted in the image. Then we reconstruct appearance and depth maps using volume rendering from the predicted 3D representation. We use the difference between the reconstructed image and the original input image, and the difference between the predicted depth map and its pseudo ground truth as our training objectives. We do not utilize any manual labels during training as we only require an unordered (i.e., not from videos or stereo pairs) collection of monocular images and their corresponding estimated depth from a previously trained depth prediction model (Bhat et al., 2023) as input. To minimize the training loss, the learned image representation needs to capture details about the shapes of the objects depicted in the input scenes.

While conceptually simple, the advantage of our approach is that it works with monocular image collections and does not make strong assumptions about the types of images it is trained on. As a result, we can train it using standard representation learning datasets such as ImageNet (Russakovsky et al., 2015). Quantitative and qualitative results illustrate that our shape-aware representations are more robust compared to variants that are not shape aware on a variety of downstream tasks. See Figure 4.1

for a qualitative example.

In summary, we make the following contributions: (i) We explore the role of 3D information when performing self-supervised learning on unordered monocular image collections. (ii) We propose a new method that enhances self-supervised learned representations via a proxy task that explicitly encodes 3D knowledge during training. (iii) When applied to a range of robustness tasks, our approach obtains superior performance compared to baselines that do not make use of 3D information at training time.

4.2 Related Work

In this section, we discuss related work in self-supervised learning, monocular shape understanding, and the role of shape in visual recognition.

Self-supervised learning. Recent approaches for deep learning-based self-supervised learning (SSL) in computer vision can be categorized into two groups: (i) predictive methods, where the learning objective depends solely on the input image, and (ii) discriminative approaches, which use additional images as inputs.

Predictive tasks include context prediction (e.g., patch or pixel prediction from a masked input image) (Doersch et al., 2015; Chen et al., 2020a; Bao et al., 2022; Zhou et al., 2022; He et al., 2022), colorization of grayscale input images (Zhang et al., 2016a), in-painting of randomly selected areas (Pathak et al., 2016), predicting image rotation (Gidaris et al., 2018), or object counting in the input image (Noroozi et al., 2017). In contrast, discriminative approaches aim to learn representations that make the input image, and an augmented version of it, more similar to each other compared to other randomly selected images (Hadsell et al., 2006; Wu et al., 2018; Dosovitskiy et al., 2014; Chen et al., 2020b; Oquab et al., 2024; Oord et al., 2018; Grill et al., 2020; Zbontar et al., 2021). Regularization to prevent trivial solutions (Grill et al., 2020; Zbontar et al., 2021; Oquab et al., 2024) and selecting challenging negative examples (He et al., 2020; Chen et al., 2020c) are important considerations for these methods. It is worth noting that some of the above methods make use of both types of losses. For a more comprehensive overview of SSL approaches, we direct the reader to (Jing and Tian, 2020; Gui et al., 2023).

One limitation of the above approaches is that their focus is on 2D representation learning. In this work, we aim to enhance the robustness of self-supervised networks by utilizing a 3D proxy task during training. Recently, (Yu et al., 2023) introduced a

new dataset consisting of common everyday objects containing multiple images, from different camera viewpoints, for each object instance. Their dataset is significantly larger than existing comparable multi-view datasets (e.g., (Henzler et al., 2021)). They use this data to perform view-consistent self-supervised fine-tuning and show that this pseudo-3D supervision results in better downstream image classification performance on their dataset. However, multi-view data of this form is still very cumbersome and time-consuming to collect and thus current datasets are still limited in their scope.

Recently, a new synthetic dataset named Photorealistic Unreal Graphics (PUG) (Bordes et al., 2023) was introduced. It could be used as a source of multi-view data as the images are rendered using 3D assets. However, the images lack realism and the diversity of objects is still not on par with large scale 2D image collections. In this work, we show that it is possible to inject 3D information into a self-supervised model by training on single-view (i.e., not multi-view) image collections alone.

Single-view 3D understanding. Our approach uses a proxy monocular 3D reconstruction task during training to enhance SSL performance. There is also a body of work that aims to estimate 3D shape from monocular images where their focus is on generation and not representation learning.

Example existing works estimate partial 3D shape in the form of depth maps, i.e., per-pixel continuous depth predictions. These methods either use pseudo ground truth depth supervision during training (Ranftl et al., 2022; Bhat et al., 2023) or are trained without depth supervision via image reconstruction losses (Garg et al., 2016; Godard et al., 2017; Zhou et al., 2017; Godard et al., 2019). Another line of work attempts to estimate the full 3D geometry of objects using 3D category priors using explicit representations like meshes (Kanazawa et al., 2018a), implicit representations like surface maps (Güler et al., 2018), or with skeletons (Wu et al., 2023b). The disadvantage of these methods is that they require strong category shape priors (e.g., a 3D deformable model of a human). More recently, there have been some category-centric works that attempt to relax the need for strong shape priors (Monnier et al., 2022; Aygün and Mac Aodha, 2024). However, these are still category focused and are thus limited to specific classes of objects that have well-defined shapes (e.g., animals or humans).

The specific choice of 3D representation (e.g., volume, mesh, or points) used by these methods can have a big impact on the quality of the 3D generated outputs and the computation required to train the model. In the last few years, implicit 3D representations parameterized via neural networks have become widely adopted for a range of 3D tasks (Niemeyer et al., 2020; Yu et al., 2021; Mildenhall et al., 2021). How-

ever, conventional implicit networks can be very slow to train, which hinders their applicability to large-scale SSL. To address this, in this work, we make use of efficient implicit representations popularized by methods that perform 3D generative modeling from single image collections (Chan et al., 2022; Skorokhodov et al., 2023).

Shape and semantics. Finally, we review work that utilizes shape information for visual recognition. It is well established, especially in the early years of cognitive development, that infants more heavily rely on shape cues compared to other cues such as texture during early category learning (Landau et al., 1988; Spelke, 1990; Spelke and Kinzler, 2007). However, computational methods like CNNs (Geirhos et al., 2019) and Vision Transformers (Naseer et al., 2021; Geirhos et al., 2021) do the opposite. With more data, and bigger models, there is some evidence to suggest that this over-reliance on texture may decrease (Dehghani et al., 2023), but it still does not fully disappear.

Prior to the wide adoption of deep-learning methods in computer vision, there were a large number of works that utilized (2D) shape information for recognition tasks. Examples include seminal works such as pictorial structures (Fischler and Elschlager, 1973) and deformable templates (Dalal and Triggs, 2005; Felzenszwalb et al., 2010; Jain and Li, 2011; Pepik et al., 2012). Subsequently, end-to-end trained approaches that did not use any structure or shape overtook these methods. However, recently a new set of methods have been developed that illustrate the benefit of using explicit shape information when combined with end-to-end learning methods for tasks like tracking (Rajasegaran et al., 2022) and action recognition (Rajasegaran et al., 2023).

Furthermore, recent studies have employed alternative forms of training data, such as styled images (Geirhos et al., 2019) and edge maps (Mummadi et al., 2021), to enhance shape awareness, albeit in a supervised context. In this work, we take inspiration from human cognition to add more shape information into our models by developing a proxy 3D reconstruction task to enhance SSL. To solve the resulting 3D reconstruction task, our model needs to learn more about the shape, and not just the texture, of objects during training.

4.3 Method

The aim of visual representation learning is to learn a function that maps an input image to a compact and informative feature vector in a high-dimensional space, capturing important visual properties such as shape, texture, semantics, and object identity. This is achieved by optimizing an objective function on a set of training data.

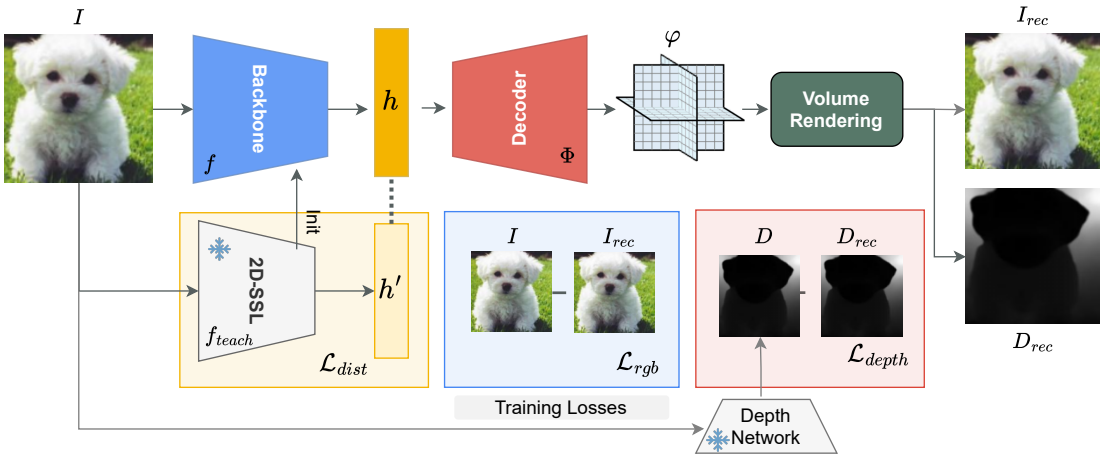


Figure 4.2. Overview of our self-supervised single-view 3D reconstruction approach. Given an input image, I, we first extract a representation of the image using an encoder network, $h = f(I)$. Then using a decoder network, Φ, we generate triplane features (Chan et al., 2022; Skorokhodov et al., 2023). Using volume rendering (Mildenhall et al., 2021), conditioned on a fixed camera location, we reconstruct the input image, I_{rec}, and its depth D_{rec}. We optimize all networks using a combination of reconstruction losses on the input image, $\mathcal{L}_{rgb}$, and estimated depth, $\mathcal{L}_{depth}$, along with a distillation loss, $\mathcal{L}_{dist}$, from a frozen 2D self-supervised learning model to prevent the forgetting of already learned informative representations.

For self-supervised learning (SSL), the objective function is optimized without using human-provided supervision (Wu et al., 2018; Chen et al., 2020b; Bao et al., 2022; He et al., 2022; Oquab et al., 2024). However, given the lack of large-scale and semantically diverse datasets containing 3D information, current self-supervised methods are typically limited to using 2D unordered image collection during training. As a result, the learned representations that emerge from models trained on 2D images are not necessarily fully capable of capturing all the properties of the 3D world (Geirhos et al., 2021). In this work, we aim to improve these learned representations by using an additional proxy 3D task during training. Our aim is not to learn a new function from scratch, but to instead improve an existing pre-trained one.

4.3.1 Background

Our goal is to learn a representation function $f(.)$, represented as a neural network, that can map an input image I into a representation $h = f(I)$, where $h \in \mathcal{R}^d$. This will be achieved by optimizing an objective function $\mathcal{L}$ on a set of training data, without using any manually labeled data. We want to improve networks that are trained with-

out supervision, since it has been shown in (Goldblum et al., 2023) that SSL-based backbones like DINOv2 (Oquab et al., 2024) outperform supervised counterparts as a global image representation.

There are a large number of publicly available self-supervised models that can extract useful representations from 2D images. Therefore, instead of learning a new representation function from scratch, we utilize a backbone that is already pre-trained with 2D self-supervised methods such as DINOv2 (Oquab et al., 2024), and improve it by training it on a new proxy 3D task.

4.3.2 3D Aware Robust Representation Learning

We use 3D reconstruction as our proxy 3D task, i.e., given an input monocular image, at training time, the network is tasked with reconstructing the 3D scene/objects depicted in the image. The intuition behind this is that for the network to successfully perform reconstruction, it must also learn 3D aware features from the input images. As we want the network to learn image representations that transfer well to a large variety of scenes, the input images should be visually diverse and the 3D representation should be able to model complex scenes with multiple objects and diverse backgrounds. Moreover, the reconstruction task should be relatively computationally efficient to enable large-scale training. While there are alternative approaches for generating 3D predictions from 2D images (Nguyen-Phuoc et al., 2019; Henzler et al., 2019; Pan et al., 2021), motivated by the need for efficiency we opted to use triplanes (Chan et al., 2022) as our 3D representation. Triplanes explicitly encode latent network features on axis-aligned planes. These features can then be aggregated via lightweight implicit feature decoders to perform efficient volume rendering for 3D reconstruction. Recently, (Skorokhodov et al., 2023) showed that triplanes can be used to generate 3D depictions of images for various types of scenes from visually diverse datasets such as ImageNet (Russakovsky et al., 2015).

Formally, given an input image I, we first extract a global image representation $h = f(I)$ using a backbone feature extractor network $f(.)$. This backbone can be pre-trained using a 2D self-supervised method. Then we use a decoder $\Phi(.)$ to generate triplane features from the representations of the input image. Note that we only require the decoder and triplane at training time and they can be discarded at inference, as we only need to retain the backbone.

This decoder takes the backbone features as input and produces triplane features

$\varphi = \Phi(h), \varphi \in \mathcal{R}^{H \times W \times C \times 3}$. The decoder consists of two components, a set of learnable triplane embeddings $\xi \in \mathcal{R}^{(h \cdot w \cdot 3) \times D}$ and upsampling blocks. As done in (Shen et al., 2023), we first apply cross-attention between the triplane embeddings and the image representation to obtain low resolution triplane features, $\varphi' = \text{cross}(\xi, h), \varphi' \in \mathcal{R}^{h \times w \times C \times 3}$. Then we apply upsampling layers, which consist of bilinear upsampling and convolution operations, to obtain full resolution triplane features $\varphi = \text{upsample}(\varphi')$. Different than (Shen et al., 2023), we do not employ any quantization or style mapping (Karras et al., 2020) as our goal is not to learn an unconditional generator, but to estimate 3D representation from the input image.

To perform volume rendering, we compute the radiance field using a simple two-layer MLP similar to (Skorokhodov et al., 2023; Chan et al., 2022) using features from the triplane at specified 3D points. Note that as (Chan et al., 2022) and (Skorokhodov et al., 2023) are generative methods, and not representation learning approaches, they generate triplanes from random codes. In contrast, our approach generates triplanes conditioned on the input image's representation h which is obtained from the backbone network. Using volume rendering from the triplane features, we produce the reconstructed image, I_{rec}, and its corresponding depth map, D_{rec},

$$I_{rec}, D_{rec} = \Pi(\Gamma(\varphi, \pi)), \tag{4.1}$$

where Γ is the function that queries the radiance fields from triplanes conditioned on camera pose π which contains extrinsic and intrinsic parameters, and Π is a volume rendering operation (Mildenhall et al., 2021). We use a fixed camera pose in our experiments since we want to learn a viewer-centered 3D representation, which is shown to be more generalizable compared to object-centric representations (Shin et al., 2018; Tatarchenko et al., 2019). Moreover, as we reconstruct the whole scene with potentially multiple objects, the canonical pose is ambiguous.

Given that 3D reconstruction from a 2D image is an ill-posed problem, like 3DGP (Skorokhodov et al., 2023), we make use of 2D depth information to produce plausible 3D predictions. As ground truth depth maps are not available for large-scale, in-the-wild, datasets like ImageNet (Russakovsky et al., 2015), we use pseudo ground truth depth maps obtained from off-the-shelf monocular depth methods such as ZoeDepth (Bhat et al., 2023). Different from 3DGP (Skorokhodov et al., 2023), we do not modify the depth reconstruction with an adapter, but use it as it is for computing a depth reconstruction loss.

Given the input image I^i and its pseudo depth D^i, we simply train the decoder such

that it generates a plausible 3D prediction that is capable of reconstructing them both using the following losses

$$\mathcal{L}_{rgb} = \frac{1}{B}\sum_{i=1}^{B} ||I^i - I^i_{rec}||_2, \tag{4.2}$$

$$\mathcal{L}_{depth} = \frac{1}{B}\sum_{i=1}^{B} ||D^i - D^i_{rec}||_2, \tag{4.3}$$

where B is the batch size. Here, $\mathcal{L}_{rgb}$ and $\mathcal{L}_{depth}$ represent mean squared error losses, that depend on the input image and its depth. An illustration of our overall pipeline is depicted in Figure 4.2. In addition to the reconstruction losses, we apply a L1 normalization loss for the density values the radiance fields, $\mathcal{L}_{norm} = \sum_{i=1}^{B} ||\Delta^i||_1$, where Δ^i is the set of density values that calculated for all the queried 3D points for the image I^i.

4.3.3 Preventing Forgetting

The benefit of our approach is that we can apply it to any self-supervised network that has already been pre-trained using 2D objectives. However, our 3D reconstruction objective might inadvertently bias the model towards the 3D task and force it to 'forget' the useful representation that it has already encoded. To prevent this, we add a knowledge distillation loss (Hinton et al., 2015),

$$\mathcal{L}_{dist} = \frac{1}{B}\sum_{i=1}^{B} ||f(I^i) - f_{teach}(I^i)||_2. \tag{4.4}$$

Here, $f(.)$ is the representation function that we are optimizing and f_{teach} is a frozen backbone that is already trained using a 2D self-supervised objective.

Our final overall training objective consists of a combination of four losses

$$\mathcal{L} = \lambda_{rgb}\mathcal{L}_{rgb} + \lambda_{depth}\mathcal{L}_{depth} + \lambda_{dist}\mathcal{L}_{dist} + \lambda_{norm}\mathcal{L}_{norm}, \tag{4.5}$$

where the λ values are weights for each of the respective loss terms.

4.3.4 Implementation Details

Backbone and Feature Extraction. We build our approach on standard Vision Transformers (ViTs) (Kolesnikov et al., 2021), specifically using the DINOv2 pre-trained variants (Oquab et al., 2024). In all experiments, we explore ViT-S/14, ViT-B/14, and ViT-L/14 backbones, which produce 384-, 768-, and 1024-dimensional features, respectively. From each image, we extract class and patch tokens from the last four

layers and concatenate them to obtain a global image representation. Additionally, we concatenate the globally averaged patch features with the CLS token and project the result to 1024 dimensions using a 1×1 convolution. This representation is used as input to the decoder.

Decoder and Triplane Representation. To generate triplane features, we employ a decoder Φ consisting of cross-attention blocks and 2D upsampling convolutional layers, following the design of (Shen et al., 2023). The decoder takes global features $h \in \mathbb{R}^{1024}$ and learnable triplane embeddings $\xi \in \mathbb{R}^{16\times16\times1024}$, and produces triplane features of size $256 \times 256 \times 32 \times 3$. We use these features to reconstruct images and depth maps via volume rendering (Barron et al., 2021), which outputs reconstructions of size $256 \times 256 \times 3$ (RGB) and 256×256 (depth). For each pixel, we sample a ray and evaluate 16 points along the ray to query the triplane representation. Using bilinear sampling, we obtain features at each 3D point, which are passed through a two-layer MLP to predict radiance (RGB and occupancy) for final rendering. We adopt importance sampling as in prior work (Skorokhodov et al., 2023; Barron et al., 2021).

Training Setup. Our model is trained end-to-end, optimizing the backbone, decoder, and triplane embeddings jointly. We use the Adam optimizer (Kingma and Ba, 2014) with a fixed learning rate of $1e^{-4}$ for 10 epochs. The loss weights are set as $\lambda_{rgb} = 0.1$, $\lambda_{depth} = 1$, $\lambda_{dist} = 1$, and $\lambda_{norm} = 1e^{-3}$ for all experiments. We use pseudo depth maps generated by ZoeDepth (Bhat et al., 2023) with the DPT backbone as supervision for training on ImageNet-1k (Russakovsky et al., 2015), which contains around 1.2 million images across 1,000 classes. These depth maps are generated automatically and provide purely geometric supervision, without any semantic cues. Training is performed on 64 GPUs with a batch size of 12 per GPU. Each training run takes approximately 10 hours for 10 epochs. We apply basic data augmentations, including resizing images to 256×256, random cropping to 224×224, and horizontal flipping with 0.5 probability.

Ablation Without Triplane. To assess the benefit of the explicit triplane representation, we include a baseline where the decoder directly upsamples global image features using 2D convolutional layers, without volume rendering. This model is trained with the same reconstruction and distillation objectives, allowing a fair comparison against our full triplane-based method.

4.4 Experiments

The main goal of our proposed method is to enhance the robustness of existing representation learning methods. We first show how our method results in improved performance on several robustness benchmarks such as ImageNet-Rendition (Im-R) (Hendrycks et al., 2021), ImageNet-Sketch (Im-Sketch) (Wang et al., 2019), and Photorealistic Unreal Graphics (PUG) (Bordes et al., 2023).

We also perform experiments on conventional tasks like image recognition (Russakovsky et al., 2015), fine-grained image classification (Van Horn et al., 2021), and depth estimation (Silberman et al., 2012). This is to illustrate that our approach does not decrease performance for other tasks at the expense of improving robustness.

After training the network with the proxy 3D task, we discard the 3D estimation components of the network and use the backbone representation function to extract global image representation from images $h = f(I)$. For evaluation, we train per-task decoder networks using a fixed representation function $y = \psi(h)$, where the form of y and $\psi(h)$ depends on the specific downstream task. In each experimental section, we provide details about the decoding function and training details. In all of our experiments, the representation functions are frozen, unless stated otherwise.

4.4.1 Robustness

Datasets. We present experimental results on benchmarks that are designed to test the robustness of methods in the face of various appearance related shifts. ImageNet-Rendition (Im-R) (Hendrycks et al., 2021) contains 30,000 images of art, cartoon, graffiti, *etc*. from 200 ImageNet classes. ImageNet-Sketch (Im-Sketch) (Wang et al., 2019) contains 50 sketch images for each of the 1000 original ImageNet classes. These two datasets contain examples where the texture of the objects is significantly different compared to real in-the-wild photographs, which leads to a significant drop in performance for previous SSL methods (Oquab et al., 2024).

Photorealistic Unreal Graphics (PUG) (Bordes et al., 2023) is a dataset that is designed to evaluate the robustness of visual recognition models. It contains synthetically generated examples from 3D assets by controlling for factors like object texture, background, lighting *etc*.

It has been demonstrated that state-of-the-art visual recognition models obtain inadequate performance on this dataset due to changes in appearance factors like object texture and size (Bordes et al., 2023).

We report performance on these two main factors in our experiments. The 3D Common Corruptions (ImageNet-3DCC) (Kar et al., 2022) dataset was created with synthetic corruptions with varying levels of difficulty using ImageNet validation images. Compared to previous datasets, it was constructed with synthetic augmentations, but it contains real images and realistic corruptions such as low lighting, flash, and motion blur which reflect real-world challenges for visual recognition models.

Protocol. All of the experiments that we present here are designed to measure the robustness of classifiers that are trained on ImageNet (Russakovsky et al., 2015) classification data. Given this, we first train a linear classifier on top of various frozen backbones from DINOv2 (Oquab et al., 2024) that are either enhanced via our method (denoted as '+ 3D-Prior') or not, using 1k ImageNet classes from the original training set. We then test the respective linear classifiers on various robustness datasets.

Results. In Table 4.1, we observe that our proposed method ('3D-Prior') improves the robustness of SSL methods on all robustness benchmarks tested, irrespective of architecture type. For instance, we improve the performance for the different backbone architectures on both ImageNet-Rendition and ImageNet-Sketch datasets, which contain highly challenging out-of-distribution examples. In particular, the performance of DINOv2 (Oquab et al., 2024) using the ViTB/14 architecture is improved by 2% on both benchmarks. Moreover, although the objects in the Im-Sketches images are not truly 3D, models incorporating a 3D prior demonstrate greater shape reasoning, exhibit increased shape bias, and achieve more accurate predictions. Furthermore, for the PUG benchmark, our method improved the performance of the models for object size and texture variation in all cases.

We also present results on the ImageNet-3DCC dataset for various synthetic corruption types in Table 4.2. For each level, there are 5 different corruption levels, for simplicity, we report the averaged top-1 accuracy for each corruption type. We observe slight improvement for corruption types like far focus, xy motion blur, and z motion blurs. However, for other factors like low light, iso noise, we achieve performance that is comparable to the baselines.

We also present qualitative results in Figure 4.3. We illustrate some top-5 predictions from linear classifiers that were trained on top of representations from DINOv2, either with or without our method. For instance, the top left example is misclassified as a 'starfish' by the DINOv2-based classifier due to the color of the input image while our *shape-aware* approach correctly identifies the images as containing a 'goldfish' due to improved shape-bias.

Method	Im-R	Im-Sketch	PUG-Texture	PUG-Size
ViT-S/14	53.7	41.2	20.7	26.8
ViT-S/14 + 3D-Prior	**54.6**	**41.8**	**21.2**	**26.9**
ViT-B/14	63.3	50.6	25.3	32.2
ViT-B/14 + 3D-Prior	**65.9**	**52.4**	**26.2**	**33.4**
ViT-L/14	74.4	59.3	34.5	42.7
ViT-L/14 + 3D-Prior	**75.9**	**59.5**	**36.4**	**43.2**

Table 4.1. Robustness evaluation using frozen backbone features from DINOv2 (Oquab et al., 2024) and their enhanced versions from our method ('+ 3D-Prior'). Here we report top-1 accuracy for all benchmarks. Irrespective of backbone architecture type, our 3D-Prior method improves performance across all datasets. For the PUG experiments, we re-run the DINOv2 baselines with our evaluation setting.

Method	color quant	far focus	flash	fog 3d	iso noise
ViT-B/14	72.5	71.6	**60.8**	62.5	**63.9**
ViT-B/14 + 3D-Prior	**72.6**	**72.0**	60.7	**62.7**	63.3
Method	low light	near focus	xy motion blur	z motion blur	
ViT-B/14	72.3	75.5	58.5	58.3	
ViT-B/14 + 3D-Prior	72.3	**75.8**	**59.0**	**58.7**	

Table 4.2. Robustness evaluation using frozen backbone features from DINOv2 (Oquab et al., 2024) and their enhanced versions from our method on the ImageNet-3DCC dataset (Kar et al., 2022) using a ViT-B/14-based architecture. While our method improves robustness for corruptions such as 'motion blur' and 'far focus', there are cases such as 'flash' where we are slightly worse.

Figure 4.3. Here we compare top-5 predictions from linear classifiers that are trained on original DINOv2 (Oquab et al., 2024) backbone features (shown in red) and our 3D enhanced approach (shown in blue) on various challenging examples from ImageNet-Rendition (Hendrycks et al., 2021) and ImageNet-Sketch (Wang et al., 2019). Our method results in more shape information being encoded in the representation and, hence, leads to classifiers that are more robust for these challenging out-of-distribution examples.

4.4.2 Downstream Tasks

Tasks and datasets. We present results on additional downstream tasks to show that our method does not lead to worse performance for other tasks at the expense of improved robustness. We report results for visual recognition on ImageNet (Russakovsky et al., 2015), fine-grained classification using the iNaturalist 2021 (Van Horn et al., 2021), and depth estimation on NYU-DepthV2 (Silberman et al., 2012).
Protocol. We follow the same evaluation protocol as in DINOv2 (Oquab et al., 2024). For ImageNet and iNat21 experiments we froze the backbone, and trained a single-layer classifier using the respective training sets and reported the top-1 validation accuracy. For depth estimation on NYU-DepthV2, we trained two different decoders on top of frozen backbone features, a single linear layer and a more complex DPT (Ranftl et al., 2022) decoder, and followed the same training recipe from DINOv2 (Oquab et al., 2024). We also compare to the non-3D baseline numbers from (Oquab et al., 2024).
Results. Results are presented in Table 4.3. For the visual recognition task on ImageNet-1k, we observe that linear classification performance is slightly improved for all the backbone architectures evaluated, and the performance of the models on fine-grained classification for iNat21 is maintained. Furthermore, the performance on depth estimation is improved compared to the baselines, especially when we use a high-resolution DPT decoder on top of our learned representation.

4.4.3 Shape Bias

Similar to humans, we want our visual recognition models to pay more attention to shape cues compared to texture. As our proxy 3D task requires learning more shape-oriented representations, our hypothesis is that it should lead to representations that have more shape bias. We use the same experimental protocol and dataset from (Geirhos et al., 2019) to measure the shape bias of different models. The dataset contains various synthetically generated examples, where the shape of the object comes from one class and the texture of the object comes from another.

We measure the shape bias of representations from DINOv2 (Oquab et al., 2024) before and after it is trained with our proxy 3D objective. The results are visualized in Figure 4.4. We observe that our method improves the shape bias of the original representations, which is the objective of our shape-centric 3D reconstruction task. Qualitative examples, where we compare predictions of models with and without our

Method	ImageNet-1k	iNat21	NYU-DepthV2 ↓	
			linear	DPT
ViT-S/14	81.1	**74.2**	0.499	0.356
ViT-S/14 + 3D-Prior	**81.4**	73.6	**0.438**	**0.346**
ViT-B/14	84.5	81.1	0.399	0.317
ViT-B/14 + 3D-Prior	**85.1**	**82.0**	**0.398**	**0.300**
ViT-L/14	86.3	85.1	**0.384**	0.293
ViT-L/14 + 3D-Prior	**86.5**	**85.2**	0.389	**0.286**

Table 4.3. Downstream task evaluation using frozen backbone features on various tasks using DINOv2 (Oquab et al., 2024) with and without our 3D-Prior method. We report top-1 accuracy for the ImageNet-1k (Russakovsky et al., 2015) and iNat21 (Van Horn et al., 2021) datasets (higher is better), and RMSE for NYU-DepthV2 (Silberman et al., 2012) dataset (lower is better). Our method leads to improvements in visual recognition performance on ImageNet and for depth estimation on NYU-DepthV2, and does not negatively impact performance on the fine-grained iNat21 dataset.

method, can be seen in Figure 4.1.

These results, combined with the robustness experiments, show that the hypothesis of improving shape bias to obtain more robust representations is valid. Furthermore, these results may further encourage future lines of work in SSL to develop methods that are designed to explicitly consider 3D representations during training.

4.4.4 Ablations

To quantify the importance of individual components of our model, we present ablation experiments on the robustness tasks in Table 4.4.

Removing the triplane. First, we investigate if using a 3D representation in the form of a triplane with volume rendering is necessary or if training a basic depth and image decoder network on top of representations is sufficient. Here, we added a decoder which consists of multiple upsampling and convolution layers to predict depth and images. We observe a drop in performance on all benchmarks, but with a smaller drop on ImageNet. This experiment indicates that using an explicit 3D representation is crucial to improve the robustness of the learned representations. Using a 3D representation

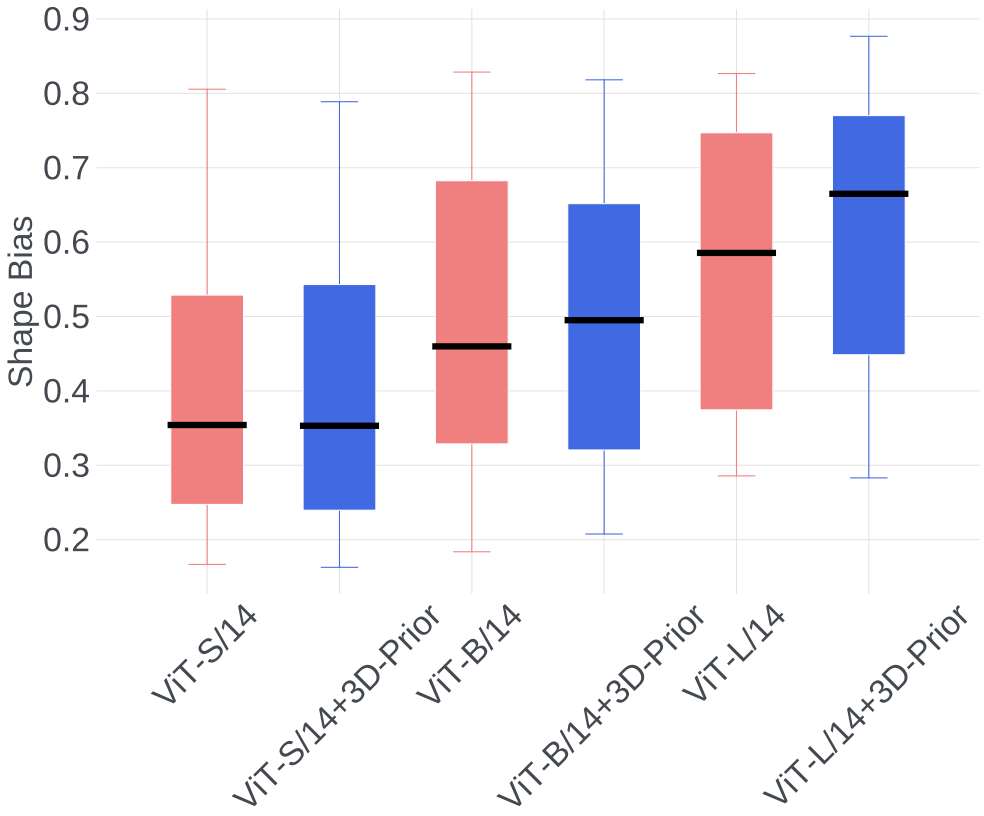


Figure 4.4. Quantification of the shape bias of different DINOv2 (Oquab et al., 2024) representations with and without our 3D-Prior method. We calculate the shape bias using the data and protocol from (Geirhos et al., 2019). Our approach increases the shape bias of visual recognition models and we observe that with larger backbones, the difference grows.

is essential because it incorporates geometric priors such as camera viewpoint, depth, and occlusion handling.

Removing distillation. Next, we try to understand what happens if we do not employ a distillation loss. Without distillation, the model is free to forget useful representation that are already encoded in the 2D SSL backbone. To test this, we simply trained a separate model without the distillation loss. Removing distillation leads to a significant drop in performance across all benchmarks. This experiment shows that preventing the forgetting of already learned useful representations is essential.

Training from scratch. We also investigate if we can learn a global image representation using only the proxy 3D task. Here, we initialized the backbone network randomly (i.e., they are no longer pre-trained) and trained the network using only the image and depth reconstruction losses. The experimental results show that the learned representation is not meaningful and, by itself, the 3D reconstruction task is not a sufficient way

Method	Im-1k	Im-R	Im-Sketch	PUG-Texture	PUG-Size
DinoV2	84.5	63.3	50.6	25.3	32.2
Ours	85.1	65.9	52.4	26.2	33.4
Ours w/o triplane	84.3	63.1	50.7	24.9	31.1
Ours w/o $\mathcal{L}_{dist}$	70.7	34.2	22.9	12.1	16.4
Ours from scratch	14.4	8.5	7.4	0.2	0.1

Table 4.4. Ablation experiments on various robustness benchmarks using a DinoV2 ViT-B/14 model, where these benchmarks measure out-of-distribution (OOD) robustness across scenarios such as sketches, renditions, texture variations, and object size changes. We investigate the importance of using an explicit 3D representation during training (i.e., w/o triplane), disabling our distillation loss (i.e., w/o $\mathcal{L}_{dist}$), and we evaluate if we can learn reasonable representations when training from scratch without using a pre-trained backbone or distillation loss (i.e., from scratch).

to learn a global image representation.

Amount of data. Finally, we explore the impact of varying the size of the training dataset that is used for the 3D proxy task. (Cole et al., 2022) showed that 2D-based SSL methods benefit from being trained on larger unlabeled datasets, but that there are diminishing returns after a certain amount for the methods they tested. Similarly, we quantify how efficient our method is in terms of the training data size. For this experiment, we randomly selected 100k and 500k images from the ImageNet training set, and trained different instances of our model on these subsets using the same number of iterations as the full model. We report the results in Figure 4.5. Interestingly, compared to 2D self-supervised methods (Cole et al., 2022), the performance of our 3D enhanced models are not significantly impacted by the reduction in training data.

Effects of $\mathcal{L}_{rgb}$ and $\mathcal{L}_{depth}$. In Table 4.5, we show how the reconstruction losses, $\mathcal{L}_{rgb}$ and $\mathcal{L}_{depth}$, impact performance. Removing the rgb loss leads to slightly enhanced performance on the Im-R and Im-Sketch robustness datasets, albeit resulting in marginally lower scores on the PUG dataset. However, omitting the rgb loss causes the model's performance to deteriorate for the visual recognition task on the ImageNet-1K dataset. The model trained without the depth loss performs worse in all cases. Given the ill-posed nature of the single-view 3D reconstruction problem, the absence of depth supervision during training can lead to predicted 3D representations being insufficient,

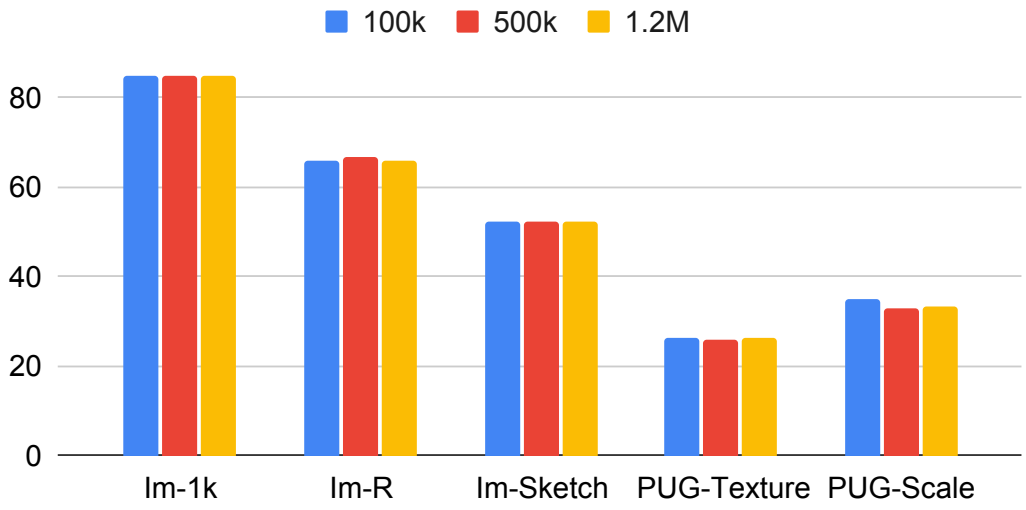


Figure 4.5. We compare the performance of our approach using different amounts of training data from that same source for the 3D proxy task with DinoV2 ViT-B/14 backbones. Surprisingly, we observe that more data does not change the performance drastically, which shows that our method is data efficient.

thereby resulting in sub-optimal representations for the input image.

Different Backbone Architecture. Furthermore, we evaluate whether our method is still effective when coupled with a different backbone architecture (i.e., a ResNet50 (He et al., 2016)) trained with an alternative self-supervised learning objective (i.e., MoCov3 (Chen et al., 2021)). In Table 4.6, we present visual recognition and robustness metrics. We evaluated both the MoCov3 baseline and our method's extensions using identical linear probing hyperparameter space and reported the best. Our approach improved the baseline results, demonstrating its adaptability across architectures and diverse SSL methods.

4.4.5 Qualitative Results

While it is not our main objective, we display reconstruction results in Figure 4.6. Our method is able to reconstruct different kinds of objects. However, the quality of our reconstructions is not on par with the state-of-the-art generative approaches. In this work, our aim is not to obtain high-quality reconstructions, but to learn 3D-aware representations to improve global image representations. Moreover, we want to avail of an efficient and scalable 3D representation to train our network with a large-scale dataset and while design choices can result in slightly lower quality reconstructions

Method	Im-1k	Im-R	Im-Sketch	PUG-Texture	PUG-Size
DinoV2	84.5	63.3	50.6	25.3	32.2
Ours	85.1	65.9	52.4	26.2	33.4
Ours w/o $\mathcal{L}_{rgb}$	84.7	66.5	52.3	25.8	33.4
Ours w/o $\mathcal{L}_{depth}$	84.5	63.8	50.3	25.4	32.3

Table 4.5. Ablation experiments on various robustness benchmarks using DinoV2 ViT-B/14. We investigate the impact of removing different reconstruction losses (i.e., w/o $\mathcal{L}_{rgb}$ and w/o $\mathcal{L}_{depth}$).

Method	Im-1k	Im-R	Im-Sketch	PUG-Texture	PUG-Size
MoCov3 R50	70.7	36.2	**24.4**	10.3	12.6
MoCov3 R50 + 3D Prior	70.7	**36.3**	23.9	**10.6**	**12.8**

Table 4.6. We test if our method is able to improve the representation obtained from MoCov3 (Chen et al., 2021) objective using a different architecture (i.e., ResNet50 (He et al., 2016)).

the learned image representations have been demonstrated to improve the robustness of the backbone networks which was our core objective.

4.5 Discussion and Limitations

One of the limitations of our approach is its reliance on pseudo depth maps for each input image during training. To obtain these, we use existing pre-trained monocular depth estimation models, such as those introduced by (Ranftl et al., 2022) and (Bhat et al., 2023). These models provide only geometric supervision in the form of depth and do not contribute any semantic or category-specific information. Despite this, pseudo depth maps can be generated automatically and at minimal cost, making them practical for large-scale 2D image collections. In addition, these monocular depth models have been shown to be robust and generalizable across a wide range of scenes and object categories. This makes them a suitable choice for our framework, which is designed to scale across diverse and unconstrained visual data.

Another important consideration is the design and integration of our 3D prior, which is represented as a triplane feature field. While the triplane offers a compact

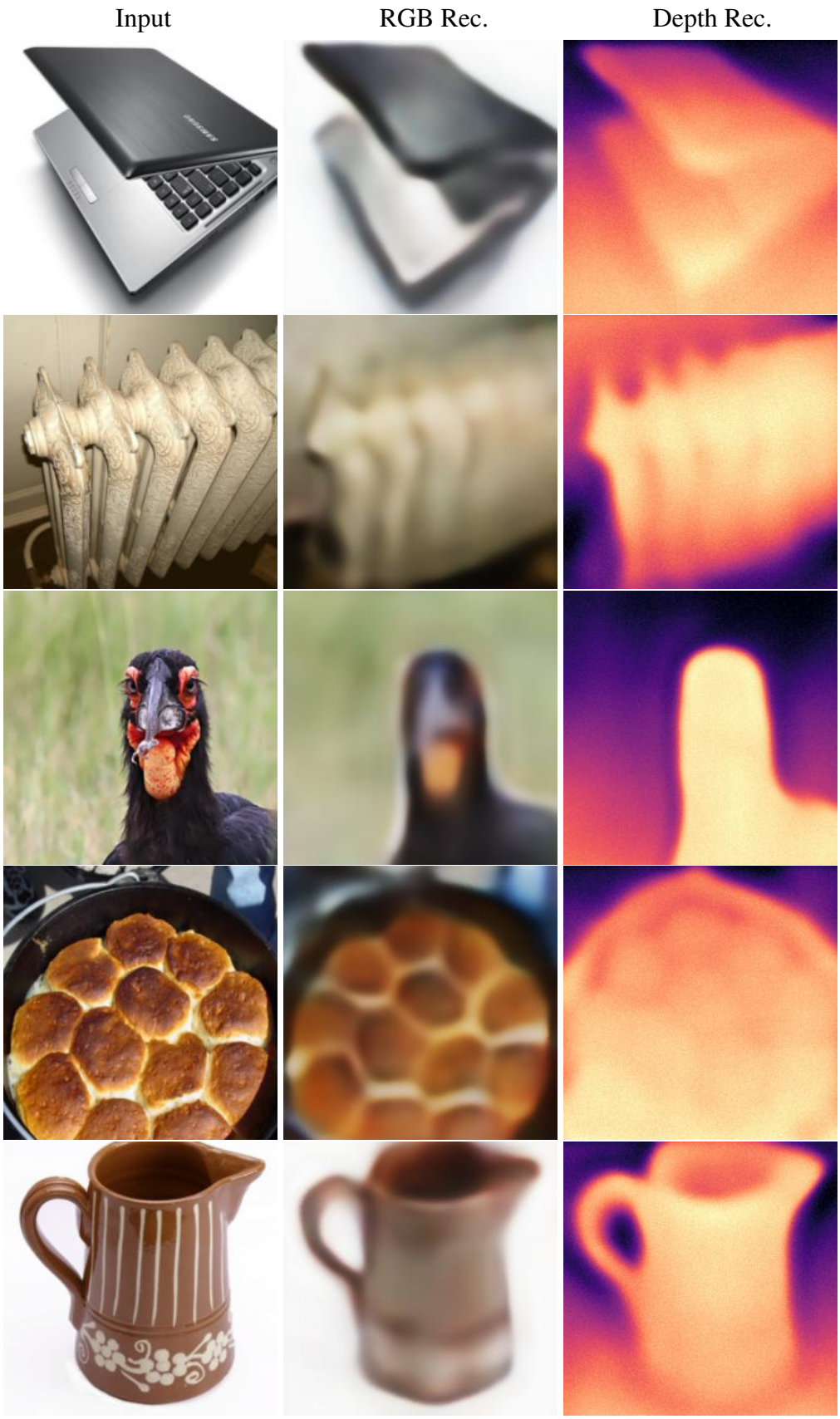


Figure 4.6. Visualization of 3D reconstruction by our model for a subset of ImageNet validation images.

Input(s)	Operation	Output
$h \in \mathcal{R}^{1024}$ $\xi \in \mathcal{R}^{16\times16\times1024}$	Cross Attention	$\varphi' \in \mathcal{R}^{16\times16\times1024}$
$\varphi' \in \mathcal{R}^{16\times16\times1024}$	Up Sample + 2D Convolution	$\mathcal{R}^{32\times32\times512}$
$\mathcal{R}^{64\times64\times256}$	Up Sample + 2D Convolution	$\mathcal{R}^{128\times128\times256}$
$\mathcal{R}^{128\times128\times128}$	Up Sample + 2D Convolution	$\mathcal{R}^{256\times256\times128}$
$\mathcal{R}^{256\times256\times128}$	1D Convolution + Reshaping	$\varphi \in \mathcal{R}^{256\times256\times32\times3}$

Table 4.7. Triplane decoder architecture (Φ). The decoder takes learnable embeddings and the global image feature as input and calculates cross-attention between them and upsamples the output with multiple blocks to obtain the final axis-aligned triplane features.

way to encode spatial structure and has proven useful in 3D-aware image synthesis, it comes with inherent limitations. Most notably, it struggles to simultaneously represent both fine-grained object geometry and large-scale scene layouts—a challenge that becomes especially pronounced in datasets with high variability, such as those resembling ImageNet, where images may depict either isolated objects or complex, cluttered environments.

In our framework, the 3D prior is introduced only after the main 2D representation has been trained. As a result, it plays a secondary role, functioning primarily as a refinement mechanism rather than a central component of the learning process. Its integration leads to improved robustness as it increases shape bias in the learned features. This inductive bias is more closely aligned with human visual perception, which prioritizes shape over texture. However, because the 3D prior is incorporated at a later stage and is subject only to fine-tuning, its ability to substantially alter the overall representation is inherently limited. Therefore, while the improvements it brings are meaningful, they remain relatively modest in scope.

4.6 Conclusion

We presented a new approach to enhance the robustness of visual representations from 2D self-supervised networks. Our method utilizes a conceptually simple single-view 3D reconstruction task to encourage learning more shape-aware 3D-centric represen-

tations. One of the distinct advantages of our approach is that it can be applied to unordered single image collections as it does not impose any stringent assumptions on the types of images it is trained on. We show that incorporating shape-aware knowledge into the representation learning process enhances robustness when compared to alternatives that are not shape aware across a range of visual understanding benchmarks. We hope that our results will encourage a new line of self-supervised works that are designed to consider 3D representations during training.

Chapter 5

Conclusion

5.1 Summary

This thesis explores key advancements in semantic correspondence estimation, 3D shape reconstruction, and self-supervised learning, introducing novel methodologies that enhance the robustness and effectiveness of visual representation learning.

First, we investigated unsupervised semantic correspondence estimation, evaluating state-of-the-art methods under a standardized protocol. Our proposed diagnostic framework and performance metric provide deeper insights into failure cases, informing the development of improved approaches. We then introduced a novel semantic correspondence method that leverages pre-trained features and optimizes for superior matches, achieving performance that surpasses existing techniques at the time of writing.

Second, we presented SAOR, a novel approach for estimating the 3D shape, texture, and viewpoint of articulated objects from a single image. Unlike traditional methods that rely on rigid 3D priors, SAOR learns to articulate shapes in a skeleton-free manner while maintaining cross-instance consistency. Its novel components make SAOR the first single-view, multi-category articulated object reconstruction method to demonstrate improved qualitative and quantitative results on challenging real-world datasets.

Finally, we addressed the limitations of conventional self-supervised learning, which predominantly relies on monocular 2D data. Inspired by the robustness of human visual perception, we proposed a method that explicitly integrates a strong 3D structural prior into self-supervised training. Our experiments demonstrate that this approach yields more robust visual representations across multiple datasets.

Together, these contributions advance the fields of semantic matching, 3D reconstruction, and self-supervised learning by introducing innovative methodologies that bridge critical gaps in existing approaches. We hope that our findings will inspire further research toward more robust and generalizable visual learning systems.

5.2 Key Takeaways

5.2.1 The Need for Improved Metrics and Better Datasets

In Chapters 2 and 3, we explored the challenges of semantic correspondence and 3D reconstruction, respectively. For both problems, the Percentage of Key Points (PCK) metric has traditionally been the standard for evaluating model performance within the research community in recent years. However, our investigation revealed significant shortcomings in the PCK metric, highlighting its limitations as a reliable indicator of semantic correspondence capabilities. Additionally, a recent study (Mariotti et al., 2024) has further exposed flaws in this metric and proposed several enhancements to address these issues. While the community continues to make substantial efforts to develop improved models and algorithms for a wide range of tasks, comparatively little attention has been directed toward refining the evaluation metrics themselves. The reliance on flawed metrics can mislead the interpretation of results and hinder the identification of genuinely effective solutions.

A similar challenge exists in the selection and utilization of datasets for benchmarking. Although datasets play a crucial role in evaluating model performance, many commonly used d atasets suffer from biases and limited diversity. For instance, the most commonly used dataset in semantic correspondence, SPair-71K (Min et al., 2019b), only contains sparsely annotated yet semantically distinct keypoints, such as the left eye of a bird, the right ankle of a cat, or the corner of a TV monitor. These keypoints are derived from object landmarks that are intentionally designed to be both detectable and salient. A more effective approach would involve incorporating additional annotations for object parts that may not be as easily identifiable but still exhibit meaningful semantic correspondences. This would provide a richer and more comprehensive understanding of object relationships, facilitating the development of models with improved generalization and robustness.

We believe the community would benefit from more dedicated efforts toward establishing robust benchmarks and improving dataset representativeness. In this thesis,

we introduced a novel evaluation framework in Chapter 2 designed to address some of these challenges for the problem of semantic correspondence. However, despite its potential, this framework has seen limited adoption, as most existing work continues to focus on improving performance metrics within established benchmarks.

Promoting the creation of diverse and challenging datasets can complement the development of enhanced metrics, resulting in more meaningful and reliable evaluations of model performance. By addressing both the flaws in evaluation metrics and the limitations of datasets, the research community can foster more accurate and transparent progress in the fields of semantic correspondence, 3D reconstruction, and beyond. Encouraging discussions around metric improvement and dataset development will ultimately lead to a stronger foundation for future advancements in computer vision and related areas.

5.2.2 The Importance of Data-Driven Approaches and Reducing Reliance on Explicit Priors

In recent years, significant advancements in computer vision have been primarily driven by the increasing scale of datasets and models. While scaling data is relatively straightforward for tasks such as image classification and segmentation, it presents substantial challenges for 3D-related tasks. Unlike 2D data, which can often be easily collected from the web, acquiring large-scale realistic 3D data is inherently difficult and resource-intensive.

To address these challenges, many previous studies have relied on 2D image collections to alleviate the need for extensive 3D data. These approaches typically involve learning priors from 2D data to infer 3D structures. However, they often incorporate explicit assumptions and constraints to model these priors, which considerably limits the generalizability of the resulting systems. Moreover, such assumptions restrict the scalability of models as they impose limitations on the types and diversity of data that can be effectively utilized. For instance, (Wu et al., 2023b) employed a skeleton-based prior that required training on a single image category. This dependence on domain-specific priors hinders the development of more robust and generalizable 3D models capable of leveraging diverse datasets.

In our work, we aim to address these limitations by reducing reliance on explicit priors. Specifically, we eliminate the use of skeleton priors, which enables our approach to be applied to significantly larger and more diverse datasets. Consequently,

we are able to train our models on image collections spanning hundreds of animal categories. This comprehensive training enhances reconstruction performance within a single model, highlighting the advantages of learning more flexible and generalizable 3D priors directly from data. By adopting this data-driven approach, we contribute to advancing the scalability and generalizability of 3D models. Our results emphasize the importance of minimizing restrictive assumptions and underscore the potential of large-scale training to learn richer representations of the 3D world. Our model works without such assumptions thanks to novel technical innovations in the objective functions and advances in model design, which enable effective utilization of large-scale available 2D datasets.

In a similar spirit, a recent study by Shtedritski et al. (2023), which investigates unsupervised semantic correspondence, relaxes the assumption made in Chapter 2 that training image pairs must belong to the same semantic class. By training a novel model on a large-scale dataset without this constraint, the authors achieve significantly improved performance in semantic matching. This further supports the view that relaxing strong assumptions and leveraging large-scale, diverse data can lead to more robust and generalizable models. Moreover, (Oquab et al., 2024), a recent self-supervised network trained without any labels and not specifically trained for semantic correspondence, but on a larger dataset, obtained even better results

As model and dataset scaling continue, it is critical to focus on reducing reliance on overly restrictive assumptions and priors. By embracing more flexible, data-driven approaches, we can foster the development of models that are more generalizable and capable of addressing a wider range of tasks and challenges. We hope our findings inspire further research into novel ways of overcoming the limitations imposed by explicit priors, encouraging collaboration and innovation across disciplines. Ultimately, such efforts will lead to the creation of more robust and scalable systems, expanding the boundaries of what is possible in 3D modeling and beyond.

5.2.3 Rethinking Representation Learning with 3D

There is significant and growing interest within the research community in developing methods to learn 3D representations. However, much of this work has primarily focused on estimating the 3D properties of the world from 2D images (Godard et al., 2017; Wu et al., 2023b; Danier et al., 2025; Wang et al., 2025; Szymanowicz et al., 2024). While this approach has its merits, it differs from how humans perceive and

understand the world. In contrast to relying on 2D images, humans acquire their visual priors and perceptual abilities by directly navigating and interacting with the 3D environment. Through physical experience, humans gain an embodied understanding of spatial relationships, depth, and object properties. This real-world interaction allows us to develop a robust and dynamic perception of the 3D world, which current models struggle to replicate. Most existing vision methods, in contrast, typically learn visual representations from single-view image collections without other supervision (Oquab et al., 2024) or with language supervision (Radford et al., 2021; Rombach et al., 2022). While this approach has been effective in many domains, it limits the depth of understanding related to 3D spatial relationships that these models can develop.

Although these models have demonstrated some capabilities in understanding 3D concepts (El Banani et al., 2024; Danier et al., 2025; Azad et al., 2024; Zhan et al., 2024), they are still far from achieving human-level perception. The ability to estimate or infer the 3D properties of the world from 2D inputs remains limited and lacks the nuanced understanding that comes from direct interaction with the 3D environment. While there have been attempts to extend existing learning algorithms to 3D, such as through the use of multi-view learning or by incorporating additional depth data (Zhang et al., 2024; Yue et al., 2024), these models were not originally designed with 3D reasoning in mind. They often treat 3D data as an afterthought or an additional layer, rather than integrating 3D considerations into their core architecture from the outset. This results in a mismatch between the models' design and the complexity of the 3D world.

We believe that to advance the development of perceptual systems capable of understanding and reasoning about 3D environments, it is essential to design new representation learning methods that are specifically tailored to reason in 3D. Such methods should not only consume 3D data but also reason about it in a way that reflects the complexities and nuances of real-world 3D spaces. This shift in approach will be crucial for building perception systems that can be used by embodied agents, such as robots or autonomous vehicles, which need to navigate and interact with the world in a dynamic, 3D context.

5.3 Limitations and Future Work

In Chapter 2, we explored the interplay between shape and semantics in the context of semantic correspondence estimation, introducing a novel evaluation framework. This

task requires matching sparsely annotated yet semantically distinct keypoints, such as the left eye of a bird, the right ankle of a cat, or the corner of a TV monitor. These keypoints are derived from object landmarks that are intentionally designed to be both detectable and salient.

A natural next step is to extend this framework by incorporating additional annotations for object parts that may not be as easily identifiable but still exhibit meaningful semantic correspondences. Expanding the annotation set in this way could improve the robustness of semantic matching models and provide deeper insights into the relationship between shape and semantics.

Furthermore, with the exception of SPair-71K (Min et al., 2019b), the datasets used in our experiments were not specifically designed for semantic correspondence. The creation of more diverse and large-scale semantic correspondence datasets could enable the development of more robust and generalizable models, facilitate benchmarking under a broader range of conditions, and provide new insights into the fundamental principles governing semantic alignment across different object categories.

In Chapter 3, we introduced SAOR, a novel method for single-view 3D object reconstruction, specifically designed for articulated objects such as animals. Our approach leverages meshes as the underlying 3D representation, benefiting from their structured nature and widespread adoption. However, meshes impose fixed-topology constraints, which limit their ability to capture complex deformations and topological variations. A recent work (Li et al., 2024) utilizes a hybrid SDF-mesh representation to avoid topology constraints and enable detailed representations; however, this choice incurs a heavy computational expense, as it requires extracting a mesh from the SDF as an intermediate step.

Future work could explore alternative 3D representations, such as 3D Gaussians (Kerbl et al., 2023) or Radiant Foam (Govindarajan et al., 2025), both of which are compatible with differentiable rendering and have demonstrated impressive fidelity in neural rendering tasks. Additionally, incorporating generative priors from diffusion models, similar to RealFusion (Melas-Kyriazi et al., 2023), could improve reconstruction quality by leveraging data-driven priors to resolve ambiguities in single-view inputs.

Another promising direction is to leverage large-scale synthetic datasets (Deitke et al., 2023; Hong et al., 2024) or high-quality generated images (Jakab et al., 2024; Kaye et al., 2025) to improve supervision and enhance generalization across diverse object categories. Such datasets could help address key challenges, such as the limited

viewpoint coverage in real-world training data, and improve the robustness of single-view 3D reconstruction methods for articulated objects. Here, using recent multi-view diffusion models like MVDream (Shi et al., 2023) can be used to generate multi-view training data.

In Chapter 4, we investigated the integration of a 3D prior into a self-supervised learning framework to enhance high-level visual understanding. Specifically, we started with a pre-trained self-supervised network and incorporated an implicit triplane representation (Chan et al., 2022) as a 3D prior. This approach, however, has two key limitations. First, since we began with an already trained network, the final learned representation remained too similar to the original one, potentially limiting its capacity for significant improvement. Second, while the triplane representation (Chan et al., 2022) is computationally efficient, it lacks expressiveness and is primarily designed for object-centric representations. In contrast, our goal was to learn representations that capture both objects and scenes. The dataset used in our experiments (e.g., ImageNet (Deng et al., 2009)) contains a mix of objects and scenes, making this limitation even more pronounced. Additionally, we relied solely on generated depth maps as the 3D signal.

Future work could benefit from leveraging multi-view datasets such as Multi-View ImageNet (Yu et al., 2023; Han et al., 2024) and Objaverse-XL (Deitke et al., 2023) to facilitate learning more 3D-aware representations. A more promising direction is to explore self-supervised learning frameworks that incorporate structured 3D representations as mid-level features. Recent research, such as Gaussian MAE (Rajasegaran et al., 2025), has demonstrated the potential of this approach by integrating 3D Gaussians (Kerbl et al., 2023) with the Masked Autoencoder (MAE) framework (He et al., 2022) to learn self-supervised representations with a 3D bottleneck. Explicitly enforcing 3D structure during representation learning may lead to more robust and meaningful feature extraction.

Further investigation in this direction could yield significant improvements. In contrast, prior approaches that focus on enhancing the 3D nature of already learned features, such as Zhang et al. (2024) and (Yue et al., 2024), have resulted in only marginal gains. This suggests that simply refining pre-trained representations may not be sufficient, and a more fundamental shift toward learning 3D-aware representations from the outset could be a more effective strategy. Future work could explore different architectures, loss functions, and training paradigms that leverage structured 3D priors to further improve self-supervised learning. For instance, a recent framework (Wang

et al., 2024), which enables the estimation of various 3D parameters directly from images, might be a promising candidate to explore, as it can reason about 3D while using an encoder model that is commonly employed and easy to adopt for standard 2D visual understanding tasks.

Another promising research avenue is to investigate how and when 3D-related information emerges in neural networks trained on 2D data. Recent work has provided intriguing hints in this regard. Danier et al. (2025) examined the emergence of monocular depth cues across different architectures, while Zhan et al. (2024); Sarkar et al. (2024) showed that generative diffusion models encode aspects of physical scene properties despite being trained without explicit 3D supervision. These findings suggest that self-supervised networks, generative models, and multi-modal vision-language models implicitly capture some degree of 3D structure, yet the mechanisms underlying this phenomenon remain poorly understood.

A systematic investigation into the origins of this implicit 3D understanding could yield valuable insights. Key factors such as the statistical properties of training data, architectural biases, and optimization dynamics likely shape these emergent representations. Understanding these mechanisms could inform the design of new learning frameworks that explicitly harness and refine such 3D priors. Beyond theoretical insights, these efforts have direct implications for the development of future embodied and physical AI systems. By better encoding 3D structure from images, models could gain stronger spatial reasoning abilities, enabling more robust perception and interaction in real-world settings. Such advances are critical for applications in robotic manipulation, autonomous navigation, and human-AI collaboration, where reasoning about 3D environments is fundamental.

Bibliography

Alwassel, H., Heilbron, F. C., Escorcia, V., and Ghanem, B. (2018). Diagnosing error in temporal action detectors. In *ECCV*.

Amir, S., Gandelsman, Y., Bagon, S., and Dekel, T. (2022). Deep vit features as dense visual descriptors. *ECCVW What is Motion For?*

Anguelov, D., Srinivasan, P., Koller, D., Thrun, S., Rodgers, J., and Davis, J. (2005). Scape: shape completion and animation of people. In *SIGGRAPH*.

Apple (2024). About face id advanced technology. Accessed: 2025-02-24.

Araslanov, N., Schaub-Meyer, S., and Roth, S. (2021). Dense unsupervised learning for video segmentation. *NeurIPS*.

Aygün, M. and Mac Aodha, O. (2024). SAOR: Single-View Articulated Object Reconstruction. In *CVPR*.

Azad, S., Jain, Y., Garg, R., Rawat, Y. S., and Vineet, V. (2024). Geometer: Probing depth and height perception of large visual-language models. *arXiv:2408.11748*.

Banik, P., Li, L., and Dong, X. (2021). A novel dataset for keypoint detection of quadruped animals from images. *arXiv:2108.13958*.

Bao, H., Dong, L., Piao, S., and Wei, F. (2022). Beit: Bert pre-training of image transformers. In *ICLR*.

Barron, J. T., Mildenhall, B., Tancik, M., Hedman, P., Martin-Brualla, R., and Srinivasan, P. P. (2021). Mip-nerf: A multiscale representation for anti-aliasing neural radiance fields. In *ICCV*.

Beery, S., Morris, D., and Yang, S. (2019). Efficient pipeline for camera trap image review. In *Data Mining and AI for Conservation Workshop at KDD*.

Beery, S., Van Horn, G., and Perona, P. (2018). Recognition in terra incognita. In *ECCV*.

Bhat, S. F., Birkl, R., Wofk, D., Wonka, P., and Müller, M. (2023). Zoedepth: Zero-shot transfer by combining relative and metric depth. *arXiv:2302.12288*.

Biggs, B., Boyne, O., Charles, J., Fitzgibbon, A., and Cipolla, R. (2020). Who left the dogs out? 3d animal reconstruction with expectation maximization in the loop. In *ECCV*.

Blanz, V. and Vetter, T. (1999). A morphable model for the synthesis of 3d faces. In *SIGGRAPH*.

Bochkovskii, A., Delaunoy, A., Germain, H., Santos, M., Zhou, Y., Richter, S. R., and Koltun, V. (2024). Depth pro: Sharp monocular metric depth in less than a second. *arXiv:2410.02073*.

Bordes, F., Shekhar, S., Ibrahim, M., Bouchacourt, D., Vincent, P., and Morcos, A. S. (2023). Pug: Photorealistic and semantically controllable synthetic data for representation learning. *arXiv:2308.03977*.

Brendel, W. and Bethge, M. (2019). Approximating cnns with bag-of-local-features models works surprisingly well on imagenet. In *ICLR*.

Bristow, H., Valmadre, J., and Lucey, S. (2015). Dense semantic correspondence where every pixel is a classifier. In *ICCV*.

Bruce, V., Georgeson, M. A., and Green, P. R. (2014). *Visual perception: Physiology, psychology and ecology*. Psychology Press.

Caron, M., Touvron, H., Misra, I., Jégou, H., Mairal, J., Bojanowski, P., and Joulin, A. (2021). Emerging properties in self-supervised vision transformers. In *ICCV*.

Cashman, T. J. and Fitzgibbon, A. (2012). What shape are dolphins? building 3d morphable models from 2d images. *PAMI*.

Chan, E. R., Lin, C. Z., Chan, M. A., Nagano, K., Pan, B., De Mello, S., Gallo, O., Guibas, L. J., Tremblay, J., Khamis, S., et al. (2022). Efficient geometry-aware 3d generative adversarial networks. In *CVPR*.

Chen, M., Radford, A., Child, R., Wu, J., Jun, H., Luan, D., and Sutskever, I. (2020a). Generative pretraining from pixels. In *ICML*.

Chen, T., Kornblith, S., Norouzi, M., and Hinton, G. (2020b). A simple framework for contrastive learning of visual representations. In *ICML*.

Chen, X., Fan, H., Girshick, R., and He, K. (2020c). Improved baselines with momentum contrastive learning. *arXiv:2003.04297*.

Chen, X., Xie, S., and He, K. (2021). An empirical study of training self-supervised vision transformers. In *ICCV*.

Cheng, Z., Su, J.-C., and Maji, S. (2021). On equivariant and invariant learning of object landmark representations. In *ICCV*.

Cho, S., Hong, S., Jeon, S., Lee, Y., Sohn, K., and Kim, S. (2021). Cats: Cost aggregation transformers for visual correspondence. *NeurIPS*.

Choe, J., Oh, S. J., Lee, S., Chun, S., Akata, Z., and Shim, H. (2020). Evaluating weakly supervised object localization methods right. In *CVPR*.

Choy, C. B., Gwak, J., Savarese, S., and Chandraker, M. (2016a). Universal correspondence network. *NeurIPS*.

Choy, C. B., Xu, D., Gwak, J., Chen, K., and Savarese, S. (2016b). 3d-r2n2: A unified approach for single and multi-view 3d object reconstruction. In *ECCV*.

Cole, E., Yang, X., Wilber, K., Mac Aodha, O., and Belongie, S. (2022). When does contrastive visual representation learning work? In *CVPR*.

Dalal, N. and Triggs, B. (2005). Histograms of oriented gradients for human detection. In *CVPR*.

Danier, D., Aygün, M., Li, C., Bilen, H., and Mac Aodha, O. (2025). Depthcues: Evaluating monocular depth perception in large vision models. In *CVPR*.

David, M. (2016). The correspondence theory of truth. *The Oxford Handbook of Truth*.

Dehghani, M., Djolonga, J., Mustafa, B., Padlewski, P., Heek, J., Gilmer, J., Steiner, A. P., Caron, M., Geirhos, R., Alabdulmohsin, I., et al. (2023). Scaling vision transformers to 22 billion parameters. In *ICML*.

Deitke, M., Liu, R., Wallingford, M., Ngo, H., Michel, O., Kusupati, A., Fan, A., Laforte, C., Voleti, V., Gadre, S. Y., et al. (2023). Objaverse-xl: A universe of 10m+ 3d objects. *NeurIPS*.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. (2009). Imagenet: A large-scale hierarchical image database. In *CVPR*.

Desbrun, M., Meyer, M., Schröder, P., and Barr, A. H. (1999). Implicit fairing of irregular meshes using diffusion and curvature flow. In *SIGGRAPH*.

Doersch, C., Gupta, A., and Efros, A. A. (2015). Unsupervised visual representation learning by context prediction. In *ICCV*.

Dosovitskiy, A., Springenberg, J. T., Riedmiller, M., and Brox, T. (2014). Discriminative unsupervised feature learning with convolutional neural networks. *NeurIPS*.

El Banani, M., Raj, A., Maninis, K.-K., Kar, A., Li, Y., Rubinstein, M., Sun, D., Guibas, L., Johnson, J., and Jampani, V. (2024). Probing the 3d awareness of visual foundation models. In *CVPR*.

Everingham, M., Eslami, S. A., Van Gool, L., Williams, C. K., Winn, J., and Zisserman, A. (2015). The pascal visual object classes challenge: A retrospective. In *IJCV*.

Felzenszwalb, P. F., Girshick, R. B., McAllester, D., and Ramanan, D. (2010). Object detection with discriminatively trained part-based models. *PAMI*.

Fergus, R., Perona, P., and Zisserman, A. (2003). Object class recognition by unsupervised scale-invariant learning. In *CVPR*.

Fischler, M. A. and Elschlager, R. A. (1973). The representation and matching of pictorial structures. *IEEE Transactions on computers*.

Garg, R., Bg, V. K., Carneiro, G., and Reid, I. (2016). Unsupervised cnn for single view depth estimation: Geometry to the rescue. In *ECCV*.

Geirhos, R., Narayanappa, K., Mitzkus, B., Thieringer, T., Bethge, M., Wichmann, F., and Brendel, W. (2021). Partial success in closing the gap between human and machine vision. *NeurIPS*.

Geirhos, R., Rubisch, P., Michaelis, C., Bethge, M., Wichmann, F. A., and Brendel, W. (2019). Imagenet-trained cnns are biased towards texture; increasing shape bias improves accuracy and robustness. In *ICLR*.

Gershkoff-Stowe, L. and Smith, L. B. (2004). Shape and the first hundred nouns. *Child Development*.

Gibson, J. J. (1950). The perception of the visual world.

Gidaris, S., Singh, P., and Komodakis, N. (2018). Unsupervised representation learning by predicting image rotations. In *ICLR*.

Godard, C., Mac Aodha, O., and Brostow, G. J. (2017). Unsupervised monocular depth estimation with left-right consistency. In *CVPR*.

Godard, C., Mac Aodha, O., Firman, M., and Brostow, G. J. (2019). Digging into self-supervised monocular depth estimation. In *ICCV*.

Goel, S., Kanazawa, A., and Malik, J. (2020). Shape and viewpoint without keypoints. In *ECCV*.

Goldblum, M., Souri, H., Ni, R., Shu, M., Prabhu, V., Somepalli, G., Chattopadhyay, P., Ibrahim, M., Bardes, A., Hoffman, J., et al. (2023). Battle of the backbones: A large-scale comparison of pretrained models across computer vision tasks. *NeurIPS*.

Gonzalez-Garcia, A., Modolo, D., and Ferrari, V. (2018). Do semantic parts emerge in convolutional neural networks? *IJCV*.

Goodfellow, I., Shlens, J., and Szegedy, C. (2015). Explaining and harnessing adversarial examples. In *ICLR*.

Govindarajan, S., Rebain, D., Yi, K. M., and Tagliasacchi, A. (2025). Radiant foam: Real-time differentiable ray tracing. *arXiv:2502.01157*.

Grenander, U. (1978). *Lectures in Pattern Theory: Volume 2 Pattern Analysis*.

Grill, J.-B., Strub, F., Altché, F., Tallec, C., Richemond, P., Buchatskaya, E., Doersch, C., Avila Pires, B., Guo, Z., Gheshlaghi Azar, M., et al. (2020). Bootstrap your own latent-a new approach to self-supervised learning. *NeurIPS*.

Gui, J., Chen, T., Cao, Q., Sun, Z., Luo, H., and Tao, D. (2023). A survey of self-supervised learning from multiple perspectives: Algorithms, theory, applications and future trends. *arXiv:2301.05712*.

Güler, R. A., Neverova, N., and Kokkinos, I. (2018). Densepose: Dense human pose estimation in the wild. In *CVPR*.

Hadsell, R., Chopra, S., and LeCun, Y. (2006). Dimensionality reduction by learning an invariant mapping. In *CVPR*.

Ham, B., Cho, M., Schmid, C., and Ponce, J. (2017). Proposal flow: Semantic correspondences from object proposals. *PAMI*.

Han, K., Rezende, R. S., Ham, B., Wong, K.-Y. K., Cho, M., Schmid, C., and Ponce, J. (2017). Scnet: Learning semantic correspondence. In *ICCV*.

Han, X., Wu, Y., Shi, L., Liu, H., Liao, H., Qiu, L., Yuan, W., Gu, X., Dong, Z., and Cui, S. (2024). Mvimgnet2. 0: A larger-scale dataset of multi-view images. In *SIGGRAPH Asia 2024*.

He, K., Chen, X., Xie, S., Li, Y., Dollár, P., and Girshick, R. (2022). Masked autoencoders are scalable vision learners. In *CVPR*.

He, K., Fan, H., Wu, Y., Xie, S., and Girshick, R. (2020). Momentum contrast for unsupervised visual representation learning. In *CVPR*.

He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. In *CVPR*.

Hendrycks, D., Basart, S., Mu, N., Kadavath, S., Wang, F., Dorundo, E., Desai, R., Zhu, T., Parajuli, S., Guo, M., et al. (2021). The many faces of robustness: A critical analysis of out-of-distribution generalization. In *ICCV*.

Henzler, P., Mitra, N. J., and Ritschel, T. (2019). Escaping plato's cave: 3d shape from adversarial rendering. In *ICCV*.

Henzler, P., Reizenstein, J., Labatut, P., Shapovalov, R., Ritschel, T., Vedaldi, A., and Novotny, D. (2021). Unsupervised learning of 3d object categories from videos in the wild. In *CVPR*.

Hinton, G., Vinyals, O., and Dean, J. (2015). Distilling the knowledge in a neural network. *arXiv:1503.02531*.

Hoiem, D., Chodpathumwan, Y., and Dai, Q. (2012). Diagnosing error in object detectors. In *ECCV*.

Hong, Y., Zhang, K., Gu, J., Bi, S., Zhou, Y., Liu, D., Liu, F., Sunkavalli, K., Bui, T., and Tan, H. (2024). Lrm: Large reconstruction model for single image to 3d.

Hu, T., Wang, L., Xu, X., Liu, S., and Jia, J. (2021). Self-supervised 3d mesh reconstruction from single images. In *CVPR*.

Huang, S., Wang, Q., Zhang, S., Yan, S., and He, X. (2019). Dynamic context correspondence network for semantic alignment. In *ICCV*.

iNaturalist (2023). inaturalist. `www.inaturalist.org`, accessed 8 November 2023.

Jacobson, A., Deng, Z., Kavan, L., and Lewis, J. P. (2014). Skinning: Real-time shape deformation. In *SIGGRAPH Courses*.

Jain, A., Tancik, M., and Abbeel, P. (2021). Putting nerf on a diet: Semantically consistent few-shot view synthesis. In *ICCV*.

Jain, A. K. and Li, S. Z. (2011). *Handbook of face recognition*. Springer.

Jakab, T., Gupta, A., Bilen, H., and Vedaldi, A. (2018). Unsupervised learning of object landmarks through conditional image generation. *NeurIPS*.

Jakab, T., Gupta, A., Bilen, H., and Vedaldi, A. (2020). Self-supervised learning of interpretable keypoints from unlabelled videos. In *CVPR*.

Jakab, T., Li, R., Wu, S., Rupprecht, C., and Vedaldi, A. (2024). Farm3d: Learning articulated 3d animals by distilling 2d diffusion. In *3DV*.

Jiang, W., Trulls, E., Hosang, J., Tagliasacchi, A., and Yi, K. M. (2021). Cotr: Correspondence transformer for matching across images. In *ICCV*.

Jing, L. and Tian, Y. (2020). Self-supervised visual feature learning with deep neural networks: A survey. *PAMI*.

Kanazawa, A., Black, M. J., Jacobs, D. W., and Malik, J. (2018a). End-to-end recovery of human shape and pose. In *CVPR*.

Kanazawa, A., Jacobs, D. W., and Chandraker, M. (2016). Warpnet: Weakly supervised matching for single-view reconstruction. In *CVPR*.

Kanazawa, A., Tulsiani, S., Efros, A. A., and Malik, J. (2018b). Learning category-specific mesh reconstruction from image collections. In *ECCV*.

Kar, A., Tulsiani, S., Carreira, J., and Malik, J. (2015). Category-specific object reconstruction from a single image. In *CVPR*.

Kar, O. F., Yeo, T., Atanov, A., and Zamir, A. (2022). 3d common corruptions and data augmentation. In *CVPR*.

Karmali, T., Atrishi, A., Harsha, S. S., Agrawal, S., Jampani, V., and Babu, R. V. (2022). Lead: Self-supervised landmark estimation by aligning distributions of feature similarity. In *WACV*.

Karras, T., Laine, S., Aittala, M., Hellsten, J., Lehtinen, J., and Aila, T. (2020). Analyzing and improving the image quality of stylegan. In *CVPR*.

Kato, H., Ushiku, Y., and Harada, T. (2018). Neural 3d mesh renderer. In *CVPR*.

Kaye, B., Jakab, T., Wu, S., Rupprecht, C., and Vedaldi, A. (2025). Dualpm: Dual posed-canonical point maps for 3d shape and pose reconstruction. In *CVPR*.

Kerbl, B., Kopanas, G., Leimkühler, T., and Drettakis, G. (2023). 3d gaussian splatting for real-time radiance field rendering. *ACM Trans. Graph.*

Khamis, S., Taylor, J., Shotton, J., Keskin, C., Izadi, S., and Fitzgibbon, A. (2015). Learning an efficient model of hand shape variation from depth images. In *CVPR*.

Khosla, A., Jayadevaprakash, N., Yao, B., and Li, F.-F. (2011). Novel dataset for fine-grained image categorization: Stanford dogs. In *CVPR Workshop on Fine-Grained Visual Categorization*.

Kim, J., Liu, C., Sha, F., and Grauman, K. (2013). Deformable spatial pyramid matching for fast dense correspondences. In *CVPR*.

Kim, S., Lin, S., Jeon, S. R., Min, D., and Sohn, K. (2018). Recurrent transformer networks for semantic correspondence. *NeurIPS*.

Kim, S., Min, D., Ham, B., Jeon, S., Lin, S., and Sohn, K. (2017). Fcss: Fully convolutional self-similarity for dense semantic correspondence. In *CVPR*.

Kingma, D. P. and Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv:1412.6980*.

Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., Xiao, T., Whitehead, S., Berg, A. C., Lo, W.-Y., Dollár, P., and Girshick, R. (2023). Segment anything. In *ICCV*.

Koestinger, M., Wohlhart, P., Roth, P. M., and Bischof, H. (2011). Annotated facial landmarks in the wild: A large-scale, real-world database for facial landmark localization. In *ICCV workshops*.

Kokkinos, F. and Kokkinos, I. (2021a). Learning monocular 3d reconstruction of articulated categories from motion. In *CVPR*.

Kokkinos, F. and Kokkinos, I. (2021b). To the point: Correspondence-driven monocular 3d category reconstruction. In *NeurIPS*.

Kolesnikov, A., Dosovitskiy, A., Weissenborn, D., Heigold, G., Uszkoreit, J., Beyer, L., Minderer, M., Dehghani, M., Houlsby, N., Gelly, S., Unterthiner, T., and Zhai, X. (2021). An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*.

Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. *NeurIPS*.

Kulkarni, N., Gupta, A., Fouhey, D., and Tulsiani, S. (2020). Articulation-aware canonical surface mapping. In *CVPR*.

Kulkarni, N., Gupta, A., and Tulsiani, S. (2019a). Canonical surface mapping via geometric cycle consistency. In *ICCV*.

Kulkarni, T. D., Gupta, A., Ionescu, C., Borgeaud, S., Reynolds, M., Zisserman, A., and Mnih, V. (2019b). Unsupervised learning of object keypoints for perception and control. *NeurIPS*.

Kulkarni, T. D., Whitney, W. F., Kohli, P., and Tenenbaum, J. (2015). Deep convolutional inverse graphics network. In *NeurIPS*.

Landau, B., Smith, L. B., and Jones, S. S. (1988). The importance of shape in early lexical learning. *Cognitive development*.

Lao, D., Yang, F., Wang, D., Park, H., Lu, S., Wong, A., and Soatto, S. (2024). On the viability of monocular depth pre-training for semantic segmentation. In *ECCV*.

Lee, J., Kim, D., Ponce, J., and Ham, B. (2019). Sfnet: Learning object-aware semantic correspondence. In *CVPR*.

Lee, J. Y., DeGol, J., Fragoso, V., and Sinha, S. N. (2021). Patchmatch-based neighborhood consensus for semantic correspondence. In *CVPR*.

Lewis, J. P., Cordner, M., and Fong, N. (2000). Pose space deformation: a unified approach to shape interpolation and skeleton-driven deformation. In *SIGGRAPH*.

Li, S., Han, K., Costain, T. W., Howard-Jenkins, H., and Prisacariu, V. (2020a). Correspondence networks with adaptive neighbourhood consensus. In *CVPR*.

Li, X., Liu, S., De Mello, S., Kim, K., Wang, X., Yang, M.-H., and Kautz, J. (2020b). Online adaptation for consistent mesh reconstruction in the wild. In *NeurIPS*.

Li, X., Liu, S., Kim, K., Mello, S. D., Jampani, V., Yang, M.-H., and Kautz, J. (2020c). Self-supervised single-view 3d reconstruction via semantic consistency. In *ECCV*.

Li, Z., Litvak, D., Li, R., Zhang, Y., Jakab, T., Rupprecht, C., Wu, S., Vedaldi, A., and Wu, J. (2024). Learning the 3d fauna of the web.

Liao, Z., Yang, J., Saito, J., Pons-Moll, G., and Zhou, Y. (2022). Skeleton-free pose transfer for stylized 3d characters. In *ECCV*.

Liu, C., Yuen, J., and Torralba, A. (2010). Sift flow: Dense correspondence across scenes and its applications. *PAMI*.

Liu, R., Wu, R., Van Hoorick, B., Tokmakov, P., Zakharov, S., and Vondrick, C. (2023). Zero-1-to-3: Zero-shot one image to 3d object. In *ICCV*.

Liu, S., Li, T., Chen, W., and Li, H. (2019). Soft rasterizer: A differentiable renderer for image-based 3d reasoning. In *ICCV*.

Liu, Y., Zhu, L., Yamada, M., and Yang, Y. (2020). Semantic correspondence as an optimal transport problem. In *CVPR*.

Liu, Z., Luo, P., Wang, X., and Tang, X. (2015). Deep learning face attributes in the wild. In *ICCV*.

Long, J. L., Zhang, N., and Darrell, T. (2014). Do convnets learn correspondence? *NeurIPS*.

Loper, M., Mahmood, N., Romero, J., Pons-Moll, G., and Black, M. J. (2015). Smpl: A skinned multi-person linear model. *TOG*.

Loper, M. M. and Black, M. J. (2014). Opendr: An approximate differentiable renderer. In *ECCV*.

Lowe, D. G. (1999). Object recognition from local scale-invariant features. In *ICCV*.

Mariotti, O., Mac Aodha, O., and Bilen, H. (2024). Improving semantic correspondence with viewpoint-guided spherical maps. In *CVPR*.

Melas-Kyriazi, L., Laina, I., Rupprecht, C., and Vedaldi, A. (2023). Realfusion: 360deg reconstruction of any object from a single image. In *CVPR*.

Mescheder, L., Oechsle, M., Niemeyer, M., Nowozin, S., and Geiger, A. (2019). Occupancy networks: Learning 3d reconstruction in function space. In *CVPR*.

Mikolajczyk, K. and Schmid, C. (2005). A performance evaluation of local descriptors. *PAMI*.

Mildenhall, B., Srinivasan, P. P., Tancik, M., Barron, J. T., Ramamoorthi, R., and Ng, R. (2021). Nerf: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*.

Min, J. and Cho, M. (2021). Convolutional hough matching networks. In *CVPR*.

Min, J., Lee, J., Ponce, J., and Cho, M. (2019a). Hyperpixel flow: Semantic correspondence with multi-layer neural features. In *ICCV*.

Min, J., Lee, J., Ponce, J., and Cho, M. (2019b). Spair-71k: A large-scale benchmark for semantic correspondence. *arXiv:1908.10543*.

Min, J., Lee, J., Ponce, J., and Cho, M. (2020). Learning to compose hypercolumns for visual correspondence. In *ECCV*.

Monnier, T., Fisher, M., Efros, A. A., and Aubry, M. (2022). Share with thy neighbors: Single-view reconstruction by cross-instance consistency. In *ECCV*.

Mummadi, C. K., Subramaniam, R., Hutmacher, R., Vitay, J., Fischer, V., and Metzen, J. H. (2021). Does enhanced shape bias improve neural network robustness to common corruptions? In *ICLR*.

Musgrave, K., Belongie, S., and Lim, S.-N. (2020). A metric learning reality check. In *ECCV*.

Naseer, M. M., Ranasinghe, K., Khan, S. H., Hayat, M., Shahbaz Khan, F., and Yang, M.-H. (2021). Intriguing properties of vision transformers. *NeurIPS*.

Neverova, N., Novotný, D., and Vedaldi, A. (2020). Continuous surface embeddings. In *NeurIPS*.

Nguyen-Phuoc, T., Li, C., Theis, L., Richardt, C., and Yang, Y.-L. (2019). Hologan: Unsupervised learning of 3d representations from natural images. In *ICCV*.

Niemeyer, M. and Geiger, A. (2021). Giraffe: Representing scenes as compositional generative neural feature fields. In *CVPR*.

Niemeyer, M., Mescheder, L., Oechsle, M., and Geiger, A. (2020). Differentiable volumetric rendering: Learning implicit 3d representations without 3d supervision. In *CVPR*.

Noroozi, M., Pirsiavash, H., and Favaro, P. (2017). Representation learning by learning to count. In *ICCV*.

O Pinheiro, P. O., Almahairi, A., Benmalek, R., Golemo, F., and Courville, A. C. (2020). Unsupervised learning of dense visual representations. *NeurIPS*.

Oord, A. v. d., Li, Y., and Vinyals, O. (2018). Representation learning with contrastive predictive coding. *arXiv:1807.03748*.

Oquab, M., Darcet, T., Moutakanni, T., Vo, H. V., Szafraniec, M., Khalidov, V., Fernandez, P., HAZIZA, D., Massa, F., El-Nouby, A., Assran, M., Ballas, N., Galuba, W., Howes, R., Huang, P.-Y., Li, S.-W., Misra, I., Rabbat, M., Sharma, V., Synnaeve, G., Xu, H., Jegou, H., Mairal, J., Labatut, P., Joulin, A., and Bojanowski, P. (2024). DINOv2: Learning robust visual features without supervision. *TMLR*.

Palmer, S. E. (1999). *Vision science: Photons to phenomenology*. MIT press.

Pan, X., Dai, B., Liu, Z., Loy, C. C., and Luo, P. (2021). Do 2d gans know 3d shape? unsupervised 3d shape reconstruction from 2d image gans. In *ICLR*.

Park, K., Sinha, U., Barron, J. T., Bouaziz, S., Goldman, D. B., Seitz, S. M., and Martin-Brualla, R. (2021). Nerfies: Deformable neural radiance fields. In *ICCV*.

Pathak, D., Krahenbuhl, P., Donahue, J., Darrell, T., and Efros, A. A. (2016). Context encoders: Feature learning by inpainting. In *CVPR*.

Pepik, B., Gehler, P., Stark, M., and Schiele, B. (2012). 3D2PM – 3D Deformable Part Models. In *ECCV*.

Poole, B., Jain, A., Barron, J. T., and Mildenhall, B. (2023). Dreamfusion: Text-to-3d using 2d diffusion. In *ICLR*.

Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al. (2021). Learning transferable visual models from natural language supervision. In *ICML*.

Rajasegaran, J., Chen, X., Li, R., Feichtenhofer, C., Malik, J., and Ginosar, S. (2025). Gaussian masked autoencoders. *arXiv:2501.03229*.

Rajasegaran, J., Pavlakos, G., Kanazawa, A., Feichtenhofer, C., and Malik, J. (2023). On the benefits of 3d pose and tracking for human action recognition. In *CVPR*.

Rajasegaran, J., Pavlakos, G., Kanazawa, A., and Malik, J. (2022). Tracking people by predicting 3d appearance, location & pose. In *CVPR*.

Ranftl, R., Bochkovskiy, A., and Koltun, V. (2021). Vision transformers for dense prediction. *ICCV*.

Ranftl, R., Lasinger, K., Hafner, D., Schindler, K., and Koltun, V. (2022). Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer. *PAMI*.

Ravi, N., Reizenstein, J., Novotny, D., Gordon, T., Lo, W.-Y., Johnson, J., and Gkioxari, G. (2020). Accelerating 3d deep learning with pytorch3d. *arXiv:2007.08501*.

Roberts, L. G. (1963). *Machine perception of three-dimensional solids*. PhD thesis, MIT.

Rocco, I., Arandjelovic, R., and Sivic, J. (2017). Convolutional neural network architecture for geometric matching. In *CVPR*.

Rocco, I., Arandjelović, R., and Sivic, J. (2020). Efficient neighbourhood consensus networks via submanifold sparse convolutions. In *ECCV*.

Rocco, I., Cimpoi, M., Arandjelović, R., Torii, A., Pajdla, T., and Sivic, J. (2018). Neighbourhood consensus networks. *NeurIPS*.

Roh, B., Shin, W., Kim, I., and Kim, S. (2021). Spatially consistent representation learning. In *CVPR*.

Rombach, R., Blattmann, A., Lorenz, D., Esser, P., and Ommer, B. (2022). High-resolution image synthesis with latent diffusion models. In *CVPR*.

Rueegg, N., Zuffi, S., Schindler, K., and Black, M. J. (2022). Barc: Learning to regress 3d dog shape from images by exploiting breed information. In *CVPR*.

Ruggero Ronchi, M. and Perona, P. (2017). Benchmarking and error diagnosis in multi-instance pose estimation. In *ICCV*.

Russakovsky, O., Deng, J., Huang, Z., Berg, A. C., and Fei-Fei, L. (2013). Detecting avocados to zucchinis: what have we done, and where are we going? In *ICCV*.

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., et al. (2015). Imagenet large scale visual recognition challenge. *IJCV*.

Ryou, S. and Perona, P. (2021). Weakly supervised keypoint discovery. *arXiv:2109.13423*.

Sarkar, A., Mai, H., Mahapatra, A., Lazebnik, S., Forsyth, D. A., and Bhattad, A. (2024). Shadows don't lie and lines can't bend! generative models don't know projective geometry... for now. In *CVPR*.

Sarlin, P.-E., DeTone, D., Malisiewicz, T., and Rabinovich, A. (2020). Superglue: Learning feature matching with graph neural networks. In *CVPR*.

Shen, B., Yan, X., Qi, C. R., Najibi, M., Deng, B., Guibas, L., Zhou, Y., and Anguelov, D. (2023). Gina-3d: Learning to generate implicit neural assets in the wild. In *CVPR*.

Shen, T., Gao, J., Yin, K., Liu, M.-Y., and Fidler, S. (2021). Deep marching tetrahedra: a hybrid representation for high-resolution 3d shape synthesis. In *NeurIPS*.

Shi, Y., Wang, P., Ye, J., Mai, L., Li, K., and Yang, X. (2023). Mvdream: Multi-view diffusion for 3d generation. *arXiv:2308.16512*.

Shin, D., Fowlkes, C. C., and Hoiem, D. (2018). Pixels, voxels, and views: A study of shape representations for single view 3d object shape prediction. In *CVPR*.

Shtedritski, A., Vedaldi, A., and Rupprecht, C. (2023). Learning universal semantic correspondences with no supervision and automatic data curation. In *ICCV*.

Sigurdsson, G. A., Russakovsky, O., and Gupta, A. (2017). What actions are needed for understanding human actions in videos? In *ICCV*.

Silberman, N., Hoiem, D., Kohli, P., and Fergus, R. (2012). Indoor segmentation and support inference from rgbd images. In *ECCV*.

Simonyan, K. and Zisserman, A. (2015). Very deep convolutional networks for large-scale image recognition. *ICLR*.

Skorokhodov, I., Siarohin, A., Xu, Y., Ren, J., Lee, H.-Y., Wonka, P., and Tulyakov, S. (2023). 3d generation on imagenet. In *ICLR*.

Smith, L. and Gasser, M. (2005). The development of embodied cognition: Six lessons from babies. *Artificial life*.

Spelke, E. S. (1990). Principles of object perception. *Cognitive science*.

Spelke, E. S. and Kinzler, K. D. (2007). Core knowledge. *Developmental science*.

Standley, T., Zamir, A., Chen, D., Guibas, L., Malik, J., and Savarese, S. (2020). Which tasks should be learned together in multi-task learning? In *ICML*.

Stathopoulos, A., Pavlakos, G., Han, L., and Metaxas, D. N. (2023). Learning articulated shape with keypoint pseudo-labels from web images. In *CVPR*.

Szymanowicz, S., Rupprecht, C., and Vedaldi, A. (2024). Splatter image: Ultra-fast single-view 3d reconstruction. In *CVPR*.

Tatarchenko, M., Richter, S. R., Ranftl, R., Li, Z., Koltun, V., and Brox, T. (2019). What do single-view 3d reconstruction networks learn? In *CVPR*.

Taylor, J., Stebbing, R., Ramakrishna, V., Keskin, C., Shotton, J., Izadi, S., Hertzmann, A., and Fitzgibbon, A. (2014). User-specific hand modeling from monocular depth sequences. In *CVPR*.

Thewlis, J., Albanie, S., Bilen, H., and Vedaldi, A. (2019). Unsupervised learning of landmarks by descriptor vector exchange. In *ICCV*.

Thewlis, J., Bilen, H., and Vedaldi, A. (2017a). Unsupervised learning of object frames by dense equivariant image labelling. *NeurIPS*.

Thewlis, J., Bilen, H., and Vedaldi, A. (2017b). Unsupervised learning of object landmarks by factorized spatial embeddings. In *ICCV*.

Tretschk, E., Kairanda, N., R, M. B., Dabral, R., Kortylewski, A., Egger, B., Habermann, M., Fua, P., Theobalt, C., and Golyanik, V. (2022). State of the art in dense monocular non-rigid 3d reconstruction. In *Computer Graphics Forum*.

Tulsiani, S., Kulkarni, N., and Gupta, A. (2020). Implicit mesh reconstruction from unannotated image collections. *arXiv:2007.08504*.

Turing, A. M. and Haugeland, J. (1950). Computing machinery and intelligence.

Ufer, N. and Ommer, B. (2017). Deep semantic feature matching. In *CVPR*.

Van den Oord, A., Li, Y., and Vinyals, O. (2018). Representation learning with contrastive predictive coding. *arXiv:1807.03748*.

Van der Maaten, L. and Hinton, G. (2008). Visualizing data using t-sne. *JMLR*.

Van Horn, G., Cole, E., Beery, S., Wilber, K., Belongie, S., and Mac Aodha, O. (2021). Benchmarking representation learning for natural world image collections. In *CVPR*.

Van Horn, G. and Perona, P. (2017). The devil is in the tails: Fine-grained classification in the wild. *arXiv:1709.01450*.

Vasudev, K. A., Gupta, A., and Tulsiani, S. (2022). Pre-train, self-train, distill: A simple recipe for supersizing 3d reconstruction. In *CVPR*.

Wah, C., Branson, S., Welinder, P., Perona, P., and Belongie, S. (2011). The caltech-ucsd birds-200-2011 dataset.

Wang, J., Chen, M., Karaev, N., Vedaldi, A., Rupprecht, C., and Novotny, D. (2025). Vggt: Visual geometry grounded transformer. In *CVPR*.

Wang, N., Zhang, Y., Li, Z., Fu, Y., Liu, W., and Jiang, Y.-G. (2018). Pixel2mesh: Generating 3d mesh models from single rgb images. In *ECCV*.

Wang, S., Leroy, V., Cabon, Y., Chidlovskii, B., and Revaud, J. (2024). Dust3r: Geometric 3d vision made easy. In *CVPR*.

Wang, X., Jabri, A., and Efros, A. A. (2019). Learning correspondence from the cycle-consistency of time. In *CVPR*.

Wang, X., Zhang, R., Shen, C., Kong, T., and Li, L. (2021a). Dense contrastive learning for self-supervised visual pre-training. In *CVPR*.

Wang, Z., Li, Q., Zhang, G., Wan, P., Zheng, W., Wang, N., Gong, M., and Liu, T. (2021b). Exploring set similarity for dense self-supervised representation learning. *arXiv:2107.08712*.

Weber, M., Welling, M., and Perona, P. (2000). Unsupervised learning of models for recognition. In *ECCV*.

Wei, F., Gao, Y., Wu, Z., Hu, H., and Lin, S. (2021). Aligning pretraining for detection via object-level contrastive learning. *NeurIPS*.

Wu, S., Jakab, T., Rupprecht, C., and Vedaldi, A. (2023a). Dove: Learning deformable 3d objects by watching videos. *IJCV*.

Wu, S., Li, R., Jakab, T., Rupprecht, C., and Vedaldi, A. (2023b). Magicpony: Learning articulated 3d animals in the wild. In *CVPR*.

Wu, Z., Xiong, Y., Yu, S. X., and Lin, D. (2018). Unsupervised feature learning via non-parametric instance discrimination. In *CVPR*.

Xian, Y., Lampert, C. H., Schiele, B., and Akata, Z. (2018). Zero-shot learning—a comprehensive evaluation of the good, the bad and the ugly. *PAMI*.

Xu, J., Zhang, Y., Peng, J., Ma, W., Jesslen, A., Ji, P., Hu, Q., Zhang, J., Liu, Q., Wang, J., et al. (2023). Animal3d: A comprehensive dataset of 3d animal pose and shape. In *ICCV*.

Yang, G., Sun, D., Jampani, V., Vlasic, D., Cole, F., Chang, H., Ramanan, D., Freeman, W. T., and Liu, C. (2021a). Lasr: Learning articulated shape reconstruction from a monocular video. In *CVPR*.

Yang, G., Sun, D., Jampani, V., Vlasic, D., Cole, F., Liu, C., and Ramanan, D. (2021b). Viser: Video-specific surface embeddings for articulated 3d shape reconstruction. In *NeurIPS*.

Yao, C.-H., Hung, W.-C., Li, Y., Rubinstein, M., Yang, M.-H., and Jampani, V. (2022). Lassie: Learning articulated shape from sparse image ensemble via 3d part discovery. In *NeurIPS*.

Yao, C.-H., Hung, W.-C., Li, Y., Rubinstein, M., Yang, M.-H., and Jampani, V. (2023). Hi-lassie: High-fidelity articulated shape and skeleton discovery from sparse image ensemble. In *CVPR*.

Ye, Y., Tulsiani, S., and Gupta, A. (2021). Shelf-supervised mesh prediction in the wild. In *CVPR*.

Yu, A., Ye, V., Tancik, M., and Kanazawa, A. (2021). pixelnerf: Neural radiance fields from one or few images. In *CVPR*.

Yu, F., Seff, A., Zhang, Y., Song, S., Funkhouser, T., and Xiao, J. (2015). LSUN: Construction of a Large-scale Image Dataset using Deep Learning with Humans in the Loop. *arXiv:1506.03365*.

Yu, X., Xu, M., Zhang, Y., Liu, H., Ye, C., Wu, Y., Yan, Z., Zhu, C., Xiong, Z., Liang, T., et al. (2023). Mvimgnet: A large-scale dataset of multi-view images. In *CVPR*.

Yue, Y., Das, A., Engelmann, F., Tang, S., and Lenssen, J. E. (2024). Improving 2d feature representations by 3d-aware fine-tuning. In *ECCV*.

Yunus, R., Lenssen, J. E., Niemeyer, M., Liao, Y., Rupprecht, C., Theobalt, C., Pons-Moll, G., Huang, J.-B., Golyanik, V., and Ilg, E. (2024). Recent trends in 3d reconstruction of general non-rigid scenes. In *Computer Graphics Forum*.

Zamir, A. R., Sax, A., Shen, W., Guibas, L. J., Malik, J., and Savarese, S. (2018). Taskonomy: Disentangling task transfer learning. In *CVPR*.

Zbontar, J., Jing, L., Misra, I., LeCun, Y., and Deny, S. (2021). Barlow twins: Self-supervised learning via redundancy reduction. In *ICML*.

Zeiler, M. D. and Fergus, R. (2014). Visualizing and understanding convolutional networks. In *ECCV*.

Zhan, G., Zheng, C., Xie, W., and Zisserman, A. (2024). A general protocol to probe large vision models for 3d physical understanding. *NeurIPS*.

Zhang, R., Isola, P., and Efros, A. A. (2016a). Colorful image colorization. In *ECCV*.

Zhang, R., Isola, P., Efros, A. A., Shechtman, E., and Wang, O. (2018a). The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR*.

Zhang, S., Benenson, R., Omran, M., Hosang, J., and Schiele, B. (2016b). How far are we from solving pedestrian detection? In *CVPR*.

Zhang, X., Wang, Z., Zhou, H., Ghosh, S., Gnanapragasam, D., Jampani, V., Su, H., and Guibas, L. (2024). Condense: Consistent 2d/3d pre-training for dense and sparse features from multi-view images. In *ECCV*.

Zhang, Y., Guo, Y., Jin, Y., Luo, Y., He, Z., and Lee, H. (2018b). Unsupervised discovery of object landmarks as structural representations. In *CVPR*.

Zhang, Z., Luo, P., Loy, C. C., and Tang, X. (2015). Learning deep representation for face alignment with auxiliary attributes. *PAMI*.

Zhao, D., Song, Z., Ji, Z., Zhao, G., Ge, W., and Yu, Y. (2021). Multi-scale matching networks for semantic correspondence. In *ICCV*.

Zhong, Y., Yuan, B., Wu, H., Yuan, Z., Peng, J., and Wang, Y.-X. (2021). Pixel contrastive-consistent semi-supervised semantic segmentation. In *ICCV*.

Zhou, B., Khosla, A., Lapedriza, A., Oliva, A., and Torralba, A. (2016). Learning deep features for discriminative localization. In *CVPR*.

Zhou, J., Wei, C., Wang, H., Shen, W., Xie, C., Yuille, A., and Kong, T. (2022). ibot: Image bert pre-training with online tokenizer. In *ICLR*.

Zhou, T., Brown, M., Snavely, N., and Lowe, D. G. (2017). Unsupervised learning of depth and ego-motion from video. In *CVPR*.

Zuffi, S., Kanazawa, A., Berger-Wolf, T., and Black, M. J. (2019). Three-d safari: Learning to estimate zebra pose, shape, and texture from images "in the wild". In *ICCV*.

Zuffi, S., Kanazawa, A., Jacobs, D. W., and Black, M. J. (2017). 3d menagerie: Modeling the 3d shape and pose of animals. In *CVPR*.

Appendix A

SAOR: Single-View Articulated 3D Object Reconstruction

A.1 Architecture

We use a ResNet-50 (He et al., 2016) as our image encoder f_{enc} in our CUB(Wah et al., 2011) experiments and the smaller ResNet-18 in quadruped animal experiments. This is in contrast to much larger ViT-based backbones used in other work (Wu et al., 2023b). We initialize these encoders from scratch, i.e., no supervised or self-supervised pre-training is used. The architecture details are presented in the following tables: deformation network f_d in Table A.1, articulation network f_a in Table A.2, texture network f_t in Table A.3, and pose network f_p in Table A.4. We note the weights used in our experiments for each loss in Table A.5.

Layer	Input	Output	Dim
Linear (3,512)	S°	l_x	$N \times 512$
Linear (512,512)	ϕ_{im}	l_z	1×512
$2 \times$ Linear (512,128)	$l_x + l_z$	L	$N \times 128$
Linear (128,3)	l	D	$N \times 3$

Table A.1. Architecture details of our Deformation Net f_d.

Layer	Input	Output	Dim
Linear (3,512)	S°	l_x	$N \times 512$
Linear (512,512)	ϕ_{im}	l_z	1×512
Linear (512,128)	$l_x + l_z$	L	$N \times 128$
Linear (128,128)	L	L	$N \times 128$
Linear (128,K)	L	W	$N \times K$
K $\times$ Linear (512, 9)	ϕ_{enc}	π	$K \times 9$

Table A.2. Architecture details of our Articulation Net f_a. K is the number of parts and N is the number of vertices, π is camera parameters.

Layer	Input	Output	Dim
Linear (512,512)	ϕ_{im}	L	$512 \times 1 \times 1$
Upsample	L	L_{up}	$512 \times 4 \times 4$
Upsample + Conv2D	L_{up}	L_{up}	$256 \times 8 \times 8$
Upsample + Conv2D	L_{up}	L_{up}	$128 \times 16 \times 16$
Upsample + Conv2D	L_{up}	L_{up}	$64 \times 32 \times 32$
Upsample + Conv2D	L_{up}	L_{up}	$32 \times 64 \times 64$
Upsample + Conv2D	L_{up}	L_{up}	$16 \times 128 \times 128$
Conv2D	L_{up}	T	$3 \times 128 \times 128$

Table A.3. Architecture details of our Texture Net f_t.

Layer	Input	Output	Dim
1 $\times$ Linear (512,128)	ϕ_{im}	L	128
C $\times$ Linear (128,6)	L	$\boldsymbol{r}_p, \boldsymbol{t}_p$	128
Linear (128,C)	L	$\boldsymbol{\alpha}$	128

Table A.4. Architecture details of our Pose Net f_p. C is the number of cameras, and $\boldsymbol{\alpha}$ are the associated scores for each camera (Wu et al., 2023b).

A.2 Training

In our experiments, we trained two different models: SAOR-101 and SAOR-Birds. The bird model is trained from scratch on CUB (Wah et al., 2011) for 500 epochs. In the first 100 epochs, we only learn deformation, and then enable articulation afterward.

The SAOR-101 model is trained in two steps. We first train the model using only Horse data from LSUN (Yu et al., 2015), then finetune it on all 101 animal categories downloaded from the iNaturalist website (iNaturalist, 2023). In a similar fashion to the SAOR-Birds model, we only learn deformation in the first 100 epochs, then allow articulation for about 300 epochs on the horse data. Finally, we finetune the model on all categories on iNaturalist data for 150 epochs. We utilize Adam (Kingma and Ba, 2014) with a fixed learning rate for optimizing our networks. We note the hyperparameters used in Table A.5.

Our simplified swap loss leads to easy hyper-parameter selection compared to Unicorn (Monnier et al., 2022). For instance, in their swap loss term, the following parameters need to be decided: i) feature bank size, ii) minimum and maximum viewpoint difference, and iii) number of bins to divide samples in the feature bank depending on the viewpoint. Moreover, they need to do multistage training where they increase the latent dimensions for the shape and texture codes to obtain similar shapes during training. Here the number of stages and the dimension of latent codes in each stage are also hyperparameters. In our method, we eliminated all of these hyperparameters. Moreover, as we do not use all of the hypotheses cameras to estimate loss during a forward pass as in (Wu et al., 2023b) and as a result of our simplified swap loss, model training is six times faster than Unicorn, as they use six cameras during training, for the same number of epochs.

Parameter	Value/Range
Optimization	
Optimizer	Adam
Learning Rate	1e-4
Batch Size	96
Epochs	500
Image Size	128×128
Mesh	
Number of Vertices	2562
Number of Faces	5120
UV Image Size	$64 \times 128 \times 3$
Number of Parts	12
Initial Position	(0,0,0)
Camera	
Translation Range	(-0.5, 0.5)
Azim Range	(-180,180)
Elev Range	(-15, 30)
Roll Range	(-30, 30)
FOV	30
Number of Cameras	4
Loss Weights	
λ_{rgb}	1
λ_{percp}	10
λ_{mask}	1
λ_{depth}	1
λ_{swap}	1
λ_{smooth}	0.1
λ_{normal}	0.1
λ_{part}	1
λ_{pose}	0.05

Table A.5. Training hyperparameters.