
Improving Deep Image Feature Matching



Xinghui Li
Lincoln College
University of Oxford

A thesis submitted for the degree of
Doctor of Philosophy
Michaelmas 2025

“We can’t solve problems by using the same kind of thinking we used when we created them.” – Albert Einstein

Abstract

Image matching establishes pixel correspondences between two images. It is usually achieved in two steps, where the high-dimensional local image features are first extracted across images and correspondences are established based on features' similarities. Image matching can be categorized into two sub-domains: geometric and semantic matching, where the former matches pixels describing the same 3D point while the latter finds pixels having the same semantic meaning. This thesis focuses on advancing both types of matching from various perspectives and explores another application of image features; namely learning 3D attributes from 2D images.

For geometric matching, we propose improvements in accuracy through refined network architectures. NCNet by Rocco et al. [1], demonstrates that 4D convolution can be used to filter incorrect matches and improve matching accuracy. However, the quadratic complexity of this module makes it impractical for matching high-resolution images, a potentially crucial factor in achieving accurate geometric matching. To address this limitation, we propose a dual-resolution network architecture. Our method applies 4D convolution to filter matches on a coarse scale and uses this result to guide the matching on a finer scale. This approach significantly enhances accuracy by increasing the matching resolution while maintaining low computational overhead.

For semantic matching, we first propose a simple and effective training framework that significantly reduces model complexity. Previous works typically follow connecting a pre-trained feature backbone with a complex matching module to handle appearance differences between semantically similar points. Inspired by metric learning, we show that the feature backbone itself can achieve comparable performance without any additional module, hence greatly simplifying the overall design. Building on this framework, we next explore adapting large vision foundation models for semantic matching. Earlier works found that Stable Diffusion (SD), known for its text-to-image generation capabilities, is also able to produce highly discriminative features for image matching. We demonstrate that the matching capability of SD can be enhanced through prompt tuning, and propose a conditional prompting module that infuses prior knowledge of the image pair as the SD's prompt to further improve matching accuracy.

Beyond feature matching, we investigate inferring 3D information from 2D images, a different use case of image features. Previous studies have shown that a

3D model of a dressed person can be derived from a 2D image feature map. We found that a person’s pose, shape, and garment attributes can be disentangled from this feature map. These attributes can be swapped or replaced, allowing the creation of a new 3D model from the modified feature map. Unlike typical disentanglement tasks like face swapping, which operate in the same data domain, our work transitions from 2D to 3D and is able to generate 3D models from 2D image attributes.

To summarise, this thesis enhances both geometric and semantic matching, and also advances the inference of 3D information from 2D image features.

Keywords – Image Feature Matching, Semantic Matching, Geometric Matching, Diffusion Model, Attribute Disentanglement.

Declaration

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

Xinghui Li, December 2024.

Acknowledgment

This is it—I have reached the destination I set out for five years ago. The life of a DPhil student is vastly different from my undergraduate years; it demands endurance and courage to navigate through frustration, stagnation, and self-doubt in pursuit of those moments of joy and pride hidden within the mist. I could not have reached the end of this journey without the care and support of many.

First and foremost, I would like to express my deepest gratitude to my supervisor, Professor Victor Adrian Prisacariu. His guidance and wisdom have been invaluable throughout my DPhil journey, and this thesis would not have been possible without his unwavering support.

I am also profoundly grateful to Professor Kai Han, with whom I co-authored nearly all my papers. In many ways, he has been my de facto second supervisor. His patience and advice over the past five years have left an indelible mark on my work and my growth as a researcher.

I owe special thanks to Dr. Shuda Li, whom I met during my summer internship at AVL. He patiently answered all my naive questions, and I was delighted to become his lab mate a year later, collaborating with him on my first project. I will always remember his kindness.

I cannot forget my friends at AVL: Ryan, Shuai, Wenjing, Lixiong, Theo, Michael, Marcelo, Kate, Henry, Wesley, Jing, Brandon and Xianzheng. I have shared countless hours of happiness with them—every joke we laughed about together, every hurdle we jumped over together, will remain in my heart. I also want to extend special thanks to Han and Xingchen. We have been classmates, roommates, friends, and brothers for nearly ten years. Thank you both for your steadfast support.

Lastly, I extend my deepest love to my parents, Hejie and Hongying. I am so fortunate to be their son and to have received their unconditional love. The final place in this acknowledgment is reserved for Junlan, the special person in my life who brightens my world. Thank you for accompanying me throughout my DPhil journey.

Contents

1	Introduction	8
1.1	Key Contributions	10
1.1.1	Improving Geometric Matching	11
1.1.2	Improving Semantic Matching	11
1.1.3	3D Attribute Disentanglement	12
1.2	Thesis Outline	13
1.3	List of Publications	13
2	Literature Review	16
2.1	Geometric Matching	16
2.1.1	The Detector-Descriptor Method	17
2.1.2	End-to-End Method	19
2.1.3	Match Filtering	20
2.2	Semantic Matching	20
2.2.1	Improving Model Performance	21
2.2.2	Learning without Keypoint Annotations	23
2.3	Human Body 3D Attribute Disentanglement	24
2.3.1	Disentanglement	24
2.3.2	Parametric Human Modelling	25
3	Improving Geometric Matching	27
3.1	Introduction	27
3.2	Publication	29
3.3	Conclusion	48
4	Improving Semantic Matching	49
4.1	SimSC: A Simple and Effective Learning Framework for Semantic Correspondence	49
4.1.1	Introduction	49
4.1.2	Publication	51

4.1.3	Conclusion	68
4.2	SD4Match: Learning to prompt Stable Diffusion Model for Semantic Matching	69
4.2.1	Introduction	69
4.2.2	Publication	70
4.2.3	Conclusion	87
5	Disentangle 3D Attributes from 2D Image Features	88
5.1	Introduction	88
5.2	Publication	89
5.3	Conclusion	105
6	Conclusion	106
6.1	Discussion and Future Work	107

Chapter 1

Introduction

Establishing pixel correspondences across two images is a fundamental problem in computer vision. Techniques are commonly divided into two categories: geometric matching and semantic matching. Geometric matching focuses on aligning pixels representing the same 3D world point across different images, it enables the recovery of 3D information about a scene and is essential for applications such as Simultaneous Localization and Mapping (SLAM) [2, 3, 4] and Structure-from-Motion (SfM) [5, 6]. In contrast, semantic matching aligns the pixels that share the same semantic meaning, regardless of whether they correspond to the same 3D object or not. This technique is useful for tasks like appearance transfer [7, 8], where pixels of one object are mapped to another.

Finding accurate correspondence is challenging due to significant visual differences between images. In geometric matching, these differences arise primarily from variations in viewpoint and illumination, as one would find when matching images of the same building are captured at different times of the day. Changes in viewpoint can also introduce partial occlusions, further complicating the matching process. In semantic matching, the main challenge is intra-class appearance variation. For instance, both a teddy dog and a Labrador are classified as dogs,

yet their appearances differ greatly, which leads to significant challenges when we learn image feature.

Matching relies on image features, which are high-dimensional vectors that encode local visual content at various positions across images. Correspondences are established based on similarities between image features across two images. Therefore, the key to success is to extract reliable image features, and how to use them for matching. Early approaches utilized handcrafted features [9, 10, 11], and found correspondences based on L2 distances between features. More recent methods employ neural networks to learn features [12, 13, 14, 15, 16], which have shown greater robustness. Along with the image features itself, some works focus on refining the initial matches, typically by enforcing neighborhood consensus constraints [1, 17, 18] or learning an inlier-outlier predictor [19, 20, 21, 22].

Despite the similarities between geometric and semantic matching, their learned features differ substantially. Features for geometric matching retain low-level geometric properties [14], such as edges and corners, while features for semantic matching capture high-level semantic information about objects [23]. This difference has the effect that geometric matchers are usually trained from scratch [24, 15, 14, 13] from multi-view images while semantic matchers [17, 18, 16] rely on feature backbones [25] pre-trained on the large-scale image classification datasets [26] to leverage high-level embedded in them.

This thesis primarily explores ways to improve both geometric and semantic matching. In Chapter 3, we focus on geometric matching and propose a dual-resolution network architecture to increase the matching resolution when applying 4D neighborhood consensus filtering, a powerful module to suppress incorrect matches. This approach enables correspondences to be localized at a finer resolution, thereby improving matching accuracy. In Chapter 4, we turn to semantic matching. We begin by introducing a simple and effective semantic matching

learning framework in Section 4.1. Inspired by metric learning, this framework improves the image feature by searching for an optimal temperature for the cross-entropy loss. It significantly simplifies network design while maintaining the same performance. After that, in Section 4.2, we integrate prompt tuning with Stable Diffusion (SD) [27], a large-scale text-to-image generative model, to improve SD’s feature for semantic matching in a similar lightweight fashion. This approach leverages SD’s extensive knowledge of objects to enhance the discriminative power of features used in semantic matching.

Moving beyond our primary focus: the pixel-wise feature matching, Chapter 5 explores another application of image features: learning 3D attributes from 2D images in the human domain. Previous studies [28, 29] have shown that the 3D model of a dressed person can be derived from its 2D image feature map. We found that a person’s pose, shape, and garment attributes can be disentangled from this 2D feature map. These learned attributes can be swapped and replaced, generating a new feature map with updated information and enabling the creation of a new 3D model based on the modified feature map. This enables more streamlined 3D pose, shape, or garment swapping between people, especially if we only have images of them.

This chapter is organized as follows. In Section 1.1, we elaborate our key contributions. In Section 1.2, we present the outline of the thesis and in Section 1.3, we provide the publication list.

1.1 Key Contributions

This thesis offers contributions to the fields of geometric and semantic matching from multiple perspectives, as well as to 3D inference from 2D image features through a novel 2D-to-3D disentanglement framework.

1.1.1 Improving Geometric Matching

The neighborhood consensus constraint [1, 30, 31] is effective in filtering out incorrect matches, and is based on the observation that the correspondences of clustered query points are likely to be close to each other. Therefore, it is unlikely that the correspondences of two adjacent points are far apart in image pixel space.

One way to enforce this constraint is to apply a 4D convolutional module to the feature correlation tensor between the two images [1], which would also enable the overall pipeline to be differentiable. However, the 4D convolution has quadratic complexity with respect to image size, which makes them computationally expensive and challenging to apply at high-resolution matching. High-resolution matching is critical to accurate geometric correspondence, as it retains more local details and allows finer localization of correspondences. To address this limitation, we propose a dual-resolution network architecture in Chapter 3. This architecture applies 4D convolution at low resolution and uses the resulting information to guide high-resolution matching. This design significantly improves matching resolution and accuracy with minimal computational overhead. We published this work in NeurIPS 2020 [32], and an extension work in TPAMI 2024 [33].

1.1.2 Improving Semantic Matching

Methods tackling semantic matching typically follow the design of a feature backbone appended by a matching module [31, 1, 17, 18, 8]. The dense feature maps of both images are first extracted by a feature backbone pre-trained on a large-scale dataset, such as ImageNet [26]. They are then processed by the matching module, which is fine-tuned on semantic matching datasets [34, 35].

Many studies have attempted to improve matching accuracy by developing more advanced matching modules. While these approaches have shown some ef-

fectiveness, they tend to increase the complexity and computational demands of the model. In Section 4.1, we revisit metric learning [36, 37], and propose a straightforward framework that allows the feature backbone itself to achieve comparable accuracy to methods with more complex architecture. The key is to find an appropriate temperature value for the softmax component when training with the cross-entropy loss, which we enable by introducing a lightweight temperature learning module to automatically identify the optimal temperature for the feature backbone. This work is presented to the public as an ArXiv preprint in 2023 [38].

Inspired by our formulation in Section 4.1, we investigate the adaptation of large vision foundation models for semantic matching without adding complex modules in Section 4.2. Stable Diffusion (SD) [27] has exhibited remarkable abilities in text-to-image generation. Previous works [39] have also demonstrated that SD can produce strong image features for semantic matching, leveraging the prior knowledge of objects learned from billions of images [40]. However, further improvement of SD for this task remains underexplored.

We demonstrate that the performance of SD on semantic matching can be straightforwardly improved by tuning a single prompt for SD using our proposed learning framework. Additionally, we show the benefits of prior information about objects to the matching, and propose a conditional prompting module that conditions the prompt to SD on the local features of the image pair. This approach injects object priors into SD, and further enhances matching accuracy. We published this work in CVPR 2024 [41].

1.1.3 3D Attribute Disentanglement

Prior studies [29, 28] have shown that the 3D model of a dressed person can be directly inferred from the 2D feature map of its image. However, these methods lack a way to manipulate the resultant 3D model due to its nonparametric nature.

In Chapter 5, we address this gap by presenting a method to learn the implicit representation of 3D attributes directly from the image feature map.

Specifically, we learn to disentangle pose, shape, and garment attributes from the feature map using an auto-encoder network architecture. These attributes can be replaced or swapped, and then re-assembled into a new feature map from which a new 3D model can be inferred. This results in a 2D-to-3D disentanglement pipeline, distinguishing itself from ordinary 2D-to-2D [42, 43] or 3D-to-3D [44, 45] disentanglement frameworks. We published this work in BMVC 2022 [46].

1.2 Thesis Outline

This integrated thesis is structured as follows: In Chapter 2, we include the literature review of relevant topics; In Chapters 3 to 5, we present one research paper per chapter to elaborate our contribution and finally in Chapter 6, we summarise our contributions, evaluate their impact and outline the direction for future work.

1.3 List of Publications

We outline the publications included in this thesis, together with their corresponding chapters:

Chapter 3: “DualRC: A Dual-Resolution Learning Framework With Neighbourhood Consensus for Visual Correspondences”. Xinghui Li, Kai Han, Shuda Li, Victor Adrian Prisacariu. In *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2024.

Section 4.1: “SimSC: A Simple and Effective Framework for Semantic Correspondence”. Xinghui Li, Kai Han, Xingchen Wan, Victor Adrian Prisacariu. In *arXiv preprint arXiv:2305.02385*, 2023.

Section 4.2: “SD4Match: Learning to Prompt Stable Diffusion Model for Semantic Matching”. Xinghui Li, Jingyi Lu, Kai Han, Victor Adrian Prisacariu. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

Chapter 5: “Disentangling 3D Attributes from a Single 2D Image: Human Pose, Shape and Garment”. Xinghui Li*, Xue Hu*, Benjamin Busam, Yiren Zhou, Ales Leonardis, Shanxin Yuan. In *The British Machine Vision Conference (BMVC)*, 2022.

Publications Not Included

During my DPhil, we also contributed to the following publications. These publications are not included in this thesis either because they are preliminary results to relevant chapters in this thesis or their topics are loosely connected to the thesis.

⇒ **“Dual-Resolution Correspondence Networks”.** Xinghui Li, Kai Han, Shuda Li, Victor Adrian Prisacariu. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

Abstract: A dual-resolution network that adapts 4D neighbourhood consensus filtering to geometric matching. This is the preliminary work to Chapter 3 of this thesis.

⇒ **“DFNet: Enhance Absolute Pose Regression with Direct Feature Matching”.** Shuai Chen, Xinghui Li, Zirui Wang, Victor Adrian Prisacariu. In *European Conference on Computer Vision (ECCV)*, 2022.

Abstract: Leveraging luma-prior NeRF and direct feature matching to enhance 6-DoF camera pose regression approaches.

⇒ **“NeFeS: Refinement for Absolute Pose Regression with Neural Feature Synthesis”.** Shuai Chen, Yash Bhalgat, Xinghui Li, Jiawang

Bian, Kejie Li, Zirui Wang, Victor Adrian Prisacariu. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

Abstract: Evolving from DFNet, NeFeS directly renders feature maps during novel view synthesis, leading to more robust and faster APR.

⇒ **“GaussCtrl: Multi-View Consistent Text-Driven 3D Gaussian Splatting Editing”**. Jing Wu^{*}, Jia-Wang Bian^{*}, **Xinghui Li**, Guangrun Wang, Ian Reid, Philip Torr, Victor Adrian Prisacariu. In *European Conference on Computer Vision (ECCV)*, 2024.

Abstract: Text-driven multiview-consistent 3D Gaussian Splatting Editing through depth control and attention alignment leveraging pre-trained Stable Diffusion.

⇒ **“RegionEdit: Region-based Image Editing with Diffusion Models”**. Jingyi Lu, **Xinghui Li**, Kai Han. In *European Conference on Computer Vision (ECCV)*, 2024.

Abstract: A region-based efficient image editing method through copy-and-paste on the latent representation of the image. It achieves more flexible editing and a significantly faster inference speed than keypoint-dragging-based image editing methods such as DragDiffusion [47].

Chapter 2

Literature Review

This literature review provides an overview of relevant works to the thesis. We first review geometric matching in Section 2.1, followed by semantic matching in Section 2.2 and finally attribute disentanglement in Section 2.3. As Chapters 3 to 5 contain individual research papers, a more detailed literature review can be found in each chapter respectively.

2.1 Geometric Matching

Geometric matching methods can be broadly classified into two categories: *detector-descriptor* methods and *end-to-end* methods. The detector-descriptor approach begins by identifying keypoints in both images using a keypoint detector. Next, it uses a descriptor to encode the local image content around these keypoints into high-dimensional feature vectors. Image correspondences are then established by matching keypoints based on their respective feature vectors. In contrast, the end-to-end approach bypasses the explicit stages of keypoint detection and encoding, aiming instead to generate correspondences directly through a single forward pass of a deep learning model. Beyond these two approaches that predict matches from

images, a third line of work concentrates on filtering unreliable correspondences from existing matches. We review each of these approaches in Sections 2.1.1 to 2.1.3 respectively. In the context of this thesis, Chapter 3 aligns with the end-to-end method. As geometric matching is a broad, vibrant, and fast-evolving field. It is difficult to discuss every aspect of this area and go beyond the scope of this work, so we refer interested readers to a recent survey [48] for a more comprehensive background.

2.1.1 The Detector-Descriptor Method

Keypoint Detectors play a critical role in identifying salient correspondences between two images. Keypoints often represent regions of abrupt visual change, and traditional handcrafted detectors [9, 49, 50] use intensity gradients to locate these points. However, these methods commonly encounter the “missed detection problem” where a keypoint is detected in only one of the images, for example, due to environmental changes such as variations in viewpoint or illumination. This inconsistency can negatively impact downstream tasks. To address this, more recent works have focused on designing detectors that are invariant to such changes. Some approaches learn invariance to viewpoint [51] or illumination [52] by training on contrastive data, while others focus on learning abstract, high-level representations of keypoints [15, 53] to achieve robustness against diverse appearance variations.

Descriptors are designed to generate high-dimensional feature vectors that capture the local information surrounding each keypoint. To achieve this, a small patch centered on the keypoint is extracted and passed through the descriptor processing pipeline for encoding. The resulting feature vector should ideally be invariant to changes in orientation, illumination, and viewpoint. Traditional handcrafted descriptors leverage various types of local information, such as orientation

histograms [9, 54, 55] or Haar wavelet transformations [10], and apply normalization techniques to enhance their robustness against rotation and lighting variations. However, despite these efforts, handcrafted descriptors often struggle under challenging conditions. To address this limitation, more recent works have focused on learning-based descriptors. Most of them [56, 57, 58, 59, 15] employ convolutional neural network (CNN) architectures and use training pipelines similar to those in representation learning for image classification [37, 36]. In these approaches, each query feature is paired with its ground-truth corresponding feature and multiple negative samples, with a loss function designed to minimize the distance between the query and its ground-truth feature, while maximizing the distance between the query and negative pairs. Various advancements have been proposed to improve the robustness of these descriptors, including multi-instance training [56], hard-case mining [57], maximizing second-order similarity [59], and matching intermediate network outputs [58]. Additionally, non-contrastive learning approaches, such as those focused on maximizing average precision [60, 61], have also been explored.

Combined Detector-Descriptor models integrate both keypoint detection and feature description into a single model, aiming for a more streamlined approach, that can produce overall improved results. One popular design is to employ a deep CNN architecture to learn image features, with keypoints detected by either identifying consistent high responses across multiple intermediate layers [12, 62] or by jointly learning keypoint detection and encoding via a unified process [63, 64, 65]. Another paradigm utilizes a shared encoder for both keypoint detection and description, followed by separate prediction heads. In this setup, keypoints are represented either as learnable score maps indicating potential keypoint quality [14, 66] or as combinations of reliability and repeatability scores [13].

The advantages of detector-descriptor methods are their potentially fast speed, and their seamless integration into many standard pipelines, such as COLMAP [6, 5] for structure-from-motion or HLoc [67] for visual localization. However, as such approaches rely on a keypoint detector, their performance may be compromised in scenes with plain texture, where point detection can become more difficult.

2.1.2 End-to-End Method

End-to-end methods bypass traditional keypoint detection and description stages by directly predicting correspondences between pairs of images. These approaches typically begin by extracting dense feature maps from both images using a CNN or ViT-based [68] backbone. Next, a matching module aggregates information from the dense feature maps to generate correspondence predictions. Various types of matching modules have been proposed, including 4D convolutional kernels, which enforce neighborhood consensus constraints [1, 69]; transformer-based modules that facilitate global-scale feature interactions [24, 70, 71, 72]; and Gaussian Processes, which enables a probabilistic formulation [73, 74]. Beyond the matching module, end-to-end methods often employ a coarse-to-fine matching pipeline [24, 70, 73, 75, 76, 77]. In this approach, correspondences are initially established at a coarse, low-resolution level and progressively refined to a finer, high-resolution level. Our proposed method, detailed in Chapter 3, is among the first to employ this strategy to achieve notable improvements in correspondence accuracy. More recently, Barroso-Laguna et al. introduced Mickey [78], a method that lifts images into 3D through scene coordinate regression, finds correspondences within 3D space, and then projects them back to the image domain. This approach demonstrates significant improvement in handling images with extreme scale variations.

Compared with detector-descriptor methods, end-to-end methods may perform better in scenes with plain texture, as they do not have to detect keypoints first

and are capable of finding correspondences for all pixels in images. The trade-off is that they may not be as easily integrated into standard pipelines like COLMAP as detector-descriptor methods.

2.1.3 Match Filtering

Unlike the aforementioned two lines of work, match filtering does not concentrate on establishing correspondence given a pair of images, but focuses on identifying unreliable correspondences within an existing set of matches. Early methods distinguish positive and negative pairs based on local smoothness [79, 80, 81, 82] or neighborhood consensus [30, 83, 84, 85]. In deep learning-based approaches, some methods formulate the task as an inlier-outlier classification problem [19, 20, 21, 22], where they take the coordinates of correspondences and predict the likelihood of each pair being an inlier. Other methods [86, 87] frame it as a feature refinement problem, pairing each keypoint with its feature and passing them through a refinement module—often a graph neural network—to produce refined features, effectively correcting initial false matches.

Since these methods require both keypoint coordinates and features, they are typically used in conjunction with detector-descriptor methods as a post-processing stage to enhance performance. Noteworthy combinations, such as SuperPoint + SuperGlue, have demonstrated robust performance across many challenging scenarios [88].

2.2 Semantic Matching

In contrast to geometric matching, where dense correspondence annotations can be obtained through structure-from-motion or epipolar geometry once depth maps of images are available, annotations for semantic matching must be labeled man-

ually. Although several datasets with correspondence annotations have been released [34, 35, 89], the annotation density and dataset scale remain far below those available for geometric matching. Consequently, research in semantic matching has taken two primary directions: one focuses on enhancing model performance using existing data, while the other explores learning semantic matching without correspondence annotations. Chapter 4 of this thesis focus on improving the model performance; however, we review both approaches for completeness in the following sections.

2.2.1 Improving Model Performance

Similar to geometric matching, early attempts at semantic matching focused on handcrafted features such as HOG [11], SIFT [9], and SIFT Flow [90]. Since semantic matching finds correspondences between different instances of an object category, such as matching the left ears of two different cats—which requires the model to have strong prior knowledge of various object categories. As a result, many methods are built upon feature backbones, such as ResNet [25] or DINO [91], pretrained on large-scale image recognition datasets [26] to leverage their inherent object knowledge. Overall, we group current research into three categories: multi-feature assembly, feature adaptation and correlation tensor filtering.

Multi-feature Assembly This approach collects features across different model layers and concatenates them to create a "hyperfeature" [16, 92, 93], instead of relying solely on the final layer’s output from the feature backbone. This is inspired by the feature pyramid concept [94] in CNNs: earlier layers capture local information, while later layers provide high-level insights. By combining them, the resulting feature retains both low-level and high-level information. Feature selection may be handpicked [16] or optimized for semantic matching objectives

[92, 93]. Recently, studies like [7, 95] have explored combining features from different backbones, showing that these combinations often complement each other, enhancing robustness.

Feature Adaptation This approach appends an additional module after the feature backbone [8, 95, 96]. The feature backbone extracts general-purpose features, which are versatile across tasks [97]. By training an adaptation module specifically on semantic matching loss, these general-purpose features are transformed into semantic matching-specific features, which improves matching accuracy.

Correlation Tensor Filtering A correlation tensor is a 4D tensor storing correlations between every possible feature pair between two images, constructed via the outer product of the feature maps of the two images. This 4D matching space provides raw match distribution, expanding beyond traditional 2D features. A pioneering work in this area, NCNet [1] applies a 4D convolutional kernel to enforce neighborhood consensus and suppress unreliable match scores. This inspired further research proposing non-isotropic 4D [31] and 6D convolutional kernels [17] to handle object scale variations in images. Transformer-based modules [98] have also been developed to enable global perception within the matching space. Additionally, some methods [18, 99, 100] construct multi-level correlation tensors using feature maps from various backbone layers, enhancing filtering module efficacy.

Recently, Stable Diffusion (SD) [27], a vision foundation model for image generation, has attracted significant attention for its extensive object knowledge, acquired from training on billions of images. SD has shown exceptional performance in semantic matching, setting new state-of-the-art on benchmarks for this task [39, 7, 95]. Our work, presented in Section 4.2, investigates ways to enhance SD’s performance on semantic matching through prompt tuning.

Since semantically salient keypoints may not always lie in regions of rapid pixel value change (e.g., the center of an eye), evaluations are typically based on locating correspondences for a predefined set of keypoints. Consequently, most methods resemble geometric end-to-end approaches, as described in Section 2.1, and are capable of finding correspondences for any query position.

2.2.2 Learning without Keypoint Annotations

This research area focuses on developing training methods that do not rely on keypoint annotations. Some studies use image-level supervision, aiming to increase the correlation between images containing objects from the same category while reducing it between images of different categories. Various approaches achieve this goal, ranging from simply maximizing the sum [1] or entropy [92] of the entire 4D correlation tensor for positive image pairs and minimizing it for negative pairs, to selectively identifying reliable regions [101, 102] and maximizing correlations only within those regions.

Other methods [8] incorporate segmentation masks, ensuring that the model’s dense object wrapping falls within object masks. This is often supported by a flow consistency and a smoothness loss to regularize the learning process. A third commonly applied approach is cycle consistency [103, 104, 105]: a query point in image A mapped to image B and then back to image A should return to its original location. The discrepancy between the initial and final mappings serves as a supervision signal, guiding the model. Data augmentation is optionally used to improve the model’s generalizability post-training.

Despite these advancements, models trained without keypoint annotations generally underperform their fully supervised counterparts. This performance gap has led to research in generating pseudo-annotations from existing sparse annotations to increase supervision density and improve model performance. Some approaches

[96, 106, 107] employ a teacher-student training framework: the sparse keypoint annotations are used to train a teacher model, and the teacher’s predictions then serve as ground-truth labels to train a student model. Other methods [108] use pairs of strong and weak data augmentations to generate pseudo-labels: a target image is augmented twice, creating a weakly augmented and a strongly augmented version. The prediction for the weakly augmented image supervises the prediction for the strongly augmented one, given the known transformation between the two versions.

2.3 Human Body 3D Attribute Disentanglement

Since Chapter 5 focuses on disentangling 3D attributes from 2D images in the human domain, we briefly review both disentanglement and human body modeling in this section.

2.3.1 Disentanglement

The objective of disentanglement is to identify underlying latent representations that control data variation. A primary application is controlled generation: once the latent distribution for each attribute is learned, we can manipulate each generation aspect by adjusting its corresponding latent code. The pioneering work in this field, InfoGAN [109], modifies the classic GAN [110] by adding an extra latent variable to the generator to learn the disentangled attribute, and maximizes the mutual information between this latent variable and the data point. It demonstrated disentanglement by learning handwriting style and digit numbers from the MNIST dataset [111] and inspired numerous follow-up studies which improved InfoGAN by various types of latent space regularization [112, 113, 114]. VAE [115] is another commonly used model for disentanglement. Similar to GAN-based methods, VAE’s

encoder disentangles the input data to several latent codes and regularization is applied to the latent space to separate each attribute [116, 117, 115, 118, 119, 120]. Disentanglement techniques enable attribute transfer across both 2D and 3D domains, helping tasks like face identity swapping [42, 43] and hairstyle transfer [121, 116] in 2D images, as well as pose and shape transfer [122, 44, 45] for 3D inputs.

All these methods learn attributes directly from raw data, maintaining the same dimensionality and format for inputs and outputs. In Chapter 5, we explore learning 3D attributes from 2D images by disentangling image feature maps. The dimensional gap and the abstract nature of feature maps introduce additional challenges compared to previous approaches, so we adopt a supervised learning paradigm and explicitly supervise attribute separation by building a synthetic dataset. Disentanglement in attribute learning is a broad, complex area; for a comprehensive background, we recommend a recent survey [123].

2.3.2 Parametric Human Modelling

Parametric models targeting humans initially focus on the parameterization of the naked human body. Various body templates have been proposed in the past decade [124, 125, 126, 127]. These methods built their models on thousands of scans of human bodies, and parameterize them by a pose vector and a shape vector. The transformation of models between different vectors is learned statistically. With the advancement in this field, the community began to explore the parameterization of the garment. Compared to the naked human body, the parameterization of the garment is much more difficult due to the complicated local details such as wrinkles and foldings. The garment models are usually associated with the naked body model such as SMPL [126]. They are either represented as separated SMPL-like templates [128, 129, 130] or displacement fields to the body model

[131, 132]. Some works [133, 128] additionally model the intra-class garment variation using statistical tools such as principle component analysis (PCA). Although parametric models are simple and easy to manipulate, they are less flexible. Our method in Chapter 5 uses an implicit function to learn dressed human modeling [28], enabling a more flexible pipeline.

We have reviewed relevant topics in the literature and provided a general introduction to background knowledge in this chapter. In the next three chapters, we will elaborate on each of our contributions.

Chapter 3

Improving Geometric Matching

3.1 Introduction

This chapter explores enhancing geometric matching accuracy. As discussed in Section 2.1, the neighborhood consensus constraint [30, 83, 84, 85] is a powerful technique for filtering unreliable correspondences. The core intuition is that correspondences of adjacent keypoints should be close to each other. NCNet [1] integrated this concept into an end-to-end pipeline. The method first extracts dense feature maps of two images using a ResNet [25] backbone. A 4D correlation tensor is then constructed from these feature maps, where each entry represents the matching score between every possible pair of features across the two images. Since the correlation tensor is inherently filled with incorrect scores, NCNet employs a 4D convolutional module to learn how to distinguish between positive and negative pairs, based on neighborhood information in the 4D space. This approach suppresses the scores of the unreliable pairs and produces state-of-the-art results at the time of publication. However, the use of 4D convolution makes this approach computationally expensive and difficult to scale to high matching resolutions due to its quadratic complexity relative to image resolution.

High matching resolution would not only capture more accurate local details but would also enhance correspondence localization precision. We pursue this goal and propose a dual-resolution image matching pipeline that finds correspondences in a coarse-to-fine manner. Inspired by the feature pyramid concept [94], we modify the feature backbone to extract both low-resolution and high-resolution feature maps for each image. The low-resolution feature map encodes high-level abstract information, while the high-resolution feature map captures detailed low-level information. We first construct a coarse 4D correlation tensor using the low-resolution feature maps from both images and apply the 4D convolutional module to filter it. This filtered result is then used as a prior to guide the matching process on the high-resolution feature maps. This approach allows us to scale neighborhood consensus filtering to high matching resolutions with minimal computational overhead. Our design has been validated on multiple public benchmarks, demonstrating significant performance gains over the baseline approach, and the state-of-the-art accuracy at the time of submission.

To summarize, the contributions of this chapter are:

1. We introduce a novel neural network architecture which generates dual-resolution feature maps allowing to match correspondences in a coarse-to-fine manner.
2. We apply a 4D convolutional module to enforce neighborhood consensus constraint on the coarse-level matches, and use this result to guide the matching on the fine level. Such a design increases the image matching resolution with minimal computational overhead.
3. We comprehensively evaluate our method on large-scale public benchmarks to demonstrate the effectiveness of our method.

We published our approach in NeurIPS 2020 [32]. We then accelerate our

pipeline using sparse 4D convolution [69] and extend our work to semantic matching using a dynamically adaptive anisotropic 4D kernel. We published these extensions with our main contribution in TPAMI 2024 [33]. Therefore, we include the TPAMI paper in this thesis for completeness, but the main focus is still our dual-resolution architecture in geometric matching.

DualRC: A Dual-Resolution Learning Framework With Neighbourhood Consensus for Visual Correspondences

Xinghui Li^{1b}, Kai Han^{1b}, *Member, IEEE*, Shuda Li^{1b}, and Victor Prisacariu^{1b}

Abstract—We address the problem of establishing accurate correspondences between two images. We present a flexible framework that can easily adapt to both geometric and semantic matching. Our contribution consists of three parts. Firstly, we propose an end-to-end trainable framework that uses the coarse-to-fine matching strategy to accurately find the correspondences. We generate feature maps in two levels of resolution, enforce the neighbourhood consensus constraint on the coarse feature maps by 4D convolutions and use the resulting correlation map to regulate the matches from the fine feature maps. Secondly, we present three variants of the model with different focuses. Namely, a universal correspondence model named DualRC that is suitable for both geometric and semantic matching, an efficient model named DualRC-L tailored for geometric matching with a lightweight neighbourhood consensus module that significantly accelerates the pipeline for high-resolution input images, and the DualRC-D model in which we propose a novel dynamically adaptive neighbourhood consensus module (DyANC) that dynamically selects the most suitable non-isotropic 4D convolutional kernels with the proper neighbourhood size to account for the scale variation. Last, we thoroughly experiment on public benchmarks for both geometric and semantic matching, showing superior performance in both cases.

Index Terms—Correspondence estimation, dense matching, geometric matching, semantic matching.

I. INTRODUCTION

ESTABLISHING correspondences across images is a fundamental problem in computer vision. A common objective is to find the projections of the same 3D geometric points of the objects in different images. Many computer vision applications rely on robust correspondences, such as structure from motion [1], [2] and visual localization [3], [4]. This problem is accompanied by several critical challenges, including viewpoint change, illumination variation, and the presence of repetitive

patterns. The key to tackling this problem is generating discriminative features that distinguish different keypoints. Many handcrafted features, such as SIFT [5], SURF [6], BRISK [7], have been proposed in the past two decades. However, they fail to demonstrate sufficient robustness under the aforementioned challenging conditions. Modern methods [8], [9], [10], [11], [12], [13], [14], [15], [16], [17] are dominated by features generated by the convolutional neural network (CNN), demonstrating superior performance over handcrafted features. Despite improving the performance of establishing correspondences between two different images of the same object (a.k.a. geometric correspondences), the advancement of the learned CNN features also enable the possibility of establishing correspondences between images of different instances from the same category (a.k.a. semantic correspondences), which is more challenging than geometric correspondences due to the extra challenges of huge intra-class appearance and scale variation. Many efforts [18], [19], [20], [21], [22], [23], [24], [25] have been devoted to tackling this challenge in the past few years, showing encouraging results.

A classic pipeline to this problem consists of three stages, namely, detection, description, and matching, including methods like [5], [6], [8], [9], [11], [12], [17]. These methods first detect keypoints across two images and then describe these keypoints by extracting feature vectors from local regions around them. Correspondence pairs are finally established based on similarities between features. Although these methods have demonstrated great successes, they still suffer from the missing detection problem [26]. Matches can only be found between detected keypoints, limiting the matching space. Alternatively, works like [18], [19], [20], [21], [22], [23], [24], [25], [26], [27] focus on establishing dense pixel correspondences, i.e. finding the correspondence for every pixel in the query image. The searching space in these works is the entire target image, remedying the missing detection problem, which, however, introduces more computational overhead, limiting their practical applications.

Among these approaches, the Neighbourhood Consensus (NC) family [24], [25], [26], [28] has shown promising results. These methods employ a CNN to extract features from two images before calculating a 4D correlation tensor representing the entire matching space where each cell records the cosine correlation score between a pair of feature vectors. This tensor is then processed by a neighbourhood consensus module, which consists of a sequence of 4D convolution operations,

Manuscript received 8 November 2021; revised 21 July 2023; accepted 31 August 2023. Date of publication 19 September 2023; date of current version 5 December 2023. This work was supported in part by Hong Kong Research Grant Council - Early Career Scheme under Grant 27208022, in part by the National Natural Science Foundation of China under Grant 62306251, and in part by the HKU Seed Fund for Basic Research. Recommended for acceptance by M. Leordeanu. (*Corresponding author: Kai Han.*)

Xinghui Li, Shuda Li, and Victor Prisacariu are with the Active Vision Lab, University of Oxford, OX1 3AZ Oxford, U.K. (e-mail: xinghui@robots.ox.ac.uk; shuda.dexter.li@gmail.com; victor@robots.ox.ac.uk).

Kai Han is with The University of Hong Kong, Hong Kong (e-mail: kaihhan@hku.hk).

This article has supplementary downloadable material available at <https://doi.org/10.1109/TPAMI.2023.3316770>, provided by the authors.

Digital Object Identifier 10.1109/TPAMI.2023.3316770

0162-8828 © 2023 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See <https://www.ieee.org/publications/rights/index.html> for more information.

to effectively filter incorrect matches. However, there are two main limitations in the vanilla 4D convolution [24]. (i) Due to the $O(n^4)$ complexity, it is prohibitively expensive to process high-resolution images, which is vital for geometric correspondence estimation. (ii) The neighbourhoods of the same size are considered for any pair of locations from the two images. This is inappropriate because the scales and shapes of objects vary a lot, especially for semantic correspondence estimation, as the irrelevant neighbourhood may negatively affect matching.

In this work, we address the above limitations of existing methods based on neighbourhood consensus and make the following three main contributions. (1) We propose a dual-resolution framework to generate feature maps at two levels of resolution for matching. The coarse-resolution feature maps are used to form a complete 4D correlation tensor, which is refined by a 4D neighbourhood consensus module. The refined 4D tensor is used to guide matching on fine-resolution feature maps. This significantly improves the matching performance without requiring the expensive 4D tensor on the fine-resolution feature maps. (2) We present three variants of our framework. The first one is a universal model named DualRC which has the basic NC module consisting of isotropic 4D convolutional kernels. It is suitable for both geometric and semantic correspondence estimation. The second one is named as DualRC-L, which consists of a lightweight NC module based on sparse correlation [26] with the aim to accelerate the pipeline in case of high-resolution input images in geometric matching. The third one is DualRC-D, in which we propose the novel dynamically adaptive neighbourhood consensus (DyANC) module that leverages non-isotropic 4D convolutional kernels and dynamically selects the best neighbourhood region to handle scale variation in semantic tasks. (3) We thoroughly evaluate our approach on public benchmarks for both geometric and semantic matching obtaining superior performance, including HPatches [29], InLoc [4], Aachen Day-Night [30], PF-PASCAL [31], PF-WILLOW [31] and SPair-71k [32].

Preliminary results of our basic DualRC have been published in [33]. In this paper, we have made three main extensions. First, we propose DualRC-L, a lightweight variant of DualRC using sparse 4D convolution kernels to reduce computational complexity significantly; second, we propose DualRC-D, a scale-aware variant of DualRC, at the core of which is our novel dynamically adaptive neighbourhood consensus (DyANC) module, allowing the model to dynamically choose 4D filters of varying sizes, which is different from [25] that only possesses fixed-size neighbourhoods without a self-adaptation mechanism to the actual context; third, we extend the evaluation of our approach from the geometric correspondence estimation task to the semantic correspondence estimation task, and also carry out cross-task evaluation for the state-of-the-art methods developed respectively for each of the tasks. Unlike existing methods that can only work well on one task, our method consistently shows promising results on both tasks, demonstrating that DualRC is a strong method for general visual correspondence estimation.

Our code is publicly available at <https://code.active.vision>.

II. RELATED WORKS

The classic pipeline for correspondence estimation consists of three stages: detection, description, and matching. In the early years, the works in the literature focused on manually developing robust local feature vectors, which is the description stage. A common objective of the design [5], [6], [34] is the invariance of the feature to the rotation and scale of the local patches extracted around the keypoints. These are achieved through rotating the patches back to canonical angles and scale normalization on patches at multiple scales. Common types of information encoded in the features are orientation histogram [5], output of the haar wavelet [6] or result of random binary test [35]. Correspondences are then established based on the nearest neighbour between features, using metrics such as the L2 distance or the cosine similarity. Additional verification like the mutual nearest neighbour or the second nearest neighbour ratio test could be applied to increase the robustness and accuracy of the matching. Although handcrafted features are popular due to their speed and simplicity of implementation, they fail to demonstrate sufficient robustness and distinctiveness under challenging cases like large viewpoint and illumination changes as well as the intra-category appearance change.

Learned local features [11], [12], [13], [14], [36] are usually generated by the convolutional neural network (CNN). The majority of the works are supervised by triplet loss, which is designed to minimise the distance between features of the positive pair of correspondences while maximise the distance between features of the negative pair. A number of variations of the triplet loss have been proposed. Tian et al. [11] proposed to maximise positive feature pairs from intermediate layers of the model alongside the pairs from the output layer. Later, they further proposed to maximise the second-order similarity of the positive feature pair, the similarity between distances of the select feature to all other features [13]. Compared with handcrafted features, learned features have shown superiority under challenging conditions. However, since features are only based on local image patches, without contextual information, they are indiscriminative to the repetitive pattern whose local information is almost identical.

Recently, there are works [8], [9], [15], [16], [17] that combine detection and description stages. Both positions of keypoints and feature vectors are produced by a single forward pass of the image through the network, rather than the detection first then followed by the description. They start with extracting a dense feature map from image using feature extractor like VGG [37] or ResNet [38], and keypoints' locations and feature vectors are produced by different modules. Dusmanu et al. [8] and Luo et al. [15] interpret the feature map not only as the feature map but also as a response map like that in DoG [5] detector. Keypoints are selected at the location where the response is maximum in both spatial and channel dimensions. Works done by Detone et al. [17] and Revaud et al. [9] append two modules after the feature extractor, one to predict keypoints location and one to generate feature respectively. The advantage of this line of work is that features are encoded with not only local information but also contextual information. Nonetheless, the pipeline may still

suffer from the missing detection problem as matching is still confined within detected keypoints.

Detector-free dense correspondence matching has drawn more and more attention in the last few years. The model is required to find the correspondence for every pixel in the query image. Similar to combined detection and description methods, the model normally starts with a feature extractor to generate feature maps of both images, followed by the dense matching module. Since matching space is extended to the entire target image, the missing detection issue is trivial under this pipeline, with the trade off being the speed of the model. Some works focus specifically on geometric matching problem. Germain et al. [39] computed and merged a collection of matching score maps from outputs of different layers of the feature backbone. Wang et al. [40] proposed a coarse-to-fine architecture to find correspondences progressively and used epipolar lines as a part of the supervision. Wiles et al. [10] generated features conditioned on target images using the attention mechanism [41] to increase the accuracy of the matching. Very recently, inspired by the success of employing the transformer [41] on image classification and object detection [42], [43], [44], several works [45], [46] that apply the transformer to the correspondence task have been proposed. They vectorize the feature map, send the sequence to transformer and aim to capture the relationship between features. The results of these works are very promising, but the computing expense of the transformer forbids the use of high resolution feature maps. Although the final positions of the correspondences can be fine-tuned by recursively applying the model, such strategy would induce a very long computing time.

There are also works which focus on semantic matching. Han et al. [18] incorporated Probabilistic Hough Matching into their trainable model to vote on extracted patch pairs, based on their geometric offsets and appearance similarity. Min et al. [20] employed a similar idea, but improved it with learned geometric offset from 6D convolution, greatly increasing the accuracy. Liu et al. [19] computed a primary correlation map from feature maps and formulated problems as an optimal transport problem. Rocco et al. [22] instead appended a regressor after the correlation map to estimate the parameters of a geometric transformation. Inspired by RANSAC, they improved the pipeline in [23] by adding a soft-inlier counting module to reduce the impact the outliers on the result. Seo et al. [21] used an order-aware attentive module to predict the global transformation.

Our DualRC and its variants belong to the line of detector-free dense correspondence matching methods. The most relevant work to our paper is the Neighbourhood Consensus Network [24] and its variants [25], [26], [47]. Rocco et al. [26] replaced the dense 4D correlation tensor with a sparse one constructed by online k nearest neighbour search, which greatly accelerated the model. However, the performance in semantic matching dropped significantly. Li et al. [25] used a collection of non-isotropic 4D kernels to deal with different sizes in neighbourhood areas. Although it boosts the accuracy in semantic matching, the additional computation brought by extra 4D kernels makes the use of high resolution feature maps even harder. Tinchev et al. [47] de-parameterised the neighbourhood consensus filtering modules and systematically studied the effects

of different image resolutions for geometric correspondence estimation.

III. METHOD

In this section, we elaborate on our method in detail. The architecture of our model is illustrated in Fig. 1. Our model mainly consists of three modules, namely, dual-resolution features generation (Section III-A), neighbourhood consensus (NC) module (Section III-B), and dense matching (Section III-C). The model starts with extracting coarse-resolution feature maps $\hat{\mathbf{F}}^A$, $\hat{\mathbf{F}}^B$ and fine-resolution feature maps $\hat{\mathbf{F}}^A$, $\hat{\mathbf{F}}^B$ from images $\mathbf{I}^A$, $\mathbf{I}^B$. The coarse feature maps form a 4D correlation tensor $\mathbf{C}$, which is further processed by the NC module, obtaining refined correlation tensor $\hat{\mathbf{C}}$. We present three options for NC module, which are described in detail in Section III-B. After filtering, the matching score maps $\hat{\mathbf{S}}^{AB}$ of queries are extracted from $\hat{\mathbf{C}}$ which are upsampled to reweigh the matching score maps $\hat{\mathbf{S}}^{AB}$ computed from fine-resolution feature maps, obtaining the final score map $\mathbf{S}^{AB}$. In such a way, we can obtain a high-resolution correlation map to accurately localize correspondences, without computing and filtering the expensive high-resolution 4D correlation tensor.

A. Dual Resolution Feature Generation

It has been reported in the literature [24], [26], [32] that the resolution of feature maps that constitute the 4D correlation tensor affects the accuracy. However, the high memory and computation cost of 4D tensor prevent vanilla 4D convolution from scaling to large feature maps. To avoid calculating the expensive 4D tensor of the high-resolution feature maps, we propose to use the dual-resolution feature maps, from which we can enjoy the benefits of both 4D convolution based neighbourhood consensus on coarse-resolution feature maps and more reliable matching from the fine-resolution feature maps.

In particular, we adopt a FPN-like [48] feature backbone, which is illustrated in Fig. 2. The feature extractor generates a pyramid of feature maps in forward direction. Top layers are spatially smaller but contain rich contextual information, while bottom layers are spatially larger with low-level details. The intuition of using FPN is that it fuses the contextual information from top into bottom layers in backward direction. In this way, the bottom layers can not only preserve their resolution but also contain rich high-level information, which makes them more robust for matching.

As shown in Fig. 2, feature maps firstly pass through a 1×1 convolutional kernel to be converted into a uniform feature dimension. The top layer is then upsampled to the same size as the next layer and two feature maps are added together. This process starts from coarse-resolution feature map and is repeated until reaching the fine-resolution feature map. Two feature maps are finally processed by 3×3 convolutional kernels for smoothing. Such a structure gives great flexibility. In principle, we can select any layer as the coarse or fine resolution feature map and final feature maps can have any feature dimension.

B. Enforcing Neighbourhood Consensus

Enforcing neighbourhood consensus constraint by 4D convolutions is firstly proposed by [24]. It has shown promising results

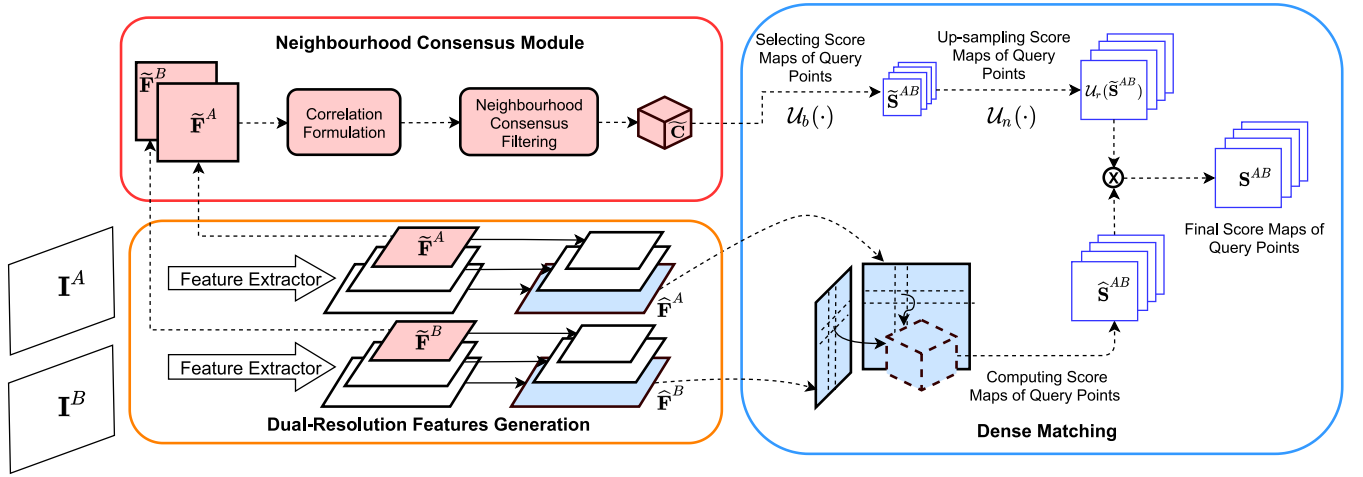


Fig. 1. Architecture of our model. It consists of three major parts: dual-resolution features generation; neighbourhood consensus module and dense matching. Firstly, feature maps at two levels of resolution are generated. The coarse resolution feature maps form 4D correlation tensor which is filtered by neighbourhood consensus module. We present three options for NC module, which are illustrated in Fig. 3. After the filtering, the matching score maps of queries are extracted from filtered 4D tensor, upsampled and multiplied with the score maps computed from fine resolution feature maps, obtaining the final score maps.

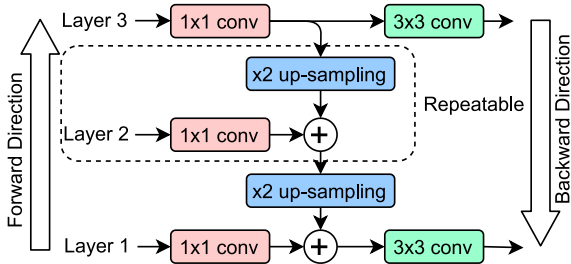


Fig. 2. Architecture of dual resolution feature generation. Features are converted to a uniform dimension by 1×1 convolutional kernel. The top feature map is upsampled and added with the bottom feature map. Finally, both feature maps are processed by 3×3 convolutional kernels for smoothing.

in both semantic and geometric tasks. By combining the basic NC module with the dual-resolution feature structure, we show that the accuracy of the correspondences can be boosted with a larger matching resolution, as demonstrated in our preliminary study [33]. Here, we present three choices for NC module. The first one is the basic NC module [24] with isotropic 4D convolutional kernels which has a balanced capability on both geometric and semantic tasks. In order to further optimize the performance in these two types of tasks, we introduce another two variants to tackle the challenges encountered in them respectively. For geometric task, which requires high-resolution input as well as fast-inference time, we present a lightweight NC module based on sparse correlation and convolution [26] to accelerate the model. For semantic task, where scale variation is prominent, we propose the DyANC module to automatically select the best neighbourhood region. We illustrate these three options in Fig. 3. We denote the model with the basic NC module as DualRC, the model with lightweight NC module as DualRC-L, and the one with DyANC module as DualRC-D. Next, we introduce each part of our framework in detail, namely, coarse correlation, mutual nearest neighbour filtering, and the three variants of NC module.

1) *Coarse Correlation Tensor*: Assume that the coarse-resolution feature maps computed by the feature backbone are $\tilde{\mathbf{F}}^A \in \mathbb{R}^{d \times h_a \times w_a}$ and $\tilde{\mathbf{F}}^B \in \mathbb{R}^{d \times h_b \times w_b}$, with $\tilde{\mathbf{f}}_{ij}^A, \tilde{\mathbf{f}}_{kl}^B \in \mathbb{R}^d$ being the features in spatial location (i, j) and (k, l) in $\tilde{\mathbf{F}}^A$ and $\tilde{\mathbf{F}}^B$ respectively. We compute the coarse 4D correlation tensor by:

$$\mathbf{C}_{ijkl} = \frac{\tilde{\mathbf{f}}_{ij}^{A\top} \tilde{\mathbf{f}}_{kl}^B}{\|\tilde{\mathbf{f}}_{ij}^A\|_2 \|\tilde{\mathbf{f}}_{kl}^B\|_2} \quad (1)$$

where $\mathbf{C} \in \mathbb{R}^{h_a \times w_a \times h_b \times w_b}$ and $\|\cdot\|_2$ represents the L_2 norm operation. Each $\mathbf{C}_{ijkl}$ in $\mathbf{C}$ is the cosine similarity between features $\tilde{\mathbf{f}}_{ij}^A$ and $\tilde{\mathbf{f}}_{kl}^B$ after L_2 normalization.

2) *Soft Mutual Nearest Neighbour Filtering*: After generating correlation tensor $\mathbf{C}$, we perform the differentiable soft mutual nearest neighbour filtering following [24]. The objective is to enforce reciprocal constraint on features. That is, if location (a, b) in image $\mathbf{I}^A$ and (c, d) in image $\mathbf{I}^B$ are true correspondence, the feature $\tilde{\mathbf{f}}_{ab}^A$ and $\tilde{\mathbf{f}}_{cd}^B$ should be their mutual nearest neighbour. Mathematically, this can be expressed as:

$$(\tilde{\mathbf{f}}_{ab}^A, \tilde{\mathbf{f}}_{cd}^B) \text{ M.N.N.} \iff \begin{cases} (a, b) = \operatorname{argmax}_{ij} \|\tilde{\mathbf{f}}_{ij}^A - \tilde{\mathbf{f}}_{cd}^B\| \\ (c, d) = \operatorname{argmax}_{kl} \|\tilde{\mathbf{f}}_{ab}^A - \tilde{\mathbf{f}}_{kl}^B\| \end{cases} \quad (2)$$

However, such (hard) mutual nearest neighbour filtering may result in excessive sparsity in the correlation tensor by eliminating majority of the correlation scores. Therefore, soft mutual nearest neighbour filtering $\hat{\mathbf{C}} = \mathcal{M}(\mathbf{C})$ is applied instead with:

$$\hat{\mathbf{C}}_{ijkl} = r_{ijkl}^A r_{ijkl}^B \mathbf{C}_{ijkl} \quad (3)$$

where

$$r_{ijkl}^A = \frac{\mathbf{C}_{ijkl}}{\max_{ab} \mathbf{C}_{abkl}} \text{ and } r_{ijkl}^B = \frac{\mathbf{C}_{ijkl}}{\max_{cd} \mathbf{C}_{ijcd}} \quad (4)$$

If (i, j) and (k, l) are in fact the mutual nearest neighbour to each other, both r_{ijkl}^A and r_{ijkl}^B would be 1 and the original correlation score would be preserved. Otherwise, either r_{ijkl}^A or r_{ijkl}^B would be low and consequently the correlation score is

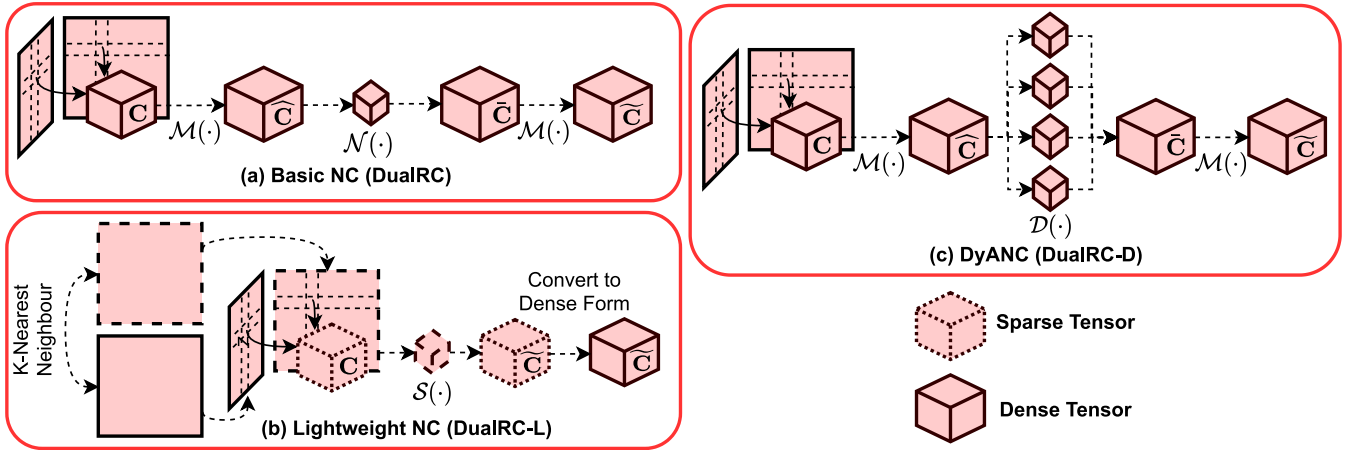


Fig. 3. Three options for neighbourhood consensus module. The first one is the basic neighbourhood consensus module [24] which works on both semantic and geometric matching. The second is a lightweight NC module based on sparse 4D correlation [26] which is to accelerate the model in the case of high resolution input images in geometric matching. The third one is our newly proposed DyANC module to tackle scale variation in semantic matching.

suppressed. Such an operation can be regarded as a filter which alleviates the impact of the incorrect matches in the correlation tensor. It is applied to the 4D tensor before and after the NC module.

3) *Basic NC Module*: The DualRC model uses the basic NC module which consists of a sequence of isotropic 4D convolutional kernels to filter the correlation tensor. The intuition is that if two features are correctly matched, the features themselves and their surrounding neighbourhood should have high consistency and subsequently the high responses in the 4D correlation tensor. Therefore, 4D convolutions are applied to learn such consistency from training data. Let $\mathcal{N}(\cdot)$ denote the basic NC module, which consists of a sequence of 4D kernels (we refer readers to [24] for the mathematical details of $\mathcal{N}(\cdot)$). The filtered correlation tensor is then $\mathcal{N}(\hat{\mathbf{C}})$. In order to make the model invariant to the order of the input image pair, we follow [24] and apply the NC module symmetrically:

$$\bar{\mathbf{C}} = \mathcal{N}(\hat{\mathbf{C}}) + (\mathcal{N}(\hat{\mathbf{C}}^\top))^\top \quad (5)$$

where $\top$ denotes swapping matching direction following the notation of [24], i.e., $\hat{\mathbf{C}}_{ijkl}^\top = \hat{\mathbf{C}}_{klij}$. The final tensor $\bar{\mathbf{C}}$ is obtained by applying another mutual nearest neighbour filtering $\bar{\mathbf{C}} = \mathcal{M}(\bar{\mathbf{C}})$.

4) *Lightweight NC Module*: Although the dual-resolution feature generation increases the matching resolution without the significant extra cost, we find that the speed of the pipeline is still somewhat unsatisfying for high resolution input images. It is because that coarse-resolution correlation tensors for high-resolution images are already computationally expensive to be processed. This problem is particularly significant in geometric matching which relies on high-resolution inputs for higher accuracy. Therefore, it is desired to have a more efficient model for geometric matching while having strong performance. To do so, instead of applying the basic NC module, we introduce a computationally efficient neighbourhood consensus module, which draws inspiration from [26]. Particularly, we replace the dense correlation tensor with a sparse one, which is then

filtered by sparse 4D convolutions [49]. The soft mutual nearest neighbour filtering is also dropped to improve efficiency.

Concretely, instead of maintaining scores for all possible feature pairs with a dense 4D correlation tensor, for each $\tilde{\mathbf{f}}_{ij}^A$ in source image, we only keep the correlation scores of its k nearest neighbours in $\tilde{\mathbf{F}}^B$, forming a sparse correlation tensor $\mathbf{C}^{AB}$:

$$\mathbf{C}_{ijkl}^{AB} = \begin{cases} \tilde{\mathbf{f}}_{ij}^{A\top} \tilde{\mathbf{f}}_{kl}^B & \text{if } \tilde{\mathbf{f}}_{kl}^B \text{ is within } k \text{ N.N. of } \tilde{\mathbf{f}}_{ij}^A \\ 0 & \text{otherwise} \end{cases} \quad (6)$$

Similarly, we can obtain $\mathbf{C}^{BA}$ by swapping the matching direction. The correlation tensor considering both matching direction is then given by:

$$\mathbf{C} = \mathbf{C}^{AB} + \mathbf{C}^{BA}. \quad (7)$$

Maintaining such a sparse 4D tensor $\mathbf{C}$ requires much less memory footprint than its dense counterpart. The sparse tensor $\mathbf{C}$ is then filtered by sparse 4D convolution filters $\mathcal{S}(\cdot)$ with a similar symmetric arrangement to ensure the order-invariant nature to the input image pair:

$$\tilde{\mathbf{C}} = \mathcal{S}(\mathbf{C}) + (\mathcal{S}(\mathbf{C}^\top))^\top. \quad (8)$$

5) *Dynamically Adaptive NC Module*: The DualRC model uses isotropic 4D convolutional kernels, such as $3 \times 3 \times 3 \times 3$ or $5 \times 5 \times 5 \times 5$ which have been proven to be effective in filtering out incorrect matches. Li et al. [25] pointed out that isotropic kernels fail to adapt to scale variation across two images, a problem commonly exists in semantic tasks. They instead proposed to use a collection of both isotropic and non-isotropic kernels, kernels like $3 \times 3 \times 5 \times 5$, to tackle such an issue. Although this improves the accuracy, they directly combine the outputs of different kernels into a single 4D tensor, and each output is treated equally. We argue such a combination is sub-optimal since the scale variation changes across different pixel locations, i.e., cells in the 4D correlation map may have largely different scale changes. Therefore, it is desired to allow the model to dynamically adopt the most suitable combination of different kernels for each cell automatically, such that each

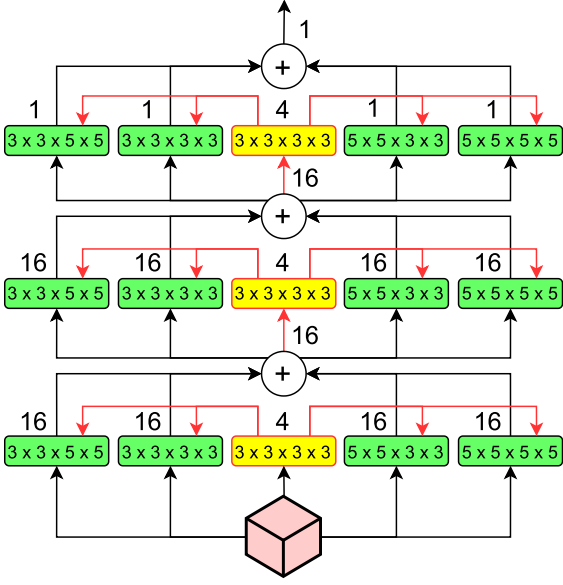


Fig. 4. Example of the dynamically adaptive NC module. The example consists of two hidden layers with 16 channels.

pair of locations in the two images can be treated with the most suitable isotropic or non-isotropic kernels. To this end, we develop the dynamically adaptive neighbourhood consensus (DyANC) module for semantic tasks.

Our DyANC module $\mathcal{D}(\cdot)$ consists of four different 4D kernels as the basic components, $3 \times 3 \times 3 \times 3$, $3 \times 3 \times 5 \times 5$, $5 \times 5 \times 3 \times 3$ and $5 \times 5 \times 5 \times 5$ to account for different possible neighbourhood variations. To allow the model to automatically choose the most suitable combination of kernels, we additionally define another $3 \times 3 \times 3 \times 3$ kernel to predict four values for each cell in the 4D tensor. These values are used to weigh the outputs after applying the four different isotropic and non-isotropic kernels. Let $\mathcal{N}_i^l(\cdot)$ be the 4D convolutional operation at l -th layer with i -th kernel, where $i = 1, 2, \dots, 4$ corresponds to the four different kernels, and $\mathcal{W}^l(\cdot)$ be the weight-learning kernel operation at layer l . The output $\hat{\mathbf{C}}^l$ at layer l can be obtained by:

$$\hat{\mathbf{C}}^l = \sum_{i=1}^4 \mathbf{W}_i^l \mathcal{N}_i^l(\hat{\mathbf{C}}^{(l-1)}), \quad (9)$$

where $\mathbf{W}^l = \text{softmax}(\mathcal{W}^l(\hat{\mathbf{C}}^{(l-1)}))$, $\mathbf{W}_i^l$ is the i^{th} channel of $\mathbf{W}^l$. The softmax operation is applied along the channel dimension to normalize the 4 weight values for each cell in the 4D tensor. We illustrate an example of $\mathcal{D}(\cdot)$ in Fig. 4. Again, we apply our DyANC module symmetrically to make our model invariant to the order of the input image pair:

$$\tilde{\mathbf{C}} = \mathcal{D}(\hat{\mathbf{C}}) + (\mathcal{D}(\hat{\mathbf{C}}^\top))^\top. \quad (10)$$

C. Dense Matching

The intuition of our dual-resolution structure is to use the result of NC module to refine the correlation scores computed from fine-resolution feature maps. This allows us to establish correspondences at a fine-resolution level while considering neighbourhood consensus. Since we do not apply NC module

in the fine level, the computing of the expensive fine-resolution 4D correlation tensor is avoided. Given a query point, we only need to compute the 2D matching score map for it, and reweigh the score map by the output of NC module.

Given any coordinate (a, b) on $\hat{\mathbf{F}}^A$, we can compute a correlation score w.r.t any location (k, l) on $\hat{\mathbf{F}}^B$ using:

$$\hat{\mathbf{S}}_{ab,kl} = \frac{\hat{\mathbf{f}}_{ab}^A \hat{\mathbf{f}}_{kl}^B}{\|\hat{\mathbf{f}}_{ab}^A\|_2 \|\hat{\mathbf{f}}_{kl}^B\|_2}, \quad (11)$$

where $\hat{\mathbf{f}}_{ab}^A$ and $\hat{\mathbf{f}}_{kl}^B$ are features at the spatial locations (a, b) and (k, l) on $\hat{\mathbf{F}}^A$ and $\hat{\mathbf{F}}^B$, respectively. After obtaining $\hat{\mathbf{S}}_{ab,kl}$ for all (k, l) , we have the fine-resolution correlation map $\hat{\mathbf{S}}_{ab} \in \mathbb{R}^{H_b \times W_b}$. Next, the coordinate (a, b) is scaled down to the resolution of the coarse feature map, obtaining the corresponding coordinate (a', b') . Note that a', b' are float numbers rather than integers hence we use bilinear interpolation $\mathcal{B}(\cdot)$ to extract the coarse-resolution correlation map $\tilde{\mathbf{S}}_{a'b'} \in \mathbb{R}^{h_b \times w_b}$ from coarse correlation tensor $\tilde{\mathbf{C}}$:

$$\tilde{\mathbf{S}}_{a'b',k'l'} = \mathcal{B}(\tilde{\mathbf{C}}_{\lceil b' \rceil \lceil a' \rceil k'l'}, \tilde{\mathbf{C}}_{\lfloor b' \rfloor \lfloor a' \rfloor k'l'}), \quad (12)$$

where $\lceil \cdot \rceil$ and $\lfloor \cdot \rfloor$ denote the ceiling and floor operation respectively. $\tilde{\mathbf{S}}_{a'b'}$ is then upsampled to the same size as the high-resolution correlation map $\hat{\mathbf{S}}_{ab}$ using the nearest neighbour upsampling method, obtaining the final correlation map $\mathbf{S}_{ab} \in \mathbb{R}^{H_b \times W_b}$:

$$\mathbf{S}_{ab} = \mathcal{U}(\tilde{\mathbf{S}}_{a'b'}) \odot \hat{\mathbf{S}}_{ab}, \quad (13)$$

where $\mathcal{U}(\cdot)$ denotes as nearest neighbour upsampling operation and $\odot$ represents the element-wise multiplication between two tensors. For each point (a, b) on $\hat{\mathbf{F}}^A$, its correspondence (c, d) on $\hat{\mathbf{F}}^B$ is then retrieved by:

$$(c, d) = \text{argmax}_{kl} \mathbf{S}_{ab,kl}. \quad (14)$$

This allows us to obtain dense correspondences from source to target or target to source.

At inference time, the model may be provided with a set of query points. If no query points is provided, we can use the coarse-resolution feature maps to ignore the unreliable query points.

Inference With Given Queries: Let $\{(x_i, y_i)\}$ be the set of given queries in the image. They are firstly scaled down to the spatial resolution of fine-resolution features $\hat{\mathbf{F}}^A$ and $\hat{\mathbf{F}}^B$, denoted as $\{(\hat{x}_i, \hat{y}_i)\}$, which are float numbers. One can choose to round the coordinates to their nearest integer neighbours and use the nearest neighbours as the queries. However, this will induce undesirable error and reduce the accuracy of the estimated correspondences. Therefore, for a given query $(\hat{x}_i, \hat{y}_i)$ we instead find its four nearest integer neighbours $(\hat{x}_j^n, \hat{y}_j^n)$, $\forall j = 1, \dots, 4$ and their correspondences $(\hat{x}_j^{n*}, \hat{y}_j^{n*})$ by (14). Finally, we use bilinear interpolation to obtain the correspondence of $(\hat{x}_i, \hat{y}_i)$ by:

$$(\hat{x}_i^*, \hat{y}_i^*) = \frac{\sum_{j=1}^4 q_j (\hat{x}_j^{n*}, \hat{y}_j^{n*})}{\sum_{j=1}^4 q_j}, \quad (15)$$

where $q_j = |\hat{x}_i - \hat{x}_j^n| |\hat{y}_i - \hat{y}_j^n|$.

Inference Without Given Queries: If no query is provided, we treat all integer coordinates $\{(\hat{x}, \hat{y})\}, \forall \hat{x} = 0, \dots, W_a - 1, \forall \hat{y} = 0, \dots, H_a - 1$ on the fine-resolution feature map $\hat{\mathbf{F}}^A$ of the query image as queries and find their correspondences $\{(\hat{x}^*, \hat{y}^*)\}$. In order to reduce the impact of incorrect matches to the downstream task, we impose a cyclic consistency on the matches to filter out incorrect pairs. We treat $\{(\hat{x}^*, \hat{y}^*)\}$ as queries and find their correspondences $\{(\hat{x}^{**}, \hat{y}^{**})\}$ by swapping the order of query and target images and obtain two sets of matching pairs $\{((\hat{x}, \hat{y}), (\hat{x}^*, \hat{y}^*))\}$ and $\{((\hat{x}^{**}, \hat{y}^{**}), (\hat{x}^*, \hat{y}^*))\}$. We pick the intersection between $\{((\hat{x}, \hat{y}), (\hat{x}^*, \hat{y}^*))\}$ and $\{((\hat{x}^{**}, \hat{y}^{**}), (\hat{x}^*, \hat{y}^*))\}$ as the result of the matching. However, querying the entire set of integer coordinates in the fine-resolution is time-consuming. Thanks to the subtle design of our dual-resolution structure, we can significantly accelerate the process. Firstly, we establish a set of matches for all integer coordinates in coarse-resolution feature map $\tilde{\mathbf{F}}^A$, denoted as $\{((\tilde{x}, \tilde{y}), (\tilde{x}^*, \tilde{y}^*)), \forall \tilde{x} = 0, \dots, w_a - 1, \forall \tilde{y} = 0, \dots, h_a - 1\}$, which can be directly retrieved from $\tilde{\mathbf{C}}$ by $(\tilde{x}^*, \tilde{y}^*) = \text{argmax}_{kl} \tilde{\mathbf{C}}_{\tilde{x}\tilde{y}kl}$. We then sort the correlation scores of these matches in descending order and select top 50% of them. Since each spatial location on $\tilde{\mathbf{F}}^A$ corresponds to a $r \times r$ region on $\hat{\mathbf{F}}^A$, where r is the spatial ratio between $\tilde{\mathbf{F}}^A$ and $\hat{\mathbf{F}}^A$, we treat all integer coordinates in $\tilde{\mathbf{F}}^A$ that lie in the corresponding local region of the selected top 50% in $\{((\tilde{x}, \tilde{y}), (\tilde{x}^*, \tilde{y}^*))\}$ as queries and then follow the aforementioned matching procedure using the cyclic consistency. This effectively reduces the number of queries by half and increases inference speed.

D. Training Loss

Due to the absence of dense keypoint annotations, existing methods normally use either image-level pairwise annotations [24], [26], [28] or sparse keypoint annotations [25] for training. We follow [25] to adopt the training loss based on sparse keypoint annotations.

Given a set of ground-truth correspondence pairs $\{((x_i, y_i), (x_i^*, y_i^*))\}_{i=1}^N$ in the image resolution, we convert them to the scale of the fine-resolution feature map. We then obtain the 2D correlation map for each query following (13) and generate the corresponding ground-truth 2D correlation map. As the coordinates are float numbers in the fine-resolution, we dilute the one-hot value to its four nearest neighbours in the ground-truth 2D map and apply a Gaussian kernel to smooth the label. By stacking and flattening all the N correlation maps, we can form the predicted and ground-truth assignment matrices $\mathbf{M}^{AB}, \mathbf{M}_{gt}^{AB} \in \mathbb{R}^{N \times T}$ where $T = H_b \times W_b$. Each row in the two matrices is a flattened 2D correlation map for a given query point. Considering both $A \rightarrow B$ and $B \rightarrow A$ matching directions, the loss is defined as:

$$\mathcal{L}_k = \|\mathbf{M}^{AB} - \mathbf{M}_{gt}^{AB}\|_F + \|\mathbf{M}^{BA} - \mathbf{M}_{gt}^{BA}\|_F, \quad (16)$$

where $\|\cdot\|_F$ denotes the Frobenius norm of matrix. Besides, we include the orthogonal loss which we proposed in [25] to enforces one-to-one matching constraint and use it as a regularization term. If $\mathbf{M}$ is a perfect assignment matrix, which means that each query is assigned to only one target and no double assignment, then $\mathbf{M}\mathbf{M}^\top$ would be an identity matrix. As such,

the orthogonal loss is defined as:

$$\begin{aligned} \mathcal{L}_o = & \|\mathbf{M}^{AB}\mathbf{M}^{AB\top} - \mathbf{M}_{gt}^{AB}\mathbf{M}_{gt}^{AB\top}\|_F \\ & + \|\mathbf{M}^{BA}\mathbf{M}^{BA\top} - \mathbf{M}_{gt}^{BA}\mathbf{M}_{gt}^{BA\top}\|_F. \end{aligned} \quad (17)$$

As we dilute the ground-truth correlation map, $\mathbf{M}_{gt}^{AB}$ is not a perfect assignment matrix and $\mathbf{M}_{gt}^{AB}\mathbf{M}_{gt}^{AB\top}$ is a diagonally dominant matrix rather than an identity matrix. The over all training loss can then be written as

$$\mathcal{L} = \mathcal{L}_k + \lambda \mathcal{L}_o, \quad (18)$$

where λ is a weight term.

IV. EXPERIMENTS

A. Implementation Details

Our model is implemented in Pytorch [50]. We use ResNet101 [38] pretrained on ImageNet [51] as the feature extractor and truncate the modules after the layer conv4_23. The feature extractor is frozen during training. For the dual-resolution feature map generation module, we select the output of the layer conv4_23 of the feature extractor as the coarse-resolution feature map, hence the coarse-resolution feature map is 1/16 of the input image size. Since geometric matching requires a high matching resolution to achieve good accuracy, we select the output of the layer conv3_4 (1/8 of the input image size) as the fine-resolution feature map for semantic matching but the output of the layer conv2_3 (1/4 of the input image size) as the fine-resolution feature map for geometric matching. All feature maps are converted to 1024 channels by 1×1 convolution in the dual-resolution feature generation module.

Following [24], [26], the geometric and semantic tasks use two different configurations of the NC module. For the geometric task, the DualRC model consists of two layers of 4D kernels $3 \times 3 \times 3 \times 3$ with channels of $\{16, 1\}$. The DualRC-L and DualRC-D follows the same configuration. For the semantic task, the DualRC consists of three layers of 4D kernels $5 \times 5 \times 5 \times 5$ with channels of $\{16, 16, 1\}$. The DualRC-L and DualRC-D adopt this configuration as well. The different choices are mainly attributed to two facts. Firstly, semantic matching requires a larger neighbourhood area and deeper structure to capture the good semantic understanding of the scene due to the significant intra-category appearance change. Secondly, geometric matching requires images with higher resolution for higher accuracy, hence the smaller kernel is favoured for its low memory requirement and faster speed. The k is set to 10 in the DualRC-L.

Our code can be found at <https://code.active.vision>.

B. Geometric Matching

1) *Training Data:* We evaluate our model on three geometric datasets: HPatches, InLoc and Aachen Day-Night. There is no training or testing data split in these three datasets, hence we follow [8] and train our model on MegaDepth [52] dataset. It consists of a large number of internet images about 196 scenes with corresponding sparse 3D point clouds constructed by COLMAP [1], [2]. The camera intrinsics and extrinsics

together with the depth maps are also included. We also follow the procedure in [8] to generate sparse ground truth labels. First, we compare the overlap in sparse SfM point cloud between all image pairs to select the pairs whose overlap is over 50%. Next, for all selected pairs, the second image with depth information is projected into the first image and occlusion is removed by depth check. Then, we randomly collect 128 correspondences from each image pair to train our model. We use the scenes with more than 500 valid image pairs for training and the rest scenes for validation. To avoid scene bias, 110 image pairs are randomly selected from each training scene to constitute our training set. In total, we obtain 15,070 training pairs and 14,638 validation pairs. All of three models are trained with Adam optimizer [53] for 15 epoch with initial learning rate of 0.001. The learning rate is halved for every 5 epochs. The coefficient to orthogonal loss is set to be 0.05. The images are resized to 400×400 during training.

2) *HPatches*: HPatches dataset [29] is designed to evaluate the accuracy of the correspondences under viewpoint and illumination changes. It contains 108 image sequences. 56 of them undergo a viewpoint change and 52 of them undergo an illumination change. Each sequence consists of one reference image and 5 query images. Each query image pairs with the reference image hence five image pairs are obtained for each sequence. A homography w.r.t. to the reference image is provided for each query image. We evaluate our method with two metrics for this dataset. The first one is Mean Matching Accuracy (MMA). This metric projects the keypoints $\mathbf{p}_i^A$ in the query image to the target image using the ground-truth homography $\mathcal{H}^*$, and calculates the percentage of predicted correspondences $\mathbf{p}_i^B$ in the target image whose distances to the ground-truth lies within a threshold t . Mathematically, it is defined as:

$$\text{MMA}(\{\mathbf{p}_i^A, \mathbf{p}_i^B\}_{i=1}^N; t) = \frac{1}{N} \sum_{i=1}^N \mathbf{1}(t - \|\mathcal{H}^*(\mathbf{p}_i^A) - \mathbf{p}_i^B\|_2), \quad (19)$$

where t is the threshold of 2D distance. $\mathbf{1}(\cdot)$ is a binary indicator function whose output is 1 for a non-negative value and 0 otherwise. $\mathcal{H}^*(\cdot)$ denotes the warping by homography. MMA is calculated for each image pair and is averaged over the entire dataset. The second metric is the corner projection error (CPE) [17] to measure the accuracy of the correspondence by homography estimation. We use OpenCV to calculate the homography $\mathcal{H}$ between two images based on the found correspondences. Then four corners $\{c_1, c_2, c_3, c_4\}$ of the query image are projected using both the predicted homography $\mathcal{H}$ and the ground-truth homography $\mathcal{H}^*$, obtaining $\{\hat{c}_1, \hat{c}_2, \hat{c}_3, \hat{c}_4\}$ and $\{\hat{c}_1^*, \hat{c}_2^*, \hat{c}_3^*, \hat{c}_4^*\}$. The CPE is defined as:

$$\text{CPE} = \frac{1}{4} \sum_{i=1}^4 \mathbf{1}(\|\epsilon - \|\hat{c}_i - \hat{c}_i^*\|_2\|) \quad (20)$$

where ϵ is a threshold. CPE is calculated individually for each image pair and is then averaged over the entire dataset. The result is reported as the average CPE under thresholds $\epsilon = 3\text{px}, 5\text{px}, 7\text{px}, 10\text{px}$.

During the evaluation, we interpolate the longer side of the input image to 1600 and keep the original aspect ratio before

TABLE I
AVERAGE MMA ON HPATCHES DATASET

Method	Average MMA			# matches
	$t = 3\text{px}$	$t = 5\text{px}$	$t = 10\text{px}$	
D2-Net [8]	44.5	66.3	83.6	1k
R2D2 [9]	73.0	81.1	84.5	1k
S.P.+S.G. [17]	64.6	72.5	78.2	1k
DELFI [16]	47.7	54.8	70.5	1k
Sparse-NCNet [26]	79.6	88.0	91.8	1k
DualRC	<u>78.8</u>	90.3	95.9	1k
DualRC-L	75.9	87.4	93.3	1k
DualRC-D	77.5	<u>89.5</u>	<u>95.7</u>	1k
NCNet [24]	62.6	79.9	90.8	2k
Sparse-NCNet [26]	78.5	86.7	90.8	2k
DualRC	78.5	90.1	95.7	2k
DualRC-L	75.3	87.1	93.0	2k
DualRC-D	<u>77.3</u>	<u>89.4</u>	<u>95.6</u>	2k

DualRC outperforms others for $t = \{5\text{px}, 10\text{px}\}$ and performs on par with Sparse-NCNet for $t = 3\text{px}$.

The bold entities indicate the best result and underscore the second best result.

TABLE II
AVERAGE CPE ON HPATCHES DATASET

Method	Average CPE			
	$\epsilon = 3\text{px}$	$\epsilon = 5\text{px}$	$\epsilon = 7\text{px}$	$\epsilon = 10\text{px}$
Sparse-NCNet [26]	49.4	57.2	62.7	68.3
DualRC	55.9	62.6	67.3	72.6
DualRC-L	56.7	<u>63.3</u>	<u>68.3</u>	<u>73.4</u>
DualRC-D	<u>56.6</u>	63.8	68.9	74.4

DualRC-L has a very closed performance to other two variants, indicating the lightweight NC module being able to achieve efficient filtering without compromising the performance.

The bold entities indicate the best result and underscore the second best result.

passing them through the network. The established correspondences are then rescaled to the original scale. For average MMA metric, we follow the evaluation procedure in [26] that 1000 matches are extracted by all methods for a fair comparison. We also include the results of top 2000 matches between methods related to the 4D neighbourhood consensus. For average CPE (AUC), the homographies are estimated by top 1000 matches for all methods. The results on two metrics are summarized in Tables I and II respectively.

Our method outperforms the existing works in both MMA and CPE metrics, especially over various neighbourhood consensus methods. The results indicate that the basic NC module and other two variants work very well in producing robust correspondences. The results of DualRC-L are slightly lower than the other two types in average MMA metric, this is not a surprising result since the sparse correlation tensor only stores the top 10 candidates for each feature, which limits the matching area. However, from results on average CPE metric, which focuses on the overall capability of extracted correspondences on the downstream task, rather than individual accuracy of each correspondence, all of our three types have similar performance, notably outperforming Sparse-NCNet.

We also evaluate the run-time for different NC-based methods and report the results in Table III. All methods are tested on an Nvidia TITAN RTX GPU. We compare the run-time for one single forward pass through the network. The time for one forward

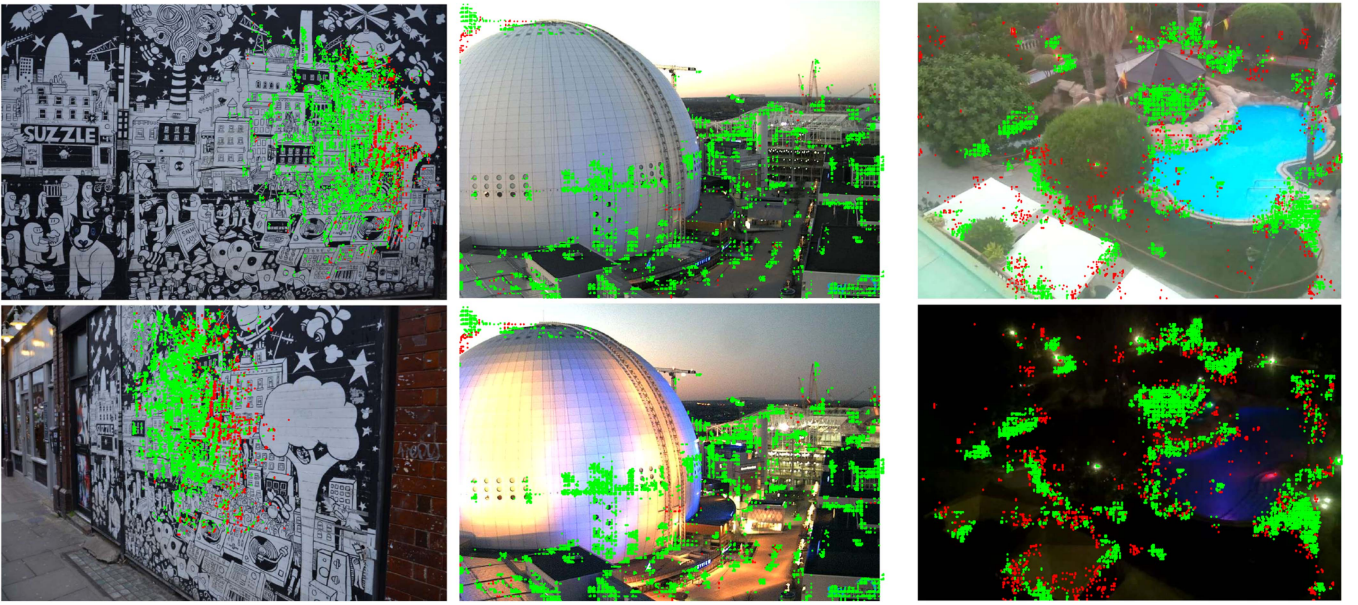


Fig. 5. Qualitative results on HPatches dataset. We select the top 6000 matches and set the projection error threshold as three pixels. Green parts are correctly localized matches while red parts are incorrect matches.

TABLE III
RUN-TIME COMPARISON

Method	Input Size px × px	Time second	MMA $t = 5\text{px}$
NCNet [24]	1600 × 1200	4.65	79.9
Sparse-NCNet [26]	3200 × 2400	1.64	<u>87.6</u>
Sparse-NCNet [26]	1600 × 1200	0.37	82.3
DualRC	1600 × 1200	3.28	90.3
DualRC-L	1600 × 1200	<u>0.47</u>	87.4

DualRC-L is almost ten times faster than DualRC but has comparable performance with DualRC. In contrast, Sparse-NCNet requires the input images to have a larger size, which in return slows down the speed of the model. The bold entities indicate the best result and underscore the second best result.

TABLE IV
RESULTS ON INLOC DATASET

Method	Distance Error			
	$\tau = 0.25\text{m}$	$\tau = 0.5\text{m}$	$\tau = 1\text{m}$	$\tau = 2\text{m}$
D2-Net [24]	43.2	61.1	74.2	-
NCNet [24]	<u>44.1</u>	63.8	76.0	78.4
Sparse-NCNet [26]	45.4	<u>66.2</u>	<u>79.9</u>	<u>82.1</u>
DualRC	<u>44.1</u>	67.5	82.4	84.8
DualRC-L	<u>44.1</u>	63.2	79.9	81.5

Our DualRC achieves the best results in distance error of 0.5m, 1m and 2m and is slightly outperformed by SparseNC-Net in 0.25.

The bold entities indicate the best result and underscore the second best result.

pass is reduced from 3.28 seconds to 0.47 s, almost 10 times faster when switching from the basic NC module to lightweight NC module, significantly outperforming Sparse-NCNet with the same input resolution and marginally being outperformed by DualRC. To maintain good performance, Sparse-NCNet requires the input images to have a larger size, which in return slows down the speed of the model. For the DualRC-D model, the model is too large to be fit into a single GPU under the image resolution of 1600×1200 , hence we evaluated it on CPU. Thus it is unfair to include it in this comparison. Qualitative results on HPatches are shown in Fig. 5.

3) *InLoc*: InLoc dataset [4] is designed to test the performance on long-term indoor relocalization. It consists of a large collection of images with depth acquired by a 3D scanner. Query images are taken by a smartphone camera a few months later. The dataset contains very challenging viewpoint and illumination changes. We follow the evaluation protocol in [4]. For each query image, the dataset provides a shortlist of 10 candidate database images. Our method is then used to establish the correspondence between the query image and the shortlisted

image. Finally the P_nP solver is used to estimate the pose of the query image. Similar to the evaluation on the HPatches dataset, we interpolate the longer side of the input images to 1600 and keep the same aspect ratio during the evaluation, and scale the correspondences back to original scale afterward. The accuracy of the relocalization is reported in the form of the percentage of the relocalized poses whose distance errors are within τ meters and angular errors are within 10° . The results are shown in Table IV. We achieve the best results among methods related to 4D neighbourhood consensus, indicating the superiority of our architecture. Several qualitative results can be found in Fig. 6.

4) *Aachen Day Night*: Aachen Day-Night dataset [30] is designed to evaluate performance on outdoor relocalization under day night illumination change. It contains 4479 d time images of the Aachen city, with 98 night query images. We use the official evaluation script provided by [30]. A shortlist of image pairs is provided and our method extracts correspondences from the pairs. Then a sparse 3D pointcloud of the scene is constructed using COLMAP [1], [2]. The poses of the query images are estimated based on reconstructed pointcloud. The accuracy is

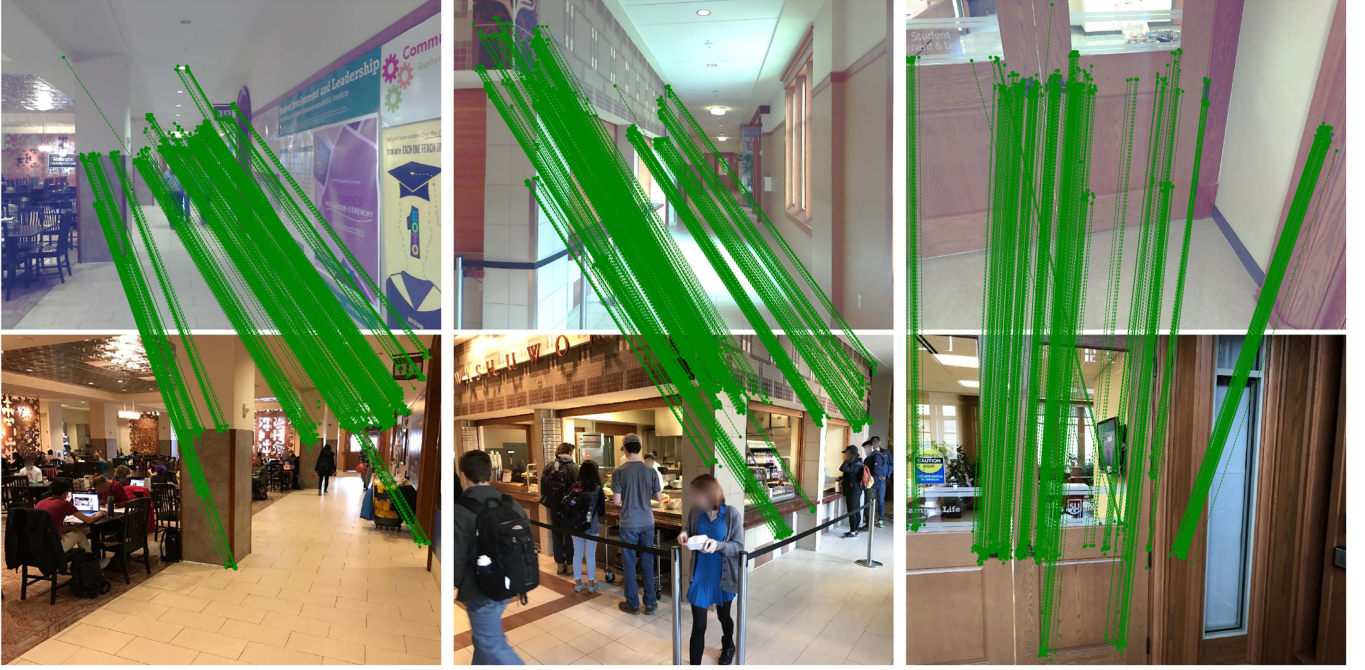


Fig. 6. Qualitative results on InLoc dataset. Top 500 matches are drawn.

TABLE V
RESULTS ON AACHEN DAY NIGHT DATASET

Method	(Distance Error, Angular Error)		
	(0.25m, 2°)	(0.5m, 5°)	(5m, 10°)
SP+SG [54]	79.6	90.8	100
ASLFeat+OANet [55]	<u>77.6</u>	<u>89.8</u>	100
Sparse-NCNet [26]	76.5	84.7	<u>98.0</u>
DualRC	79.6	88.8	100
DualRC-L	<u>77.6</u>	87.8	100

Our DualRC achieves the best results in threshold (0.5m, 2°) and (5m, 10°). The bold entities indicate the best result and underscore the second best result.

measured in the percentage of relocalized pose within the distance error τ meters and the angular error θ° . The results are presented in Table V. Our DualRC model achieves the best result in thresholds (0.5 m, 2°) and (5 m, 10°) among the existing works. Qualitative results are shown in Fig. 7.

C. Semantic Matching

We evaluate our model on three semantic matching datasets: PF-PASCAL, PF-WILLOW, and SPair-71 k datasets.

PF-PASCAL and PF-WILLOW are parts of Proposal Flow dataset [31]. PF-PASCAL contains 1351 images from 20 categories, while PF-WILLOW has 100 images equally distributed into 10 categories. Both datasets have sparse keypoint annotation. We follow the standard evaluation procedure [18], [20], [24], [25], [56] and generate 2940 training pairs, 300 validation pairs and 300 testing pairs for PF-PASCAL, as well as 900 testing pairs for PF-WILLOW. Since PF-WILLOW does not have the training data, the evaluation on both datasets uses the model trained on the PF-PASCAL dataset. The model is trained with

Adam optimizer for 20 epochs with learning rate of 0.001 and orthogonal loss weight is set to 0.005. The input image is resized to 256×256 for both training and testing on both PF-PASCAL and PF-WILLOW datasets.

SPair-71 k is a far more challenging dataset. It consists of 53340 training image pairs, 5384 validation pairs and 12234 testing pairs, spanning 18 categories with provided keypoint annotations and object bounding boxes. Compared with PF-PASCAL, SPair-71 k contains much larger viewpoint and scale variations between images, making correspondence localization very hard. We evaluate two types of model on SPair-71 k: one is fine-tuned on the training data of SPair-71 k and the other is the transferred model trained on PF-PASCAL. The model is trained with Adam optimizer with the learning rate of 0.001 for 10 epochs with orthogonal loss weight is set to 0.005. The model is trained and tested with an image size of 400×400 .

The evaluation metric adopted for both datasets is the standard percentage of corrected keypoints (PCK). Given a set of prediction and groundtruth keypoints $\mathcal{K} = \{(k_i, k_i^*)\}_{i=1}^N$, it measures the percentage of predictions whose distances to the ground-truth values lies within a certain threshold value. Mathematically, it is defined as:

$$PCK(\mathcal{K}) = \frac{1}{N} \sum_{i=1}^N \mathbf{1}(\|k_i - k_i^*\|_2 \leq \alpha \cdot \max(w, h)) \quad (21)$$

where (w, h) are the width and height of either the input image or the bounding box of the object. In our case, PF-PASCAL and PF-WILLOW use the image and SPair-71 k uses the bounding box.

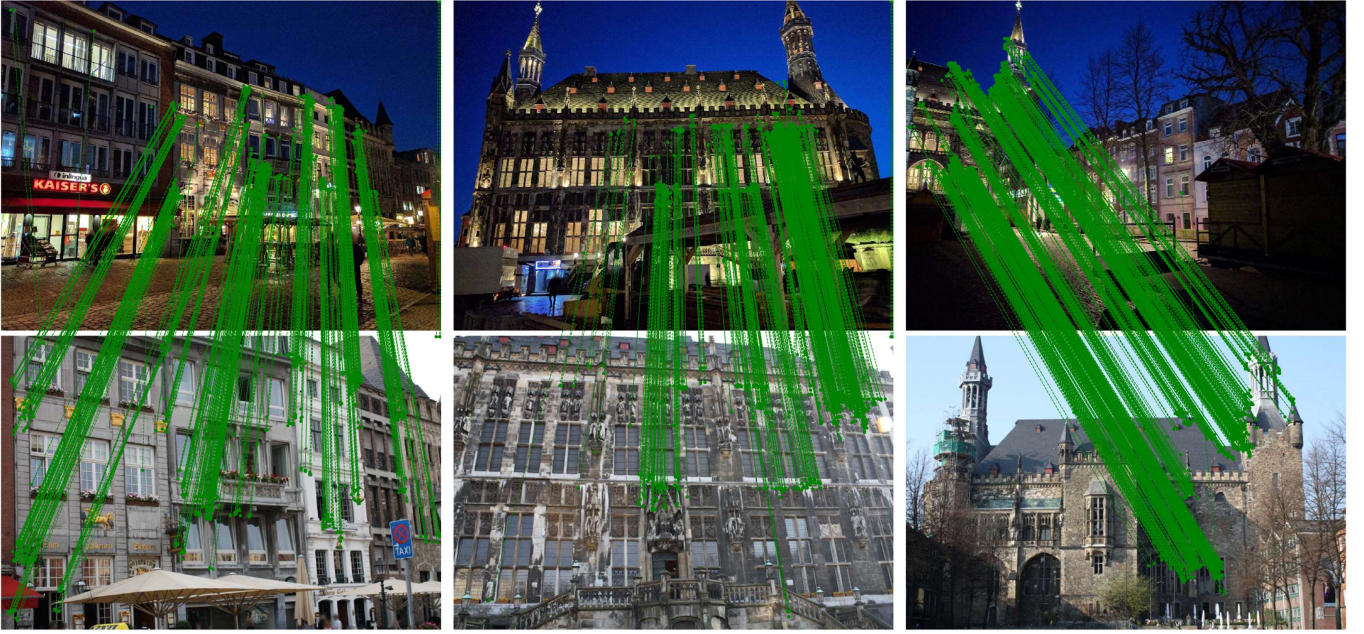


Fig. 7. Qualitative Results on Aachen Day Night Dataset. Top 500 matches are drawn. Note that our model can achieve good matching under significant illumination and viewpoint change.

TABLE VI
RESULTS FOR SEMANTIC MATCHING

	PF-PASCAL PCK($\alpha = 0.1$)	PF-WILLOW PCK($\alpha = 0.1$)	SPair-71k ($\alpha = 0.1$)	
			(Fine-tuned)	(Transferred)
UCN [56]	75.1	-	-	17.7
NCNet [24]	78.9	67.0	-	20.1
SCOT [19]	85.6	76.0	35.6	-
DCC-Net [28]	82.3	73.8	-	26.7
ANC-Net [25]	86.1	-	-	<u>28.7</u>
HPF [32]	84.8	<u>74.4</u>	28.2	-
DHPF [57]	90.7	71.0	37.3	27.4
DualRC	85.8	56.3	<u>42.0</u>	26.4
DualRC-D	<u>86.7</u>	60.0	42.9	29.4

Our DualRC-D model achieves the best results in both transferred and finetuned experiments on SPair-71k datasets and the second best result on PF-PASCAL dataset. Note that there is a clear boost in accuracy when switching from DualRC to DualRCD, indicating the effectiveness of DyANC module.

The bold entities indicate the best result and underscore the second best result.

We summarize the results in Table VI. The DualRC-D model achieves the best results in SPair-71 k datasets and the second best result in PF-PASCAL dataset, when compared with existing methods. We can see that there is a clear boost in accuracy across all datasets when switching from the basic NC module (DualRC) to the DyANC module, which demonstrates the effectiveness of DyANC module. We find that, for the same object category, the keypoints in PF-WILLOW are annotated slightly differently from the other two datasets. Meanwhile, PF-WILLOW only contains 3 categories, while PF-PASCAL and SPair-71 K contain around 20 categories. Thus, there exist distribution shifts between PF-WILLOW and other datasets. Therefore, we observe that methods performing well on PF-WILLOW do not perform as well on PF-PASCAL and SPair-71 k. Since PF-PASCAL and SPair-71 k are much larger and more challenging than PF-WILLOW, the performance of PF-PASCAL and SPair-71 k is a stronger indicator.

We have also evaluated DualRC-L and Sparse-NCNet on the semantic tasks, however, the results are unsatisfactory. We believe it is due to the fact that the semantic matching requires a greater area of neighbourhood for high-level semantic understanding. The lightweight NC module in DualRC-L model only stores top 10 matching candidates for each feature and hence the neighbourhood is very sparse with little semantic information. This consequently leads to the poor performance in semantic tasks.

To verify whether our DyANC module is able to learn to choose proper neighbourhood sizes, we visualize weights of 4D kernel components and demonstrate them in Fig. 8. In these two examples, the first pair has small scale variation and the second one has significant scale variation. From the weight learned by the input layer, it is clear that the non-isotropic kernel $3 \times 3 \times 5 \times 5$ of the pair with significant scale variation has a greater response than that of the pair with less scale variation. This clearly indicates that DyANC module successfully captures the scale variation between objects and automatically selects the suitable neighbourhood region. Qualitative results on three semantic datasets are presented in Fig. 9.

D. More Discussion and Comparison

Correspondence estimation is a fundamental problem and the field advances very fast. During the preparation of this paper, a few new works appear and show better performance than our DualRC in either the geometric or semantic correspondence estimation task. Particularly, for the geometric task, LoFTR [46], a transformer based method sharing a similar dual-resolution design as our DualRC, achieves the current state-of-the-art performance. Meanwhile, for the semantic task, CHM [20], which

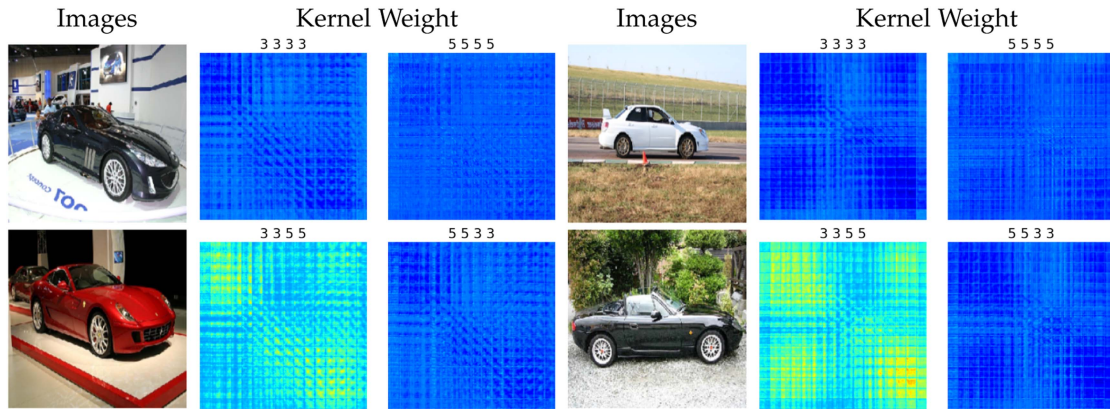


Fig. 8. Visualization of learned 4D Kernels. The weights of the input layer are drawn. It is clear that the non-isotropic kernel $3 \times 3 \times 5 \times 5$ have a greater response when scale variation between two images is greater.

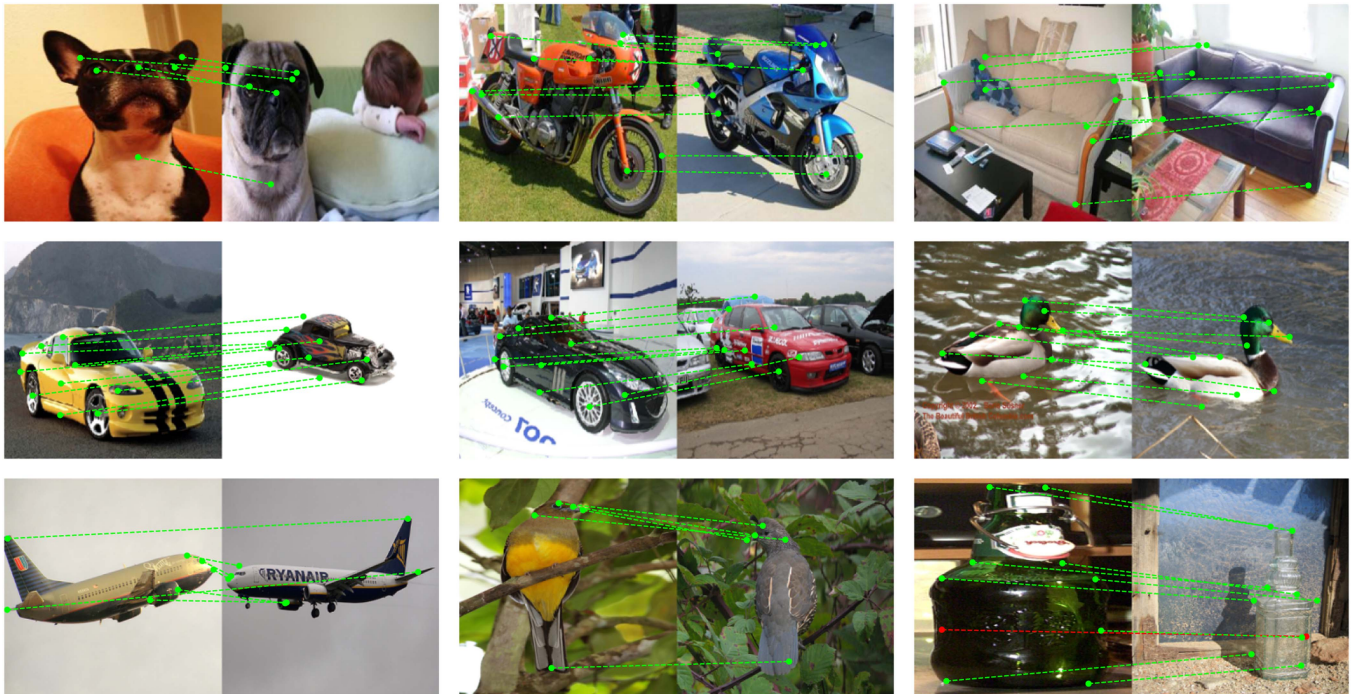


Fig. 9. Qualitative results on semantic matching datasets. From top to bottom: PF-PASCAL, PF-WILLOW (middle), SPair-71 k.

introduces 6D convolution kernels to construct deep Hough matching networks, and CATs [58] and its followup [59], which introduce cost aggregation transformers for matching, achieve the state-of-the-art performance. Our DualRC is designed for general visual correspondence estimation including both geometric and semantic tasks, while the aforementioned methods are developed with a specific focus either on geometric or semantic task. Thus, to have a full picture of how these methods work on the general visual correspondence estimation problem, we retrain LoFTR on PF-PASCAL to measure its performance on the semantic task, and CHM and CATs on MegaDepth to measure their performance on the geometric task. The experimental details are provided in the supplementary material. We found that DualRC notably outperforms LoFTR on the semantic task (PCK w/ $\alpha = 0.1$ on PF-PASCAL: 85.8 vs 63.1). CHM and

CATs struggle on the geometric task and are significantly outperformed by DualRC (Table II in the supp.), partly due to their huge memory consumption, prohibiting the models to process images of a large resolution (e.g., 1600×1600), which is critical for geometric matching. Our method outperforms CHM and CATs on all resolutions that can be handled under the same hardware. For LoFTR, we further evaluate it on the PhotoTourism dataset of Image Matching Challenge [60] and our method achieves overall on-par performance with the LoFTR when both methods are trained under the same hardware resource (Table I in the supp.). In contrast to these recent advances, which show state-of-the-art performance on either of the two tasks, our DualRC consistently achieves promising results on both tasks. Detailed comparison on performance and memory consumption can be found in the supplementary material.

V. CONCLUSION

In this paper, we propose a dense matching model which can be easily adapted to both semantic and geometric matching. We propose a model named DualRC that consists of a dual-resolution structure to establish correspondences in a coarse-to-fine manner. The coarse-resolution feature maps form a coarse-correlation tensor which is filtered by the neighbourhood consensus (NC) module. Besides, we propose two NC variants to tackle the specific demands for geometric and semantic matching respectively. To accelerate the model in the presence of high-resolution input images in geometric matching, we present DualRC-L which has a lightweight NC module based on sparse correlation [26] and convolution to run efficient filtering. For semantic matching where scale variation is predominant, we propose DualRC-D with the novel DyANC module to select the best neighbourhood region during filtering. The output of the filtering is then used to regularize the matching using fine-resolution feature maps. Our model is evaluated on both semantic matching datasets as well as geometric datasets. In all cases, our method achieves promising results, outperforming all other NC based methods. Although dense matching avoids missing-detection problem, it is slow when applied to large image pairs since all pixels are needed to be treated as queries. Naively prepending a keypoint detector instead may results in the incompatibility between keypoint detector and dense matcher, which degrades the performance. Therefore, one possible future direction is to combining keypoint detection and matching module, leveraging the advantages from both sides.

REFERENCES

- [1] J. L. Schönberger and J.-M. Frahm, "Structure-from-motion revisited," in *Proc. IEEE Intl. Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 4104–4113.
- [2] J. L. Schönberger, E. Zheng, M. Pollefeys, and J.-M. Frahm, "Pixelwise view selection for unstructured multi-view stereo," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 501–518.
- [3] P.-E. Sarlin, C. Cadena, R. Siegwart, and M. Dymczyk, "From coarse to fine: Robust hierarchical localization at large scale," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 12708–12717.
- [4] H. Taira et al., "InLoc: Indoor visual localization with dense matching and view synthesis," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 1293–1307.
- [5] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, pp. 91–110, 2004.
- [6] H. Bay, T. Tuytelaars, and L. V. Gool, "SURF: Speeded up robust features," in *Proc. Eur. Conf. Comput. Vis.*, 2006, pp. 404–417.
- [7] S. Leutenegger, M. Chli, and R. Y. Siegwart, "BRISK: Binary robust invariant scalable keypoints," in *Proc. Int. Conf. Comput. Vis.*, 2011, pp. 2548–2555.
- [8] M. Dusmanu et al., "D2-Net: A trainable CNN for joint detection and description of local features," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 8084–8093.
- [9] J. Revaud et al., "R2D2: Repeatable and reliable detector and descriptor," in *Proc. Conf. Neural Inf. Process. Syst.*, 2019, pp. 12414–12424.
- [10] O. Wiles, S. Ehrhardt, and A. Zisserman, "Co-attention for conditioned image matching," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 15915–15924.
- [11] Y. Tian, B. Fan, and F. Wu, "L2-Net: Deep learning of discriminative patch descriptor in Euclidean space," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 6128–6136.
- [12] Y. Ono, E. Trulls, P. Fua, and K. M. Yi, "LF-Net: Learning local features from images," in *Proc. Conf. Neural Inf. Process. Syst.*, 2018, pp. 6237–6247.
- [13] Y. Tian, X. Yu, B. Fan, F. Wu, H. Heijnen, and V. Balntas, "SOSNet: Second order similarity regularization for local descriptor learning," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 11008–11017.
- [14] A. Mishchuk, D. Mishkin, F. Radenovic, and J. Matas, "Working hard to know your neighbor's margins: Local descriptor learning loss," in *Proc. Conf. Neural Inf. Process. Syst.*, 2017, pp. 4829–4840.
- [15] Z. Luo et al., "ASLFeat: Learning local features of accurate shape and localization," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 6588–6597.
- [16] H. Noh, A. Araujo, J. Sim, T. Weyand, and B. Han, "Large-scale image retrieval with attentive deep local features," in *Proc. Int. Conf. Comput. Vis.*, 2017, pp. 3476–3485.
- [17] D. DeTone, T. Malisiewicz, and A. Rabinovich, "SuperPoint: Self-supervised interest point detection and description," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 337–33712.
- [18] K. Han et al., "SCNet: Learning semantic correspondence," in *Proc. Int. Conf. Comput. Vis.*, 2017, pp. 1831–1840.
- [19] Y. Liu, L. Zhu, M. Yamada, and Y. Yang, "Semantic correspondence as an optimal transport problem," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 4463–4472.
- [20] J. Min and M. Cho, "Convolutional hough matching networks," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 2939–2949.
- [21] P. H. Seo, J. Lee, D. Jung, B. Han, and M. Cho, "Attentive semantic alignment with offset-aware correlation kernels," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 349–364.
- [22] I. Rocco, R. Arandjelovic, and J. Sivic, "Convolutional neural network architecture for geometric matching," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 6148–6157.
- [23] I. Rocco, R. Arandjelović, and J. Sivic, "End-to-end weakly-supervised semantic alignment," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 6917–6925.
- [24] I. Rocco, M. Cimpoi, R. Arandjelović, A. Torii, T. Pajdla, and J. Sivic, "Neighbourhood consensus networks," in *Proc. Conf. Neural Inf. Process. Syst.*, 2018, pp. 1651–1662.
- [25] S. Li, K. Han, T. W. Costain, H. Howard-Jenkins, and V. Prisacariu, "Correspondence networks with adaptive neighbourhood consensus," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 10196–10205.
- [26] I. Rocco, R. Arandjelović, and J. Sivic, "Efficient neighbourhood consensus networks via submanifold sparse convolutions," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 605–621.
- [27] P. Truong, M. Danelljan, and R. Timofte, "GLU-Net: Global-local universal network for dense flow and correspondences," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 6258–6268.
- [28] S. Huang, Q. Wang, S. Zhang, S. Yan, and X. He, "Dynamic context correspondence network for semantic alignment," in *Proc. Int. Conf. Comput. Vis.*, 2019, pp. 2010–2019.
- [29] V. Balntas, K. Lenc, A. Vedaldi, and K. Mikolajczyk, "HPatches: A benchmark and evaluation of handcrafted and learned local descriptors," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 5173–5182.
- [30] T. Sattler et al., "Benchmarking 6DoF outdoor visual localization in changing conditions," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 8601–8610.
- [31] B. Ham, M. Cho, C. Schmid, and J. Ponce, "Proposal flow: Semantic correspondences from object proposals," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 7, pp. 1711–1725, Jul. 2018.
- [32] J. Min, J. Lee, J. Ponce, and M. Cho, "Hyperpixel flow: Semantic correspondence with multi-layer neural features," in *Proc. Int. Conf. Comput. Vis.*, 2019, pp. 3395–3404.
- [33] X. Li, K. Han, S. Li, and V. Prisacariu, "Dual-resolution correspondence networks," in *Proc. Conf. Neural Inf. Process. Syst.*, 2020, pp. 17346–17357.
- [34] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "ORB: An efficient alternative to sift or SURF," in *Proc. Int. Conf. Comput. Vis.*, 2011, pp. 2564–2571.
- [35] M. Calonder, V. Lepetit, C. Strecha, and P. Fua, "BRIEF: Binary robust independent elementary features," in *Proc. Eur. Conf. Comput. Vis.*, 2010, pp. 778–792.
- [36] E. Simo-Serra, E. Trulls, L. Ferraz, I. Kokkinos, P. Fua, and F. Moreno-Noguer, "Discriminative learning of deep convolutional feature point descriptors," in *Proc. Int. Conf. Comput. Vis.*, 2015, pp. 118–126.
- [37] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proc. Int. Conf. Learn. Representations*, 2015.

- [38] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 770–778.
- [39] H. Germain, G. Bourmaud, and V. Lepetit, "S2DNet: Learning accurate correspondences for sparse-to-dense feature matching," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 626–643.
- [40] Q. Wang, X. Zhou, B. Hariharan, and N. Snavely, "Learning feature descriptors using camera pose supervision," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 757–774.
- [41] A. Vaswani et al., "Attention is all you need," in *Proc. Conf. Neural Inf. Process. Syst.*, 2017, pp. 5998–6008.
- [42] A. Dosovitskiy et al., "An image is worth 16x16 words: Transformers for image recognition at scale," in *Proc. Int. Conf. Learn. Representations*, 2021.
- [43] H. Chen et al., "Pre-trained image processing transformer," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 12294–12305.
- [44] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-end object detection with transformers," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 213–229.
- [45] W. Jiang, E. Trulls, J. Hosang, A. Tagliasacchi, and K. M. Yi, "COTR: Correspondence transformer for matching across images," in *Proc. Int. Conf. Comput. Vis.*, 2021, pp. 6207–6217.
- [46] J. Sun, Z. Shen, Y. Wang, H. Bao, and X. Zhou, "LoFTR: Detector-free local feature matching with transformers," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 8922–8931.
- [47] G. Tinchev, S. Li, K. Han, D. Mitchell, and R. Kouskouridas, "resolution correspondence networks," in *Proc. Brit. Mach. Vis. Conf.*, 2021.
- [48] T.-Y. Lin et al., "Feature pyramid networks for object detection," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 2117–2125.
- [49] C. Choy, J. Gwak, and S. Savarese, "4D spatio-temporal convnets: Minkowski convolutional neural networks," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 3075–3084.
- [50] A. Paszke et al., "PyTorch: An imperative style, high-performance deep learning library," in *Proc. Conf. Neural Inf. Process. Syst.*, 2019, pp. 8026–8037.
- [51] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, 2009, pp. 248–255.
- [52] Z. Li and N. Snavely, "Megadepth: Learning single-view depth prediction from internet photos," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 2041–2050.
- [53] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. Int. Conf. Learn. Representations*, 2015.
- [54] P.-E. Sarlin, D. DeTone, T. Malisiewicz, and A. Rabinovich, "SuperGlue: Learning feature matching with graph neural networks," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 4938–4947.
- [55] J. Zhang et al., "Learning two-view correspondences and geometry using order-aware network," in *Proc. Int. Conf. Comput. Vis.*, 2019, pp. 5845–5854.
- [56] C. B. Choy, J. Gwak, S. Savarese, and M. Chandraker, "Universal correspondence network," 2016, *arXiv:1606.03558*.
- [57] J. Min, J. Lee, J. Ponce, and M. Cho, "Learning to compose hypercolumns for visual correspondence," in *Proc. 16th Eur. Conf. Comput. Vis.*, 2020, pp. 346–363.
- [58] S. Cho, S. Hong, S. Jeon, Y. Lee, K. Sohn, and S. Kim, "CATs: Cost aggregation transformers for visual correspondence," in *Proc. 35th Conf. Neural Inf. Process. Syst.*, 2021, pp. 9011–9023.
- [59] S. Hong, S. Cho, J. Nam, and S. Kim, "Cost aggregation is all you need for few-shot segmentation," 2021, *arXiv:2112.11685*.
- [60] "Image matching challenge." [Online]. Available: <https://www.cs.ubc.ca/research/image-matching-challenge/2021/>



Xinghui Li received master of engineering degree (a 4-year program) with first class honor from the University of Oxford, in 2019. He is currently working toward the PhD degree with Active Vision Lab (AVL) of the University of Oxford under the supervision of Professor Victor Prisacariu. His research interests are image matching, visual localization and multi-modal learning.



Kai Han (Member, IEEE) received the PhD degree in computer science from The University of Hong Kong, in 2018. He has since held positions as a postdoctoral researcher with the Visual Geometry Group (VGG), the University of Oxford, a lecturer in computer vision with the University of Bristol, and a visiting faculty researcher with Google Research. He is currently an assistant professor with the Department of Statistics and Actuarial Science, The University of Hong Kong.



Shuda Li received the PhD degree in computer science from the University of Bristol, U.K., in 2016 and worked as a post-doctoral research assistant in the Active Vision Lab (AVL) with the University of Oxford, U.K., between 2017 and 2020. He is currently a research scientist with Common Sense Machine Inc. His research interests include SLAM, Lie and Multi-view Geometry, Real2Sim2Real, and Spatial AI.



Victor Prisacariu received the graduation degree (with first class honors) in computer engineering from Gheorghe Asachi Technical University, Iasi, Romania, in 2008, and the PhD degree in engineering science from the University of Oxford, in 2012. He is currently an Associate Professor in Information Engineering with the Department of Engineering Science with the University of Oxford, leading the Active Vision Lab (AVL). He is also the Chief Scientist of Niantic Inc.

DualRC: A Dual-Resolution Learning Framework with Neighbourhood Consensus for Visual Correspondences

–Supplementary Material–

Xinghui Li[†] Kai Han^{‡*} Shuda Li[†] Victor Prisacariu[†]

[†]University of Oxford [‡]The University of Hong Kong

1. Comparing with LoFTR

We further compare our DualRC-L with LoFTR [4] on the PhotoTourism dataset of IMC 2021 [1]. This dataset provides training data, allowing us to train both methods under the same hardware resource for a fair comparison. We follow the official dataset split and train LoFTR using the officially released code. During training, we re-scale the longer side of the images to 400, and the models are trained using $2 \times$ Nvidia TITAN RTX GPUs (with 24GB memory each) for 30 epochs.

For evaluation, we do not compare with the results in the leaderboard because those results may heavily rely on engineering tricks, extensive hyper-parameter tuning, hardware advantages, and the combination of multiple state-of-the-art methods. This makes it difficult to make a fair comparison. Therefore, we adopt the metric used by LoFTR to evaluate both methods. Specifically, LoFTR measures the accuracy of the relative pose between query-target image pairs. The pose is calculated by estimating the essential matrix based on established correspondences, and the error is represented by the angle between the estimated and ground-truth rotational vectors and translational vectors. The metric calculates the percentage of query-target image pairs whose rotational and translational errors are both smaller than a certain threshold. The final results are reported as the “Area Under Curve” of this accuracy curve under three thresholds: 5° , 10° , and 20° . During correspondence establishment, the maximum value of the longer side of images that our hardware can process for LoFTR is 1400. Therefore, we test both DualRC-L and LoFTR using this image size for a fair comparison. Correspondences are then scaled back to the sizes of the original images before estimating poses.

Table 1. Evaluation of DualRC-L and LoFTR on PhotoTourism dataset (image longer size: 1400)

Scene	DualRC-L	LoFTR
	$5^\circ / 10^\circ / 20^\circ$	$5^\circ / 10^\circ / 20^\circ$
British Museum	0.360 / 0.570 / 0.758	0.475 / 0.673 / 0.807
Florence Cathedral Side	0.477 / 0.624 / 0.747	0.452 / 0.602 / 0.714
Lincoln Memorial Statue	0.529 / 0.671 / 0.768	0.378 / 0.510 / 0.611
Milan Cathedral	0.386 / 0.603 / 0.764	0.450 / 0.645 / 0.793
Mount Rushmore	0.171 / 0.312 / 0.479	0.166 / 0.271 / 0.410
Piazza San Marco	0.328 / 0.534 / 0.688	0.375 / 0.540 / 0.671
Sagrada Familia	0.407 / 0.574 / 0.701	0.329 / 0.459 / 0.575
St Pauls Cathedral	0.438 / 0.630 / 0.786	0.503 / 0.678 / 0.792
Average	0.387 / 0.574 / 0.711	0.391 / 0.547 / 0.671

As shown in Table 1, our method achieves on-par performance to LoFTR when both methods are trained under the same hardware resource. Specifically, our lightweight DualRC-L performs slightly better than LoFTR at thresholds 10° and 20° , while LoFTR performs slightly better than DualRC-L at the threshold of 5° . The original LoFTR is trained with the longer side of images scaled to 840. Since LoFTR uses transformers and the memory cost of training grows quadratically ($O(n^2)$)

*Correspondence author.

with the image size, it is trained on $16\times$ Nvidia V100 GPUs in the original paper. However, due to limited hardware resources, we train LoFTR on the same image size and hardware resource as DualRC-L, which is using an image size of 400 and $2\times$ Nvidia TITAN RTX GPUs. A smaller image size indicates more coarse and sparse supervision, which may degrade performance. Moreover, during evaluation, LoFTR scales the longer side of the image to 1400. The ratio between the training and testing image size is less than 2. By reducing the training image size, this ratio increases, potentially making it harder for the weights learned from training images to generalize to testing images.

Our DualRC is a unified framework that can work well on both geometric and semantic matching tasks. Therefore, to provide a comprehensive comparison with LoFTR (originally developed for geometric tasks), we additionally evaluate LoFTR on the semantic task. We train LoFTR on the PF-Pascal dataset, resizing the images to 400×400 during both the training and testing stages. The results are reported as the average PCK with $\alpha = 0.1$, following the experiments in our main manuscript. The result for LoFTR is 63.1, significantly lower than the 85.8 achieved by DualRC. We believe there are two reasons for this performance difference: 1) The semantic task has fewer keypoint matches (usually < 10) available for training compared to the geometric task, which benefits from dense supervision. 2) The semantic task measures the accuracy of correspondences for given query points, whereas the geometric task does not explicitly define query points and instead evaluates the accuracy of downstream tasks (e.g., pose estimation) using the correspondences generated by the method. From the official implementation of LoFTR, we discovered that it cannot query subpixel-level keypoints in the source image, although it can find subpixel-level correspondences in the target image. All subpixel-level query points are rounded to the nearest integer positions on LoFTR’s coarse-level grid, resulting in deviated semantic correspondences.

2. Comparing with CHM and CATs

We compare our method with two new SoTA semantic matching methods: CHM [3] and CATs [2], in the geometric matching tasks to evaluate their performances as generic correspondence matchers. We train both CHM and CATs on MegaDepth dataset following the same configuration as DualRC in the main manuscript and test their performance on HPatches dataset with two metrics: Mean Matching Accuracy (MMA) and Corner Projection Error (CPE) in the accumulative form (AUC) as in the main manuscript. The results are reported in Table 2.

Due to the high memory cost of CHM and CATs, we are unable to train them with an image size of 400, which is the original training image size of DualRC-L. Therefore, we train all three methods, CHM, CATs, and our DualRC-L, with an image size of 256 using $2\times$ Nvidia TITAN RTX for fair comparison. For evaluation, since the testing image size of CATs must be the same as the training image size by the design of its architecture, we first test all three methods on the image size of 256. The largest size we are able to evaluate CHM is 800 so we further test DualRC-L and CHM on the image size of 800. We also include the result of our originally trained DualRC-L (w/ a training image size of 400) on testing images with the size of 1600. We evaluate three methods under two settings: with external query points and without external query points. For the first setting, we use the SIFT detector to find queries. For the second setting, we use the cyclic consistency (CC) described in Section 3.3 to find correspondences. We do not evaluate CC on the testing image size of 256 since the matching feature resolution is too small (16×16 for CATs, 32×32 for DualRC-L, 32×32 for CHM) to extract enough pairs with CC for evaluation.

From the results of Table 2, under the testing image resolution of 256, CATs has the best performance in both metrics followed by DualRC-L and CHM. As we increase the resolution of the testing images, the performance of both DualRC-L and CHM are improved dramatically. However, due to the high memory cost of CATs, it cannot take in larger images to validate under the same hardware resource. We also notice that for DualRC-L and CHM, the performance using CC is much better than using external query points. We believe that this is due to the fact that the external query points do not always lie in the region where the method has high confidence. CC instead selects the correspondences through cycle-consistency and it always keeps the most confident correspondences by the method. At the image size of 1600, CHM is out of memory while DualRC-L can still easily accommodate, which in turn highlights the efficacy of our method. We also report the peak allocated memory of three methods under different image sizes in Table 3 to illustrate our advantages in memory cost. It can be seen that our method requires significantly less memory cost than CHM and CATs in all cases.

Table 2. Evaluation of DualRC-L, CHM and CATs on HPatches dataset. “Trn” and “Test” represent the size of the training and testing image respectively. “SIFT kps” means correspondences are selected by the SIFT detector and “CC” means correspondences are selected through filtering of dense matching using cyclic consistency. All results are obtained using 1k correspondences.

	Metrics	DualRC-L	CHM	CATs
Trn:256, Test:256 (SIFT kps)	MMA	0.055/0.186/0.348	0.00/0.00/0.00	0.190/0.341/0.442
	3/5/10 (px)			
	CPE (AUC) 3/5/7 (px)			
Trn:256, Test:800 (SIFT kps)	MMA	0.408/0.608/0.745	0.00/0.00/0.00	Not Applicable
	3/5/10 (px)			
	CPE (AUC) 3/5/7 (px)			
Trn:400, Test:1600 (SIFT kps)	MMA	0.573/0.675/0.737	Out of Memory	Not Applicable
	3/5/10 (px)			
	CPE (AUC) 3/5/7 (px)			
Trn:256, Test:800 (CC)	MMA	0.672/0.824/0.912	0.373/0.486/0.753	Not Applicable
	3/5/10 (px)			
	CPE (AUC) 3/5/7 (px)			
Trn:400, Test:1600 (CC)	MMA	0.761/0.876/0.936	Out of Memory	Not Applicable
	3/5/10 (px)			
	CPE (AUC) 3/5/7 (px)			

Table 3. Peak memory allocation of DualRC-L, CHM and CATs under different testing image sizes.

	Test Size: 256	Test Size: 800	Test Size: 1600
DualRC-L	720 MB	1385 MB	4279 MB
CHM	826 MB	6797 MB	Out of Memory
CATs	831 MB	Not Applicable	Not Applicable

References

- [1] Image matching challenge. <https://www.cs.ubc.ca/research/image-matching-challenge/>. 1
- [2] Seokju Cho, Sunghwan Hong, Sangryul Jeon, Yunsung Lee, Kwanghoon Sohn, and Seungryong Kim. Cats: Cost aggregation transformers for visual correspondence. In *Thirty-Fifth Conference on Neural Information Processing Systems*, 2021. 2
- [3] Juhong Min and Minsu Cho. Convolutional hough matching networks. In *Proceedings of IEEE Intl. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2021. 2
- [4] Jiaming Sun, Zehong Shen, Yuang Wang, Hujun Bao, and Xiaowei Zhou. LoFTR: Detector-free local feature matching with transformers. *Proceedings of IEEE Intl. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2021. 1

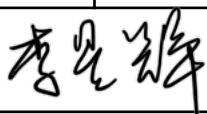
Statement of Authorship for joint/multi-authored papers for PGR thesis

To appear at the end of each thesis chapter submitted as an article/paper

The statement shall describe the candidate's and co-authors' independent research contributions in the thesis publications. For each publication there should exist a complete statement that is to be filled out and signed by the candidate and supervisor (**only required where there isn't already a statement of contribution within the paper itself**).

Title of Paper	DualRC: A Dual-Resolution Learning Framework With Neighbourhood Consensus for Visual Correspondences
Publication Status	☒ Published ☐ Accepted for Publication ☐ Submitted for Publication ☐ Unpublished and unsubmitted work written in a manuscript style
Publication Details	Xinghui Li , Kai Han, Shuda Li, Victor Adrian Prisacariu. In IEEE Transactions On Pattern Analysis And Machine Intelligence (TPAMI), 2024

Student Confirmation

Student Name:	Xinghui Li		
Contribution to the Paper	First Author Contribution: • Conception of research ideas. • Design and implementation of models. • Running of large-scale experiments. • Writing and presentation of the paper.		
Signature		Date	07/Jan/2025

Supervisor Confirmation

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

Supervisor name and title: Prof Victor Adrian Prisacariu			
Supervisor comments I certify that the student made a substantial contribution to the chapter and that the description described above is accurate.			
Signature		Date	9/1/25

This completed form should be included in the thesis, at the end of the relevant chapter.

3.3 Conclusion

In this chapter, we presented an end-to-end dual-resolution architecture that can establish dense pixel-wise correspondences between a pair of images in a coarse-to-fine manner. We first obtained a full 4D correlation tensor from the coarse-resolution feature maps, which was refined by the learnable 4D convolutional neighborhood consensus module. The refined correlation tensor was then used to guide the model to obtain robust matching from fine-resolution feature maps. In this way, we could dramatically increase matching reliability and localisation accuracy, while avoiding applying the expensive 4D convolution kernels on fine-resolution feature maps. We demonstrated the effectiveness of our approach on several public benchmarks.

Although we extended our work to semantic matching using our dynamically adaptive 4D kernel, it significantly increased the model’s complexity. This prompted us to investigate semantic matching, and we proposed a simple and effective learning pipeline in the next chapter.

Chapter 4

Improving Semantic Matching

This chapter focuses on improving semantic matching. Specifically, we investigate two areas: First, in Section 4.1, we re-examine metric learning and propose a simple yet effective learning framework to finetune pretrained feature backbones on semantic matching objective. This allows feature backbones to achieve on par performance to models with much more complex architecture. Second, in Section 4.2, we explore prompt tuning to adapt Stable Diffusion, a text-to-image generator, to the semantic matching task using a similar metric learning framework introduced in Section 4.1. In addition, we propose a conditional prompting module to include image information in the prompt and further improve the matching accuracy.

4.1 SimSC: A Simple and Effective Learning Framework for Semantic Correspondence

4.1.1 Introduction

This section presents a simple and effective pipeline for semantic matching. One major challenge for semantic matching is to overcome the intra-category appear-

ance difference. This requires the model to have strong and high-level knowledge of different categories of objects. Therefore, as already discussed in Section 2.2.1, many methods rely on feature backbones pretrained on large-scale image recognition datasets to extract object-aware dense feature maps [91, 97, 23], followed by their own matching modules to adapt feature maps to semantic matching tasks. Various architectures have been proposed, ranging from high-dimensional convolutions [17, 1, 100, 134] to transformers [18, 99, 98]. These methods have demonstrated good results in capturing intra-category appearance differences. However, the design of the network keeps increasing the complexity of the model and becomes less intuitive, and the computational cost grows dramatically.

To avoid adding further complexity to the network, we revisit a largely overlooked approach: learning correspondences directly through the backbone using metric learning [36, 37]. This perspective had been discussed before, and can be dated back to UCN [135], but has been overlooked in more recent works due to the later substantial improvement brought by the development of downstream matching modules. Under this formulation, we extract local features of keypoints from a dense feature map built from the image, and explicitly pull the local features of correct correspondences closer while pushing the features of unrelated keypoints away. Despite its simplicity, we discover that the backbone is able to learn surprisingly accurate correspondences by employing softmax and the cross-entropy loss, and the key to this success is to select an appropriate temperature value for the softmax operation to amplify the supervision signal at the early stage of training. This yields a significantly simpler framework with no complex downstream module and only a single forward pass through the backbone.

To further automate the framework, we introduce a temperature learning module that predicts the optimal temperature based on the dense feature maps of input images. This module is trained jointly with the backbone, adjusting the

temperature dynamically. Experiments demonstrate that we are able to achieve the matching accuracy comparable to the state-of-the-art methods with the same backbone but much more complicated architecture. Additionally, our approach works well with both convolutional and ViT-based backbones.

To summarize, our contributions in this section are:

1. We introduce a simple learning framework for semantic correspondence. By employing the softmax operation with an appropriate temperature value, the backbone itself is sufficient to learn accurate correspondence. Unlike previous methods, our framework does not have any downstream task module or multi-level feature exploitation.
2. We propose a temperature learning module that is trained together with feature backbones and updates the temperature online in order to avoid the manual search for the optimal temperature and further simplify the framework.
3. We evaluate our method on three public benchmarks to demonstrate its effectiveness on both CNN-based and ViT-based feature backbones.

We published this work as an ArXiv 2023 technical report in [38].

SimSC: A Simple and Effective Framework for Semantic Correspondence

Xinghui Li[†] Kai Han^{‡*} Xingchen Wan[†] Victor Adrian Prisacariu[†]
[†]University of Oxford [‡]University of Hong Kong
xinghui@robots.ox.ac.uk kaihanx@hku.hk
xwan@robots.ox.ac.uk victor@robots.ox.ac.uk

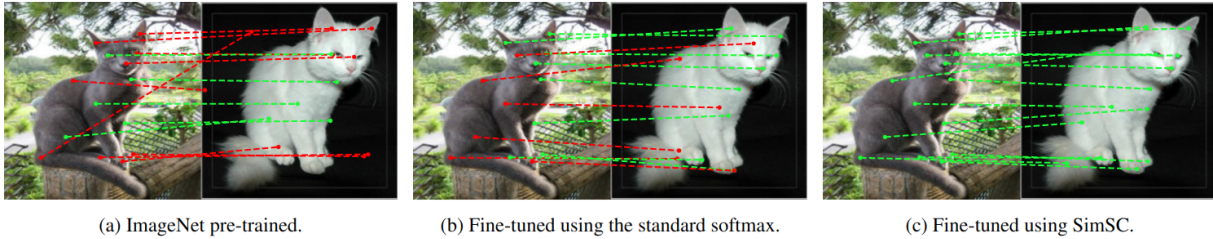


Figure 1. Semantic correspondence matching using ResNet101 features. As illustrated, SimSC enables the backbone to learn accurate correspondence on its own without any additional modules.

Abstract

*We present a simple yet very effective framework to address the problem of semantic correspondence. Existing works usually rely on intricate downstream modules or leveraging multi-level deep features for matching. We revisit and reassess the potential of metric learning, which has been largely overlooked in this field, to address this problem. We demonstrate that, with an appropriate temperature value for softmax, commonly used feature backbones, like ResNet or ViT, can surprisingly learn accurate correspondences by themselves, without any additional module or multi-layer exploitation. To further automate the framework, we employ a learning module to predict the optimal temperature for backbones, avoiding the grid search. We designate our framework as SimSC, short for **Simple framework for Semantic Correspondence**. Through thorough evaluations on public benchmarks, using both convolutional and transformer-based feature backbones, SimSC achieves state-of-the-art results, showcasing its remarkable simplicity and effectiveness.*

1. Introduction

Matching semantically similar points across two images is a challenging task in computer vision due to significant appearance differences between two instances. The majority of methods in the literature contain two parts [4, 17, 18, 22, 24, 25, 32, 38, 53]: a feature backbone,

predominately ResNet [15], that is pre-trained on the large-scale dataset such as ImageNet [7], and a downstream task module. The general-purpose feature maps are extracted from image pairs by feature backbones, and they are subsequently sent to the downstream module which is trained on a semantic correspondence dataset.

Recently, attention has been given to developing powerful downstream task modules. Various architectures have been proposed, ranging from high-dimensional convolutions [24, 25, 32, 38, 53] to transformers [4, 20]. The head is either directly applied to feature maps themselves [18, 22, 33, 35], the 4D matching score space formed by feature maps [20, 24, 25, 32, 38] or a combination of both [4]. Multi-level feature exploitation is another popular technique and frequently coupled with the downstream task modules to enhance the power of models [4, 32, 35]. These methods have demonstrated promising results in capturing intra-category appearance differences. However, the design of the network becomes more complicated and less intuitive, and the computational cost grows dramatically.

To avoid further stacking up the complexity of the network, we revisit and reassess a largely overlooked perspective: learning correspondences by the backbone itself with metric learning. This perspective had been discussed before, dated back to UCN [5]. However, due to the later substantial improvement by the development of downstream matching modules, It has been overlooked ever since. Under such a formulation, we extract local features of key-points from a dense feature map of the image, and explicitly pull the local features of correct correspondences closer

and push the features of unrelated keypoints away. Despite its simplicity, we discover that the backbone is able to learn surprisingly accurate correspondences by employing softmax and the cross-entropy loss, and the key to success is to select an appropriate temperature value for softmax to amplify the supervision signal at the early stage of training. We further discover that the formulation can be extended beyond the cross-entropy loss and generalized to matching-specific losses like the flow loss [4, 32]. This yields a significantly clean framework with no complex downstream module but only a single forward pass through the backbone.

However, a manual search for the optimal temperature may be required for different datasets and backbones. To further automate the framework, we propose a temperature learning module that predicts the optimal temperature based on the dense feature maps of input images. The temperature learning module is jointly trained with the backbone, and the temperature is updated and adjusted online. Experiments demonstrate that we are able to achieve the matching accuracy comparable to the state-of-the-art methods with the same backbone but much more complicated architecture. Our method is versatile and can be applied to both convolutional and ViT-based backbones.

We summarize our contributions as follows: (1) We introduce a remarkably simple learning framework for semantic correspondence. By employing the softmax with an appropriate temperature value, the backbone itself is sufficient to learn accurate correspondence. Unlike previous methods, our framework does not have any downstream task module or multi-level feature exploitation. (2) We propose a temperature learning module that is trained together with feature backbones and updates the temperature online in order to avoid the manual search for the optimal temperature and further simplify the framework. (3) Despite the simplicity, our method achieves state-of-the-art performance on public benchmarks, including PF-Pascal and SPair-71K datasets, using both convolutional and transformer-based backbones. Notably, by coupling with the more powerful iBOT [54] backbone, our framework outperforms previous methods on PF-Pascal and SPair-71K datasets by a significant margin.

2. Related work

Semantic Correspondence Semantic matching initially relied on handcrafted features such as SIFT [31] or HOG [6]. SIFT Flow [29] is the pioneering work that matches semantically similar objects using SIFT descriptor. Recently, dense feature maps extracted by deep neural networks [15, 42] have shown great success in many computer vision tasks [9, 11, 26, 39, 44, 47]. SCNet [14] is one of the first works that adopts the deep learning framework for the semantic correspondence problem. The majority of literature shares a similar pipeline: dense feature extraction followed by semantic correspondence downstream

task modules. One family of modules is directly applied to the feature map and enhances features directly. This can be achieved through feature binning [2], multi-level feature assembly [33, 35] or learned feature fusion [22]. The other family of modules is applied to the 4D correlation maps formed by dense feature maps. NCNet [38] is the pioneering work in this direction. It filters the correlation maps by enforcing the neighborhood consensus constraint. Follow-up works improve the accuracy by employing anisotropic 4D convolutional kernels [25]; performing Hough matching with 6D convolutions [32]; applying transformers to multi-level correlation maps [20] or solving the matching as an optimal transport problem [30]. Later works [4, 19] also attempt to fuse correlation maps and features which demonstrate promising results as well. Although downstream task modules are becoming more advanced and powerful, the complexity of the model grows dramatically as well, which motivates us to explore correspondence learning from a different perspective. Parallel to innovation on downstream task modules, a second line of work focuses on learning semantic correspondence using image-level supervision [22, 27, 40, 48]. However, the accuracy of these methods still lags behind keypoint-supervised methods.

Metric Learning Metric learning focuses on aggregating key features from data. If two pieces of data describe the same concept, features extracted from them should be similar. Otherwise, the features should be different. It has been extensively studied in the context of image classification [16, 43] and facial recognition [8, 49, 50]. In the context of image feature matching, works have been devoted to learning the description of local image patches [36, 41, 45, 46] extracted by a pre-defined keypoint detector. Our method is different from theirs as we work on the end-to-end matching where no keypoint detector is used. Our method is closely related to UCN [5], the pioneering work solving the end-to-end matching from the perspective of metric learning. However, the main contribution of UCN is its architectural innovation and it only discussed losses explicitly applied to features but overlooked the formulation involving softmax and cross-entropy loss. Another related line of work investigates the role of the temperature of softmax on the training of the network when paired with the cross-entropy loss [1, 51, 52]. Nevertheless, the scope is confined to extracting global information and classification tasks. We instead focus on its impact on extracting local features and matching tasks.

3. Method

In this section, we introduce SimSC, which is illustrated in Fig. 2. Our framework has a simple architecture: a feature backbone $f(\cdot)$ and a temperature learning module $g(\cdot)$.

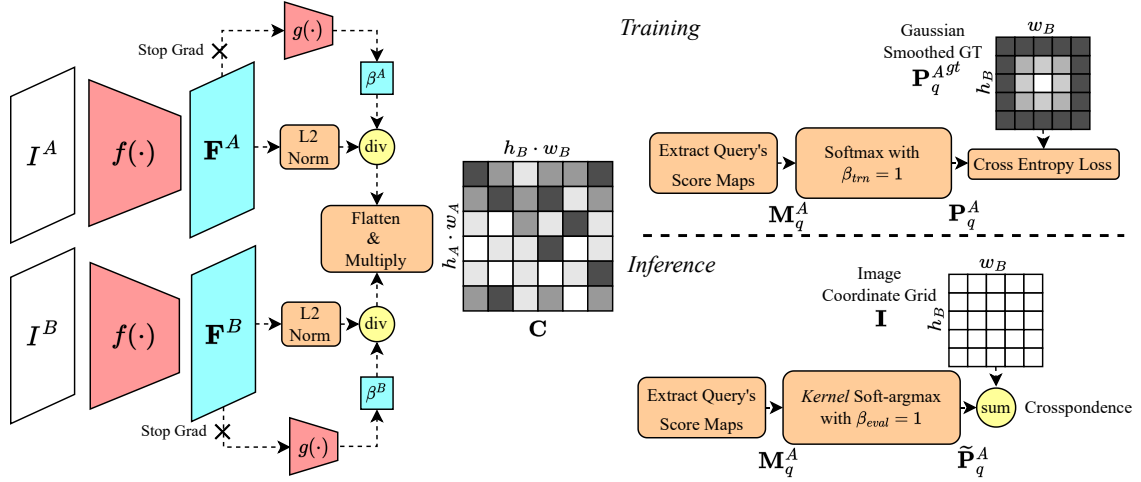


Figure 2. Overview of SimSC. (i) The feature backbone $f(\cdot)$ extracts feature maps F^A and F^B from images I^A and I^B respectively. (ii) The temperature learning module $g(\cdot)$ predicts partial temperatures β^A and β^B from F^A and F^B . (iii) F^A and F^B are normalized and divided by their corresponding partial temperatures and construct the matching score tensor C . (iv) For training, the matching score map M_q^A of the query point is extracted from C and converted to a probability distribution by the standard softmax before being supervised by ground truth distribution $P_q^{A^{gt}}$ using the cross-entropy loss. (v) For inference, the matching score map M_q^A is processed by kernel soft-argmax proposed in Lee et al. [22]. The final correspondence is the marginalization of the probability map $\hat{P}_q^A$ multiplied with an image coordinate grid I . Overall, our method has a simple architecture with no additional module compared with existing methods.

The backbone extracts the dense feature maps from the input image pair, and the temperature learning module predicts the temperature for the softmax operation based on the feature maps. Correspondences are found by the kernel soft-argmax proposed by [22]. Compared with existing methods, our method is remarkably simple without any complicated downstream task module.

In Sec. 3.1, we briefly review how semantic matching is formulated as a metric learning problem. In Sec. 3.2, we show the importance of temperature in learning correspondence and the structure of the temperature learning module. Finally we show how to localize the final correspondence in Sec. 3.3.

3.1. Semantic Matching as Metric Learning

Given a pair of images $I^A \in \mathbb{R}^{3 \times H_A \times W_A}$ and $I^B \in \mathbb{R}^{3 \times H_B \times W_B}$, the feature backbone $f(\cdot)$ extracts dense feature maps $F^A \in \mathbb{R}^{C \times h_A \times w_A}$ and $F^B \in \mathbb{R}^{C \times h_B \times w_B}$ from them respectively. We then perform L2 normalization along the channel dimension, obtaining $\hat{F}^A \in \mathbb{R}^{C \times h_A \times w_A}$ and $\hat{F}^B \in \mathbb{R}^{C \times h_B \times w_B}$:

$$\hat{F}_{ij}^A = \frac{F_{ij}^A}{\|F_{ij}^A\|} \quad \text{and} \quad \hat{F}_{kl}^B = \frac{F_{kl}^B}{\|F_{kl}^B\|}, \quad (1)$$

where i, j and k, l are spatial indices for feature maps F^A and F^B respectively. Suppose we have a set of sparse ground-truth correspondences $\mathbf{X} = \{(\mathbf{x}_q^A, \mathbf{x}_q^B) \mid q =$

$1, \dots, n\}$ where $\mathbf{x}_q^A = (x_q^A, y_q^A)$ and $\mathbf{x}_q^B = (x_q^B, y_q^B)$. For each $\mathbf{x}_q^A$, the objective is to maximize the similarity between $\hat{F}_{\mathbf{x}_q^A}^A$ and $\hat{F}_{\mathbf{x}_q^B}^B$, and minimize the similarity between $\hat{F}_{\mathbf{x}_q^A}^A$ and the rest of features of $\hat{F}^B$.

The simplest solution is employing the commonly used triplet loss $\mathcal{L}_{triplet}$, which directly minimizes L2 distances between features of positive correspondences and maximizes the distances between features of negative pairs:

$$\mathcal{L}_{triplet} = \frac{1}{n} \sum_{q=1}^n \|\hat{F}_{\mathbf{x}_q^A}^A - \hat{F}_{\mathbf{x}_q^B}^B\| + \max(0, m - \frac{1}{h_B w_B} \sum_{\forall kl \neq \mathbf{x}_q^B} \|\hat{F}_{\mathbf{x}_q^A}^A - \hat{F}_{kl}^B\|). \quad (2)$$

A similar idea has been adopted in [5]. However, this loss does not yield a satisfactory result, and cannot compete with the improvement brought by advanced downstream matching modules, which is partially the reason for why the community shifted away from this direction.

After the emergence of NCNet [38], filtering 4D correlation tensor gains more attention due to its effectiveness in enforcing neighbourhood consensus. Many follow-up works have been devoted to how to extract the most information from the correlation tensor, using high dimensional convolution [25, 26, 32] or transformer [20, 32]. As the output of their downstream task module is a 4D tensor, feature loss is no longer applied. Instead, they convert the filtered

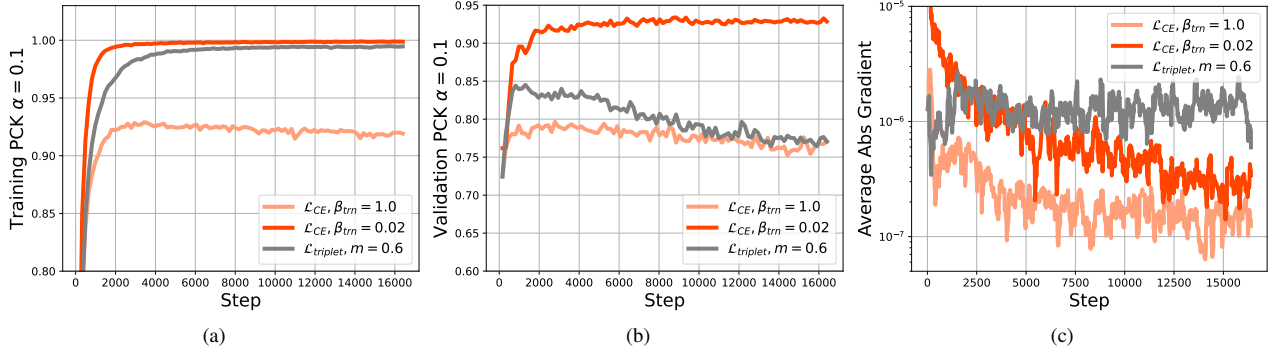


Figure 3. Finetuning ResNet101 on PFPascal dataset with different losses and β_{trn} . (a) Training accuracy; (b) Validation accuracy; (c) Average absolute value of the gradient to the last convolutional layer.

correlation into a matching probability map which is supervised by either the cross-entropy loss [19, 53], the probability L2 loss [25, 26], or predominantly keypoint flow loss [4, 20, 22, 32]. Inspired by them, we also construct such a correlation tensor but removing all extra modules. The 4D correlation tensor $\mathbf{C} \in \mathbb{R}^{h_A \times w_A \times h_B \times w_B}$ is:

$$\mathbf{C}_{ijkl} = (\hat{\mathbf{F}}_{ij}^A)^\top (\hat{\mathbf{F}}_{kl}^B). \quad (3)$$

For each query $\mathbf{x}_q^A$, its 2D correlation map $\mathbf{M}_q^A \in \mathbb{R}^{h_B \times w_B}$ is then extracted from $\mathbf{C}$. Map $\mathbf{M}_q^A$ contains correlations between $\hat{\mathbf{F}}_{\mathbf{x}_q^A}^A$ and all features in $\mathbf{F}^B$. After that, the softmax with temperature β_{trn} converts $\mathbf{M}_q^A$ into a probability distribution $\mathbf{P}_q^A$:

$$\mathbf{P}_{q\ kl}^A = \frac{\exp(\mathbf{M}_{q\ kl}^A / \beta_{trn})}{\sum_{kl} \exp(\mathbf{M}_{q\ kl}^A / \beta_{trn})}. \quad (4)$$

We then construct the ground-truth distribution $\mathbf{P}_q^{Agt}$ with the ground-truth correspondence $\mathbf{x}_q^B$. The backbone can then be tuned using the cross-entropy loss:

$$\mathcal{L}_{CE} = -\frac{1}{n} \sum_q \sum_{kl} \mathbf{P}_{q\ kl}^{Agt} \cdot \log \mathbf{P}_{q\ kl}^A. \quad (5)$$

3.2. Importance of Temperature

Despite the simplicity of this formulation, we observe that the backbone itself is able to learn surprisingly accurate semantic correspondences if using a small β_{trn} when trained by $\mathcal{L}_{CE}$ without any additional module. Although such a result is encouraging, the role which the temperature plays during training remain unexplained. In literature, Lee et al. [23] also reported a similar observation regarding the temperature, but they did not provide a in-depth analysis either.

We choose ResNet101 as our backbone for investigation to align with the literature. We compare three loss configurations: $\mathcal{L}_{triplet}$ with margin $m = 0.6$, $\mathcal{L}_{CE}$ with a moderate temperature $\beta_{trn} = 1.0$ and $\mathcal{L}_{CE}$ with with a small

temperature $\beta_{trn} = 0.02$, which is chosen based on [4, 22]. We plot the training and validation accuracy of finetuning ResNet101 on PF-Pascal [12] dataset using these configurations in Fig. 3(a) and (b) respectively. In addition, we also plot the average absolute gradient of the last convolutional layer of ResNet101 in Fig. 3(c).

As shown in Fig. 3(a) and (b), ResNet101 fits to training data tightly when trained with $\mathcal{L}_{triplet}$ and $\mathcal{L}_{CE}$ with $\beta_{trn} = 0.02$. However, the plot of validation accuracy shows that the generalizability of $\mathcal{L}_{triplet}$ is worse than that of $\mathcal{L}_{CE}$, indicating that $\mathcal{L}_{triplet}$ is not the optimal choice for semantic matching. Focusing on $\mathcal{L}_{CE}$, if no temperature is set, i.e., $\beta_{trn} = 1.0$, both training and validation accuracy deteriorate significantly compared to a small temperature $\beta_{trn} = 0.02$, implying that the learning is suppressed. Focusing on Fig. 3(c), the gradient of $\mathcal{L}_{CE}, \beta_{trn} = 0.02$ is significantly larger than that of $\mathcal{L}_{CE}, \beta_{trn} = 1.0$ at the early stage of the training, before decreasing rapidly and converging to the similar magnitude to $\mathcal{L}_{CE}, \beta_{trn} = 1.0$. The gradient of the $\mathcal{L}_{triplet}$ remain constant at the same scale throughout training.

We further write down the derivatives of $\mathcal{L}_{CE}$ with respect to features. Assume $\mathbf{P}_q^{Agt}$ is a Dirac-delta distribution, the derivative of $\mathcal{L}_{CE}$ to the query feature $\mathbf{F}_{\mathbf{x}_q^A}^A$ is:

$$\frac{\partial \mathcal{L}_{CE}}{\partial \mathbf{F}_{\mathbf{x}_q^A}^A} = -\frac{1}{n} \sum_{q=1}^n \left(\frac{\partial \mathcal{L}_{CE}}{\partial \mathbf{P}_{q\ \mathbf{x}_q^B}^A} \sum_{kl} \frac{\partial \mathbf{P}_{q\ \mathbf{x}_q^B}^A}{\partial \mathbf{M}_{q\ kl}^A} \frac{\partial \mathbf{M}_{q\ kl}^A}{\partial \mathbf{F}_{\mathbf{x}_q^A}^A} \right) \quad (6)$$

where:

$$\frac{\partial \mathcal{L}_{CE}}{\partial \mathbf{P}_{q\ \mathbf{x}_q^B}^A} = \frac{1}{\mathbf{P}_{q\ \mathbf{x}_q^B}^A} \quad (7)$$

$$\frac{\partial \mathbf{P}_{q\ \mathbf{x}_q^B}^A}{\partial \mathbf{M}_{q\ kl}^A} = \frac{1}{\beta_{trn}} \mathbf{P}_{q\ \mathbf{x}_q^B}^A (1 - \mathbf{P}_{q\ \mathbf{x}_q^B}^A) \quad \text{if } kl = \mathbf{x}_q^B \quad (8)$$

$$= -\frac{1}{\beta_{trn}} \mathbf{P}_{q\ \mathbf{x}_q^B}^A \mathbf{P}_{q\ kl}^A \quad \text{if } kl \neq \mathbf{x}_q^B \quad (9)$$

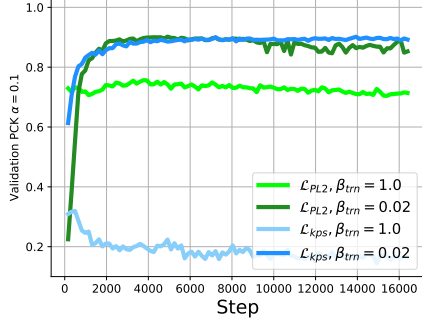


Figure 4. A small temperature improves backbone’s accuracy not only trained by $\mathcal{L}_{CE}$, but also the keypoint flow loss $\mathcal{L}_{kps}$ and the probability L2 loss $\mathcal{L}_{PL2}$.

$$\frac{\partial \mathbf{M}_{q\ kl}^A}{\partial \mathbf{F}_{\mathbf{x}_q^A}^A} = \frac{1}{\|\mathbf{F}_{\mathbf{x}_q^A}^A\|} (\hat{\mathbf{F}}_{kl}^B - \mathbf{M}_{q\ kl}^A \hat{\mathbf{F}}_{\mathbf{x}_q^A}^A) \quad (10)$$

and the derivative to all features in $\mathbf{F}^B$ are:

$$\frac{\partial \mathcal{L}_{CE}}{\partial \mathbf{F}_{kl}^B} = -\frac{1}{n} \sum_{q=1}^n \left(\frac{\partial \mathcal{L}_{CE}}{\partial \mathbf{P}_{q\ \mathbf{x}_q^B}^A} \frac{\partial \mathbf{P}_{q\ \mathbf{x}_q^B}^A}{\partial \mathbf{M}_{q\ kl}^A} \frac{\partial \mathbf{M}_{q\ kl}^A}{\partial \mathbf{F}_{kl}^B} \right) \quad (11)$$

where:

$$\frac{\partial \mathbf{M}_{q\ kl}^A}{\partial \mathbf{F}_{kl}^B} = \frac{1}{\|\hat{\mathbf{F}}_{kl}^B\|} (\hat{\mathbf{F}}_{\mathbf{x}_q^A}^A - \mathbf{M}_{q\ kl}^A \hat{\mathbf{F}}_{kl}^B). \quad (12)$$

Eq. (7) is the derivative of the cross-entropy loss function; Eq. (8) (9) are the derivatives of the softmax function and finally Eq. (10) is the derivatives of the correlation to the raw feature itself.

Evidently, Eq. (7) explains the downturn of the gradient of $\mathcal{L}_{CE}$ in Fig. 3(c). As training progresses, the matching becomes more accurate and the matching probability at the ground-truth position, i.e. $\mathbf{P}_{q\ \mathbf{x}_q^B}^A$, increases. Eq. (7) therefore becomes smaller. Eq. (8) (9) reveal that β_{trn} acts as a scaling factor to the gradient and a small β_{trn} amplifies the gradient, which explains why the gradient of $\mathcal{L}_{CE}, \beta_{trn} = 0.02$ is larger than that of $\mathcal{L}_{CE}, \beta_{trn} = 1.0$.

From Fig. 3(a) and (b), most of the performance gain occurs at the early stage of the training. This is also when the gradient of $\mathcal{L}_{CE}, \beta = 0.02$ leads that of $\mathcal{L}_{CE}, \beta_{trn} = 1.0$ the most before it rapidly decreases. This implies that a small temperature achieves better accuracy by amplifying the supervision signal during the most critical stage of the training. Without it, the gradient is too small for the backbone to capture sufficient knowledge at the early stage of the training, and the reasons for such a small are the term $1/\|\mathbf{F}_{\mathbf{x}_q^A}^A\|$ in Eq. (10) and the product of two probabilities in Eq. (8) (9).

We further discover that this method can be extended beyond the cross-entropy loss. We additionally apply the

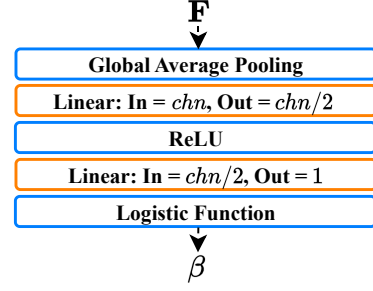


Figure 5. The architecture of the temperature learning module. It starts with the global average pooling followed by a two-layer MLP, which converts the average pooled feature map to a scalar. The logistic function transfers the scalar to the range of $[0, 1]$. In the figure, chn represents the channel dimension of $\mathbf{F}$.

probability L2 loss $\mathcal{L}_{PL2}$ and the keypoint flow loss $\mathcal{L}_{kps}$ to the framework and observe the similar pattern where a small temperature significantly improve accuracy. We illustrate the results in Fig. 4.

The simplest method to find the optimal β_{trn} is by brute-force grid search. However, this method is time-consuming since we might need to repeat the grid search for different feature backbones, loss and datasets. Therefore, we propose a temperature learning module $g(\cdot)$ to predict the optimal temperature conditioned on input features. As illustrated in Fig. 5, the module has a very simple architecture. It first performs the global average pooling on input feature map $\mathbf{F}$ and converts it into a vector. A two-layer MLP then projects the vector to a scalar β which is mapped to the range $[0, 1]$ by the logistic function $\text{sig}(x) = 1/(1 + e^x)$.

The formulation in Sec. 3.1 is then modified as follows: we process $\mathbf{F}^A$ and $\mathbf{F}^B$ with $g(\cdot)$ and obtain $\beta^A = g(\mathbf{F}^A)$ and $\beta^B = g(\mathbf{F}^B)$. The normalized feature maps $\hat{\mathbf{F}}^A$ and $\hat{\mathbf{F}}^B$ are then divided by β^A and β^B respectively. We finally calculate the matching score tensor $\mathbf{C}$ and replace Eq. (3) with:

$$\mathbf{C}_{ijkl} = \left(\frac{\hat{\mathbf{F}}_{ij}^A}{\beta^A} \right)^\top \left(\frac{\hat{\mathbf{F}}_{kl}^B}{\beta^B} \right). \quad (13)$$

By doing so, our method automatically set the temperature value as $\beta^A \cdot \beta^B$, so we do not need a second temperature in the softmax and set $\beta_{trn} = 1$.

3.3. Localization of correspondences

During inference, in order to achieve the sub-pixel accuracy, we use the kernel soft-argmax proposed in [22]. Given the query $\mathbf{x}_q^A$, its matching score map $\mathbf{M}_q^A$ is extracted from matching score tensor $\mathbf{C}$ using bilinear interpolation. It then multiplies with a Gaussian kernel $\mathbf{G} \in \mathbb{R}^{h_B \times w_B}$ with the mean being the maximum of $\mathbf{M}_q^A$ and the standard deviation of σ to suppress the non-maximum modes in $\mathbf{M}_q^A$:

$$\widetilde{\mathbf{M}}_q^A = \mathbf{G} * \mathbf{M}_q^A, \quad (14)$$

where $*$ is the element-wise multiplication between matrices. Note that, since we have already rescaled $\mathbf{C}$ by $\beta^A \cdot \beta^B$, we do not need to manually set a different temperature value β_{eval} for evaluation but use $\beta_{eval} = 1$. The standard softmax operation directly converts $\tilde{\mathbf{M}}_q^A$ to a probability distribution $\tilde{\mathbf{P}}_q^A$, and the final correspondence can be obtained by:

$$\bar{\mathbf{x}}_q^B = \sum_{kl} (\tilde{\mathbf{P}}_q^A)_{kl} * \mathbf{I}_{kl}, \quad (15)$$

where $\bar{\mathbf{x}}_q^B \in \mathbb{R}^2$ is the predicted correspondence, k and l are spatial indices of $\tilde{\mathbf{P}}_q^A$ and $\mathbf{I} \in \mathbb{Z}^{2 \times h_B \times w_B}$ is an image coordinate grid with $\mathbf{I}_{kl} = (k, l)$. The prediction $\bar{\mathbf{x}}_q^B$ is interpreted as the expected value of the coordinate grid $\mathbf{I}$ under the probability distribution $\tilde{\mathbf{P}}_q^A$.

4. Experiments

We evaluate our method on three public datasets: PF-Pascal [12], PF-Willow [13], and SPair-71K [34]. In Sec. 4.1, we briefly introduce three public datasets and the evaluation metric. In Sec. 4.2, we elaborate on our experimental setup, which includes configurations of models and hyperparameters. We show the comparison between our method and state-of-the-art methods in Sec. 4.3 and provide model analysis in Sec. 4.4.

4.1. Datasets and evaluation metric

Datasets PF-Pascal consists of 2941 training image pairs, 308 validation pairs, and 299 testing pairs spanning across 20 categories of objects. Its supplement PF-Willow contains 900 testing pairs split into 3 categories with no training and validation data. SPair-71K, on the other hand, is a larger and more challenging dataset with 53,340 training pairs, 5,384 validation pairs, and 12,234 testing pairs across 18 categories of objects with large scale and appearance variation. For all three datasets, each image pair has non-uniform numbers of ground-truth correspondences. SPair-71K additionally provides the object’s bounding box for each image.

Evaluation Metric We follow the common practice in the literature and use the percentage of corrected keypoints (PCK) as the evaluation metric. Given an image pair (I^A, I^B) and its associated correspondence set $\mathbf{X} = \{(\mathbf{x}_q^A, \mathbf{x}_q^B) \mid q = 1, 2, \dots, n\}$, for each $\mathbf{x}_q^A = (x_q^A, y_q^A)$, we find its predicted correspondence $\bar{\mathbf{x}}_q^B$ and calculate PCK for the image pair by:

$$PCK(I^A, I^B) = \frac{1}{n} \sum_q^n \mathbb{I}(\|\bar{\mathbf{x}}_q^B - \mathbf{x}_q^B\| \leq \alpha * \theta) \quad (16)$$

where θ is the base threshold, α is a number less than 1 and $\mathbb{I}(\cdot)$ is the binary indicator function with $\mathbb{I}(\text{true}) = 1$

and $\mathbb{I}(\text{false}) = 0$. For PF-Pascal, θ is set as $\theta_{img} = \max(H_B, W_B)$. For PF-Willow, base threshold is set as $\theta_{kps} = \max(\max_q(x_q^B) - \min_q(x_q^B), \max_q(y_q^B) - \min_q(y_q^B))$ and for SPair-71K, base threshold is set as $\theta_{bbox} = \max(h_{bbox}, w_{bbox})$ where h_{bbox} and w_{bbox} are height and width of the bounding box. The choices of all three base thresholds align with the convention in the literature.

4.2. Experimental details

Feature backbone We implement our framework in Python using the PyTorch [37] library. We evaluate our finetuning method on ImageNet pre-trained ResNet101 [15], which has the CNN architecture and is widely used in literature. Specifically, ResNet101 is truncated before `layer4`, so the feature dimension is 1024, and the image-to-feature spatial size ratio is 16. In addition, we also validate our method on ResNet50, DINO [3], and iBOT [54] to show that our method works for both CNN-based and ViT-based [10] feature backbones.

Training configuration We train two configurations: finetuning the *last block* and finetuning the *entire feature backbone*, which we denote as **SimSC-ResNet101** and **SimSC-ResNet101*** respectively in Tab. 1. SimSC-ResNet101 corresponds to works that only train their downstream modules and SimSC-ResNet101* corresponds to works that train both downstream modules and backbones. We use the Adam optimizer [21] for training, and the learning rate is set to 1×10^{-4} for finetuning the last block and to 2×10^{-5} for finetuning the entire feature backbone. The learning rate for the temperature learning module $g(\cdot)$ is set to 1×10^{-4} for both configurations. We stop the gradient of $g(\cdot)$ from propagating to the feature backbone to prevent interference between the two modules. We finetune feature backbones for 100 epochs for PF-Pascal dataset and 10 epochs for SPair-71K dataset. We choose cross-entropy loss $\mathcal{L}_{CE}$ as the loss for its simplicity and follow [25] by applying a 5×5 gaussian kernel to smooth $\mathbf{P}_q^{Agt}$ during training. The standard deviation σ for $\mathbf{G}$ in Eq. (14) is set to 7. Images are resized to 256×256 for all datasets and backbones during training. We train all models on 3 Nvidia GTX 1080Ti GPUs.

4.3. Results

We summarize the comparison between existing methods and our method in Tab. 1. All listed methods are trained with strongly supervised loss. However, SFNet and NCNet are originally trained with weakly supervised losses. They are re-trained with the keypoint cross-entropy loss, denoted as SFNet-C.E. Loss and NCNet-C.E. Loss, for a fair comparison.

Table 1. Comparison between previous methods and SimSC on PF-Pascal, PF-Willow and SPair-71K datasets. For PF-Pascal, methods are trained and evaluated on its training and testing splits. For PF-Willow, methods are trained on the training data of PF-Pascal and evaluated on the testing data of PF-Willow. For SPair-71K, we evaluate the models transferred from PF-Pascal and denote results as **SPair-71K (T)** and models fine-tuned on training split of SPair-71K as **SPair-71K (F)**. All methods in the table use ResNet101 as the feature backbone and trained by keypoint supervised loss. We additionally include **SimSC-iBOT*** which fine-tunes iBOT, a more powerful ViT-based feature backbone using our method. The top four results for each column are picked out and ranked by color in bold: **first, second, third, fourth**.

Methods	Eval. Reso.	PF-Pascal α with θ_{img}			PF-Willow α with θ_{kps}			SPair-71K (T) α with θ_{bbox}		SPair-71K (F) α with θ_{bbox}	
		0.05	0.1	0.15	0.05	0.1	0.15	0.05	0.1	0.05	0.1
ANC-Net [25]	256	-	86.1	-	-	-	-	-	28.7	-	-
HPF [33]	max 300	60.1	84.8	92.7	45.9	74.4	85.6	-	-	-	-
SCOT [30]	max 300	63.1	85.4	92.7	47.8	76.0	87.1	-	-	-	-
DHPF [35]	240	75.7	90.7	95.0	41.4	67.4	81.8	15.4	27.4	20.7	37.3
MMNet [53]	224 $\times$ 320	77.6	89.1	94.3	-	-	-	-	-	-	40.9
CHM [32]	240	80.1	91.6	-	-	69.6	-	-	30.1	-	46.3
PMNC [24]	400	82.4	90.6	-	-	-	-	-	28.8	-	50.4
CATs [4]	256	75.4	92.6	96.4	40.9	69.5	83.2	13.6	27.0	27.9	49.9
TransforMatcher [20]	240	80.8	91.8	-	-	65.3	-	-	30.1	32.4	53.7
SemiMatch-CATs [19]	256	80.1	93.5	96.6	-	-	-	-	-	-	50.1
NCNet-C.E. Loss [38]	ori.	78.6	91.7	95.3	43.0	70.9	83.9	17.3	32.4	29.1	50.7
SFNet-C.E. Loss [22]	ori.	78.7	92.9	96.0	43.2	72.5	85.9	13.3	27.9	26.2	50.0
PWarpC-CATs [48]	ori.	79.8	92.6	96.4	48.1	75.1	86.6	15.4	27.9	28.2	48.4
PWarpC-NCNet [48]	ori.	79.2	92.1	95.6	48.0	76.2	86.8	21.5	37.1	31.6	52.0
PWarpC-DHPF [48]	ori.	79.1	91.3	96.1	48.5	74.4	85.4	16.4	28.6	23.3	38.7
PWarpC-SFNet [48]	ori.	78.3	92.2	96.2	47.5	77.7	88.8	17.3	32.5	27.0	48.8
SCorrSAN [18]	256	81.5	93.3	96.6	-	-	-	-	-	-	55.3
SimSC-ResNet101	256	84.4	92.3	95.6	37.1	60.6	73.3	18.1	33.5	30.9	51.0
SimSC-ResNet101	ori.	82.2	92.7	95.2	36.1	63.5	76.6	24.5	36.8	35.4	51.0
SimSC-ResNet101*	256	86.2	92.6	95.3	42.5	65.4	78.1	18.2	31.4	31.3	51.6
SimSC-ResNet101*	ori.	85.3	93.3	96.0	40.7	68.0	80.6	24.5	36.0	36.6	51.5
SimSC-iBOT*	256	88.4	95.6	97.3	44.9	71.4	84.5	22.0	37.9	43.0	63.5

Following the convention in the literature, we report four results: finetuned result on PF-Pascal (**PF-Pascal**), transferred result from PF-Pascal to PF-Willow (**PF-Willow**), transferred result from PF-Pascal to SPair-71K (**SPair-71K (T)**), and finetuned result on SPair-71K (**SPair-71K (F)**). The majority of the existing methods are evaluated on either an image size of 256×256 or the original image size, so we evaluate **SimSC-ResNet101** and **SimSC-ResNet101*** on both sizes. Under the same feature backbone, we achieve accuracy on par with state-of-the-art methods despite the remarkable simplicity of our method. We achieve the best results at threshold $\alpha = 0.05$ for PF-Pascal, SPair-71K (T) and SPair-71K (F). We note that our transferred result from PF-Pascal to PF-Willow is not as significant as to SPair-71K. At the same time, we also notice that methods that transfer strongly from PF-Pascal to PF-Willow tend to transfer deficiently from PF-Pascal to SPair-71K. We suspect that this is because the data distributions between PF-Willow and SPair-71K are different and represent two directions of generalization. Moving toward one means moving away from the other. We additionally include the results of finetuning a more powerful backbone named iBOT [54], which has the ViT-B/16 architecture, denoted as **SimSC-iBOT***. As shown, it outperforms other methods on PF-

Pascal, SPair-71K (T) and SPair-71K (F) at all thresholds by a notable margin.

Focusing on SimSC itself, finetuning the entire feature backbone shows more promise than finetuning only the last block. We also notice that performance using the original images is better than using squared images. We believe this is attributed to the fact that objects are not distorted if using original images.

4.4. Analysis

We include several additional analyses of our method in this section. All evaluation is done with an image size of 256×256 . A model with the symbol “*” indicates finetuning the entire feature backbone, otherwise finetuning only the last block. More experiments and qualitative results can be found in the supplementary materials.

4.4.1 Learning curve of temperature

We provide the plots of temperatures, validation accuracy and loss of ResNet101*-SimSC on PF-Pascal in Fig. 6 to show how the temperature and backbone are jointly updated. As illustrated, β_A and β_B have very similar learning curves. From 0-2k step, the temperature and loss drop rapidly and accuracy is improved, due to the matching dis-

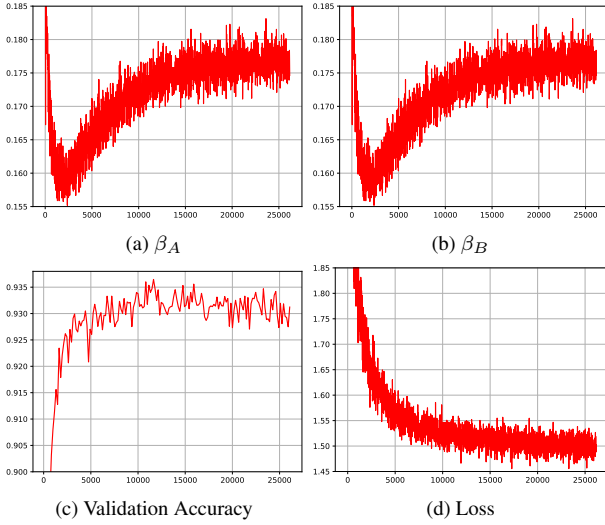


Figure 6. Learning curves of β_A , β_B , validation accuracy and loss of SimSC-ResNet101* on the PF-Pascal dataset. We train the model for 200 epochs in order to provide the entire trend of the curve.

tribution drawn closer to the target by both reducing temperature and learning better features. From 2k-12k step, the temperature increases while accuracy keeps improving. This shows that the temperature gradually changes to its optimal position and helps tuning of the backbone. The accuracy starts dropping after 12k step due to overfitting and the temperature is stable after 20k step.

4.4.2 Effectiveness on different feature backbones

We test our method on four feature backbones: ResNet50 [15], ResNet101 [15], DINO [3] and iBOT [54]. ResNet50 and ResNet101 are CNN-based, and DINO and iBOT have the ViT-B/8 and ViT-B/16 architecture respectively. Following the setup in Sec. 4.2, we finetune both the last block and the entire module for each type of feature backbone, except for DINO, where we only finetune the last block. This is because the image-to-feature spatial size ratio of DINO is 8 rather than 16 (the latter one is the ratio of all other feature backbones). The memory cost would be too large to fit into our GPU if we finetune the entire DINO under the same image size. We summarize the performance of each type of feature backbone in Tab. 2. All evaluation metrics are the same as described in Sec. 4.1.

As shown in Tab. 2, our method works well on all four feature backbones. iBOT appears to be the most effective one due to its raw features’ superiority and transformer architecture. DINO takes the second position in PF-Pascal and SPair-71K (F), followed by ResNet101 and ResNet50.

Table 2. Performance of SimSC using different feature backbones.

Feat. Ext.	PF-Pas. $\alpha = 0.1$	PF-Wil. $\alpha = 0.1$	S.P. (T) $\alpha = 0.1$	S.P. (F) $\alpha = 0.1$
Res101	92.3	60.6	33.5	51.0
Res101*	92.6	65.4	31.4	51.6
Res50	90.4	59.7	31.3	47.0
Res50*	91.3	62.5	28.5	45.7
DINO	92.8	62.2	28.3	52.6
iBOT	94.9	65.4	36.0	61.8
iBOT*	95.6	71.4	37.9	63.5

Table 3. Comparison between finetuning ResNet101* on PF-Pascal using manually set β_{trn} and SimSC.

Losses	PF-Pascal $\alpha = 0.1$		
	$\beta_{trn} = 0.02$	SimSC	β by SimSC
$\mathcal{L}_{CE}$	92.1	92.6	0.03
$\mathcal{L}_{kps}$	89.7	91.5	0.07
$\mathcal{L}_{PL2}$	89.3	91.5	0.09

4.4.3 Temperature Learning on Different Losses

We compare the results of finetuning ResNet101* on PF-Pascal by a manually set temperature and SimSC using different training losses mentioned in Sec. 3.2. The results are presented in Tab. 3. As demonstrated, SimSC consistently outperforms manually set β_{trn} across all three types of losses. This is due to the fact that the optimal β_{trn} is different for each type of loss, and SimSC is able to track the optimal β under different settings. With $\mathcal{L}_{CE}$, the SimSC yields a similar result to $\beta_{trn} = 0.02$ which indicates that the optimal β for $\mathcal{L}_{CE}$ is around 0.02 to 0.03. However, for $\mathcal{L}_{kps}$ and $\mathcal{L}_{PL2}$, SimSC produces a more noticeable gain, implying β learned by SimSC is closer to their respective optimal temperature.

5. Conclusion

We propose a remarkably simple framework that tackles the problem of semantic matching. We demonstrate that, with an appropriate temperature value for softmax, commonly used feature backbones, such as ResNet, can surprisingly learn accurate correspondences by themselves, without any additional module or multi-layer exploitation. In order to avoid manually tuning the temperature, we propose a temperature learning module that is jointly optimized with feature extractors. We evaluate our method on three public datasets and achieve accuracy on par with state-of-the-art methods built on the same backbone but with complicated downstream task modules. By coupling with iBOT, a more powerful self-supervised feature extractor, we outperform existing methods on the PF-Pascal and SPair-71K datasets by a large margin.

References

- [1] Atish Agarwala, Jeffrey Pennington, Yann Dauphin, and Sam Schoenholz. Temperature check: theory and practice for training models with softmax-cross-entropy losses. *arXiv preprint arXiv:2010.07344*, 2020. 2
- [2] Shir Amir, Yossi Gandelsman, Shai Bagon, and Tali Dekel. Deep vit features as dense visual descriptors. *arXiv preprint arXiv:2112.05814*, 2021. 2
- [3] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *ICCV*, 2021. 6, 8
- [4] Seokju Cho, Sunghwan Hong, Sangryul Jeon, Yunsung Lee, Kwanghoon Sohn, and Seungryong Kim. Cats: Cost aggregation transformers for visual correspondence. In *NeurIPS*, 2021. 1, 2, 4, 7
- [5] Christopher B Choy, JunYoung Gwak, Silvio Savarese, and Manmohan Chandraker. Universal correspondence network. *Advances in neural information processing systems*, 29, 2016. 1, 2, 3
- [6] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *CVPR*, 2005. 2
- [7] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009. 1
- [8] Jiankang Deng, Jia Guo, Niannan Xue, and Stefanos Zafeiriou. Arcface: Additive angular margin loss for deep face recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4690–4699, 2019. 2
- [9] Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Superpoint: Self-supervised interest point detection and description. In *CVPR*, 2018. 2
- [10] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*, 2021. 6
- [11] Mihai Dusmanu, Ignacio Rocco, Tomas Pajdla, Marc Pollefeys, Josef Sivic, Akihiko Torii, and Torsten Sattler. D2-net: A trainable cnn for joint description and detection of local features. In *CVPR*, 2019. 2
- [12] Bumsu Ham, Minsu Cho, Cordelia Schmid, and Jean Ponce. Proposal flow. In *CVPR*, 2016. 4, 6
- [13] Bumsu Ham, Minsu Cho, Cordelia Schmid, and Jean Ponce. Proposal flow: Semantic correspondences from object proposals. *IEEE TPAMI*, 2017. 6
- [14] Kai Han, Rafael S Rezende, Bumsu Ham, Kwan-Yee K Wong, Minsu Cho, Cordelia Schmid, and Jean Ponce. Snet: Learning semantic correspondence. In *ICCV*, 2017. 2
- [15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016. 1, 2, 6, 8
- [16] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015. 2
- [17] Shuaiyi Huang, Qiuyue Wang, Songyang Zhang, Shipeng Yan, and Xuming He. Dynamic context correspondence network for semantic alignment. In *ICCV*, 2019. 1
- [18] Shuaiyi Huang, Luyu Yang, Bo He, Songyang Zhang, Xuming He, and Abhinav Shrivastava. Learning semantic correspondence with sparse annotations. In *ECCV*, 2022. 1, 7
- [19] Jiwon Kim, Kwangrok Ryoo, Junyoung Seo, Gyuseong Lee, Daehwan Kim, Hansang Cho, and Seungryong Kim. Semi-supervised learning of semantic correspondence with pseudo-labels. In *CVPR*, 2022. 2, 4, 7
- [20] Seungwook Kim, Juhong Min, and Minsu Cho. Transformer: Match-to-match attention for semantic correspondence. In *CVPR*, 2022. 1, 2, 3, 4, 7
- [21] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 6
- [22] Junhyup Lee, Dohyung Kim, Jean Ponce, and Bumsu Ham. Sfnets: Learning object-aware semantic correspondence. In *CVPR*, 2019. 1, 2, 3, 4, 5, 7
- [23] Junhyup Lee, Dohyung Kim, Wonkyung Lee, Jean Ponce, and Bumsu Ham. Learning semantic correspondence exploiting an object-level prior. *IEEE TPAMI*, 2020. 4
- [24] Jae Yong Lee, Joseph DeGol, Victor Fragoso, and Sudipta N Sinha. Patchmatch-based neighborhood consensus for semantic correspondence. In *CVPR*, 2021. 1, 7
- [25] Shuda Li, Kai Han, Theo W Costain, Henry Howard-Jenkins, and Victor Prisacariu. Correspondence networks with adaptive neighbourhood consensus. In *CVPR*, 2020. 1, 2, 3, 4, 6, 7
- [26] Xinghui Li, Kai Han, Shuda Li, and Victor Prisacariu. Dual-resolution correspondence networks. In *NeurIPS*, 2020. 2, 3, 4, 1
- [27] Xin Li, Deng-Ping Fan, Fan Yang, Ao Luo, Hong Cheng, and Zicheng Liu. Probabilistic model distillation for semantic correspondence. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7505–7514, 2021. 2
- [28] Zhengqi Li and Noah Snavely. Megadepth: Learning single-view depth prediction from internet photos. In *CVPR*, 2018. 1
- [29] Ce Liu, Jenny Yuen, and Antonio Torralba. Sift flow: Dense correspondence across scenes and its applications. *IEEE TPAMI*, 2010. 2
- [30] Yanbin Liu, Linchao Zhu, Makoto Yamada, and Yi Yang. Semantic correspondence as an optimal transport problem. In *CVPR*, 2020. 2, 7
- [31] David G Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 2004. 2
- [32] Juhong Min and Minsu Cho. Convolutional hough matching networks. In *CVPR*, 2021. 1, 2, 3, 4, 7
- [33] Juhong Min, Jongmin Lee, Jean Ponce, and Minsu Cho. Hyperpixel flow: Semantic correspondence with multi-layer neural features. In *ICCV*, 2019. 1, 2, 7
- [34] Juhong Min, Jongmin Lee, Jean Ponce, and Minsu Cho. Spair-71k: A large-scale benchmark for semantic correspondence. *arXiv preprint arXiv:1908.10543*, 2019. 6

- [35] Juhong Min, Jongmin Lee, Jean Ponce, and Minsu Cho. Learning to compose hypercolumns for visual correspondence. In *ECCV*, 2020. 1, 2, 7
- [36] Yuki Ono, Eduard Trulls, Pascal Fua, and Kwang Moo Yi. Lf-net: Learning local features from images. *Advances in neural information processing systems*, 31, 2018. 2
- [37] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *NeurIPS*, 2019. 6
- [38] Ignacio Rocco, Mircea Cimpoi, Relja Arandjelović, Akihiko Torii, Tomas Pajdla, and Josef Sivic. Neighbourhood consensus networks. In *NeurIPS*, 2018. 1, 2, 3, 7
- [39] Paul-Edouard Sarlin, Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Superglue: Learning feature matching with graph neural networks. In *CVPR*, 2020. 2
- [40] Aleksandar Shtedritski, Andrea Vedaldi, and Christian Rupprecht. Learning universal semantic correspondences with no supervision and automatic data curation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops*, pages 933–943, 2023. 2
- [41] Edgar Simo-Serra, Eduard Trulls, Luis Ferraz, Iasonas Kokkinos, Pascal Fua, and Francesc Moreno-Noguer. Discriminative learning of deep convolutional feature point descriptors. In *ICCV*, 2015. 2
- [42] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *CVPR*, 2014. 2
- [43] Kihyuk Sohn. Improved deep metric learning with multi-class n-pair loss objective. *Advances in neural information processing systems*, 29, 2016. 2
- [44] Jiaming Sun, Zehong Shen, Yuang Wang, Hujun Bao, and Xiaowei Zhou. Loftr: Detector-free local feature matching with transformers. In *CVPR*, 2021. 2, 1
- [45] Yurun Tian, Bin Fan, and Fuchao Wu. L2-net: Deep learning of discriminative patch descriptor in euclidean space. In *CVPR*, 2017. 2
- [46] Yurun Tian, Xin Yu, Bin Fan, Fuchao Wu, Huub Heijnen, and Vassileios Balntas. Sosnet: Second order similarity regularization for local descriptor learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11016–11025, 2019. 2
- [47] Georgi Tinchev, Shuda Li, Kai Han, David Mitchell, and Rigas Kouskouridas. Xresolution correspondence networks. In *BMVC*, 2021. 2
- [48] Prune Truong, Martin Danelljan, Fisher Yu, and Luc Van Gool. Probabilistic warp consistency for weakly-supervised semantic correspondences. In *CVPR*, 2022. 2, 7
- [49] Feng Wang, Xiang Xiang, Jian Cheng, and Alan Loddon Yuille. Normface: L2 hypersphere embedding for face verification. In *Proceedings of the 25th ACM international conference on Multimedia*, pages 1041–1049, 2017. 2
- [50] Hao Wang, Yitong Wang, Zheng Zhou, Xing Ji, Dihong Gong, Jingchao Zhou, Zhifeng Li, and Wei Liu. Cosface: Large margin cosine loss for deep face recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5265–5274, 2018. 2
- [51] Dingyi Zhang, Yingming Li, and Zhongfei Zhang. Deep metric learning with spherical embedding. *Advances in Neural Information Processing Systems*, 33:18772–18783, 2020. 2
- [52] Xu Zhang, Felix Xinnan Yu, Svebor Karaman, Wei Zhang, and Shih-Fu Chang. Heated-up softmax embedding. *arXiv preprint arXiv:1809.04157*, 2018. 2
- [53] Dongyang Zhao, Ziyang Song, Zhenghao Ji, Gangming Zhao, Weifeng Ge, and Yizhou Yu. Multi-scale matching networks for semantic correspondence. In *CVPR*, 2021. 1, 4, 7
- [54] Jinghao Zhou, Chen Wei, Huiyu Wang, Wei Shen, Cihang Xie, Alan Yuille, and Tao Kong. ibot: Image bert pre-training with online tokenizer. In *ICLR*, 2022. 2, 6, 7, 8

SimSC: A Simple and Effective Framework for Semantic Correspondence

Supplementary Material

Table 4. Evaluation on MegaDepth.

	AUC@5	AUC@10	AUC@20
$\beta = 0.04$	36.6	52.8	66.6
$\beta = 1.0$	6.9	14.7	26.8
DualRCNet	27.0	42.9	58.3
LoFTR	52.8	69.2	81.2

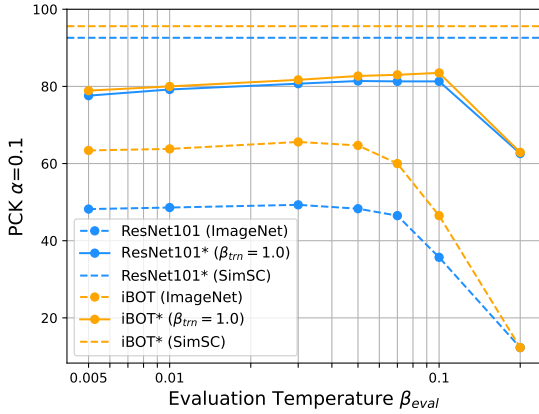


Figure 7. Comparison between ImageNet pre-trained models, fine-tuned models with $\beta_{trn} = 1$, and SimSC on PF-Pascal dataset with different β_{eval} .

6. Additional Experiments

6.1. Role of Temperature on Geometric Matching

To investigate the impact of temperature on geometric matching, we train and evaluate our method on MegaDepth [28] dataset. We follow the popular dual-resolution architecture (LoFTR [44]/DualRC-Net [26]), using the lightweight ResNet18 to learn both coarse-level matching and fine-level offset without any additional modules. We use cross-entropy loss to train both coarse-level prediction and fine-level offset. We tried two temperature values 0.04 and 1.0 and followed LoFTR reporting the accuracy of relative pose estimation in terms of AUC in Table 4. As shown, a small temperature significantly improves the performance. It achieves superior results compared with the widely recognized DualRC-Net and achieves over 70% of LoFTR performance without any transformer module. Further improvement is possible by increasing the batch size or carefully searching hyperparameters, as LoFTR did.

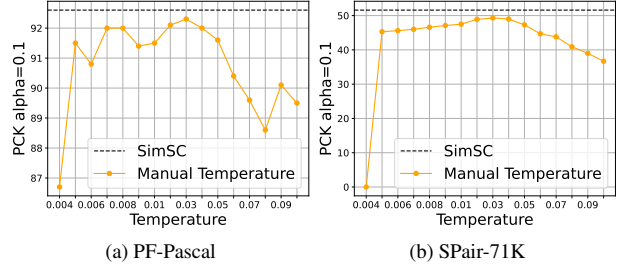


Figure 8. Comparison between fine-tuning ResNet101* with different manual temperatures and SimSC on the PF-Pascal and SPair-71K datasets. For each sample of manual temperature, the evaluation temperature is set to be the same as the training temperature.

6.2. Comparison with the finetuning baselines.

We compare the performance of ImageNet pre-trained backbones, fine-tuned backbones with $\beta_{trn} = 1$, and fine-tuned backbones using SimSC. The accuracy of kernel softmax is significantly affected by β_{eval} . Therefore, we evaluate ImageNet pre-trained backbones and fine-tuned backbones with $\beta_{trn} = 1$ using different β_{eval} on the PF-Pascal dataset and plot the results in Fig. 7. Our method improves the results of fine-tuning both ResNet101* and iBOT* by 11.2 percentage points and 12 percentage points respectively when compared with fine-tuning backbone with training temperature $\beta_{trn} = 1$. The performance of ImageNet pre-trained models and fine-tuned models with $\beta_{trn} = 1$ drops significantly when $\beta_{eval} > 0.1$. This is because β_{eval} is too big to suppress incorrect candidates and they keep too much weight in the marginalization in Eq. (15).

6.3. Manual temperature values for training

We fine-tune ResNet101* with different manual β_{trn} on the PF-Pascal and SPair-71K datasets and plot the results together with results of SimSC in Fig. 8. The evaluation temperature β_{eval} is set to be the same as β_{trn} for each sample. At $\beta_{trn} = 0.004$, we obtain NaN error during training in both PF-Pascal and SPair-71K datasets, which is the reason for the poor performance at this temperature. We suspect that this is because the temperature is too small and the gradient around the maximum is too steep, causing instability during training. The performance increases until reaching the peak at $\beta_{trn} = 0.03$ and starts decreasing. The proposed temperature learning method can successfully predict the optimal temperatures across different datasets without the exhaustive grid search.

Table 5. Effect of different learning rate in fine-tuned feature extractors with $\beta_{trn} = 1$. Apart from “SimSC”, all models are fine-tuned on PF-Pascal dataset with $\beta_{trn} = 1$ and evaluated with $\beta_{eval} = 0.02$. For models trained with learning rate of 2×10^{-5} , we also include the results that are read from Fig. 7 in the bracket because $\beta_{eval} = 0.02$ is not the optimal value to them.

ResNet101*		iBOT*	
LR	PF-Pascal $\alpha = 0.1$	LR	PF-Pascal $\alpha = 0.1$
2×10^{-6}	79.9	2×10^{-6}	76.3
2×10^{-5}	80.3 (81.4)	2×10^{-5}	81.3 (83.5)
2×10^{-4}	78.0	2×10^{-4}	74.6
2×10^{-3}	63.5	2×10^{-3}	4.5
2×10^{-2}	54.6	2×10^{-2}	4.4
SimSC	92.6	SimSC	95.6

6.4. Learning-rate vs temperature effects

As the temperature is important for the learning process by affecting the training signal (thus the gradients), one may wonder whether the same effects of the temperature can be trivially achieved by using a different learning rate. To validate this, we fine-tune ResNet101* and iBOT* with learning rates from 2×10^{-6} to 2×10^{-2} on PF-Pascal dataset using the baseline fine-tuning framework with $\beta_{trn} = 1$ and summarize the results in Tab. 5. As demonstrated, with $\beta_{trn} = 1$, the baseline framework cannot achieve the same performance as our method regardless of the learning rate. This is because the update rule of the Adam optimizer is not linearly proportional to the gradient, hence the role of β_{trn} cannot be replaced by adjusting the learning rate.

6.5. The single-parameter design

We explore a simplified design of the temperature learning module that consists of one single learnable parameter β_e , which takes no input and does not depend on anything. The effective temperature is replaced by β_e^2 . We evaluate this simplified design on ResNet101* and iBOT*. We notice that the single-parameter design is sensitive to the learning rate of the parameter and requires careful tuning. We experiment with several learning rates on the PF-Pascal dataset using ResNet101* and choose the best value 0.005 based on the performance on the validation set. For a fair comparison with the MLP design of the main manuscript, we apply this value to both ResNet101* and iBOT* across all datasets. The rest of the configuration is the same as in Sec. 4.2 in the main manuscript. We summarize the results of both the single-parameter design and the MLP design in Tab. 6.

On the PF-Pascal dataset, the single-parameter design and the MLP design achieve close results in ResNet101*, provided that the MLP design slightly outperforms the single-parameter design on PF-Pascal and SPair-71K (F) at the original image size. However, the MLP design notice-

ably outperforms the single-parameter design in iBOT* on all three datasets. When switching to the single-parameter design, we notice a moderate drop in the generalizability of transferring iBOT* trained on PF-Pascal to PF-Willow and SPair-71K. This is because different feature extractors may have different learning curves, so the learning rate found for ResNet101* may not be optimal for iBOT*. The single-parameter design does not have the contextual information of the feature map in the temperature learning. The temperature update solely depends on the learning rate. The optimal learning rate may depend on the architecture and dataset.

We also include the learned effective temperature of each feature extractor on each dataset in Tab. 7 for reference. For the MLP design, since different image pairs have different temperatures, we calculate the mean and the standard deviation of temperatures of all image pairs in testing set of each dataset.

6.6. Qualitative examples

We include more qualitative examples in Fig. 9 - Fig. 13.

Table 6. Comparison between the single-parameter design and the MLP design of the temperature learning module $g(\cdot)$. The evaluation metric is the same as described Sec. 4.1 and Table 1 of the main manuscript.

Feat. Ext. (Eval. Size)	$g(\cdot)$	PF-Pascal $\alpha(0.05/0.1/0.15)$	PF-Willow $\alpha(0.05/0.1/0.15)$	SPair-71K (T) $\alpha(0.05/0.1)$	SPair-71K (F) $\alpha(0.05/0.1)$
ResNet101* (256)	single param.	85.1 / 92.6 / 94.6	42.2 / 65.7 / 78.3	18.0 / 31.0	33.1 / 51.1
	MLP	86.2 / 92.6 / 95.3	42.5 / 65.4 / 78.1	18.2 / 31.4	31.3 / 51.6
ResNet101* (ori.)	single param.	85.0 / 92.8 / 95.5	39.5 / 68.5 / 82.2	24.5 / 35.9	35.6 / 48.2
	MLP	85.3 / 93.3 / 96.0	40.7 / 68.0 / 80.6	24.5 / 36.0	36.6 / 51.5
iBOT* (256)	single param.	89.7 / 94.9 / 96.7	44.3 / 68.9 / 82.2	20.1 / 33.4	43.3 / 62.0
	MLP	88.4 / 95.6 / 97.3	44.9 / 71.4 / 84.5	22.0 / 37.9	43.0 / 63.5

Table 7. Comparison between the effective temperatures of the single-parameter design and the MLP design. For the MLP design, we calculate the mean and the standard deviation of temperatures for all image pairs in the testing set of each dataset. They are presented in the format of “mean $\pm$ standard deviation”.

Feat. Ext. (Eval. Size)	$g(\cdot)$	PF-Pascal	PF-Willow	SPair-71K (T)	SPair-71K (F)
ResNet101* (256)	single param.	0.0333	0.0333	0.0333	0.0379
	MLP	0.0303 $\pm$ 0.0017	0.0308 $\pm$ 0.0017	0.0305 $\pm$ 0.0015	0.0341 $\pm$ 0.0030
ResNet101* (ori.)	single param.	0.0333	0.0333	0.0333	0.0379
	MLP	0.0307 $\pm$ 0.0018	0.0314 $\pm$ 0.0025	0.0370 $\pm$ 0.0016	0.0408 $\pm$ 0.0031
iBOT* (256)	single param.	0.0289	0.0289	0.0289	0.0352
	MLP	0.0251 $\pm$ 0.0025	0.0254 $\pm$ 0.0025	0.2585 $\pm$ 0.0030	0.0324 $\pm$ 0.0037

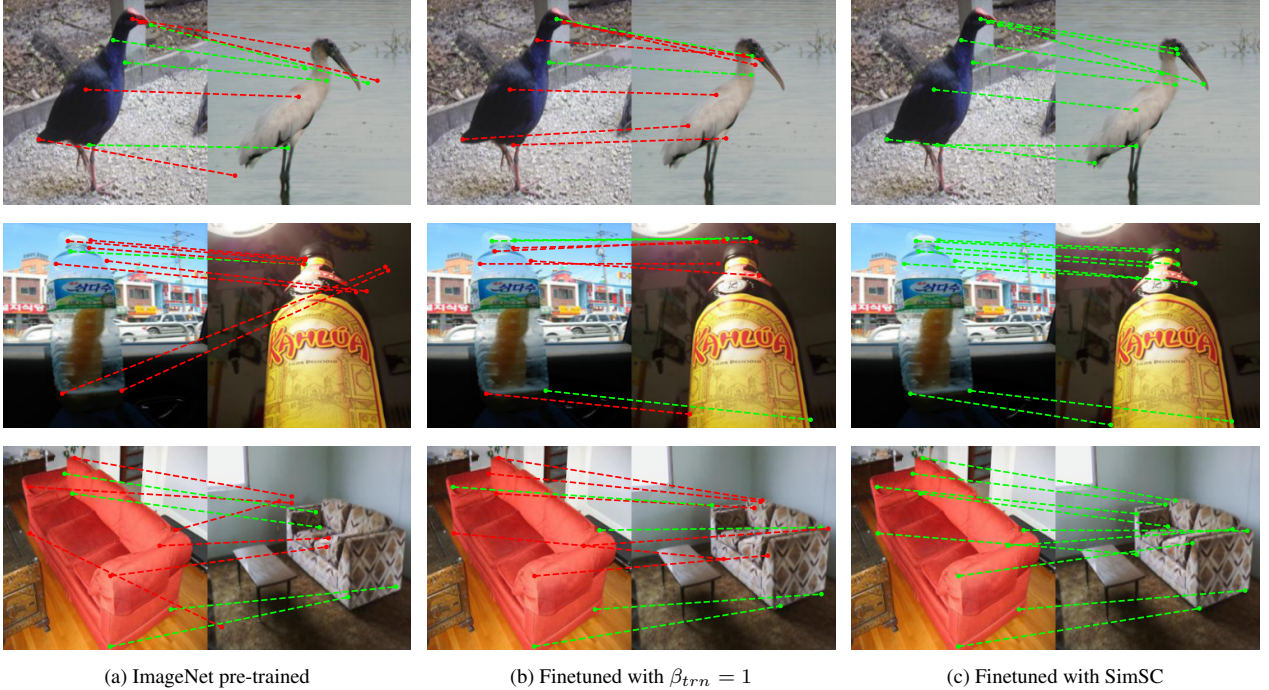


Figure 9. Comparison between semantic matching of ImageNet pre-trained ResNet101*, finetuned ResNet101* with $\beta_{trn} = 1$ and SimSC-ResNet101* on the PF-Pascal dataset.

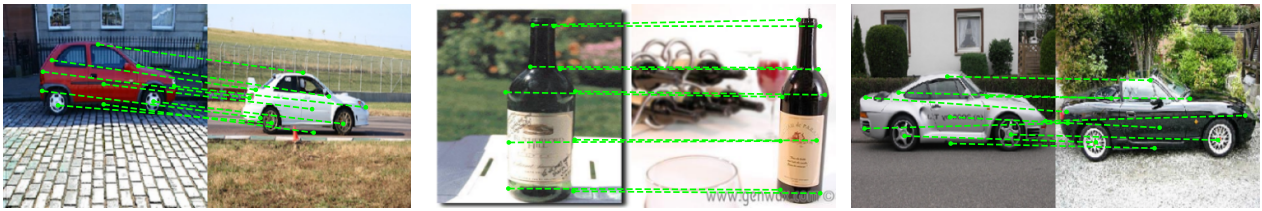


Figure 10. Examples of semantic matching by SimSC-ResNet101* on the PF-Willow dataset.

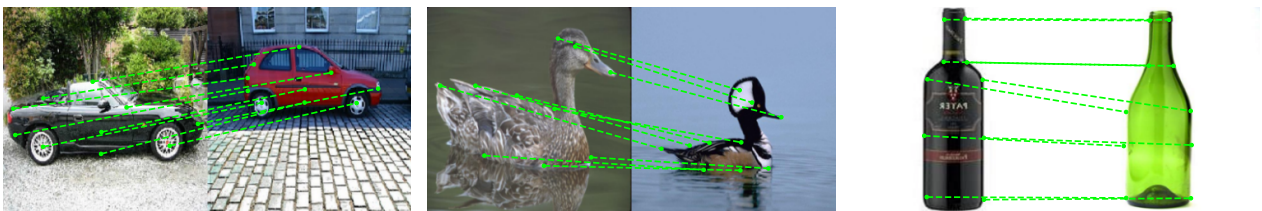


Figure 11. Examples of semantic matching by SimSC-iBOT* on the PF-Willow dataset.

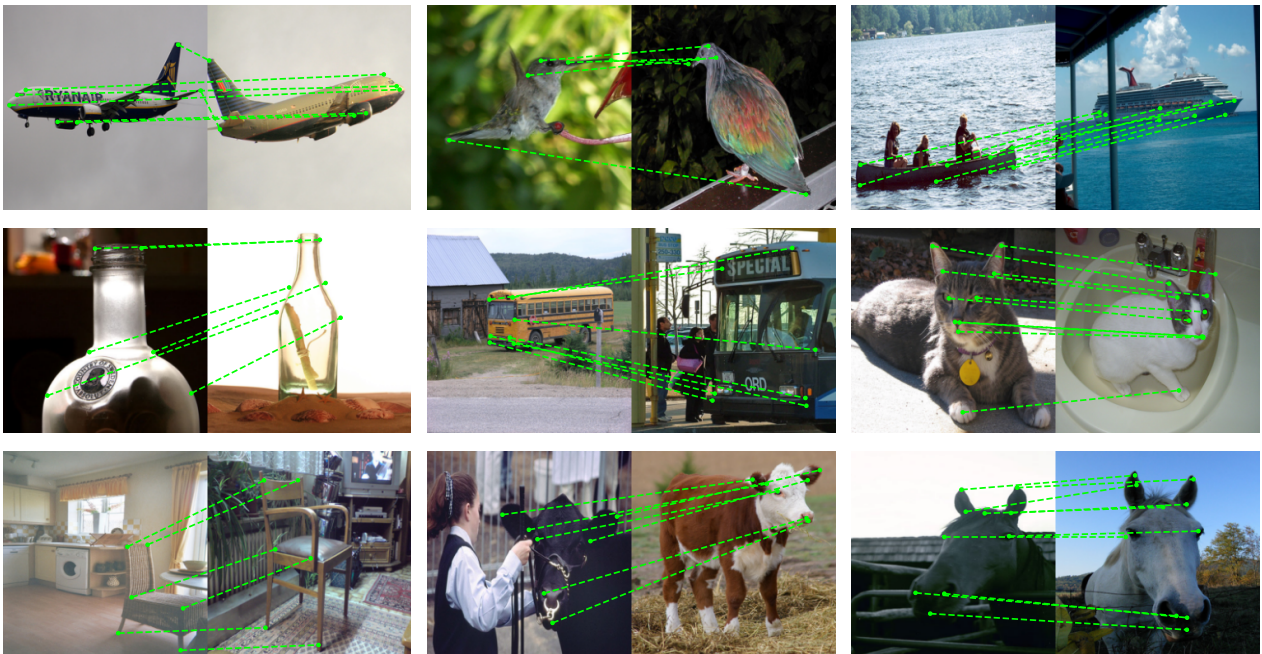


Figure 12. Examples of semantic matching by SimSC-ResNet101* on the SPair-71K dataset.

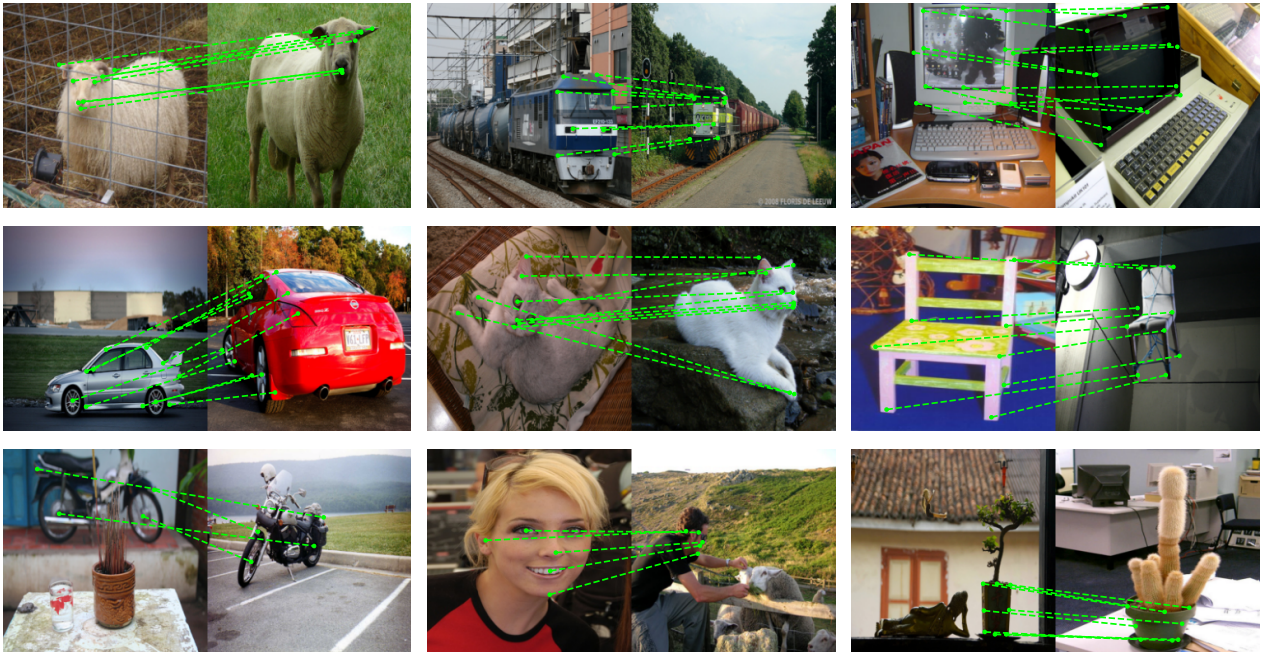


Figure 13. Examples of semantic matching by SimSC-iBOT* on the SPair-71K dataset.

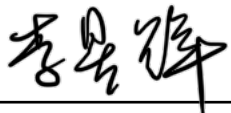
Statement of Authorship for joint/multi-authored papers for PGR thesis

To appear at the end of each thesis chapter submitted as an article/paper

The statement shall describe the candidate's and co-authors' independent research contributions in the thesis publications. For each publication there should exist a complete statement that is to be filled out and signed by the candidate and supervisor (**only required where there isn't already a statement of contribution within the paper itself**).

Title of Paper	SimSC: A Simple and Effective Framework for Semantic Correspondence
Publication Status	☐ Published ☐ Accepted for Publication ☒ Submitted for Publication ☐ Unpublished and unsubmitted work written in a manuscript style
Publication Details	Xinghui Li , Kai Han, Xingchen Wan, Victor Adrian Prisacariu. In arXiv preprint arXiv:2305.02385, 2023

Student Confirmation

Student Name:	Xinghui Li		
Contribution to the Paper	First Author Contribution: • Conception of research ideas. • Design and implementation of models. • Running of large-scale experiments. • Writing and presentation of the paper.		
Signature		Date	07/Jan/2025

Supervisor Confirmation

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

Supervisor name and title:	Prof Victor Adrian Prisacariu		
Supervisor comments	I certify that the student made a substantial contribution to the chapter and that the description described above is accurate.		
Signature		Date	9/1/25

This completed form should be included in the thesis, at the end of the relevant chapter.

4.1.3 Conclusion

We proposed a simple framework that tackled the problem of semantic matching. We demonstrated that, with an appropriate temperature value for softmax, commonly used feature backbones such as ResNet [25] could surprisingly learn accurate correspondences by themselves, without any additional module or multi-layer exploitation. In order to avoid manually tuning the temperature, we proposed a temperature learning module that was jointly optimized with the feature extractors. We evaluated our method on three public datasets and achieved accuracy on par with methods built on the same backbone but with much more complex matching modules. Our proposed pipeline is versatile and does not depend on any specific type of model. Therefore, we adopt the same pipeline when we investigate the prompt tuning of the Stable Diffusion model in the next section, where experiments further confirm the value of our pipeline.

4.2 SD4Match: Learning to prompt Stable Diffusion Model for Semantic Matching

4.2.1 Introduction

This section aims to how to leverage a large-scale vision foundation model to tackle semantic matching problems. Stable Diffusion (SD) [27] has demonstrated good results in generating high-quality images from textual prompts. Recent studies have shown that the model excels not only in generative tasks but also in discriminative vision tasks, such as image segmentation [136] and monocular depth estimation [137]. In the realm of semantic matching, early research [138, 39, 7] has demonstrated that SD performs on par with methods specifically designed for this task. However, these approaches primarily explore the capabilities of SD from a zero-shot perspective. We are therefore interested in adapting the SD model to semantic matching in a more efficient manner, avoiding traditional full-model fine-tuning.

Prompt tuning has been used to effectively adapt pre-trained vision-language models, such as CLIP [139], to new data domains in tasks like image classification [140, 141]. Inspired by this, we propose directly optimizing the prompt input of the SD model on the semantic matching objective using a similar metric learning framework introduced in Section 4.1. Despite its straightforward nature, we find that learning a single, universal prompt applicable to all images is already highly effective in adapting SD to semantic matching, and not only improves previous SD-based semantic matchers [39, 7] but also leads to the state-of-the-art performance over all types of methods.

Furthermore, we observe that including prior information in the prompt about the objects to be matched significantly affects overall matching performance. We

thus introduce two additional prompt tuning schemes: one that leverages explicit prior semantic knowledge by learning distinct prompts for each object category, and the other that uses a novel conditional prompting module to condition the prompt on the local feature patches of both matching images. Experiments show that these modules lead to further improvements in matching accuracy.

To summarize, our contributions in this section are:

1. We demonstrate that the performance of Stable Diffusion in the semantic matching task can be significantly enhanced using a straightforward prompt tuning technique.
2. We propose a novel conditional prompting module, which uses the local features of the image pair to infuse information of the pair to the prompt. This design further improve the matching accuracy.
3. We evaluate our method on three public datasets, establishing new accuracy benchmarks for each. Notably, we achieve an increase of 12 percentage points on the challenging SPair-71k dataset [35], surpassing the previous state-of-the-art.

We published this work in CVPR 2024 [41].

SD4Match: Learning to Prompt Stable Diffusion Model for Semantic Matching

Xinghui Li[†] Jingyi Lu[‡] Kai Han^{‡*} Victor Adrian Prisacariu[†]
[†]University of Oxford [‡]University of Hong Kong
 xinghui@robots.ox.ac.uk jingyi@connect.hku.hk
 kaihax@hku.hk victor@robots.ox.ac.uk

Abstract

In this paper, we address the challenge of matching semantically similar keypoints across image pairs. Existing research indicates that the intermediate output of the UNet within the Stable Diffusion (SD) can serve as robust image feature maps for such a matching task. We demonstrate that by employing a basic prompt tuning technique, the inherent potential of Stable Diffusion can be harnessed, resulting in a significant enhancement in accuracy over previous approaches. We further introduce a novel conditional prompting module that conditions the prompt on the local details of the input image pairs, leading to a further improvement in performance. We designate our approach as SD4Match, short for Stable Diffusion for Semantic Matching. Comprehensive evaluations of SD4Match on the PF-Pascal, PF-Willow, and SPair-71k datasets show that it sets new benchmarks in accuracy across all these datasets. Particularly, SD4Match outperforms the previous state-of-the-art by a margin of 12 percentage points on the challenging SPair-71k dataset. Code is available at the project website: <https://sd4match.active.vision/>.

1. Introduction

Matching keypoints between two semantically similar objects is one of the challenges in computer vision. The difficulties arise from the fact that semantic correspondences may exhibit significant visual dissimilarity due to occlusion, different viewpoints, and intra-category appearance differences. Although significant progress has been made [4, 14, 26, 30, 37, 44], the problem is far from being completely solved. Recently, Stable Diffusion (SD) [46] has demonstrated a remarkable ability to generate high-quality images based on input textual prompts. Looking specifically at semantic matching, follow-up studies [50, 55, 56] have further revealed that SD is not only proficient in generative tasks but also applicable to feature extraction. Ex-

periments demonstrate that SD can perform on par with methods especially designed for semantic matching, paving a new direction in this field. This brings up a yet unanswered question: Have we fully explored the capacity of SD in matching? Or, how should we harness the information gathered from billions of images stored within SD to further improve its performance?

Engineering the textual prompt has already been extensively utilized in numerous computer vision tasks, including image generation using Stable Diffusion. In these applications, prompts are meticulously handcrafted to achieve the desired output. Prompt tuning, or direct optimization of prompt embedding, has also been utilized to adapt pre-trained vision-language models, such as CLIP [43], to new data domains in tasks like image classification, especially when faced with limited data resources [21, 59]. Inspired by the latter strategy, and given that the accuracy of matching is quantifiable and limiting the prompt to the textual domain is unnecessary, we can directly optimize the prompt on the latent space to exploit SD’s potential for semantic matching. In spite of its straightforward nature, we find that learning a single, universal prompt applicable to all images is already highly effective in adapting SD to semantic matching, and not only improves previous SD-based semantic matchers [50, 55] but also leads to the state-of-the-art performance over all types of methods.

Current works that explore SD for visual perception tasks mimic the textual prompt input of standard image generation SD, by either handcrafting a text input [50] or by using an implicit captioner [54]. Novel to our work, we find that the choice for prompt, text, or otherwise, particularly when including prior information, significantly influences the overall matching performance. We then introduce two additional prompt tuning schemes tailored specifically for semantic matching: one that leverages explicit prior semantic knowledge and learns a distinct prompt for each object category, and a novel conditional prompting module that conditions the prompt on the local feature patches of both images in the image pair to be matched, instead of global descriptors of each individual image. Experiments show

*Corresponding author.

that these designs lead to further improvements in matching accuracy.

Our contributions in this paper are summarized as follows: (1) We demonstrate that the performance of Stable Diffusion in the semantic matching task can be significantly enhanced using a straightforward prompt tuning technique. (2) We further propose a novel conditional prompting module, which uses the local features of the image pair. Our experiments show that this design supersedes earlier models reliant on the global descriptor of individual images, leading to a noticeable improvement in matching accuracy. (3) We evaluate our approach on the PF-Pascal, PF-Willow, and SPair-71k datasets, establishing new accuracy benchmarks for each. Notably, we achieve an increase of 12 percentage points on the challenging SPair-71k dataset, surpassing the previous state-of-the-art.

2. Related Work

Semantic Correspondence Early attempts at semantic matching focused on handcrafted features such as HOG [7], SIFT [34], and SIFT Flow [31]. SCNet [14] was the first deep learning method to tackle this problem. Various network architectures have been proposed, addressing the problem from different perspectives, such as metric learning [5], neighbourhood consensus [28, 36, 44], multilayer feature assembly [37, 39], and transformer-based architecture [4, 23], etc. Another line of work in this field is learning matching from image-level annotations. SFNet [25] uses segmentation masks of images; PMD [29] employs teacher-student models and learns from synthetic data, while PWarpC [51] relies on the probabilistic consistency flow from augmented images. Although significant progress has been made, the majority of these methods are based on ResNet [15], which has inferior representation capability compared to later ViT-based feature extractors like DINO [3, 41] or iBOT [57], thus limiting their performance. SimSC [30] demonstrates an improvement of 20% in matching accuracy by switching from ResNet to iBOT in its finetuning pipeline. Semantic matching can also be approached from graph matching [19, 32, 45]. However, these works on a simpler setting where they matches two sets of predefined semantic keypoints labeled on both source and target images, while we only have annotated points on the source image and need to search for corresponding points across the entire target image.

Diffusion Model The pioneering work that formulated image generation as a diffusion process is DDPM [17]. Since then, numerous follow-up works have been proposed to improve the generation process. DDIM [49] and PNDM [33] accelerate the generation process through the development of new noise schedulers. The works by [8] and [16]

enhance the fidelity of the generation by adjusting the denoising step. Another milestone in this field is Stable Diffusion [46], which significantly increases the resolution of generated image by working on the latent space instead of pixel level, paving the way for novel methods in image editing [2, 6] and object-oriented image generation [10, 47], etc. More recently, [56] found that pre-trained Stable Diffusion can also act as a feature extractor, drawing features from images for visual perception tasks. This insight led to studies like DIFT [50] and SD+DINO [55], which delve into the impact of timestep and layer on pre-trained SD’s capabilities in semantic matching. Our work is closely related to these efforts, but we instead explore how the prompt can be optimized within an SD framework to improve its performance on semantic matching.

Prompt Tuning Prompt tuning has gained popularity due to its success in adapting pretrained language models to downstream tasks in natural language processing [20, 48]. COOP [59] was the first work to introduce prompt tuning to computer vision, adapting CLIP to different data distributions in a few-shot setting for image classification. Its successor, COCOOP [58], conditions the prompt on input images to enhance generalizability. These have inspired several other prompting methods [21, 60]. A parallel line of research explores visual prompts, often in the form of masks overlaid on images, to achieve similar objectives [1, 40]. However, most literature has solely focused on prompt tuning for the image classification task. To the best of our knowledge, our study is the first to apply prompt tuning to the SD model for semantic matching.

3. Method

In this section, we introduce our method, namely Stable Diffusion for Semantic Matching (SD4Match). We illustrate the general pipeline in Fig. 1. The UNet component within SD extracts feature maps from input images based on the prompt generated by our prompting module. We present three options for the prompt: one single universal prompt (SD4Match-Single), one prompt per object category (SD4Match-Class), and the conditional prompting module (SD4Match-CPM). The prompt is tuned by the cross-entropy loss between the predicted matching probability and the ground-truth probability of given query points. All modules are kept frozen except for the prompting module during the tuning.

We organize this section as follows: In Sec. 3.1, we briefly introduce the diffusion model and its application as a feature extractor. In Sec. 3.2, we investigate the effect of the existing prompt schemes and introduce our prompt tuning in detail. Finally, in Sec. 3.3, we elaborate on the design of the conditional prompting module and the reasoning behind it.

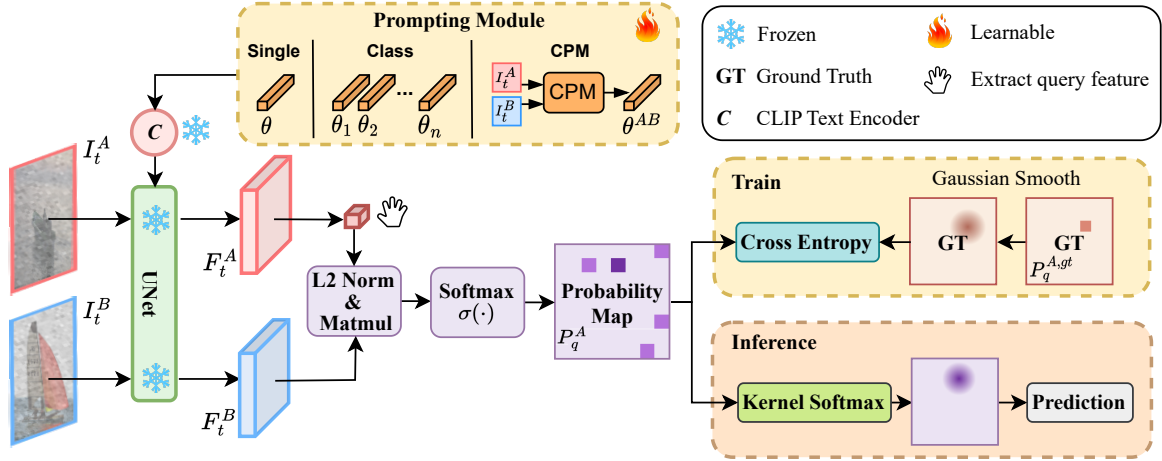


Figure 1. The general pipeline of SD4Match. We present three prompt tuning options for our method: Single, Class, and conditional prompting module (CPM). The prompt is tuned by the cross-entropy loss between the predicted probability map and the ground-truth probability map of given query points. During inference, we use Kernel-Softmax proposed by Lee et al. [25] to localize correspondences.

3.1. Preliminary

Diffusion Model The image diffusion model, as proposed in [17, 49], is a generative model designed to capture the distribution of a set of images. This model consists of two processes: forward and reverse. In the forward process, a clean image I is progressively corrupted by a sequence of Gaussian noise. This corruption follows the equation:

$$I_t = \sqrt{\alpha_t} I_{t-1} + \sqrt{1 - \alpha_t} \epsilon_{t-1} \quad (1)$$

where $\epsilon \sim \mathcal{N}(0, 1)$ and α_t is the coefficient controlling the level of corruption at timestep t . When $t = T$ is sufficiently large, image I_T is totally corrupted, resembling a sample of $\mathcal{N}(0, 1)$. In the reverse process, the diffusion model $f_\theta(I_t, t)$ learns to predict the noise ϵ_t added to the image I at timestep t in the forward process. Therefore, by drawing a sample from $\mathcal{N}(0, 1)$, we can recover its corresponding “original image” by iteratively removing the noise ϵ_t . In Stable Diffusion [46], such a reverse process can condition on various types of input, such as text or other images, to control the content of generated image.

Stable Diffusion as Feature Extractor The text-to-image Stable Diffusion is found with the capability of extracting semantically meaningful feature maps from images [35, 50, 55]. Given an input image I and a specific timestep t , I is encoded by VAE to the latent representation z which is then corrupted to z_t by Eq. (1). The UNet in Stable Diffusion then predicts the noise at timestep t . The resulting feature map is obtained from the output of an intermediate layer of the UNet’s decoder during this noise prediction phase. Observations show that the earlier layer of the decoder with a large t captures more abstract and semantic in-

formation while the later layer of the decoder with a small t focuses on local texture and details. This is similar to the feature pyramids in ResNet [15]. Therefore, careful choices of t and layer are required. For the sake of simplicity, we skip the VAE encoding step and directly refer to the UNet’s input as image I rather than latent representation z in the following paragraphs.

3.2. Prompt Tuning for Semantic Correspondence

We first investigate the impact of various existing prompts on matching accuracy. We evaluated three commonly-used prompts: an empty textual string “ ” [56]; the textual template “a photo of a {*object category*}” which requires the category of the object [50], and the implicit captioner borrowed from the image segmentation method [54] that directly converting the input image into textual embeddings [55]. We applied these prompts to two SD-based approaches: DIFT [50] and SD+DINO [55]. DIFT directly uses the feature map produced by SD2-1 to perform matching with the textual template as the prompt, whereas SD+DINO uses the implicit captioner and fuses the features from DINOv2 and SD1-5 for better accuracy. To isolate the effect of DINO on matching results, we excluded the DINO feature in SD+DINO, designating this modified setting as SD+DINO*. The results are summarized in Tab. 1. We notice that the SD model keeps the majority of its capability in semantic matching even when supplied with a non-informative empty string. This is attributed to the fact that the timestep t in both works is relatively small ($t = 261$ for DIFT and $t = 100$ for SD+DINO out of the total timestep $T = 1000$) so the information in the input image remains largely intact. Therefore, the image itself is sufficient for most matching cases. With the help of prior knowledge of

Table 1. Evaluation of existing works on SPair-71k dataset with different prompt types. The results are PCK with $\alpha = 0.1$. The definition of the metric is provided in Sec. 4.2.

Prompt Type	Empty String	“a photo of a $\{object\ category\}$ ”	Implicit Captioner
DIFT [50]	50.7	52.9	-
SD+DINO* [55]	50.3	52.2	52.4

the object of interest, either in the form of object category or the implicit captioner, the accuracy is improved by about 2 percentage points, reflecting the importance of input-related prompts.

Analogous to the empty string, we search for a single universal prompt that is applied to all images. We randomly initialize the prompt and directly finetune the prompt embeddings with the semantic matching loss proposed by [30]. Given two images I_t^A and I_t^B corrupted to timestep t , the UNet $f(\cdot)$ of Stable Diffusion extracts their corresponding feature maps $F_t^A \in \mathbb{R}^{C \times H_A \times W_A}$ and $F_t^B \in \mathbb{R}^{C \times H_B \times W_B}$ by:

$$F_t = f(I_t, t, \theta) \quad (2)$$

where $\theta \in \mathbb{R}^{N \times D}$ is the prompt embedding, N is the prompt length and D is the dimension of the embedding. F_t^A and F_t^B are then L2-normalized along the feature dimension obtaining $\hat{F}_t^A$ and $\hat{F}_t^B$. Let $\mathbf{X} = \{(\mathbf{x}_q^A, \mathbf{x}_q^B) \mid q = 1, 2, \dots, n\}$ be the ground-truth correspondences provided in the training data. For each query point $\mathbf{x}_q^A = (x_q^A, y_q^A)$ in I_t^A , we extract its corresponding feature $\hat{F}_{t,q}^A \in \mathbb{R}^C$ from $\hat{F}_t^A$ and compute a correlation map $M_q^A \in [-1, 1]^{H_B \times W_B}$ with the entire $\hat{F}_t^B$:

$$M_{q,kl}^A = (\hat{F}_{t,q}^A)^\top \hat{F}_{t,kl}^B \quad (3)$$

where $\hat{F}_{t,kl}^B \in \mathbb{R}^C$ are the feature at position (k, l) in $\hat{F}_t^B$. The correlation map M_q^A is converted to a probability distribution P_q^A by the softmax operation $\sigma(\cdot)$ with temperature β :

$$\sigma(\mathbf{z})_i = \frac{\exp(z_i/\beta)}{\sum_j \exp(z_j/\beta)} \quad (4)$$

The loss between I_t^A and I_t^B is the average of the cross-entropy between P_q^A and the ground-truth distribution $P_q^{A,gt}$ of all correspondence pairs $\mathbf{X}$:

$$\mathcal{L} = \frac{1}{n} \sum_{q=1}^n H(P_q^{A,gt}, P_q^A) \quad (5)$$

where $P_q^{A,gt}$ is the Dirac delta distribution $\delta(\mathbf{x}_q^B)$. Following [30], we apply a $k \times k$ Gaussian kernel to $P_q^{A,gt}$ for label smoothing. During inference, we use Kernel-Softmax [25] to localize the prediction. The entire UNet f is fixed and the only parameter required to update is the prompt embedding θ during tuning. We refer to this option as **SD4Match-Single**.

Just like the textual template, we can also learn one prompt for one specific category. Assume we have n classes and a set of n prompt embeddings $\{\theta_1, \theta_2, \dots, \theta_n\}$, then Eq. (2) becomes:

$$F_t = f(I_t, t, \Theta(c(I_t))) \quad (6)$$

where $c(I_t) \in \{1, 2, \dots, n\}$ is the category of object of interest in I_t and $\Theta(i) = \theta_i$. We denote this as **SD4Match-Class**. Similar to the textual template, the prompts corresponding to the category of input images are fetched and used in SD at inference.

3.3. Conditional Prompting Module

An alternative approach to SD4Match-Class is to condition the prompt on input images, thus eliminating the need for manual inspection of the object’s category. Previous studies have delved into conditional prompts for tasks like image classification [40, 58] and image segmentation [54]. In this context, the prompt is conditioned on the global descriptor of the image, typically extracted by ViT-based [9] feature extractors such as DINOv2 [41] or CLIP [43]. This descriptor is then projected to match the dimension of the prompt embedding and forwarded to the text encoder accompanied by a learnable positional embedding. While this design has shown effectiveness for the tasks mentioned above, it might not work well for finding semantic correspondence. Our reasons are:

1. Semantic matching involves a pair of images. The prompt for this specific pair should be one prompt conditioned on and applied to both images, rather than two different prompts conditioned on each individual image and applied to them separately.
2. Semantic matching relies on the local details of images. The prompt should be conditioned on the local features rather than the global descriptors of the images.
3. The prompt should incorporate a universal head that is applicable to all images. An analogy for this is the prefix “a photo of a” in the textual template.

We therefore propose a novel conditional prompting module (CPM) and illustrate its architecture in Fig. 2. It mainly consists of four components: a DINOv2 feature extractor, two linear layers $g_d(\cdot)$ and $g_n(\cdot)$ and an adaptive MaxPooling layer $p(\cdot)$. Given a pair of clean images I^A and I^B , we first use DINOv2 to extract their local feature patches $\mathcal{F}^A$ and $\mathcal{F}^B$, where $\mathcal{F}^A, \mathcal{F}^B \in \mathbb{R}^{N_{\text{dino}} \times D_{\text{dino}}}$. We then fuse the local features of two images by concatenating them

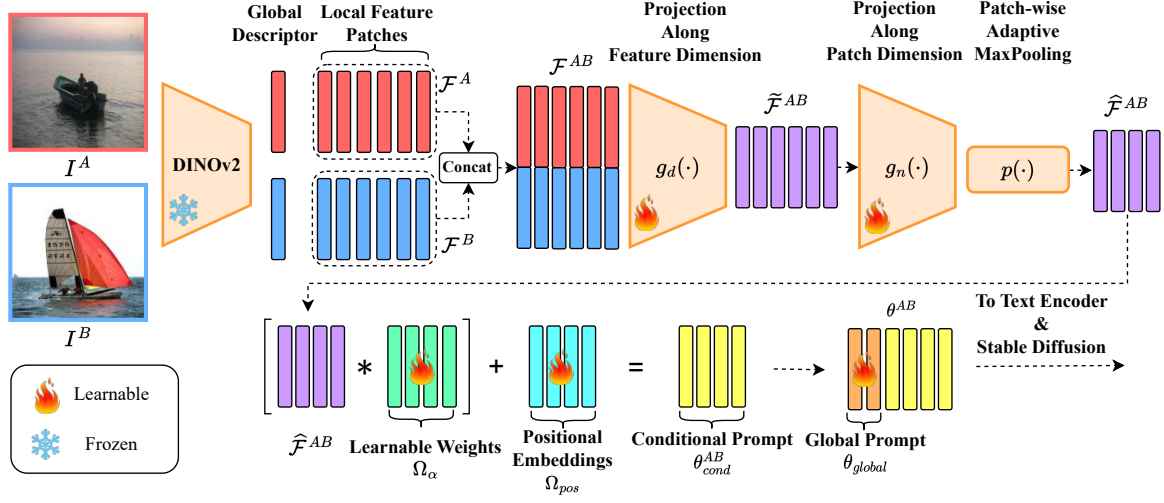


Figure 2. Illustration of the architecture of our conditional prompting module.

along the feature dimension and projecting the concatenated feature $\mathcal{F}^{AB} \in \mathbb{R}^{N_{\text{dino}} \times 2D_{\text{dino}}}$ to the dimension of the prompt embedding D by the linear layer $g_d(\cdot)$, resulting in $\tilde{\mathcal{F}}^{AB} \in \mathbb{R}^{N_{\text{dino}} \times D}$. These operations only explore the feature-wise relationship within the image pair but do not extract the inter-patches information from it. Therefore, we further process $\tilde{\mathcal{F}}^{AB}$ by another linear layer $g_n(\cdot)$ along the *patch* dimension. This allows information exchanges between different local feature patches, enhancing the capability of the prompt. The output of $g_n(\cdot)$ goes through the adaptive MaxPooling layer $p(\cdot)$ to reduce its patch dimension to N_{cond} so that the prompt will not exceed the maximum prompt length of the SD model, producing $\hat{\mathcal{F}}^{AB} \in \mathbb{R}^{N_{\text{cond}} \times D}$. We then follow the design in [54], generating the conditional prompt $\theta_{\text{cond}}^{AB}$ by $\theta_{\text{cond}}^{AB} = \hat{\mathcal{F}}^{AB} * \Omega_{\alpha} + \Omega_{\text{pos}}$, where $\Omega_{\alpha} \in \mathbb{R}^{N_{\text{cond}} \times D}$ is a conditioning weight and $\Omega_{\text{pos}} \in \mathbb{R}^{N_{\text{cond}} \times D}$ is a positional embedding. The conditional prompt $\theta_{\text{cond}}^{AB}$ is appended after a global prompt $\theta_{\text{global}} \in \mathbb{R}^{N_{\text{global}} \times D}$, producing the final prompt $\theta^{AB} \in \mathbb{R}^{N \times D}$ and $N = N_{\text{global}} + N_{\text{cond}}$. Eq. (2) subsequently becomes:

$$F_t^A = f(I_t^A, t, \theta^{AB}), \quad F_t^B = f(I_t^B, t, \theta^{AB}) \quad (7)$$

Note that the feature map F_t^A changes if its pairing image I_t^B also changes since the prompt is conditioned on both images in the image pair. A benefit of this design is that if multiple objects are present in images, the prompt would focus on the common object between the image pair. We designate this configuration as **SD4Match-CPM**.

4. Experiments

In this section, we first provide the implementation details of our method in Sec. 4.1 and introduce datasets and eval-

uation metrics in Sec. 4.2. We then present the evaluation results and ablation studies in Sec. 4.3 and Sec. 4.4 respectively.

4.1. Implementation Details

Our method is implemented in Python using the Huggingface [11, 52, 53] and PyTorch [42] libraries. We follow DIFT and adopt Stable Diffusion 2-1 as the diffusion model, where the total timestep T is 1000. We utilize the output from the 2nd up block of the UNet as the feature map, and set the timestep $t = 261$ during training. We choose DINOv2-ViT-B/14 [41] as the feature extractor in CPM. The maximum prompt length for Stable Diffusion is 77, which includes two special tokens: SOS and EOS. Therefore, we set the prompt length $N = 75$ in SD4Match-Single and SD4Match-Class, and $N_{\text{global}} = 25$ and $N_{\text{cond}} = 50$ in SD4Match-CPM, occupying all 77 token positions. For inference, we set the timestep $t = 50$ as empirical tests suggest it provides optimal results, even though our method is trained at $t = 261$. The temperature β and the Gaussian smooth kernel k are set to 0.04 and 7, respectively, during training. We train our method using the Adam optimizer [24] with a batch size of 9 for 30,000 steps across all experiments. The learning rate is set at 1×10^{-2} in all configurations, except for the two linear layers $g_d(\cdot)$ and $g_n(\cdot)$ in the CPM, where it's 1×10^{-3} . This rate remains constant throughout training. Images are resized to 768×768 for both the training and testing phases. We train SD4Match on three Quadro RTX 6000 GPUs.

4.2. Datasets and Evaluation Metrics

We evaluate our method on three popular semantic correspondence datasets: PF-Pascal [12], PF-Willow [13], and

Table 2. Evaluation on the SPair-71k dataset at $\alpha = 0.1$. Methods are classified into three categories based on their degree of supervision: (1) methods which are zero-shot and not tuned on the training data of Spair-71k, marked as **Z**. (2) methods using image-level annotations, marked as **I**. (3) methods using ground-truth keypoint annotations, marked as **K**. Best results in each category are **bolded**. Overall, Our method achieves the best results in all of 18 categories and we outperform the second-best method SimSC-iBOT [30] by 12 percentage points.

Method	Aero	Bike	Bird	Boat	Bottle	Bus	Car	Cat	Chair	Cow	Dog	Horse	Motor	Person	Plant	Sheep	Train	TV	All
Z DINO [3]	37.3	23.8	63.0	19.9	41.7	29.9	24.1	64.4	21.3	48.7	42.1	30.3	23.3	41.0	28.6	29.8	40.7	37.1	35.9
DINOv2 [41]	69.9	58.9	86.8	36.9	43.4	42.6	39.3	70.2	37.5	69.0	63.7	68.9	55.1	65.0	33.3	57.8	51.2	31.2	53.9
DIFT [50]	61.2	53.2	79.5	31.2	45.3	39.8	33.3	77.8	34.7	70.1	51.5	57.2	50.6	41.4	51.9	46.0	67.6	59.5	52.9
SD+DINO [55]	71.4	59.1	87.3	38.1	51.3	43.3	40.2	77.2	42.3	75.4	63.2	68.8	56.0	66.1	52.8	59.4	63.0	55.1	59.3
I NCNet [44]	17.9	12.2	32.1	11.7	29.0	19.9	16.1	39.2	9.9	23.9	18.8	15.7	17.4	15.9	14.8	9.6	24.2	31.1	20.1
SFNet [25]	26.9	17.2	45.5	14.7	38.0	22.2	16.4	55.3	13.5	33.4	27.5	17.7	20.8	21.1	16.6	15.6	32.2	35.9	26.3
PMD [29]	26.2	18.5	48.6	15.3	38.0	21.7	17.3	51.6	13.7	34.3	25.4	18.0	20.0	24.9	15.7	16.3	31.4	38.1	26.5
K CATs [4]	52.0	34.7	72.2	34.3	49.9	57.5	43.6	66.5	24.4	63.2	56.5	52.0	42.6	41.7	43.0	33.6	72.6	58.0	49.9
PMNC [27]	54.1	35.9	74.9	36.5	42.1	48.8	40.0	72.6	21.1	67.6	58.1	50.5	40.1	54.1	43.3	35.7	74.5	59.9	50.4
SemiMatch [22]	53.6	37.0	74.6	32.3	47.5	57.7	42.4	67.4	23.7	64.2	57.3	51.7	43.8	40.4	45.3	33.1	74.1	65.9	50.7
Trans.Mat. [23]	59.2	39.3	73.0	41.2	52.5	66.3	55.4	67.1	26.1	67.1	56.6	53.2	45.0	39.9	42.1	35.3	75.2	68.6	53.7
SCorrSAN [18]	57.1	40.3	78.3	38.1	51.8	57.8	47.1	67.9	25.2	71.3	63.9	49.3	45.3	49.8	48.8	40.3	77.7	69.7	55.3
SimSC-iBOT [30]	62.2	54.9	79.3	53.2	57.0	72.1	64.8	77.7	39.2	75.9	69.5	68.7	62.4	59.4	45.2	49.5	86.8	71.4	63.5
SD4Match-Single	72.1	66.5	82.3	62.5	57.6	76.0	73.3	81.5	62.0	85.0	71.9	76.1	68.5	76.5	68.9	58.0	89.3	83.1	72.6
SD4Match-Class	75.1	66.6	88.1	71.4	57.8	86.6	74.6	84.2	63.0	83.8	71.5	77.6	73.5	87.2	63.3	60.0	92.0	89.8	75.5
SD4Match-CPM	75.3	67.4	85.7	64.7	62.9	86.6	76.5	82.6	64.8	86.7	73.0	78.9	70.9	78.3	66.8	64.8	91.5	86.6	75.5

SPair-71k [38]. PF-Pascal consists of 2941 training image pairs, 308 validation pairs, and 299 testing pairs spanning across 20 categories of objects. PF-Willow is the supplement to PF-Pascal with 900 testing pairs only. SPair-71K is a larger and more challenging dataset with 53,340 training pairs, 5,384 validation pairs, and 12,234 testing pairs across 18 categories of objects with large scale and appearance variation. Each of the three datasets has non-uniform numbers of ground-truth correspondences.

We follow the common practice in the literature and use the Percentage of Corrected Keypoints (PCK) as the evaluation metric. Given an image pair (I^A, I^B) and its associated correspondence set $\mathbf{X} = \{(\mathbf{x}_q^A, \mathbf{x}_q^B) \mid q = 1, 2, \dots, n\}$, for each $\mathbf{x}_q^A = (x_q^A, y_q^A)$, we find its predicted correspondence $\bar{\mathbf{x}}_q^B$ and calculate PCK for the image pair by:

$$PCK(I^A, I^B) = \frac{1}{n} \sum_q \mathbb{I}(\|\bar{\mathbf{x}}_q^B - \mathbf{x}_q^B\| \leq \alpha * \theta) \quad (8)$$

where θ is the base threshold, α is a number less than 1 and $\mathbb{I}(\cdot)$ is the binary indicator function with $\mathbb{I}(\text{true}) = 1$ and $\mathbb{I}(\text{false}) = 0$. For PF-Pascal, θ is set as $\theta_{img} = \max(h_{img}, w_{img})$. For PF-Willow, the base threshold is $\theta_{kps} = \max(\max_q(x_q^B) - \min_q(x_q^B), \max_q(y_q^B) - \min_q(y_q^B))$. For SPair-71K, the base threshold is $\theta_{bbox} = \max(h_{bbox}, w_{bbox})$ where h_{bbox} and w_{bbox} are height and width of the bounding box. All three base thresholds' choices align with the literature convention.

4.3. Evaluation Results

Evaluation on SPair-71k We provide the evaluation results on SPair-71k in Tab. 2. Specifically, we achieve the best results across all 18 categories, and we outper-

form the second-best method SimSC-iBOT by 12 percentage points (from 63.5 to 75.5) when considering the overall accuracy. Compared to DIFT, which shares the same SD model with ours but uses a textual template as the prompt, SD4Match-Single improves the performance of the SD model by 37.2%, proving that the potential buried in the SD model can be harnessed by simply learning a single prompt. Among the three options of our method, SD4Match-Class outperforms SD4Match-Single by 2.9 percentage points. This echoes the results in Tab. 1, showing the benefit of the prior knowledge of the object. SD4Match-CPM achieves the same accuracy as SD4Match-Class, indicating the effectiveness of our CPM module in capturing the prior knowledge of the object without manual effort.

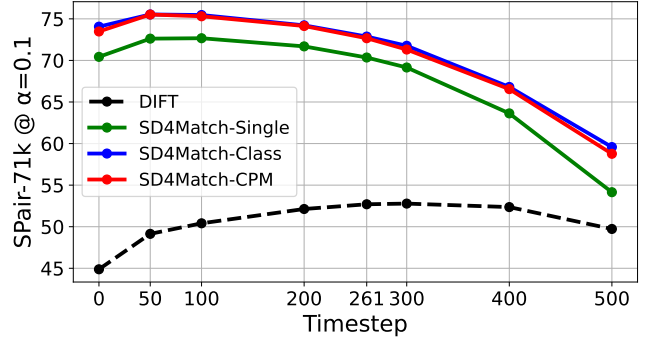
Generalizability Test Following the practice in the literature, we also test the generalizability of our method by tuning the prompt on the training data of PF-Pascal and evaluating it on the testing data of PF-Pascal, PF-Willow, and SPair-71k. The results are presented in Tab. 3. We do not evaluate SD4Match-Class since the three datasets have different categories. Among methods using image-level annotations and keypoint annotations, SD4Match-CPM achieves accuracy on par with SimSC-iBOT on PF-Pascal, and the best generalized results on PF-Willow and SPair-71k across all α . This verifies the generalizability of our method. Compared with zero-shot methods, especially SD-based methods DIFT and SD+DINO, we observe substantial improvement on PF-Pascal and PF-Willow but deterioration on SPair-71k. This is because our universal prompt and CPM overfit the smaller distributions of PF-Pascal and PF-Willow, leading to a certain degree of reduced generalizability on the much larger distribution of SPair-71k. Other

Table 3. Generalizability test of different methods. All methods are either zero-shot or trained on the PF-Pascal dataset unless labelled otherwise.

Method	PF-Pascal $\theta_{img} @ \alpha$			PF-Willow $\theta_{kps} @ \alpha$			SPair-71k $\theta_{bbox} @ \alpha$	
	0.05	0.1	0.15	0.05	0.1	0.15	0.05	0.1
Z DINOv1 [3]	55.6	74.2	81.6	39.7	64.8	77.0	23.1	35.9
DINOv2 [41]	63.4	82.6	89.9	37.3	63.4	73.1	38.4	53.9
DIFT [50]	69.0	82.2	88.1	44.8	68.0	79.8	39.7	52.9
SD+DINO [55]	68.1	85.7	91.5	44.8	72.0	85.0	44.1	59.3
I NCNet [44]	54.3	78.9	86.0	44.0	72.7	85.4	-	26.4
SFNet [25]	59.0	84.0	92.0	46.3	74.0	84.2	11.2	24.0
PWarpC-NCNet [51]	64.2	84.4	90.5	45.0	75.9	87.9	18.2	35.3
K CATs [4]	75.4	92.6	96.4	40.9	69.5	83.2	13.6	27.0
Trans.Mat [23].	80.8	91.8	-	-	65.3	-	-	30.1
DHPF [39]	77.3	91.7	95.5	44.8	70.6	83.2	15.3	27.5
SimSC-iBOT [30]	88.4	95.6	97.3	44.9	71.4	84.5	22.0	37.9
SD4Match-Single	81.3	92.4	96.6	50.7	77.8	89.5	30.5	44.4
SD4Match-CPM	84.4	95.2	97.5	52.1	80.4	91.2	27.2	40.9
SD4Match-Single (Tuned on SPair-71k)	71.8	85.5	90.4	55.5	81.3	91.2	56.5	72.6
SD4Match-CPM (Tuned on SPair-71k)	73.3	87.0	91.5	56.7	80.9	91.6	59.5	75.5

CPM conditioned on:	SPair-71k @ $\alpha = 0.1$
1. Image Pair; Local Feat.	75.5
2. Image Pair; Global Desc.	73.3
3. Ind. Image; Local Feat.	70.8
4. Ind. Image; Global Desc.	68.5
5. w/o global prompt	74.6
6. w/o $g_n(\cdot)$	74.0

(a)



(b)

Figure 3. Results of ablation studies. (a): Evaluation of our method on SPair-71k with different settings of CPM. (b): Evaluation of our method on SPair-71k at different timesteps.

methods tuned on PF-Pascal also exhibit this trend. To further address this point, we additionally provide the results of our method tuned on SPair-71k. We observe a slight improvement on PF-Pascal but a substantial boost on PF-Willow when compared to zero-shot baselines. This improvement is attributed to SD4Match being tuned on a larger dataset, which prevents overfitting on a small data distribution and enhances its generalizability.

4.4. Ablation Studies

Conditional Prompting Module We conduct a thorough ablation study to evaluate each design choice of CPM and present the results in Fig. 3 (a). We evaluate our choice of conditioning in cases 1 through 4. Case 1 involves conditioning on the image pair and local feature patches, representing our current setting in CPM. When we replace local feature patches with a global image descriptor (case 2)

or shift from conditioning on the image pair to conditioning on an individual image (case 3), there is a decline in performance for both scenarios. Notably, the performance drop from case 1 to case 3 is more significant than that from case 1 to case 2. This suggests that conditioning on the image pair has a more substantial impact than conditioning on local feature patches in semantic matching. In Case 4, where we condition on the individual image and its global image descriptor, the architecture mirrors the implicit captioner in SD+DINO and COCOOP [58]. This configuration results in the lowest accuracy, further emphasizing the advantages of conditioning on an image pair and leveraging local features in semantic matching, and the fundamental difference in design rationale between adaptation modules in other tasks and ours. We also investigate the impacts of the global prompt and the patch-wise linear layer $g_n(\cdot)$ in cases 5 and 6, respectively. Removing either of these

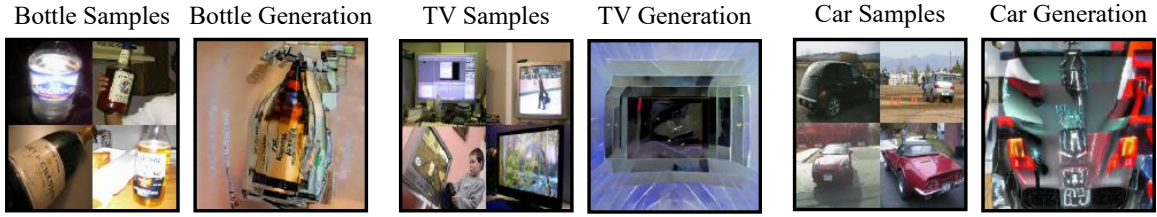


Figure 4. Visualization of the learned class-specific prompt in SD4Match-Class.



Figure 5. Visualization of the learned conditional prompt in SD4Match-CPM.

elements leads to a decrease in performance, underscoring their effectiveness.

Evaluation at Different Timesteps The timestep t is an important hyperparameter that plays a major role in the matching quality. Both DIFT and SD-DINO have performed the grid search to find the optimal timestep. We test our method using different t and plot the result in Fig. 3 (b). We select DIFT as the baseline because we share the same SD model and do not involve other types of feature. As illustrated, our method outperforms the baseline across a wide range of t over 0 to 500. Interestingly, we train our method at $t = 261$ which is the optimal value under the zero-shot setting, but the accuracy at inference instead peaks at $t=50$ and then gradually decreases. This indicates that the model favors a cleaner input image (fewer t) but the noise is also necessary to achieve a good result when using the learned prompt.

Visualization of SD4Match Prompt To further investigate what has been learned during prompt tuning, we visualize images generated by Stable Diffusion using the learned prompt. We first visualize the images generated using the class-specific prompt learned by SD4Match-Class and present samples of selected categories in Fig. 4. Interestingly, for each of the selected categories, the generated image is an abstract illustration of that category. This highlights the intriguing capability of the prompt to learn high-level category details using only keypoint supervision at the UNet’s intermediate stage. This can be loosely compared with textual inversion [10] or DreamBooth [47], which extract an object’s information from multiple images of it-

self and generate the same object in different styles. We show that, even without the explicit reconstruction supervision present in these works, Stable Diffusion can still learn the category-level information from local-level supervision. This reveals the powerful inference ability of the SD model on local information. We also visualize the conditional prompt generated by the CPM and provide selected samples in Fig. 5. The conditional prompt, similar to the class-specific prompt, captures the semantic information of objects’ categories. Moreover, the prompt emphasizes the shared object between two images. As shown in Fig. 5, multiple objects are present in the image pair, and the prompt focuses on the object with the same semantic meaning. This suggests that the CPM is effective in automatically capturing the prior knowledge of the object of interest, subsequently enhancing the matching accuracy.

5. Conclusion

In this paper, we introduce SD4Match, a prompt tuning method that adapts the Stable Diffusion for the semantic matching task. We demonstrate that the quality of features produced by the SD model for this task can be substantially enhanced by simply learning a single universal prompt. Furthermore, we present a novel conditional prompting module that conditions the prompt on the local features of an image pair, resulting in a notable increase in matching accuracy. We evaluate our method on three public datasets, establishing new benchmark accuracies for each. Notably, we surpass the previous state-of-the-art on the challenging SPair-71k dataset by a margin of 12 percentage points.

References

- [1] Hyojin Bahng, Ali Jahanian, Swami Sankaranarayanan, and Phillip Isola. Exploring visual prompts for adapting large-scale models. *arXiv preprint arXiv:2203.17274*, 2022. **2**
- [2] Tim Brooks, Aleksander Holynski, and Alexei A Efros. Instructpix2pix: Learning to follow image editing instructions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18392–18402, 2023. **2**
- [3] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9650–9660, 2021. **2, 6, 7**
- [4] Seokju Cho, Sunghwan Hong, Sangryul Jeon, Yunsung Lee, Kwanghoon Sohn, and Seungryong Kim. Cats: Cost aggregation transformers for visual correspondence. *Advances in Neural Information Processing Systems*, 34:9011–9023, 2021. **1, 2, 6, 7**
- [5] Christopher B Choy, JunYoung Gwak, Silvio Savarese, and Manmohan Chandraker. Universal correspondence network. *Advances in neural information processing systems*, 29, 2016. **2**
- [6] Guillaume Couairon, Jakob Verbeek, Holger Schwenk, and Matthieu Cord. Diffedit: Diffusion-based semantic image editing with mask guidance. *arXiv preprint arXiv:2210.11427*, 2022. **2**
- [7] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05)*, pages 886–893. Ieee, 2005. **2**
- [8] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34:8780–8794, 2021. **2**
- [9] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. **4**
- [10] Rinon Gal, Yuval Atzmon, Or Patashnik, Amit H Bermano, Gal Chechik, and Daniel Cohen-Or. An image is worth one word: Personalizing text-to-image generation using textual inversion. *arXiv preprint arXiv:2208.01618*, 2022. **2, 8**
- [11] Sylvain Gugger, Lysandre Debut, Thomas Wolf, Philipp Schmid, Zachary Mueller, Sourab Mangrulkar, Marc Sun, and Benjamin Bossan. Accelerate: Training and inference at scale made simple, efficient and adaptable. <https://github.com/huggingface/accelerate>, 2022. **5**
- [12] Bumsu Ham, Minsu Cho, Cordelia Schmid, and Jean Ponce. Proposal flow. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3475–3484, 2016. **5**
- [13] Bumsu Ham, Minsu Cho, Cordelia Schmid, and Jean Ponce. Proposal flow: Semantic correspondences from object proposals. *IEEE transactions on pattern analysis and machine intelligence*, 40(7):1711–1725, 2017. **5**
- [14] Kai Han, Rafael S Rezende, Bumsu Ham, Kwan-Yee K Wong, Minsu Cho, Cordelia Schmid, and Jean Ponce. Sfnnet: Learning semantic correspondence. In *Proceedings of the IEEE international conference on computer vision*, pages 1831–1840, 2017. **1, 2**
- [15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. **2, 3**
- [16] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022. **2**
- [17] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020. **2, 3**
- [18] Shuaiyi Huang, Luyu Yang, Bo He, Songyang Zhang, Xuming He, and Abhinav Shrivastava. Learning semantic correspondence with sparse annotations. In *European Conference on Computer Vision*, pages 267–284. Springer, 2022. **6**
- [19] Bo Jiang, Pengfei Sun, and Bin Luo. Glnnet: Graph learning-matching convolutional networks for feature matching. *Pattern Recognition*, 121:108167, 2022. **2**
- [20] Zhengbao Jiang, Frank F Xu, Jun Araki, and Graham Neubig. How can we know what language models know? *Transactions of the Association for Computational Linguistics*, 8: 423–438, 2020. **2**
- [21] Muhammad Uzair Khattak, Hanoona Rasheed, Muhammad Maaz, Salman Khan, and Fahad Shahbaz Khan. Maple: Multi-modal prompt learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19113–19122, 2023. **1, 2**
- [22] Jiwon Kim, Kwangrok Ryoo, Junyoung Seo, Gyuseong Lee, Daehwan Kim, Hansang Cho, and Seungryong Kim. Semi-supervised learning of semantic correspondence with pseudo-labels. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19699–19709, 2022. **6**
- [23] Seungwook Kim, Juhong Min, and Minsu Cho. Transmatcher: Match-to-match attention for semantic correspondence. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8697–8707, 2022. **2, 6, 7**
- [24] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. **5**
- [25] Junghyup Lee, Dohyung Kim, Jean Ponce, and Bumsu Ham. Sfnnet: Learning object-aware semantic correspondence. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2278–2287, 2019. **2, 3, 4, 6, 7**
- [26] Jongmin Lee, Yoonwoo Jeong, Seungwook Kim, Juhong Min, and Minsu Cho. Learning to distill convolutional features into compact local descriptors. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2021. **1**

- [27] Jae Yong Lee, Joseph DeGol, Victor Fragoso, and Sudipta N Sinha. Patchmatch-based neighborhood consensus for semantic correspondence. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13153–13163, 2021. 6
- [28] Shuda Li, Kai Han, Theo W Costain, Henry Howard-Jenkins, and Victor Prisacariu. Correspondence networks with adaptive neighbourhood consensus. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10196–10205, 2020. 2
- [29] Xin Li, Deng-Ping Fan, Fan Yang, Ao Luo, Hong Cheng, and Zicheng Liu. Probabilistic model distillation for semantic correspondence. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7505–7514, 2021. 2, 6
- [30] Xinghui Li, Kai Han, Xingchen Wan, and Victor Adrian Prisacariu. Simsc: A simple framework for semantic correspondence with temperature learning. *arXiv preprint arXiv:2305.02385*, 2023. 1, 2, 4, 6, 7
- [31] Ce Liu, Jenny Yuen, and Antonio Torralba. Sift flow: Dense correspondence across scenes and its applications. *IEEE transactions on pattern analysis and machine intelligence*, 33(5):978–994, 2010. 2
- [32] He Liu, Tao Wang, Yidong Li, Congyan Lang, Yi Jin, and Haibin Ling. Joint graph learning and matching for semantic feature correspondence. *Pattern Recognition*, 134:109059, 2023. 2
- [33] Luping Liu, Yi Ren, Zhijie Lin, and Zhou Zhao. Pseudo numerical methods for diffusion models on manifolds. *arXiv preprint arXiv:2202.09778*, 2022. 2
- [34] David G Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60:91–110, 2004. 2
- [35] Grace Luo, Lisa Dunlap, Dong Huk Park, Aleksander Holynski, and Trevor Darrell. Diffusion hyperfeatures: Searching through time and space for semantic correspondence. *arXiv preprint arXiv:2305.14334*, 2023. 3
- [36] Juhong Min and Minsu Cho. Convolutional hough matching networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2940–2950, 2021. 2
- [37] Juhong Min, Jongmin Lee, Jean Ponce, and Minsu Cho. Hyperpixel flow: Semantic correspondence with multi-layer neural features. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3395–3404, 2019. 1, 2
- [38] Juhong Min, Jongmin Lee, Jean Ponce, and Minsu Cho. Spair-71k: A large-scale benchmark for semantic correspondence. *arXiv preprint arXiv:1908.10543*, 2019. 6
- [39] Juhong Min, Jongmin Lee, Jean Ponce, and Minsu Cho. Learning to compose hypercolumns for visual correspondence. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XV 16*, pages 346–363. Springer, 2020. 2, 7
- [40] Changdae Oh, Hyeji Hwang, Hee-young Lee, YongTaek Lim, Geunyoung Jung, Jiyoung Jung, Hosik Choi, and Kyungwoo Song. Blackvip: Black-box visual prompting for robust transfer learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 24224–24235, 2023. 2, 4
- [41] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023. 2, 4, 5, 6, 7
- [42] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019. 5
- [43] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021. 1, 4
- [44] Ignacio Rocco, Mircea Cimpoi, Relja Arandjelović, Akihiko Torii, Tomas Pajdla, and Josef Sivic. Neighbourhood consensus networks. *Advances in neural information processing systems*, 31, 2018. 1, 2, 6, 7
- [45] Michal Rolínek, Paul Swoboda, Dominik Zietlow, Anselm Paulus, Vít Musil, and Georg Martius. Deep graph matching via blackbox differentiation of combinatorial solvers. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXVIII 16*, pages 407–424. Springer, 2020. 2
- [46] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022. 1, 2, 3
- [47] Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Yael Pritch, Michael Rubinstein, and Kfir Aberman. Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22500–22510, 2023. 2, 8
- [48] Taylor Shin, Yasaman Razeghi, Robert L Logan IV, Eric Wallace, and Sameer Singh. Autoprompt: Eliciting knowledge from language models with automatically generated prompts. *arXiv preprint arXiv:2010.15980*, 2020. 2
- [49] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020. 2, 3
- [50] Luming Tang, Menglin Jia, Qianqian Wang, Cheng Perng Phoo, and Bharath Hariharan. Emergent correspondence from image diffusion. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. 1, 2, 3, 4, 6, 7
- [51] Prune Truong, Martin Danelljan, Fisher Yu, and Luc Van Gool. Probabilistic warp consistency for weakly-supervised semantic correspondences. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8708–8718, 2022. 2, 7

- [52] Patrick von Platen, Suraj Patil, Anton Lozhkov, Pedro Cuenca, Nathan Lambert, Kashif Rasul, Mishig Davaadorj, and Thomas Wolf. Diffusers: State-of-the-art diffusion models. <https://github.com/huggingface/diffusers>, 2022. 5
- [53] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*. Association for Computational Linguistics, 2020. 5
- [54] Jiarui Xu, Sifei Liu, Arash Vahdat, Wonmin Byeon, Xiaolong Wang, and Shalini De Mello. Open-vocabulary panoptic segmentation with text-to-image diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2955–2966, 2023. 1, 3, 4, 5
- [55] Junyi Zhang, Charles Herrmann, Junhwa Hur, Luisa Polania Cabrera, Varun Jampani, Deqing Sun, and Ming-Hsuan Yang. A tale of two features: Stable diffusion complements dino for zero-shot semantic correspondence. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. 1, 2, 3, 4, 6, 7
- [56] Wenliang Zhao, Yongming Rao, Zuyan Liu, Benlin Liu, Jie Zhou, and Jiwen Lu. Unleashing text-to-image diffusion models for visual perception. *arXiv preprint arXiv:2303.02153*, 2023. 1, 2, 3
- [57] Jinghao Zhou, Chen Wei, Huiyu Wang, Wei Shen, Cihang Xie, Alan Yuille, and Tao Kong. ibot: Image bert pre-training with online tokenizer. *arXiv preprint arXiv:2111.07832*, 2021. 2
- [58] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Conditional prompt learning for vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16816–16825, 2022. 2, 4, 7
- [59] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Learning to prompt for vision-language models. *International Journal of Computer Vision*, 130(9):2337–2348, 2022. 1, 2
- [60] Beier Zhu, Yulei Niu, Yucheng Han, Yue Wu, and Hanwang Zhang. Prompt-aligned gradient for prompt tuning. *arXiv preprint arXiv:2205.14865*, 2022. 2

SD4Match: Learning to Prompt Stable Diffusion Model for Semantic Matching

Supplementary Material

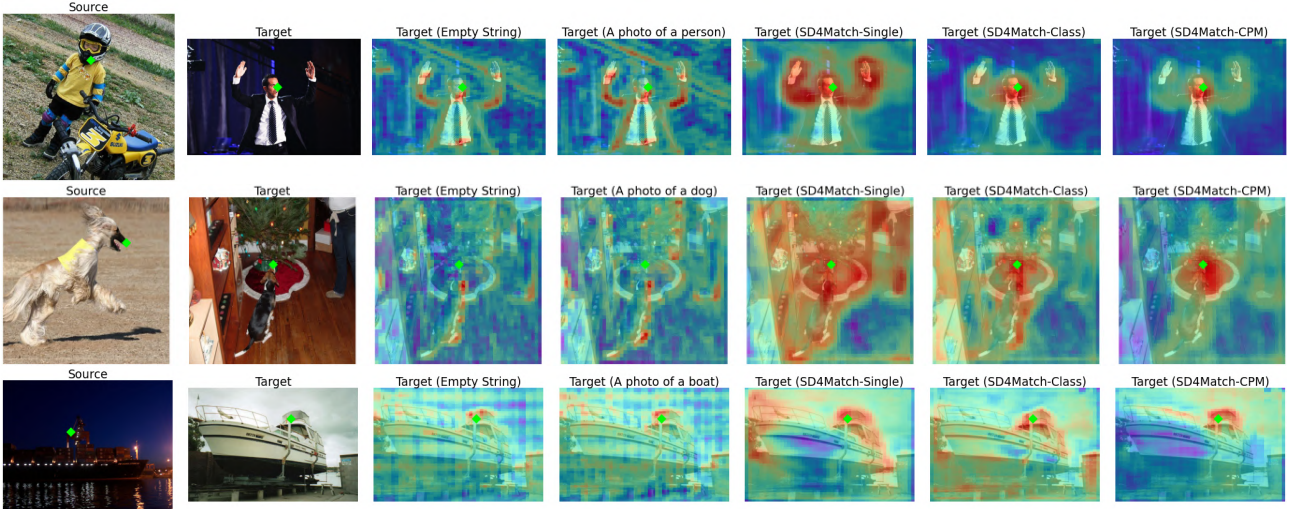


Figure 6. Correlation between the query feature and the target image with different prompts. Warmer colors indicate a higher correlation.

6. Discussion on Prompt Initialization

We discuss the impact of prompt initialization on matching accuracy in this section. We evaluate two extra initialization methods, an empty string and the text “*a photo of an object*”, on SD4Match-Single as it is a generic configuration that does not require prior knowledge of the object category. We summarize their results on SPair-71k in Tab. 4. As demonstrated, the results of an empty string and “*a photo of an object*” are slightly better than the random initialization in the main manuscript. This validates the potential of our method and rooms for further improvement.

Table 4. Evaluation of different initialization methods on SPair-71k using SD4Match-Single.

Init. Method	SPair-71k @ $\alpha = 0.1$
“ <i>a photo of an object</i> ”	73.5
Empty String	73.4
Random Init.	72.6

7. Discussion on Image Size

We follow DIFT [50] and adopt the image size 768x768 during training and inference. We discovered that the discriminative power of SD deteriorates significantly if we deviate from such a size. For instance, on SPair-71k @ $\alpha = 0.1$, the accuracy of vanilla SD2-1 drops from 52.9 to 34.6 when the image size is halved to 384x384 and to

47.9 when doubled to 1536x1536. This is because SD2-1 is trained with the size of 768x768 so the semantic knowledge learned by SD2-1 is based on this size. Larger or smaller images will undermine the extraction of semantic information and provide a worse starting point for prompt tuning.

8. Visualization of Feature Correlation

To demonstrate the impact of different prompts on matching accuracy, we visualize the feature correlation using different prompts in Fig. 6. As shown, features extracted by learned prompts (SD4Match-Single, SD4Match-Class and SD4Match-CPM) are more discriminative in differentiating objects and backgrounds than textual prompts (Empty String and “*a photo of a {category}*”). This is attributed to the feature discriminative loss in Eq. (5) as it teaches the prompt to look for foreground objects. Moreover, SD4Match-Class and SD4Match-CPM have more localized capability than the SD4Match-Single, which shows the benefit of prior knowledge of the object category.

9. More Visualization

We provide more examples of images generated by per-class prompts in Fig. 7, images generated by conditional prompts in Fig. 8 and more qualitative comparison with baselines in Fig. 9.

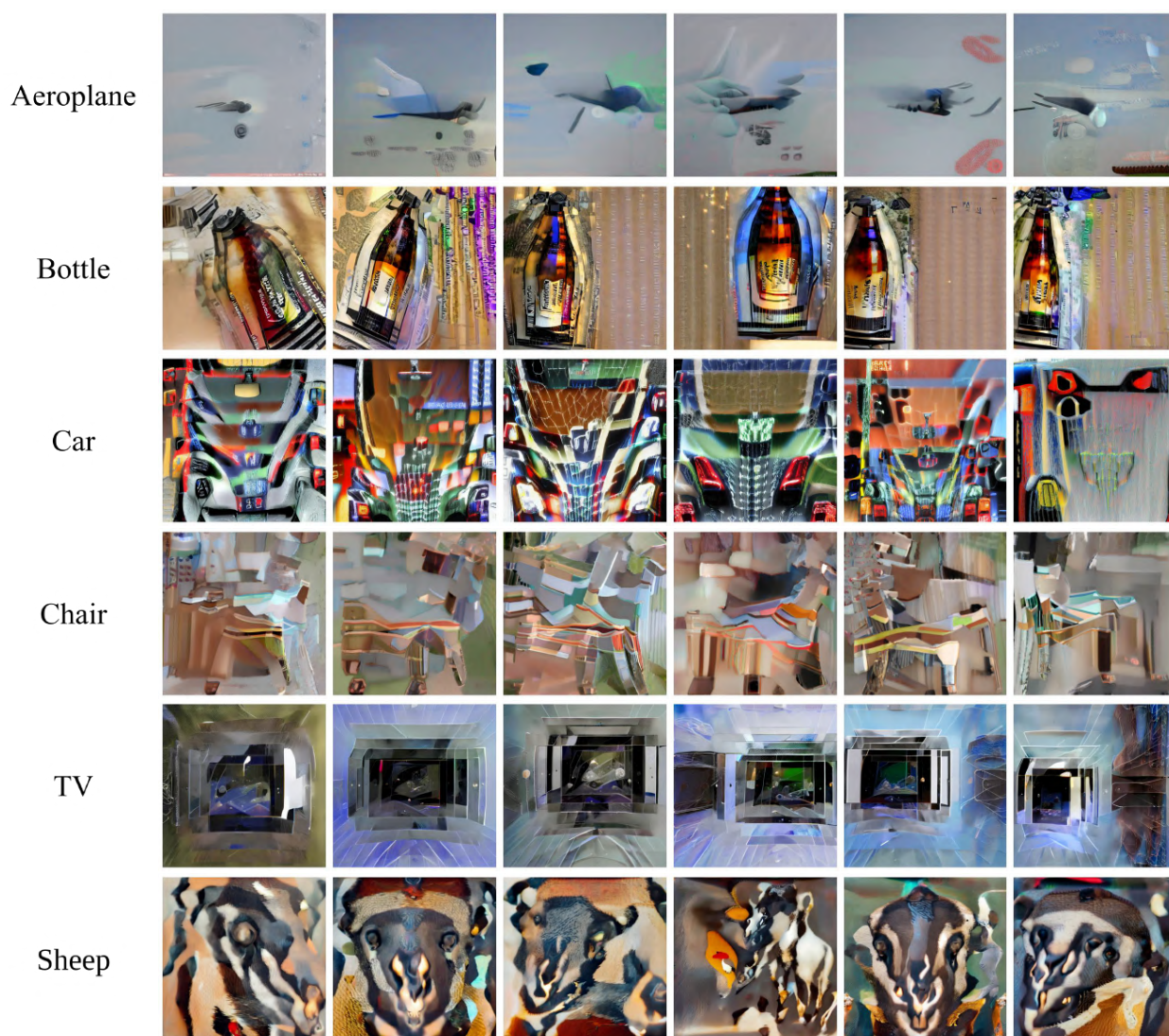


Figure 7. Visualizations of images generated by class-specific prompts learned by SD4Match-Class. For each object category, images are generated by the same prompt but with different random seeds.

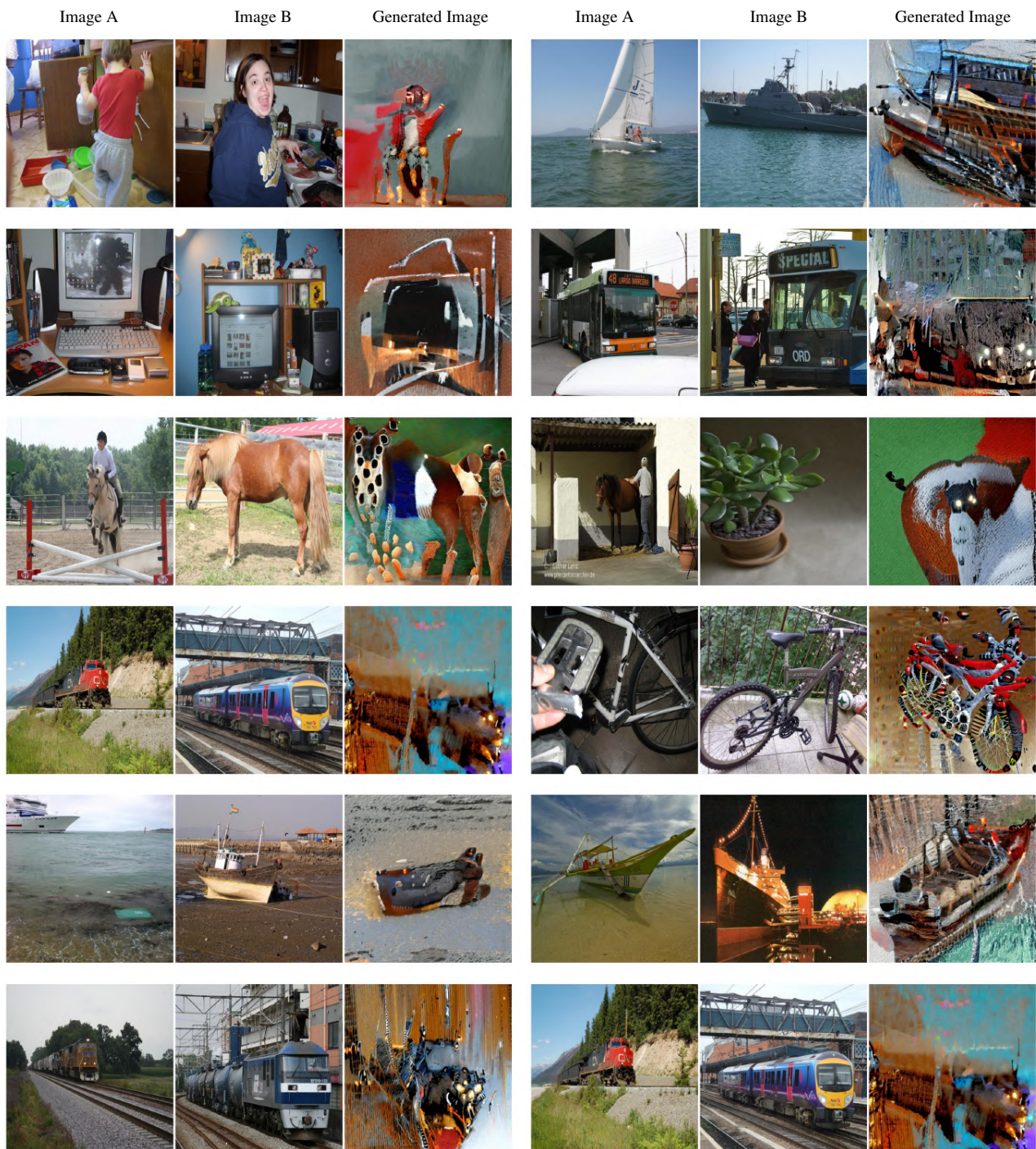


Figure 8. Visualization of images generated by conditional prompts learned by SD4Match-Class. The generated image is an abstract illustration of the category of the shared object presented in Image A and Image B.

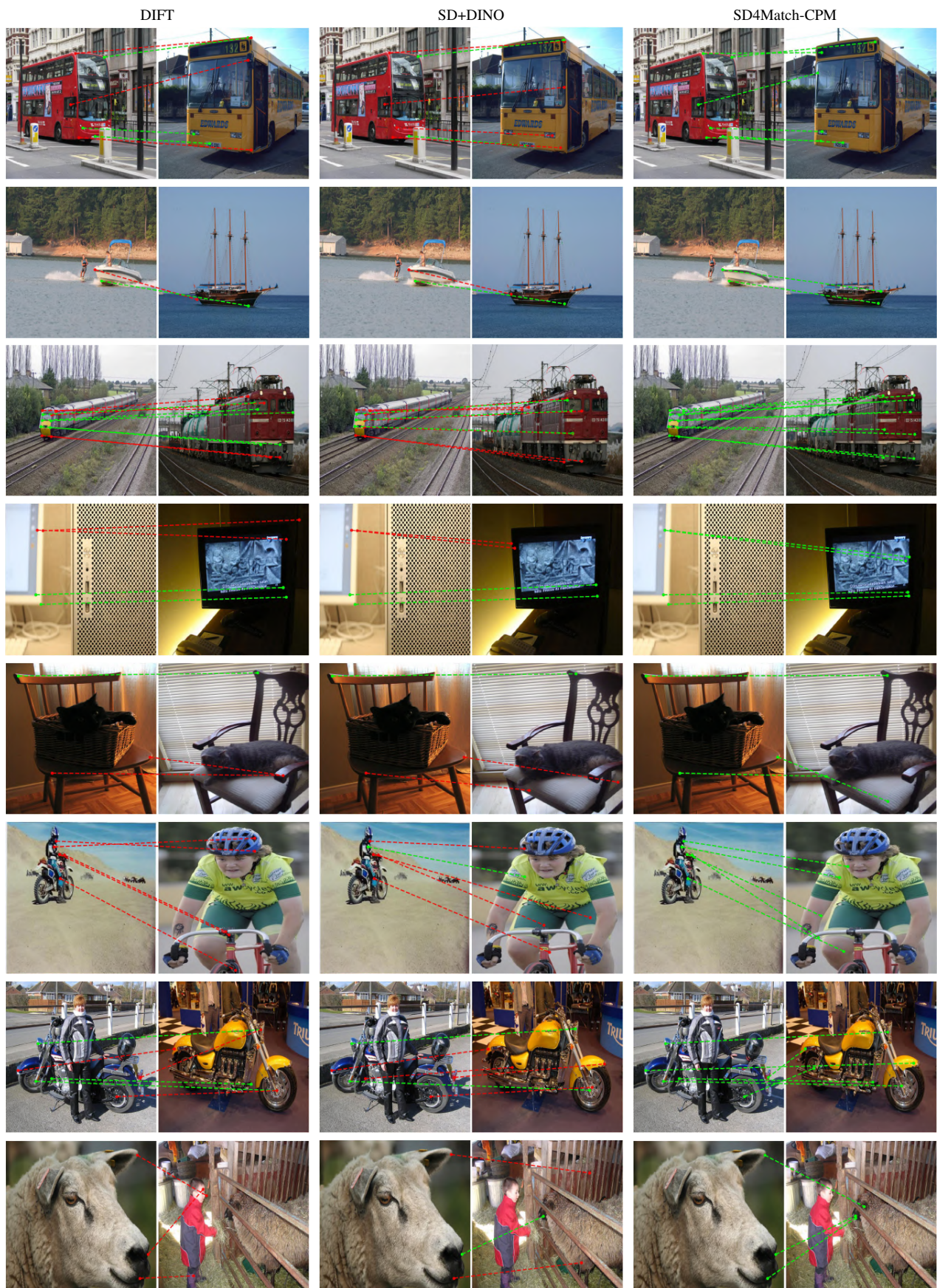


Figure 9. Qualitative comparison between DIFT, SD+DINO, and SD4Match-CPM.

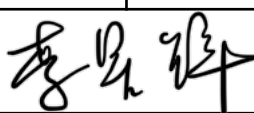
Statement of Authorship for joint/multi-authored papers for PGR thesis

To appear at the end of each thesis chapter submitted as an article/paper

The statement shall describe the candidate's and co-authors' independent research contributions in the thesis publications. For each publication there should exist a complete statement that is to be filled out and signed by the candidate and supervisor (**only required where there isn't already a statement of contribution within the paper itself**).

Title of Paper	SD4Match: Learning to Prompt Stable Diffusion Model for Semantic Matching
Publication Status	☒ Published ☐ Accepted for Publication ☐ Submitted for Publication ☐ Unpublished and unsubmitted work written in a manuscript style
Publication Details	Xinghui Li , Jinyi Lu, Kai Han, Victor Adrian Prisacariu. In Conference on Computer Vision and Pattern Recognition (CVPR), 2024

Student Confirmation

Student Name:	Xinghui Li		
Contribution to the Paper	First Author Contribution: • Conception of research ideas. • Design and implementation of models. • Running of large-scale experiments. • Writing and presentation of the paper.		
Signature		Date	07/Jan/2025

Supervisor Confirmation

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

Supervisor name and title: Prof Victor Adrian Prisacariu			
Supervisor comments I certify that the student made a substantial contribution to the chapter and that the description described above is accurate.			
Signature		Date	9/1/25

This completed form should be included in the thesis, at the end of the relevant chapter.

4.2.3 Conclusion

We introduced a prompt tuning method that adapted Stable Diffusion for the semantic matching task. We demonstrated that the quality of features produced by the SD model for this task could be substantially enhanced by simply learning a single universal prompt. Furthermore, we presented a novel conditional prompting module that conditioned the prompt on the local features of an image pair, resulting in a notable increase in matching accuracy. We evaluated our method on three public datasets, establishing new benchmark accuracies for each.

In this chapter, we explored semantic matching and presented a series of advancements to enhance its effectiveness. This thesis has covered both geometric and semantic matching so far. In the next chapter, we will explore another application of the image feature: 3D information inference.

Chapter 5

Disentangle 3D Attributes from 2D Image Features

5.1 Introduction

Deep image features are the cornerstone of image matching, and the previous two chapters focused on improving matching accuracy from various perspectives. In this chapter, we explore a practical application of image features: inferring 3D information about objects. Prior studies [28, 29] have shown that the 3D model of a clothed person can be directly derived from deep image feature maps. However, one limitation of these methods is that the final model is fixed and we cannot easily modify its pose, shape or garment. To address this, we propose a disentanglement pipeline that learns the pose, shape, and garment attributes of a dressed person from 2D feature maps and uses these attributes to directly control the 3D model.

Our method has three main components: a feature extractor, a multi-head encoder-decoder, and a surface predictor. First, the feature extractor builds a feature map from the input image. Second, the multi-head encoder then disentangles the pose, shape, and garment attributes from the feature map, encoding them into

separate latent codes. These latent codes are reassembled into a feature map by the decoder, and the surface predictor subsequently constructs the 3D model based on this feature map. To modify the pose, shape, or garment of the 3D model, we simply replace the corresponding latent code with the desired one. This results in a new feature map being assembled by the decoder, and the updates are reflected in the final 3D model.

To the best of our knowledge, at the time of this publication, our method is the first to use representations learned from 2D inputs to control 3D outputs. Compared to traditional disentanglement pipelines, such as those used in face swapping [42, 43], where the input and output share the same spatial dimensions, our task involves inputs and outputs of different dimensions. This adds an extra layer of complexity, making the task less constrained and more challenging.

To summarize, our contributions in this chapter are:

1. We propose a method to disentangle 3D shape attributes from 2D image data. We exemplify the principle by learning pose, shape and garment representations of dressed humans from 2D images and allow expressive control of reconstructed 3D model from disentangled feature sub-manifolds. To the best of our knowledge, our method is the first to achieve 2D-to-3D representation learning and output manipulation at the time of publication of this work.
2. We analyse design choices and provide experimental evidence of controlled feature manipulation for pose, shape and garment representations from 2D input on a publicly available dataset.

We publish this work in BMVC 2022 [46].

Disentangling 3D Attributes from a Single 2D Image: Human Pose, Shape and Garment

Xue Hu^{2†}

xue.hu17@imperial.ac.uk

Xinghui Li^{3‡}

xinghui@robots.ox.ac.uk

Benjamin Busam⁴

b.busam@tum.de

Yiren Zhou¹

zhouyiren@huawei.com

Ales Leonardis¹

ales.leonardis@huawei.com

Shanxin Yuan^{1‡}

shanxinyuan@gmail.com

¹ Huawei Noah's Ark Lab

² Imperial College London

³ University of Oxford

⁴ Technical University of Munich

Abstract

For visual manipulation tasks, we aim to represent image content with semantically meaningful features. However, learning implicit representations from images often lacks interpretability, especially when attributes are intertwined. We focus on the challenging task of extracting disentangled 3D attributes only from 2D image data. Specifically, we focus on human appearance and learn implicit pose, shape and garment representations of dressed humans from RGB images. Our method learns an embedding with disentangled latent representations of these three image properties and enables meaningful re-assembling of features and property control through a 2D-to-3D encoder-decoder structure. The 3D model is inferred solely from the feature map in the learned embedding space. To the best of our knowledge, our method is the first to achieve cross-domain disentanglement for this highly under-constrained problem. We qualitatively and quantitatively demonstrate our framework's ability to transfer pose, shape, and garments in 3D reconstruction on virtual data and show how an implicit shape loss can benefit the model's ability to recover fine-grained reconstruction details.

1 Introduction

If you reconstruct a 3D model from a single image, you also want the power to control its content in a meaningful way. This wish has created a wide range of applications that leverage learning of implicit representations by disentanglement, such as face identity swapping [10, 46], hairstyle transfer [16, 38] and pose and shape transfer [17, 28, 47]. The main objective

of implicit representation learning is to disentangle and encode information regarding each characteristic of input signals such that a new sample can be generated by manipulating learned representations. For example, if we separate pose and shape information from the 3D model of a person, we can achieve pose transfer by simply replacing the pose information while keeping the original shape information.

However, the current literature only discusses cases where input and output are from the same domain. This means that if the input is an image or a 3D mesh, the output would also take the same form as the input and learned representations can only control the output in the same domain. Due to this restriction, current works are difficult to be directly applied to 2D-to-3D tasks such as Augmented Reality (AR), where 3D information is often inferred from easily acquired 2D images [4, 9, 20, 26, 41].

In this paper, we propose a solution to this highly under-constrained problem. We focus on learning pose, shape and garment representations of dressed human bodies from 2D RGB images and use these representations to manipulate corresponding 3D models. Our inspiration is drawn from the fact that the 3D mesh of a dressed human can be solely estimated from the feature map of its images [35, 36] by training a shape prior. A key deduction from this observation is that the feature map of an object’s 2D signal contains sufficient information to construct its 3D model if a shape prior is provided. Therefore, if we disentangle and reconstruct feature maps rather than input signals themselves, we can subsequently control the final 3D models which are inferred from the feature maps using the shape prior.

Our method consists of three parts: a feature extractor, a multi-head encoder-decoder and an MLP, as illustrated in Fig. 1. A feature extractor firstly extracts a feature map from the input image. Further, a consecutive feature extractor learns an embedding that encodes disentangled representations for pose, shape and garment into three respective latent codes. The feature map is then recovered from the latent codes, and finally, an MLP is used as a shape prior to construct the 3D model based on the pixel-aligned feature interpolated from the feature map. To change the pose, shape or garment of the 3D model individually while keeping the other properties, we can change the corresponding latent code, which consequently changes the generated 3D mesh.

To the best of our knowledge, our model is the first method to use representations learned from 2D input to control 3D output. Although we exemplify the power of the method with dressed human bodies, such a principle can be generalised to any class of object if the shape prior can be properly trained. In contrast to standard human modelling methods [2, 19, 23, 34], we are able to control the model without using a template. To summarize, our contributions are twofold:

- We propose a method to disentangle 3D shape attributes from 2D image data. We exemplify the principle by learning pose, shape and garment representations of dressed humans from 2D images and allow expressive control of reconstructed 3D model from disentangled feature sub-manifolds. Our method is the first to achieve 2D-to-3D representation learning and output manipulation.
- We analyse design choices and provide experimental evidence of controlled feature manipulation for pose, shape and garment representations from 2D input on a publicly available dataset [32].

2 Related Work

Disentanglement The objective of disentanglement is to find underlying latent representations that control the variation of the data. The pioneering work in this area is InfoGAN [7] which is based on Generative Adversarial Networks (GANs) [11] and aims to maximize the mutual information between the latent codes and generator distribution. β -VAE [16] and its variant [6] use a Variational Autoencoder (VAE) [21] rather than GAN model and penalize a KL-Divergence term to enhance the independence within latent space.

Disentanglement can be applied to both 2D image inputs and 3D inputs, to achieve either attribute transfer like face identity swapping [10, 46] and hairstyle transfer [16, 38] on 2D images, or pose and shape transfer [17, 28, 47] on 3D inputs. Our method is more related to [47] where pose and shape representations are learned in an unsupervised manner. The most significant differences between our method and most existing pipelines are that our input and output are from different domains (2D input and 3D output), and we disentangle the image’s feature map which is an abstract representation rather than the raw data. These differences make our task much more challenging.

Parametric Human Modelling Parametric models targeting humans initially focus on the parameterization of the naked human body. Various body templates, and skeleton hands [44, 45], have been proposed in the past decade [2, 14, 23, 30]. Such a parameterization has recently been generalized to the modelling of the garment [3, 13, 22, 24, 25, 32, 39]. Compared to the naked human body, the parameterization of the garment is much more difficult due to the complicated local details such as wrinkles and foldings. The garment models are usually associated with the naked body model such as SMPL. They are either represented as separated SMPL-like templates [32, 39, 48] or displacement fields to the body model [1, 24]. Some works [3, 32] additionally model the intra-class garment variation using statistical tools such as principle component analysis (PCA).

Implicit Neural Representation Implicit neural representation aims to represent a 3D object using an implicit function which typically takes the form of an MLP. The pioneering works are OccNet [27] and DeepSDF [31] which encode the objects as an occupancy field and a signed distance field, respectively. Compared with traditional 3D representation such as voxel or mesh, neural representation allows a continuous surface representation which circumvents the loss of accuracy due to discretization. Following these two works, many variations of implicit functions [5, 18, 33, 40] and training losses [12] have been proposed. The aforementioned methods mainly focus on 3D input such as point clouds, but PiFU [35] generalizes the implicit function to 2D images by proposing a pixel-aligned implicit function. Instead of a 3D coordinate, the function takes in the corresponding 2D image feature of the 3D location and the depth value, and it can infer a wide variety of human poses and shapes. DISN [42] also has a similar structure to construct static objects. Our method is closely related to PiFU as we disentangle pose and shape information from the image’s feature map.

3 Method

Our method intends to disentangle pose, shape and garment representations of a dressed human from a single RGB image and simultaneously infer the 3D mesh of it. Existing works [35, 36] have already demonstrated that 3D mesh can be inferred solely from a single image

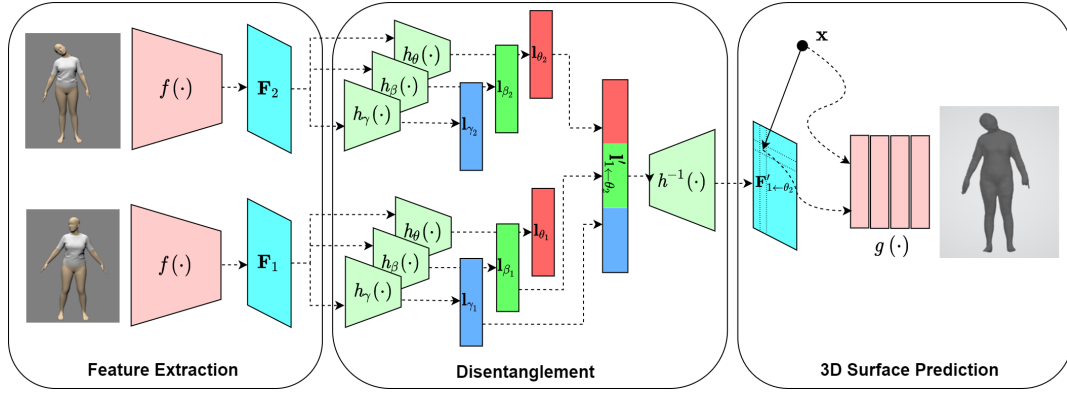


Figure 1: This figure illustrates the architecture of our method. The feature map is $\mathbf{F}$ is extracted from the image by the feature extractor $f(\cdot)$. The feature map is then disentangled into three latent codes $\mathbf{l}_\theta$, $\mathbf{l}_\beta$ and $\mathbf{l}_\gamma$ which correspond to pose, shape and garment representations, respectively. By swapping the latent code, a feature map $\mathbf{F}'$ with swapped information is recovered by $h^{-1}(\cdot)$, from which the 3D mesh with swapped property is inferred as the implicit expression by $g(\cdot)$, where $g(\cdot)$ takes both a 3D coordinate $\mathbf{x}$ and the aligned feature of the projection of $\mathbf{x}$ on the image.

by pixel-aligned implicit function based on the feature map of the image. This indicates that such a feature map contains all information regarding the target mesh; hence, pose, shape and garment representation can be disentangled. Therefore, our method consists of three parts: a feature extractor that extracts the feature map from the image, a multi-head encoder-decoder that disentangles and reconstructs the feature map, and an MLP which infers the 3D mesh from the feature map. We illustrate this pipeline in Fig. 1.

3.1 Feature Disentanglement

After feature extraction, the feature disentanglement is achieved by a multi-head autoencoder module. Given an image $\mathbf{I}_1$, its feature map $\mathbf{F}_1 \in \mathbb{R}^{H \times W \times C}$ can be extracted using the feature extractor $f(\cdot)$. Such a feature map encodes all the necessary information to construct a 3D model, as the missing depth information would be compensated by the implicit function, which acts like a 3D human shape prior [35]. Hence, we only need to disentangle the feature map so that a new feature map and thus the 3D human body can be reconstructed after latent editing using the encoded information from an image of the desired human model.

The disentanglement network follows a similar design to the work of Zhou *et al.* [47]. To disentangle the pose, shape and garment properties, the feature map $\mathbf{F}_1$ is past through three separated encoder modules $h_\theta(\cdot)$, $h_\beta(\cdot)$ and $h_\gamma(\cdot)$, which downsample, flatten and project the feature map into three latent codes: $\mathbf{l}_{\theta_1}$, $\mathbf{l}_{\beta_1}$ and $\mathbf{l}_{\gamma_1}$, which represent pose, shape and garment information of the 3D model, respectively. With three latent codes, the feature map can be reconstructed by concatenating the latent codes together into a longer latent vector $\mathbf{l}'_1$, where $\mathbf{l}'_1 = \text{cat}[\mathbf{l}_{\theta_1}, \mathbf{l}_{\beta_1}, \mathbf{l}_{\gamma_1}]$. Such a vector is then passed through the decoder module $h^{-1}(\cdot)$ of the encoder-decoder, which upsamples the vector and then reconstructs the feature map $\mathbf{F}'_1$. The pose, shape or garment transfer can be achieved by simply replacing the corresponding latent code. For example, if we want to transfer the pose of the person in $\mathbf{I}_1$ to another pose of the person in $\mathbf{I}_2$, we just need to extract the latent codes $\mathbf{l}_{\theta_2}$, $\mathbf{l}_{\beta_2}$ and $\mathbf{l}_{\gamma_2}$ from $\mathbf{I}_2$ and replace the latent code $\mathbf{l}_{\theta_1}$ with $\mathbf{l}_{\theta_2}$. As a result, the feature map $\mathbf{F}'_{1 \leftarrow \theta_2}$ reconstructed from concatenated latent code $\mathbf{l}'_{1 \leftarrow \theta_2} = \text{cat}[\mathbf{l}_{\theta_2}, \mathbf{l}_{\beta_1}, \mathbf{l}_{\gamma_1}]$ would contain the shape and garment information of $\mathbf{I}_1$

but the pose information of $\mathbf{I}_2$. According to $\mathbf{F}'_{1 \leftarrow \theta_2}$, the implicit surface expression can be predicted by the surface prediction MLP.

3.2 3D Reconstruction

In recent years, the implicit neural representation has exhibited the exceptional capability in 3D rendering [31, 35, 37, 40]. The object or scene is often encoded into an MLP as an occupancy field (OCC) or a signed distance field (SDF). Mathematically, it is defined as:

$$s_{\mathbf{x}} = g(\mathbf{x}) \quad (1)$$

where $g(\cdot)$ is the MLP and $s_{\mathbf{x}}$ is the learned implicit surface expression at the sampling position $\mathbf{x} \in \mathbb{R}^3$. The object's surface is represented by the level set of the implicit function $g(\cdot)$, from which the 3D geometry can be easily constructed by Marching Cube. Normally, a MLP could only store one scene or object [31, 37, 40]. However, a single MLP $g(\cdot)$ can be generalized to multiple objects by taking feature map of the image as an input [35, 36]:

$$s_{\mathbf{x}} = g(\mathbf{F}_x, \mathbf{x}_z) \quad (2)$$

where x is the projection of $\mathbf{x}$ on to the image plane, $\mathbf{F}_x \in \mathbb{R}^C$ is the value of encoded feature map $\mathbf{F}$ at a sampling point x , and $\mathbf{x}_z$ is the depth value of $\mathbf{x}$. By taking the pixel-aligned feature $\mathbf{F}_x$, multiple objects can be inferred from a single MLP and objects can be controlled by editing the feature map.

For 2D-3D human reconstruction task, the formulation in Eq. 2 is commonly supervised using the ground truth occupancy values [35]. However, compared with occupancy, we consider the SDF as a better supervision signal during training because using SDF we can not only supervise the distance value but also the surface orientation. Unfortunately, the formulation in Eq. 2 is not suitable for SDF supervision as the output is not differentiable to the $\mathbf{x}$ hence the surface norm is unable to be computed. In order to supervise the surface norm, and being inspired by [43], we change the formulation in Eq. 2 into:

$$s_{\mathbf{x}} = g(\mathbf{F}_x, \sigma(\mathbf{x})) \triangleq g(\mathbf{F}, \mathbf{x}) \quad (3)$$

where $\sigma(\cdot)$ is the positional encoding function defined in [43]. For the sake of simplification, we denote the MLP expression $g(\mathbf{F}_x, \sigma(\mathbf{x}))$ as $g(\mathbf{F}, \mathbf{x})$. At here, we explicitly take the 3D location $\mathbf{x}$ as an input and hence the surface norm can be computed and supervised.

3.3 Loss

Our loss includes two parts: reconstruction loss and disentanglement loss.

3D Reconstruction Loss In order to learn the feature disentanglement from 2D images, we need to first initialize the feature extractor $f(\cdot)$ and the MLP modules $g(\cdot)$ for a 3D shape prior. $f(\cdot)$ followed by $g(\cdot)$ makes up the direct reconstruction pipeline similar to [35], which can be supervised by the 3D reconstruction loss according to ground-truth 3D meshes $\mathbf{M}$. We define the reconstruction loss as $\mathcal{L}_{recon}(\mathbf{F}, \mathbf{X}, \mathbf{M})$, where $\mathbf{X}$ is a set of 3D points that are sampled around the surface of $\mathbf{M}$. We explore two implicit representations: occupancy

and signed distance function (SDF). Specifically, following [35], the occupancy-based 3D reconstruction loss is defined as:

$$\mathcal{L}_{recon}(\mathbf{F}, \mathbf{X}, \mathbf{M}) = \sum_{\mathbf{x} \in \mathbf{X}} \|g(\mathbf{F}, \mathbf{x}) - \mathbf{M}_{occ}(\mathbf{x})\|_2^2 \quad (4)$$

where $\mathbf{X}$ is a set of off-surface points that are sampled tightly around the surface in a way such that the half of the samples are outside the surface and the other half are inside the surface. $\mathbf{M}_{occ}(\mathbf{x})$ is the ground truth occupancy value of $\mathbf{M}$ at 3D position $\mathbf{x}$ and $\mathbf{x} \in \mathbf{X}$.

For SDF-based reconstruction, we adopt the implicit geometric regularization [12], which can learn the SDF surface from point clouds without ground truth SDF supervision, as the 3D reconstruction loss. We combine the sampling strategy in [40] and [35]: a set of points $\mathbf{X} = \mathbf{X}_{on} \cup \mathbf{X}_{off}$ are randomly sampled such that half number of points are on the surface of the mesh, denoted as $\mathbf{X}_{on}$ and the remaining points are off the surface, denoted as $\mathbf{X}_{off}$. In particular, $\mathbf{X}_{off}$ are randomly sampled around the ground-truth mesh and within the cubic space with a ratio of 3:1. For each image, the reconstruction loss $\mathcal{L}_{recon}(\mathbf{F}, \mathbf{X}, \mathbf{M})$ consists of three parts: the level set loss $\mathcal{L}_{ls}$, the Eikonal regularization loss $\mathcal{L}_{igr}$ and off surface loss $\mathcal{L}_o$:

$$\mathcal{L}_{recon}(\mathbf{F}, \mathbf{X}, \mathbf{M}) = \lambda_{ls}\mathcal{L}_{ls} + \lambda_{igr}\mathcal{L}_{igr} + \lambda_o\mathcal{L}_o \quad (5)$$

$$\text{with } \mathcal{L}_{ls} = \sum_{\mathbf{x} \in \mathbf{X}_{on}} (\|g(\mathbf{F}, \mathbf{x})\|_1 + 1 - \langle \nabla_{\mathbf{x}}g(\mathbf{F}, \mathbf{x}), \nabla_{\mathbf{x}}\mathbf{M}(\mathbf{x}) \rangle), \quad (6)$$

$$\mathcal{L}_{igr} = \sum_{\mathbf{x}} \|\nabla_{\mathbf{x}}g(\mathbf{F}, \mathbf{x}) - 1\|_1, \quad (7)$$

$$\mathcal{L}_o = \sum_{\mathbf{x} \in \mathbf{X}_{off}} \exp(-\alpha \cdot \|g(\mathbf{F}, \mathbf{x})\|_1), \quad \alpha \gg 0. \quad (8)$$

The level set loss $\mathcal{L}_{ls}$ forces the gradient of the MLP (*i.e.*, the predicted surface normal) $\nabla_{\mathbf{x}}g(\mathbf{x})$ to align with the ground-truth normal of the surface $\nabla_{\mathbf{x}}\mathbf{M}(\mathbf{x})$. The Eikonal regularization loss $\mathcal{L}_{igr}$ regularize the MLP to satisfy Eikonal equation $\|\nabla_{\mathbf{x}}g(\mathbf{x})\| = 1$. The off surface loss $\mathcal{L}_o$ push the points off the surface away from the level set surface. The overall reconstruction loss $\mathcal{L}_{recon}$ is the sum of these three components, weighted by coefficients λ_{ls} , λ_{igr} and λ_o respectively. In the experiments, the coefficients are set to $\lambda_{ls} = \lambda_{igr} = 1$ and $\lambda_o = 0.1$.

Disentanglement Loss With the properly pre-trained feature extractor and MLP surface predictor, we then jointly train the encoder-decoder for feature disentanglement and MLP for surface refinement. In order to properly learn the disentanglement of pose, shape and garment, we adopt the control variate strategy. We construct a dataset which consists of pairs of images and their corresponding ground-truth 3D meshes. For each pair, the difference between the images is either in pose, shape or garment, while the other two are kept same. This control strategy allows the encoder-decoder to focus on one property at each time and hence interpret the difference in latent space. We provides more details about our dataset pairing in section 4.1.

For each image pair, denote the different property between them as d , where $d \in \{\theta, \beta, \gamma\}$ and two remaining properties as $s1$ and $s2$, where $s1, s2 \in \{\theta, \beta, \gamma\} \setminus \{d\}$. Let $\mathbf{F}_1 = f(\mathbf{I}_1)$ and $\mathbf{F}_2 = f(\mathbf{I}_2)$, then $\{\mathbf{l}_{s1,1}, \mathbf{l}_{s2,1}, \mathbf{l}_{d1}\} = h(\mathbf{F}_1)$ and $\{\mathbf{l}_{s1,2}, \mathbf{l}_{s2,2}, \mathbf{l}_{d2}\} = h(\mathbf{F}_2)$. We define two procedures:

- *Disentangled self-reconstruction* If no latent code swapping is applied, we should be able to reconstruct the feature maps $\mathbf{F}_1$ and $\mathbf{F}_2$ by $\mathbf{F}'_1 = h^{-1}(\mathbf{l}'_1)$ and $\mathbf{F}'_2 = h^{-1}(\mathbf{l}'_2)$,

where $\mathbf{I}'_1 = \text{concat}[\mathbf{l}_{s1,1}, \mathbf{l}_{s2,1}, \mathbf{l}_{d1}]$ and $\mathbf{I}'_2 = \text{concat}[\mathbf{l}_{s1,2}, \mathbf{l}_{s2,2}, \mathbf{l}_{d2}]$. We should also be able to recover $\mathbf{M}_1$ and $\mathbf{M}_2$ from the reconstructed feature map $\mathbf{F}'_1$ and $\mathbf{F}'_2$.

- *Disentangled cross-reconstruction* In the case where the latent codes of property d is swapped, the modified latent codes become: $\mathbf{I}'_{1 \leftarrow d2} = \text{concat}[\mathbf{l}_{s1,1}, \mathbf{l}_{s2,1}, \mathbf{l}_{d2}]$, then two new feature maps can be reconstructed as $\mathbf{F}'_{1 \leftarrow d2} = h^{-1}(\mathbf{I}'_{1 \leftarrow d2})$ and $\mathbf{F}'_{2 \leftarrow d1} = h^{-1}(\mathbf{I}'_{2 \leftarrow d1})$. Since there is only one different property between two images, the reconstructed feature map of one image should be the same as the original feature maps of the other and so are their 3D meshes.

We define the feature map reconstruction loss before and after disentanglement during self- and cross-reconstruction:

$$\mathcal{L}_{feat} = \|\mathbf{F}_1 - \mathbf{F}'_1\|_2^2 + \|\mathbf{F}_2 - \mathbf{F}'_2\|_2^2 + \|\mathbf{F}_1 - \mathbf{F}'_{2 \leftarrow d1}\|_2^2 + \|\mathbf{F}_2 - \mathbf{F}'_{1 \leftarrow d2}\|_2^2 \quad (9)$$

We also add the latent identity loss $\mathbf{L}_{latent}$ which matches the compressed latent code of two invariant properties in each image pair:

$$\mathbf{L}_{latent} = \|\mathbf{l}_{s1,1} - \mathbf{l}_{s1,2}\|_2^2 + \|\mathbf{l}_{s2,1} - \mathbf{l}_{s2,2}\|_2^2 \quad (10)$$

We define a surface reconstruction loss (either with occupancy or SDF) between the ground truth mesh $\mathbf{M}$ and the predicted output $g(\cdot)$. This term direct supervises the quality of the 3D models generated from the reconstructed feature maps:

$$\begin{aligned} \mathcal{L}_{recon} = & \mathcal{L}_{recon}(\mathbf{F}'_1, \mathbf{X}_1, \mathbf{M}_1) + \mathcal{L}_{recon}(\mathbf{F}'_2, \mathbf{X}_2, \mathbf{M}_2) \\ & + \mathcal{L}_{recon}(\mathbf{F}'_{2 \leftarrow d1}, \mathbf{X}_1, \mathbf{M}_1) + \mathcal{L}_{recon}(\mathbf{F}'_{1 \leftarrow d2}, \mathbf{X}_2, \mathbf{M}_2) \end{aligned} \quad (11)$$

Therefore, for each image pair, the overall disentanglement loss $\mathcal{L}_{disent}$ is defined as the weighted combination of the above three terms:

$$\mathcal{L}_{disent} = \mathcal{L}_{feat} + \mathcal{L}_{latent} + \mathcal{L}_{recon} \quad (12)$$

4 Experiment

4.1 Dataset

It is extremely challenging to obtain a real-world dataset that contains two humans with different shapes and garments but in the same pose. Therefore, we modified a public physically simulated dataset, TailorNet dataset [32] to evaluate our method. The original dataset consists of physically simulated 3D sequences of motions of dressed humans, which are modelled by the SMPL model with an additional garment layer on top of it. The dataset includes humans with a range of body shapes and several types of garments, where each type of garment also has intra-class variation. We make the following modifications to the dataset: (i) We construct the image pairing by comparing the SMPL parameters of the human body and garment type. In the end, we obtain 1369 pose-vary pairs, 51 shape-vary pairs and 761 garment-varying pairs. (ii) Since the combined ground-truth meshes of dressed humans have multiple layers, the normals are in the opposite direction at overlapping regions. This may confuse the network during training with SDF-based supervision loss. Therefore, we convert them into single-layer water-tight meshes using the Manifold package. (iii) For each mesh, we render the front view using Blender as the input to our method. The final output is the three subsets that contain pairwise data variation. The three subsets are padded to the same size of 1369 and split into the training, validation and test dataset with a ratio of 6:2:2.

Experiment	Implicit Func	Chamfer (mm)	P2S (mm)	normal (mm)
self	SDF/OCC	1.43 /3.65	11.21 /22.24	11.16/ 8.93
cross - pose	SDF/OCC	1.26 /4.73	12.12 /21.63	10.63/ 8.72
cross - shape	SDF/OCC	2.46 /3.09	12.25 /19.88	11.46/ 9.28
cross - garment	SDF/OCC	1.26 /3.36	9.35 /22.61	11.53/ 8.86

Table 1: Reconstruction error (median) after swapping the latent code of body shape, pose and garment style, using SDF and occupancy.

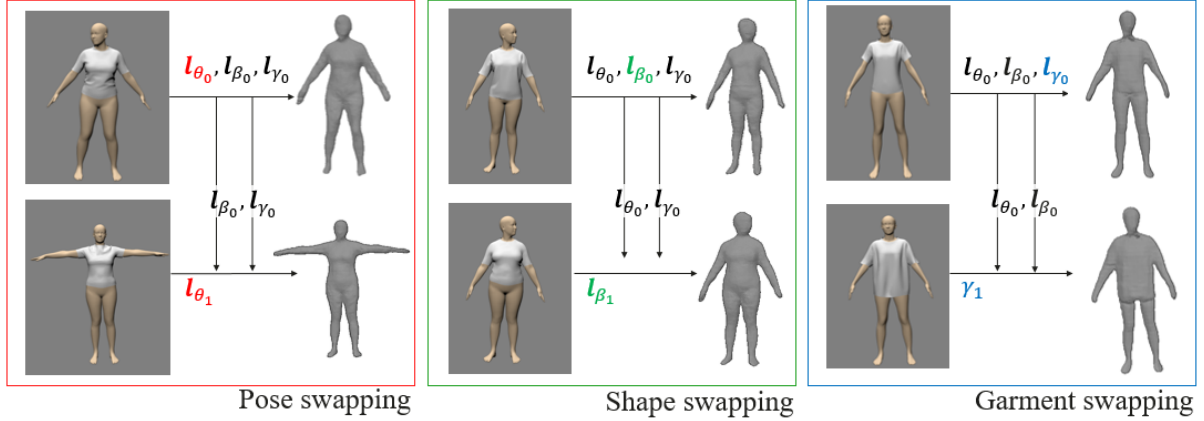


Figure 2: Example results of surface reconstruction after pairwise swapping the encoded latent components. Note that the pose and shape swapping are achieved on SDF-based pipeline, while the garment swapping is achieved on occupancy-based pipeline.

4.2 Implementation Details

Our method is implemented using PyTorch. For the feature extractor $f(\cdot)$, we follow the choice in [35] and use the Hourglass Network [29]. For the encoder $h(\cdot)$ and decoder modules $h^{-1}(\cdot)$, we use a ResNet18 [15] implemented in Bolts library [8]. We change the input dimension of the first convolutional layer to match the dimension of the feature map. The latent size for $\mathbf{l}_\theta$, $\mathbf{l}_\beta$ and $\mathbf{l}_\gamma$ are all set to be 128 according to the ablation study (see details in supplementary material). For the surface predictor, we follow the design in [43] which is a fully connected MLP with residual link between every two linear layers.

We first pre-train the feature extractor and surface predictor on the direct reconstruction pipeline. For both occupancy and SDF supervision, we use the RMSprop optimizer with a learning rate of 1×10^{-4} , the number of sampling is 1000 and batch size is 8. After epoch 150, the learning rate is reduced to 1×10^{-5} to ensure smooth convergence. During the training for feature disentanglement, the feature extractor is frozen while the encoder-decoder and surface predictor are tuned together. The same optimizer is used except that the learning rate is kept as 1×10^{-4} until the convergence reaches around 360 epochs.

4.3 Results

4.3.1 Reconstruction with latent representation

Our proposed encoder-decoder structure allows us to disentangle and modify the latent code $\mathbf{l}$ by replacing its components $\mathbf{l}_\theta$, $\mathbf{l}_\beta$ and $\mathbf{l}_\gamma$. We test the quality of self-reconstruction and pairwise cross-reconstruction defined in Section 3.3. The reconstruction quality is measured using three metrics: the Chamfer distance between reconstructed and ground-truth surfaces,

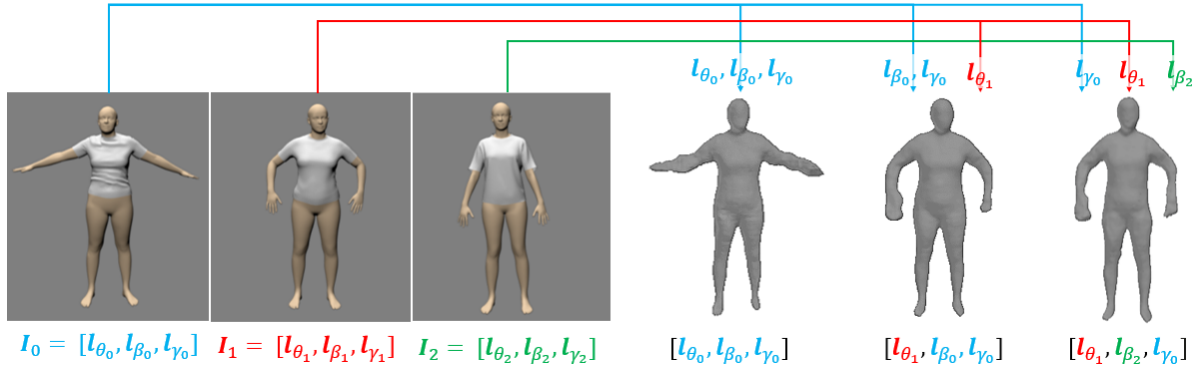


Figure 3: Example results of surface reconstruction after uncontrolled swapping of both body pose and shape. The synthetic mesh (rightmost) incorporates the tight garment style of I_0 , body pose of I_1 and slimmer body shape of I_2 .

the Euclidean distance from predicted vertices to their closest point on the ground-truth surface (P2S), and the RMS difference between the predicted and ground-truth surface normal vectors. We summarise the results in Table 1, where “self” means self-recon and “cross” means cross-recon. The SDF-based pipeline achieves a higher accuracy than the occupancy-based pipeline. However, from visual inspections, we find the SDF-based method can better distinguish between different body poses and shapes than garment styles. By contrast, occupancy-based disentangled reconstruction capture all three variations, though the reconstruction quality is lower. The difference may be because the SDF approach tends to over-smoothen the predicted surface and the garment variation in the training dataset is too subtle. Fig. 2 shows the visual demonstration for how the mesh reconstruction is controlled through pairwise latent swapping.

Pairwise swapping can only change one property each time and requires the other two to be identical, which is hardly possible in reality. Therefore, given a specific garment, we test the uncontrolled swapping where the body pose and shape latent code are replaced by new values at the same time. As shown in Fig. 3, the body pose $\mathbf{l}_\theta$, shape $\mathbf{l}_\beta$ and garment information $\mathbf{l}_\gamma$ are respectively encoded from $\mathbf{I}_1$, $\mathbf{I}_2$ and $\mathbf{I}_0$, concatenated, and decoded into a new human body. We have included more examples in the supplementary material.

4.3.2 Interpolation test

Given a starting and ending frame with different features, we are able to interpolate the latent codes to achieve a smooth transition of body pose, shape or garment style in the reconstructed mesh, as shown in Fig. 4. While the pose interpolation can be used to achieve an animation effect, the body shape and garment interpolation is helpful in generating new virtual assets in an effortless way. More examples are provided in the supplementary material.

5 Discussion and Conclusion

In this paper, we propose the first and novel method for the task of extracting disentangled 3D attributes only from 2D image data. To illustrate the feasibility, we apply the principle to learn pose, shape, and garment representations of a human body directly from images. Our method disentangles and encodes the representations of three properties into latent codes using an encoder-decoder from the feature map of the input image and reconstructs the human

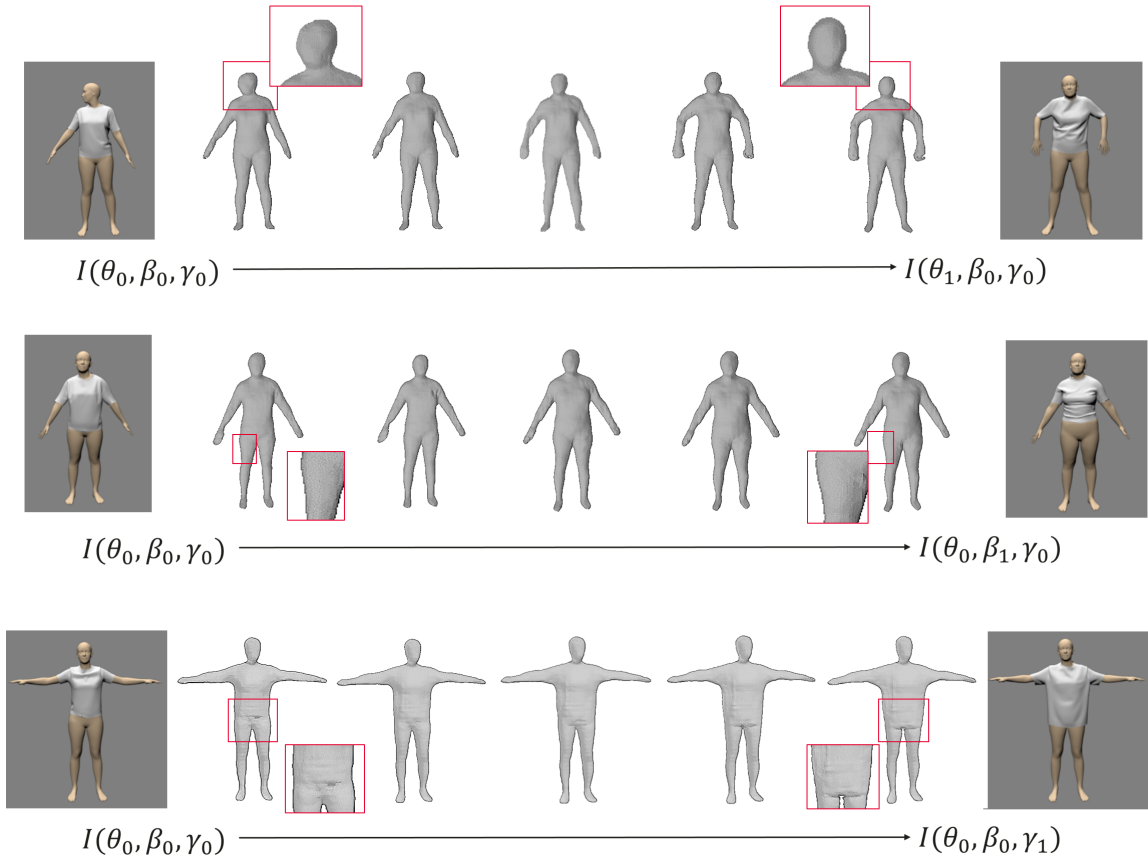


Figure 4: Example results of interpolation test: the latent code of source and target image are interpolated to reconstruct a dressed human body with new features.

body based on learned representations using the implicit function. We demonstrate the control to the reconstructed mesh by manipulating the code in the latent space. We believe that our novel approach towards 2D-to-3D disentanglement can pave the way to interpretable manipulations of 3D content from 2D images alone and therefore opens the path to more seamless and controllable interaction with 3D models without the need for explicit supervision. In our future work we plan to use TKFAC on MindSpore¹, which is a new deep learning computing framework.

¹<https://www.mindspore.cn/>

References

- [1] Thiemo Alldieck, Marcus Magnor, Bharat Lal Bhatnagar, Christian Theobalt, and Gerard Pons-Moll. Learning to reconstruct people in clothing from a single rgb camera. In *Proceedings of IEEE Intl. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2019.
- [2] Dragomir Anguelov, Praveen Srinivasan, Daphne Koller, Sebastian Thrun, Jim Rodgers, and James Davis. Scape: shape completion and animation of people. In *Proceedings of ACM Special Interest Group on GRAPHICS (SIGGRAPH)*, 2005.
- [3] Bharat Lal Bhatnagar, Garvita Tiwari, Christian Theobalt, and Gerard Pons-Moll. Multi-garment net: Learning to dress 3d people from images. In *Proceedings of Intl. Conf. on Computer Vision (ICCV)*, 2019.
- [4] Benjamin Busam, Hyun Jun Jung, and Nassir Navab. I like to move it: 6d pose estimation as an action decision process. *arXiv preprint arXiv:2009.12678*, 2020.
- [5] Rohan Chabra, Jan E Lenssen, Eddy Ilg, Tanner Schmidt, Julian Straub, Steven Lovegrove, and Richard Newcombe. Deep local shapes: Learning local sdf priors for detailed 3d reconstruction. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020.
- [6] Ricky TQ Chen, Xuechen Li, Roger B Grosse, and David K Duvenaud. Isolating sources of disentanglement in variational autoencoders. In *Proceedings of Conf. on Neural Information Processing Systems (NeurIPS)*, 2018.
- [7] Xi Chen, Yan Duan, Rein Houthoofd, John Schulman, Ilya Sutskever, and Pieter Abbeel. Infogan: Interpretable representation learning by information maximizing generative adversarial nets. In *Proceedings of Conf. on Neural Information Processing Systems (NeurIPS)*, 2016.
- [8] William Falcon and Kyunghyun Cho. A framework for contrastive self-supervised learning and designing a new approach. *arXiv preprint arXiv:2009.00104*, 2020.
- [9] Daoyi Gao, Yitong Li, Patrick Ruhkamp, Iuliia Skobleva, Magdalena Wysocki, Hyun-Jun Jung, Pengyuan Wang, Arturo Guridi, and Benjamin Busam. Polarimetric pose prediction. In *European Conference on Computer Vision (ECCV)*, October 2022.
- [10] Gege Gao, Huaibo Huang, Chaoyou Fu, Zhaoyang Li, and Ran He. Information bottleneck disentanglement for identity swapping. In *Proceedings of IEEE Intl. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2021.
- [11] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Proceedings of Conf. on Neural Information Processing Systems (NeurIPS)*, 2014.
- [12] Amos Gropp, Lior Yariv, Niv Haim, Matan Atzmon, and Yaron Lipman. Implicit geometric regularization for learning shapes. In *Proceedings of Intl. Conf. on Machine Learning (ICML)*, 2020.

- [13] Erhan Gundogdu, Victor Constantin, Amrollah Seifoddini, Minh Dang, Mathieu Salzmann, and Pascal Fua. Garnet: A two-stream network for fast and accurate 3d cloth draping. In *Proceedings of IEEE Intl. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2019.
- [14] Nils Hasler, Carsten Stoll, Martin Sunkel, Bodo Rosenhahn, and H-P Seidel. A statistical model of human pose and body shape. In *Computer graphics forum*, 2009.
- [15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of IEEE Intl. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2016.
- [16] Irina Higgins, Loic Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner. beta-vae: Learning basic visual concepts with a constrained variational framework. In *Proceedings of Intl. Conf. on Machine Learning (ICML)*, 2016.
- [17] Boyi Jiang, Juyong Zhang, Jianfei Cai, and Jianmin Zheng. Disentangled human body embedding based on deep hierarchical neural network. *IEEE Trans. on Visualization and Computer Graphics (TVCG)*, 2020.
- [18] Chiyu Jiang, Avneesh Sud, Ameesh Makadia, Jingwei Huang, Matthias Nießner, Thomas Funkhouser, et al. Local implicit grid representations for 3d scenes. In *Proceedings of IEEE Intl. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2020.
- [19] Hanbyul Joo, Tomas Simon, and Yaser Sheikh. Total capture: A 3d deformation model for tracking faces, hands, and bodies. In *Proceedings of IEEE Intl. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2018.
- [20] HyunJun Jung, Patrick Ruhkamp, Guangyao Zhai, Nikolas Brasch, Yitong Li, Yannick Verdie, Jifei Song, Yiren Zhou, Anil Armagan, Slobodan Ilic, et al. Is my depth ground-truth good enough? hammer—highly accurate multi-modal dataset for dense 3d scene regression. *arXiv preprint arXiv:2205.04565*, 2022.
- [21] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.
- [22] Zorah Lahner, Daniel Cremers, and Tony Tung. Deepwrinkles: Accurate and realistic clothing modeling. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018.
- [23] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J Black. Smpl: A skinned multi-person linear model. *ACM Trans. on Graphics (ToG)*, 2015.
- [24] Qianli Ma, Jinlong Yang, Anurag Ranjan, Sergi Pujades, Gerard Pons-Moll, Siyu Tang, and Michael J. Black. Learning to dress 3d people in generative clothing. In *Proceedings of IEEE Intl. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2020.

- [25] Qianli Ma, Shunsuke Saito, Jinlong Yang, Siyu Tang, and Michael J. Black. SCALE: Modeling clothed humans with a surface codec of articulated local elements. In *Proceedings of IEEE Intl. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2021.
- [26] Fabian Manhardt, Gu Wang, Benjamin Busam, Manuel Nickel, Sven Meier, Luca Minicullo, Xiangyang Ji, and Nassir Navab. Cps++: Improving class-level 6d pose and shape estimation from monocular images with self-supervised learning. *arXiv preprint arXiv:2003.05848*, 2020.
- [27] Lars Mescheder, Michael Oechsle, Michael Niemeyer, Sebastian Nowozin, and Andreas Geiger. Occupancy networks: Learning 3d reconstruction in function space. In *Proceedings of IEEE Intl. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2019.
- [28] Jiteng Mu, Weichao Qiu, Adam Kortylewski, Alan Yuille, Nuno Vasconcelos, and Xiaolong Wang. A-sdf: Learning disentangled signed distance functions for articulated shape representation. In *Proceedings of IEEE Intl. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2021.
- [29] Alejandro Newell, Kaiyu Yang, and Jia Deng. Stacked hourglass networks for human pose estimation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2016.
- [30] Ahmed A A Osman, Timo Bolkart, and Michael J. Black. STAR: A sparse trained articulated human body regressor. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020.
- [31] Jeong Joon Park, Peter Florence, Julian Straub, Richard Newcombe, and Steven Lovegrove. DeepSDF: Learning continuous signed distance functions for shape representation. In *Proceedings of IEEE Intl. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2019.
- [32] Chaitanya Patel, Zhouyingcheng Liao, and Gerard Pons-Moll. Tailornet: Predicting clothing in 3d as a function of human pose, shape and garment style. In *Proceedings of IEEE Intl. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2020.
- [33] Songyou Peng, Michael Niemeyer, Lars Mescheder, Marc Pollefeys, and Andreas Geiger. Convolutional occupancy networks. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020.
- [34] Gerard Pons-Moll, Javier Romero, Naureen Mahmood, and Michael J Black. Dyna: A model of dynamic human shape in motion. *ACM Trans. on Graphics (ToG)*, 2015.
- [35] Shunsuke Saito, Zeng Huang, Ryota Natsume, Shigeo Morishima, Angjoo Kanazawa, and Hao Li. Pifu: Pixel-aligned implicit function for high-resolution clothed human digitization. In *Proceedings of Intl. Conf. on Computer Vision (ICCV)*, 2019.
- [36] Shunsuke Saito, Tomas Simon, Jason Saragih, and Hanbyul Joo. Pifuhd: Multi-level pixel-aligned implicit function for high-resolution 3d human digitization. In *Proceedings of IEEE Intl. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2020.

- [37] Shunsuke Saito, Jinlong Yang, Qianli Ma, and Michael J. Black. SCANimate: Weakly supervised learning of skinned clothed avatar networks. In *Proceedings of IEEE Intl. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2021.
- [38] Eduardo Hugo Sanchez, Mathieu Serrurier, and Mathias Ortner. Learning disentangled representations via mutual information estimation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020.
- [39] Igor Santesteban, Nils Thuerey, Miguel A Otaduy, and Dan Casas. Self-supervised collision handling via generative 3d garment models for virtual try-on. In *Proceedings of IEEE Intl. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2021.
- [40] Vincent Sitzmann, Julien Martel, Alexander Bergman, David Lindell, and Gordon Wetzstein. Implicit neural representations with periodic activation functions. In *Proceedings of Conf. on Neural Information Processing Systems (NeurIPS)*, 2020.
- [41] Pengyuan Wang, HyunJun Jung, Yitong Li, Siyuan Shen, Rahul Parthasarathy Srikanth, Lorenzo Garattoni, Sven Meier, Nassir Navab, and Benjamin Busam. Phocal: A multi-modal dataset for category-level object pose estimation with photometrically challenging objects. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21222–21231, 2022.
- [42] Qiangeng Xu, Weiyue Wang, Duygu Ceylan, Radomir Mech, and Ulrich Neumann. Disn: Deep implicit surface network for high-quality single-view 3d reconstruction. In *Advances in Neural Information Processing Systems*, 2019.
- [43] Alex Yu, Vickie Ye, Matthew Tancik, and Angjoo Kanazawa. pixelnerf: Neural radiance fields from one or few images. In *Proceedings of IEEE Intl. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2021.
- [44] Shanxin Yuan, Qi Ye, Bjorn Stenger, Siddhant Jain, and Tae-Kyun Kim. Bighand2. 2m benchmark: Hand pose dataset and state of the art analysis. In *Proceedings of IEEE Intl. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2017.
- [45] Shanxin Yuan, Guillermo Garcia-Hernando, Björn Stenger, Gyeongsik Moon, Ju Yong Chang, Kyoung Mu Lee, Pavlo Molchanov, Jan Kautz, Sina Honari, Lihao Ge, et al. Depth-based 3d hand pose estimation: From current achievements to future goals. In *Proceedings of IEEE Intl. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2018.
- [46] Jianfu Zhang, Yuanyuan Huang, Yaoyi Li, Weijie Zhao, and Liqing Zhang. Multi-attribute transfer via disentangled representation. In *Proceedings of AAAI Conference on Artificial Intelligence (AAAI)*, 2019.
- [47] Keyang Zhou, Bharat Lal Bhatnagar, and Gerard Pons-Moll. Unsupervised shape and pose disentanglement for 3d meshes. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020.
- [48] Heming Zhu, Yu Cao, Hang Jin, Weikai Chen, Dong Du, Zhangye Wang, Shuguang Cui, and Xiaoguang Han. Deep fashion3d: A dataset and benchmark for 3d garment reconstruction from single images. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020.

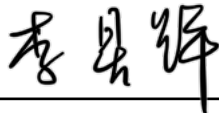
Statement of Authorship for joint/multi-authored papers for PGR thesis

To appear at the end of each thesis chapter submitted as an article/paper

The statement shall describe the candidate's and co-authors' independent research contributions in the thesis publications. For each publication there should exist a complete statement that is to be filled out and signed by the candidate and supervisor (**only required where there isn't already a statement of contribution within the paper itself**).

Title of Paper	Disentangling 3D Attributes from a Single 2D Image: Human Pose, Shape and Garment
Publication Status	☒ Published ☐ Accepted for Publication ☐ Submitted for Publication ☐ Unpublished and unsubmitted work written in a manuscript style
Publication Details	Xinghui Li* , Xue Hu*, Benjamin Busam, Yiren Zhou, Ales Leonardis, Shanxin Yuan. In The British Machine Vision Conference (BMVC), 2022

Student Confirmation

Student Name:	Xinghui Li		
Contribution to the Paper	First Author Contribution: • Conception of research ideas. • Design and implementation of models. • Running of large-scale experiments. • Writing and presentation of the paper.		
Signature		Date	07/Jan/2025

Supervisor Confirmation

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

Supervisor name and title:	Prof Victor Adrian Prisacariu		
Supervisor comments	I certify that the student made a substantial contribution to the chapter and that the description described above is accurate.		
Signature		Date	9/1/25

This completed form should be included in the thesis, at the end of the relevant chapter.

5.3 Conclusion

We introduced an approach for disentangling 3D attributes using only 2D image data. To demonstrate its feasibility, we applied our method to learn distinct representations of pose, shape, and garment of the human body directly from images. Our approach disentangled these three attributes into separate latent codes, using an encoder-decoder framework to process the input image’s feature map. The model then reconstructed the human body based on these learned representations through an implicit function. By adjusting the latent codes, we were able to effectively manipulate the reconstructed mesh. We believe this novel 2D-to-3D disentanglement approach opens up new possibilities for interpretable 3D content manipulation from 2D images alone, enabling seamless and controllable interactions with 3D models without the need for explicit supervision.

We have introduced the contributions of this thesis in detail in the last three chapters. In the next chapter, we present the conclusion of the thesis, followed by the discussion and future works.

Chapter 6

Conclusion

This thesis has primarily focused on improving both geometric and semantic matching from different perspectives and explored learning of 3D attributes from 2D image feature maps.

In Chapter 3, we concentrated on geometric matching. We proposed a dual-resolution matching model that identifies correspondences in a coarse-to-fine manner. By filtering the coarse matching space with a 4D convolutional module and using the refined output to guide finer-scale geometric matching, we increased feature matching resolution and improved accuracy with minimal computational overhead.

Next, in Section 4.1, we turned our attention to semantic matching. we revisited metric learning, formulated a semantic matching pipeline using only a pre-trained feature backbone, and achieved accuracy comparable to methods with more complex architectures. In Section 4.2, we leveraged the extensive knowledge embedded in Stable Diffusion, adapting the model for semantic matching tasks using prompt tuning. We further enhanced matching accuracy by introducing a novel conditional prompting module, which integrates prior knowledge about the images into the Stable Diffusion model.

Lastly, in Chapter 5 we explore learning 3D information from 2D image feature map in Chapter 5. we explored the extraction of 3D information from 2D image feature maps. We proposed a method that disentangles pose, shape, and garment attributes from 2D image features, enabling direct control over 3D model generation.

Overall, this thesis contributes to the fields of image feature matching and the extraction of 3D information from image features.

6.1 Discussion and Future Work

While significant progress has been made in geometric matching, there remain many interesting avenues for future research. For example, challenges such as illumination, viewpoint, and scale differences between images still persist. To address illumination variations, one potential direction is to develop a large-scale dataset with diverse lighting conditions, enabling models to learn brightness-invariant features. For viewpoint and scale differences, we may leverage the most recent 3D foundation models [142, 143] and combine both 2D and 3D features of images to enhance the robustness against these variations.

In semantic matching, a major challenge is the scarcity of training data with dense keypoint annotations. Unlike geometric matching, which can generate dense labels through SfM, semantic matching datasets [35, 89] are often too small to train models with strong generalization capabilities. A promising direction to tackle this issue is the use of diffusion models to generate synthetic data. Diffusion models [144] can generate images based on object masks, allowing the creation of various object instances by manipulating the masks into different shapes and positions. This approach naturally provides dense correspondences between the synthesized images due to the known transformations.

For learning 3D attributes from 2D image features, one potential improvement is extending the current pipeline to handle multiview images. Epipolar constraints between images could offer additional supervision during model training. Another avenue for future work is to explore unsupervised disentanglement approaches. While the current method serves as a proof-of-concept, demonstrating that 3D attributes can be disentangled from 2D image features, it relies on paired data with controlled variation. This requirement imposes significant constraints on the available data and limits the generalizability of the model. To address these limitations, future work could draw inspiration from the disentanglement literature [123] by incorporating techniques such as imposing regularization in the latent space, thereby improving the model’s flexibility and adaptability to more diverse datasets.

Bibliography

- [1] Ignacio Rocco, Mircea Cimpoi, Relja Arandjelović, Akihiko Torii, Tomas Pajdla, and Josef Sivic. Neighbourhood consensus networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2018.
- [2] Andrew J Davison, Ian D Reid, Nicholas D Molton, and Olivier Stasse. Monoslam: Real-time single camera slam. *IEEE Transaction on Pattern Analysis and Machine Intelligence (TPAMI)*, 2007.
- [3] Georg Klein and David Murray. Parallel tracking and mapping on a camera phone. In *Proceedings of IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, 2009.
- [4] Richard A Newcombe, Shahram Izadi, Otmar Hilliges, David Molyneaux, David Kim, Andrew J Davison, Pushmeet Kohi, Jamie Shotton, Steve Hodges, and Andrew Fitzgibbon. Kinectfusion: Real-time dense surface mapping and tracking. In *Proceedings of IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, 2011.
- [5] Johannes Lutz Schönberger and Jan-Michael Frahm. Structure-from-motion revisited. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.
- [6] Johannes Lutz Schönberger, Enliang Zheng, Marc Pollefeys, and Jan-Michael Frahm. Pixelwise view selection for unstructured multi-view stereo. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2016.
- [7] Junyi Zhang, Charles Herrmann, Junhwa Hur, Luisa Polania Cabrera, Varun Jampani, Deqing Sun, and Ming-Hsuan Yang. A tale of two features: Stable diffusion complements dino for zero-shot semantic correspondence. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.
- [8] Junghyup Lee, Dohyung Kim, Jean Ponce, and Bumsu Ham. Sfnet: Learning object-aware semantic correspondence. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

- [9] David G Lowe. Object recognition from local scale-invariant features. In *Proceedings of International Conference on Computer Vision (ICCV)*, 1999.
- [10] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool. Surf: Speeded up robust features. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2006.
- [11] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2005.
- [12] Mihai Dusmanu, Ignacio Rocco, Tomas Pajdla, Marc Pollefeys, Josef Sivic, Akihiko Torii, and Torsten Sattler. D2-net: A trainable cnn for joint detection and description of local features. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.
- [13] Jerome Revaud, Philippe Weinzaepfel, César De Souza, Noe Pion, Gabriela Csurka, Yann Cabon, and Martin Humenberger. R2D2: Repeatable and Reliable Detector and Descriptor. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.
- [14] Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Superpoint: Self-supervised interest point detection and description. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.
- [15] Yuki Ono, Eduard Trulls, Pascal Fua, and Kwang Moo Yi. Lf-net: learning local features from images. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2018.
- [16] Juhong Min, Jongmin Lee, Jean Ponce, and Minsu Cho. Hyperpixel flow: Semantic correspondence with multi-layer neural features. In *Proceedings of International Conference on Computer Vision (ICCV)*, 2019.
- [17] Juhong Min and Minsu Cho. Convolutional hough matching networks. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.
- [18] Seokju Cho, Sunghwan Hong, Sangryul Jeon, Yunsung Lee, Kwanghoon Sohn, and Seungryong Kim. Cats: Cost aggregation transformers for visual correspondence. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021.
- [19] Jiahui Zhang, Dawei Sun, Zixin Luo, Anbang Yao, Lei Zhou, Tianwei Shen, Yurong Chen, Hongen Liao, and Long Quan. Learning two-view correspondences and

- geometry using order-aware network. In *Proceedings of International Conference on Computer Vision (ICCV)*, 2019.
- [20] Kwang Moo Yi, Eduard Trulls, Yuki Ono, Vincent Lepetit, Mathieu Salzmann, and Pascal Fua. Learning to find good correspondences. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.
 - [21] Shihua Zhang and Jiayi Ma. Convmatch: Rethinking network design for two-view correspondence learning. *IEEE Transaction on Pattern Analysis and Machine Intelligence (TPAMI)*, 2023.
 - [22] Xin Liu and Jufeng Yang. Progressive neighbor consistency mining for correspondence pruning. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.
 - [23] Shir Amir, Yossi Gandelsman, Shai Bagon, and Tali Dekel. Deep vit features as dense visual descriptors. In *Proceedings of the European Conference on Computer Vision Workshops (ECCVW)*, 2022.
 - [24] Jiaming Sun, Zehong Shen, Yuang Wang, Hujun Bao, and Xiaowei Zhou. LoFTR: Detector-free local feature matching with transformers. *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.
 - [25] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.
 - [26] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009.
 - [27] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.
 - [28] Shunsuke Saito, Zeng Huang, Ryota Natsume, Shigeo Morishima, Angjoo Kanazawa, and Hao Li. Pifu: Pixel-aligned implicit function for high-resolution clothed human digitization. In *Proceedings of International Conference on Computer Vision (ICCV)*, 2019.
 - [29] Shunsuke Saito, Tomas Simon, Jason Saragih, and Hanbyul Joo. Pifuhd: Multi-level pixel-aligned implicit function for high-resolution 3d human digitization. In

Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2020.

- [30] JiaWang Bian, Wen-Yan Lin, Yasuyuki Matsushita, Sai-Kit Yeung, Tan-Dat Nguyen, and Ming-Ming Cheng. Gms: Grid-based motion statistics for fast, ultra-robust feature correspondence. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.
- [31] Shuda Li, Kai Han, Theo W. Costain, Henry Howard-Jenkins, and Victor Prisacariu. Correspondence networks with adaptive neighbourhood consensus. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.
- [32] Xinghui Li, Kai Han, Shuda Li, and Victor Prisacariu. Dual-resolution correspondence networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.
- [33] Xinghui Li, Kai Han, Shuda Li, and Victor Prisacariu. Dualrc: A dual-resolution learning framework with neighbourhood consensus for visual correspondences. *IEEE Transaction on Pattern Analysis and Machine Intelligence (TPAMI)*, 2023.
- [34] Bumsub Ham, Minsu Cho, Cordelia Schmid, and Jean Ponce. Proposal flow. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.
- [35] Juhong Min, Jongmin Lee, Jean Ponce, and Minsu Cho. Spair-71k: A large-scale benchmark for semantic correspondence. *arXiv preprint arXiv:1908.10543*, 2019.
- [36] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.
- [37] Kihyuk Sohn. Improved deep metric learning with multi-class n-pair loss objective. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2016.
- [38] Xinghui Li, Kai Han, Xingchen Wan, and Victor Adrian Prisacariu. Simsc: A simple framework for semantic correspondence with temperature learning. *arXiv preprint arXiv:2305.02385*, 2023.
- [39] Luming Tang, Menglin Jia, Qianqian Wang, Cheng Perng Phoo, and Bharath Hariharan. Emergent correspondence from image diffusion. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.
- [40] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell

- Wortsman, et al. Laion-5b: An open large-scale dataset for training next generation image-text models. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.
- [41] Xinghui Li, Jingyi Lu, Kai Han, and Victor Adrian Prisacariu. Sd4match: Learning to prompt stable diffusion model for semantic matching. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.
- [42] Gege Gao, Huaibo Huang, Chaoyou Fu, Zhaoyang Li, and Ran He. Information bottleneck disentanglement for identity swapping. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.
- [43] Jianfu Zhang, Yuanyuan Huang, Yaoyi Li, Weijie Zhao, and Liqing Zhang. Multi-attribute transfer via disentangled representation. In *Proceedings of AAAI Conference on Artificial Intelligence (AAAI)*, 2019.
- [44] Boyi Jiang, Juyong Zhang, Jianfei Cai, and Jianmin Zheng. Disentangled human body embedding based on deep hierarchical neural network. *IEEE Transaction on Visualization and Computer Graphics (TVCG)*, 2020.
- [45] Jiteng Mu, Weichao Qiu, Adam Kortylewski, Alan Yuille, Nuno Vasconcelos, and Xiaolong Wang. A-sdf: Learning disentangled signed distance functions for articulated shape representation. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.
- [46] Xue Hu, Xinghui Li, Benjamin Busam, Yiren Zhou, Ales Leonardis, and Shanxin Yuan. Disentangling 3d attributes from a single 2d image: Human pose, shape and garment. In *Proceedings of British Machine Vision Conference (BMVC)*, 2022.
- [47] Yujun Shi, Chuhui Xue, Jiachun Pan, Wenqing Zhang, Vincent YF Tan, and Song Bai. Dragdiffusion: Harnessing diffusion models for interactive point-based image editing. *arXiv preprint arXiv:2306.14435*, 2023.
- [48] Shibiao Xu, Shunpeng Chen, Rongtao Xu, Changwei Wang, Peng Lu, and Li Guo. Local feature matching using deep learning: A survey. *Information Fusion*, 2024.
- [49] Edward Rosten and Tom Drummond. Machine learning for high-speed corner detection. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2006.
- [50] Christopher G Harris, Mike Stephens, et al. A combined corner and edge detector. In *Alvey Vision Conference*, 1988.

- [51] Jiri Matas Dmytro Mishkin, Filip Radenovic. Repeatability is not enough: Learning discriminative affine regions via discriminability. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018.
- [52] Yannick Verdie, Kwang Yi, Pascal Fua, and Vincent Lepetit. Tilde: A temporally invariant learned detector. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.
- [53] Nikolay Savinov, Akihito Seki, Lubor Ladicky, Torsten Sattler, and Marc Pollefeys. Quad-networks: unsupervised learning to rank for interest point detection. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.
- [54] Engin Tola, Vincent Lepetit, and Pascal Fua. A fast local descriptor for dense matching. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2008.
- [55] Michael Calonder, Vincent Lepetit, Christoph Strecha, and Pascal Fua. Brief: Binary robust independent elementary features. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2010.
- [56] Edgar Simo-Serra, Eduard Trulls, Luis Ferraz, Iasonas Kokkinos, Pascal Fua, and Francesc Moreno-Noguer. Discriminative learning of deep convolutional feature point descriptors. In *Proceedings of International Conference on Computer Vision (ICCV)*, 2015.
- [57] Anastasiia Mishchuk, Dmytro Mishkin, Filip Radenovic, and Jiri Matas. Working hard to know your neighbor’s margins: Local descriptor learning loss. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.
- [58] Yurun Tian, Bin Fan, and Fuchao Wu. L2-net: Deep learning of discriminative patch descriptor in euclidean space. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.
- [59] Yurun Tian, Xin Yu, Bin Fan, Fuchao Wu, Huub Heijnen, and Vassileios Balntas. Sosnet: Second order similarity regularization for local descriptor learning. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.
- [60] Kun He, Yan Lu, and Stan Sclaroff. Local descriptors optimized for average precision. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

- [61] Jerome Revaud, Jon Almazán, Rafael S Rezende, and Cesar Roberto de Souza. Learning with average precision: Training image retrieval with a listwise loss. In *Proceedings of International Conference on Computer Vision (ICCV)*, 2019.
- [62] Zixin Luo, Lei Zhou, Xuyang Bai, Hongkai Chen, Jiahui Zhang, Yao Yao, Shiwei Li, Tian Fang, and Long Quan. Aslfeat: Learning local features of accurate shape and localization. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.
- [63] Hyeonwoo Noh, Andre Araujo, Jack Sim, Tobias Weyand, and Bohyung Han. Large-scale image retrieval with attentive deep local features. In *Proceedings of International Conference on Computer Vision (ICCV)*, 2017.
- [64] Michał Tyszkiewicz, Pascal Fua, and Eduard Trulls. Disk: Learning local features with policy gradient. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.
- [65] Xiaoming Zhao, Xingming Wu, Jinyu Miao, Weihai Chen, Peter CY Chen, and Zhengguo Li. Alike: Accurate and lightweight keypoint detection and descriptor extraction. *IEEE Transactions on Multimedia*, 2022.
- [66] Pierre Gleize, Weiyao Wang, and Matt Feiszli. Silk: Simple learned keypoints. In *Proceedings of International Conference on Computer Vision (ICCV)*, 2023.
- [67] Paul-Edouard Sarlin, Cesar Cadena, Roland Siegwart, and Marcin Dymczyk. From coarse to fine: Robust hierarchical localization at large scale. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.
- [68] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. *Proceedings of International Conference on Learning Representations (ICLR)*, 2021.
- [69] Ignacio Rocco, Relja Arandjelović, and Josef Sivic. Efficient neighbourhood consensus networks via submanifold sparse convolutions. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020.
- [70] Wei Jiang, Eduard Trulls, Jan Hosang, Andrea Tagliasacchi, and Kwang Moo Yi. COTR: Correspondence Transformer for Matching Across Images. In *Proceedings of International Conference on Computer Vision (ICCV)*, 2021.

- [71] Hongkai Chen, Zixin Luo, Lei Zhou, Yurun Tian, Mingmin Zhen, Tian Fang, David Mckinnon, Yanghai Tsin, and Long Quan. Aspanformer: Detector-free image matching with adaptive span transformer. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2022.
- [72] Qing Wang, Jiaming Zhang, Kailun Yang, Kunyu Peng, and Rainer Stiefelhagen. Matchformer: Interleaving attention in transformers for feature matching. In *Proceedings of the Asian Conference on Computer Vision (ACCV)*, 2022.
- [73] Johan Edstedt, Ioannis Athanasiadis, Mårten Wadenbäck, and Michael Felsberg. Dkm: Dense kernelized feature matching for geometry estimation. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.
- [74] Johan Edstedt, Qiyu Sun, Georg Bökman, Mårten Wadenbäck, and Michael Felsberg. Roma: Robust dense feature matching. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.
- [75] Dongli Tan, Jiang-Jiang Liu, Xingyu Chen, Chao Chen, Ruixin Zhang, Yunhang Shen, Shouhong Ding, and Rongrong Ji. Eco-tr: Efficient correspondences finding via coarse-to-fine refinement. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2022.
- [76] Junjie Ni, Yijin Li, Zhaoyang Huang, Hongsheng Li, Hujun Bao, Zhaopeng Cui, and Guofeng Zhang. Pats: Patch area transportation with subdivision for local feature matching. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.
- [77] Qunjie Zhou, Torsten Sattler, and Laura Leal-Taixe. Patch2pix: Epipolar-guided pixel-level correspondences. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.
- [78] Axel Barroso-Laguna, Sowmya Munukutla, Victor Adrian Prisacariu, and Eric Brachmann. Matching 2d images in 3d: Metric relative pose from metric correspondences. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.
- [79] Daniel Pizarro and Adrien Bartoli. Feature-based deformable surface detection with self-occlusion reasoning. *International Journal of Computer Vision (IJCV)*, 2012.

- [80] Wen-Yan Daniel Lin, Ming-Ming Cheng, Jiangbo Lu, Hongsheng Yang, Minh N Do, and Philip Torr. Bilateral functions for global motion modeling. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2014.
- [81] Wen-Yan Lin, Siying Liu, Nianjuan Jiang, Minh N Do, Ping Tan, and Jiangbo Lu. Repmatch: Robust feature matching and pose for reconstructing modern cities. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2016.
- [82] Josef Maier, Martin Humenberger, Markus Murschitz, Oliver Zendel, and Markus Vincze. Guided matching based on statistical optical flow for fast and robust correspondence analysis. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2016.
- [83] T Sattler, B Leibe, and L Kobbelt. Improving ransac’s efficiency with a spatial consistency filter [c]. In *Proceedings of International Conference on Computer Vision (ICCV)*, 2009.
- [84] Frederik Schaffalitzky and Andrew Zisserman. Automated scene matching in movies. In *Image and Video Retrieval: International Conference*, 2002.
- [85] Sivic and Zisserman. Video google: A text retrieval approach to object matching in videos. In *Proceedings of International Conference on Computer Vision (ICCV)*, 2003.
- [86] Paul-Edouard Sarlin, Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Superglue: Learning feature matching with graph neural networks. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.
- [87] Philipp Lindenberger, Paul-Edouard Sarlin, and Marc Pollefeys. Lightglue: Local feature matching at light speed. In *Proceedings of International Conference on Computer Vision (ICCV)*, 2023.
- [88] Lars Hammarstrand, Fredrik Kahl, Will Maddern, Tomas Pajdla, Marc Pollefeys, Torsten Sattler, Josef Sivic, Erik Stenborg, Carl Toft, and Akihiko Torii. Long-term visual localization. <https://www.visuallocalization.net/benchmark/>, 2020.
- [89] Yixuan Sun, Yiwen Huang, Haijing Guo, Yuzhou Zhao, Runmin Wu, Yizhou Yu, Weifeng Ge, and Wenqiang Zhang. Misc210k: A large-scale dataset for multi-instance semantic correspondence. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

- [90] Ce Liu, Jenny Yuen, and Antonio Torralba. Sift flow: Dense correspondence across scenes and its applications. *IEEE Transaction on Pattern Analysis and Machine Intelligence (TPAMI)*, 2011.
- [91] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of International Conference on Computer Vision (ICCV)*, 2021.
- [92] Juhong Min, Jongmin Lee, Jean Ponce, and Minsu Cho. Learning to compose hypercolumns for visual correspondence. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020.
- [93] Grace Luo, Lisa Dunlap, Dong Huk Park, Aleksander Holynski, and Trevor Darrell. Diffusion hyperfeatures: Searching through time and space for semantic correspondence. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024.
- [94] Tsung-yi Lin, Piotr Doll, Ross Girshick, Kaiming He, Bharath Hariharan, Serge Belongie, and Facebook Ai. Feature pyramid networks for object detection. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.
- [95] Junyi Zhang, Charles Herrmann, Junhwa Hur, Eric Chen, Varun Jampani, Deqing Sun, and Ming-Hsuan Yang. Telling left from right: Identifying geometry-aware semantic correspondence. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.
- [96] Shuaiyi Huang, Luyu Yang, Bo He, Songyang Zhang, Xuming He, and Abhinav Shrivastava. Learning semantic correspondence with sparse annotations. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2022.
- [97] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of International Conference on Computer Vision (ICCV)*, 2017.
- [98] Seungwook Kim, Juhong Min, and Minsu Cho. Transformatcher: Match-to-match attention for semantic correspondence. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.
- [99] Seokju Cho, Sunghwan Hong, and Seungryong Kim. Cats++: Boosting cost aggregation with convolutions and transformers. *IEEE Transaction on Pattern Analysis and Machine Intelligence (TPAMI)*, 2022.

- [100] Dongyang Zhao, Ziyang Song, Zhenghao Ji, Gangming Zhao, Weifeng Ge, and Yizhou Yu. Multi-scale matching networks for semantic correspondence. In *Proceedings of International Conference on Computer Vision (ICCV)*, 2021.
- [101] Ignacio Rocco, Relja Arandjelović, and Josef Sivic. End-to-end weakly-supervised semantic alignment. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.
- [102] Seungryong Kim, Stephen Lin, Sang Ryul Jeon, Dongbo Min, and Kwanghoon Sohn. Recurrent transformer networks for semantic correspondence. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2018.
- [103] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of International Conference on Computer Vision (ICCV)*, 2017.
- [104] Prune Truong, Martin Danelljan, Fisher Yu, and Luc Van Gool. Warp consistency for unsupervised learning of dense correspondences. In *Proceedings of International Conference on Computer Vision (ICCV)*, 2021.
- [105] Prune Truong, Martin Danelljan, Fisher Yu, and Luc Van Gool. Probabilistic warp consistency for weakly-supervised semantic correspondences. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.
- [106] Xin Li, Deng-Ping Fan, Fan Yang, Ao Luo, Hong Cheng, and Zicheng Liu. Probabilistic model distillation for semantic correspondence. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.
- [107] Jiwon Kim, Byeongho Heo, Sangdoo Yun, Seungryong Kim, and Dongyoon Han. Match me if you can: Semantic correspondence learning with unpaired images. In *Proceedings of the Asian Conference on Computer Vision (ACCV)*, 2023.
- [108] Jiwon Kim, Kwangrok Ryoo, Junyoung Seo, Gyuseong Lee, Daehwan Kim, Hansang Cho, and Seungryong Kim. Semi-supervised learning of semantic correspondence with pseudo-labels. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.
- [109] Xi Chen, Yan Duan, Rein Houthoofd, John Schulman, Ilya Sutskever, and Pieter Abbeel. Infogan: Interpretable representation learning by information maximizing generative adversarial nets. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2016.

- [110] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *Communications of the ACM*, 2020.
- [111] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 1998.
- [112] Zinan Lin, Kiran Thekumparampil, Giulia Fanti, and Sewoong Oh. Infogan-cr and modelcentrality: Self-supervised model training and selection for disentangling gans. In *Proceedings of International Conference on Machine Learning (ICML)*, 2020.
- [113] Xinqi Zhu, Chang Xu, and Dacheng Tao. Where and what? examining interpretable disentangled representations. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.
- [114] Yuxiang Wei, Yupeng Shi, Xiao Liu, Zhilong Ji, Yuan Gao, Zhongqin Wu, and Wangmeng Zuo. Orthogonal jacobian regularization for unsupervised disentanglement in image generation. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.
- [115] Abhishek Kumar, Prasanna Sattigeri, and Avinash Balakrishnan. Variational inference of disentangled latent concepts from unlabeled observations. In *Proceedings of International Conference on Learning Representations (ICLR)*, 2018.
- [116] Irina Higgins, Loic Matthey, Arka Pal, Christopher P Burgess, Xavier Glorot, Matthew M Botvinick, Shakir Mohamed, and Alexander Lerchner. beta-vae: Learning basic visual concepts with a constrained variational framework. *Proceedings of International Conference on Learning Representations (ICLR)*, 2017.
- [117] Ricky TQ Chen, Xuechen Li, Roger B Grosse, and David K Duvenaud. Isolating sources of disentanglement in variational autoencoders. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2018.
- [118] Minyoung Kim, Yuting Wang, Pritish Sahu, and Vladimir Pavlovic. Relevance factor vae: Learning and identifying disentangled factors. *arXiv preprint arXiv:1902.01568*, 2019.
- [119] Hugo Caselles-Dupré, Michael Garcia Ortiz, and David Filliat. Symmetry-based disentangled representation learning requires interaction with environments. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

- [120] Robin Quessard, Thomas Barrett, and William Clements. Learning disentangled representations and group structure of dynamical environments. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.
- [121] Eduardo Hugo Sanchez, Mathieu Serrurier, and Mathias Ortner. Learning disentangled representations via mutual information estimation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020.
- [122] Keyang Zhou, Bharat Lal Bhatnagar, and Gerard Pons-Moll. Unsupervised shape and pose disentanglement for 3d meshes. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020.
- [123] Xin Wang, Hong Chen, Zihao Wu, Wenwu Zhu, et al. Disentangled representation learning. *IEEE Transaction on Pattern Analysis and Machine Intelligence (TPAMI)*, 2024.
- [124] Nils Hasler, Carsten Stoll, Martin Sunkel, Bodo Rosenhahn, and H-P Seidel. A statistical model of human pose and body shape. In *Wiley Computer Graphics Forum (CGF)*, 2009.
- [125] Dragomir Anguelov, Praveen Srinivasan, Daphne Koller, Sebastian Thrun, Jim Rodgers, and James Davis. Scape: shape completion and animation of people. In *Proceedings of ACM SIGGRAPH Conference and Exhibition on Computer Graphics and Interactive Techniques (SIGGRAPH)*, 2005.
- [126] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J. Black. SMPL: A skinned multi-person linear model. *Proceedings of ACM SIGGRAPH Conference and Exhibition on Computer Graphics and Interactive Techniques (SIGGRAPH)*, 2015.
- [127] Ahmed AA Osman, Timo Bolkart, and Michael J Black. Star: Sparse trained articulated human body regressor. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020.
- [128] Chaitanya Patel, Zhouyingcheng Liao, and Gerard Pons-Moll. Tailornet: Predicting clothing in 3d as a function of human pose, shape and garment style. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.
- [129] Igor Santesteban, Nils Thuerey, Miguel A Otaduy, and Dan Casas. Self-supervised collision handling via generative 3d garment models for virtual try-on. *IEEE Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

- [130] Heming Zhu, Yu Cao, Hang Jin, Weikai Chen, Dong Du, Zhangye Wang, Shuguang Cui, and Xiaoguang Han. Deep fashion3d: A dataset and benchmark for 3d garment reconstruction from single images. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020.
- [131] Thiemo Alldieck, Marcus Magnor, Bharat Lal Bhatnagar, Christian Theobalt, and Gerard Pons-Moll. Learning to reconstruct people in clothing from a single rgb camera. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.
- [132] Qianli Ma, Jinlong Yang, Anurag Ranjan, Sergi Pujades, Gerard Pons-Moll, Siyu Tang, and Michael J Black. Learning to dress 3d people in generative clothing. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.
- [133] Bharat Lal Bhatnagar, Garvita Tiwari, Christian Theobalt, and Gerard Pons-Moll. Multi-garment net: Learning to dress 3d people from images. In *Proceedings of International Conference on Computer Vision (ICCV)*, 2019.
- [134] Jae Yong Lee, Joseph DeGol, Victor Fragoso, and Sudipta N Sinha. Patchmatch-based neighborhood consensus for semantic correspondence. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.
- [135] Christopher B Choy, JunYoung Gwak, Silvio Savarese, and Manmohan Chandraker. Universal correspondence network. *arXiv preprint arXiv:1606.03558*, 2016.
- [136] Jiarui Xu, Sifei Liu, Arash Vahdat, Wonmin Byeon, Xiaolong Wang, and Shalini De Mello. Open-vocabulary panoptic segmentation with text-to-image diffusion models. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.
- [137] Bingxin Ke, Anton Obukhov, Shengyu Huang, Nando Metzger, Rodrigo Caye Daudt, and Konrad Schindler. Repurposing diffusion-based image generators for monocular depth estimation. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.
- [138] Wenliang Zhao, Yongming Rao, Zuyan Liu, Benlin Liu, Jie Zhou, and Jiwen Lu. Unleashing text-to-image diffusion models for visual perception. In *Proceedings of International Conference on Computer Vision (ICCV)*, 2023.
- [139] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al.

- Learning transferable visual models from natural language supervision. In *Proceedings of International Conference on Machine Learning (ICML)*, 2021.
- [140] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Learning to prompt for vision-language models. *International Journal of Computer Vision (IJCV)*, 2022.
- [141] Muhammad Uzair Khattak, Hanoona Rasheed, Muhammad Maaz, Salman Khan, and Fahad Shahbaz Khan. Maple: Multi-modal prompt learning. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.
- [142] Shuzhe Wang, Vincent Leroy, Yohann Cabon, Boris Chidlovskii, and Jerome Revaud. Dust3r: Geometric 3d vision made easy. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.
- [143] Vincent Leroy, Yohann Cabon, and Jérôme Revaud. Grounding image matching in 3d with mast3r. *arXiv preprint arXiv:2406.09756*, 2024.
- [144] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *Proceedings of International Conference on Computer Vision (ICCV)*, 2023.