Summary is provided after page 12. It is recommended to read that first before diving into the details of the paper 😉😉

Emergent Correspondence from Image Diffusion

Luming Tang* **Menglin Jia*** **Qianqian Wang***
Cheng Perng Phoo **Bharath Hariharan**
Cornell University

Abstract

Finding correspondences between images is a fundamental problem in computer vision. In this paper, we show that correspondence emerges in image diffusion models *without any explicit supervision*. We propose a simple strategy to extract this implicit knowledge out of diffusion networks as image features, namely DIffusion FeaTures (DIFT), and use them to establish correspondences between real images. Without any additional fine-tuning or supervision on the task-specific data or annotations, DIFT is able to outperform both weakly-supervised methods and competitive off-the-shelf features in identifying semantic, geometric, and temporal correspondences. Particularly for semantic correspondence, DIFT from Stable Diffusion is able to outperform DINO and OpenCLIP by 19 and 14 accuracy points respectively on the challenging SPair-71k benchmark. It even outperforms the state-of-the-art supervised methods on 9 out of 18 categories while remaining on par for the overall performance. Project page: https://diffusionfeatures.github.io.

1 Introduction

Drawing correspondence between images is a fundamental problem in computer vision. Good correspondences are necessary for many applications including 3D reconstruction [74], object tracking [23], video segmentation [88], image editing [59] and image-to-image translation [82]. This problem of drawing correspondence is easy for humans: we can match object parts not only across different viewpoints, articulations and lighting changes, but even across drastically different categories (e.g., between cats and horses) or different modalities (e.g., between photos and cartoons). As humans we are able to learn these correspondence solely by watching and interacting with the world, with no or very few explicit correspondence labels. The question is, can computer vision systems learn such accurate correspondences without any labeled data at all?

Where do we need correspondence in images?

Task we want to perform

For learning from unlabeled data, unsupervised [14] and self-supervised learning [27] algorithms abound. Indeed, there is some evidence that self-supervised learning techniques produce good correspondences as a side product [10, 30]. Meanwhile, there is a recent new class of self-supervised models that has been attracting a lot of attention: diffusion-based generative models [34, 80]. While diffusion models are primarily models for image synthesis, a key observation is that these models produce good results for image-to-image translation [54, 83] and image editing [8]. For instance, they could convert a dog to a cat without changing its pose or context [62]. It would appear that to perform such editing, the model must implicitly reason about correspondence between the two categories (e.g., the model need to know where the dog's eye is in order to replace it with the cat's eye). We therefore ask, do image diffusion models learn correspondences?

Why diffusion-based models though?

We answer the question in the affirmative by construction: we provide a simple way of extracting correspondences on real images using pre-trained diffusion models. These diffusion models [42] have at the core a U-Net [72, 18, 71] that takes noisy images as input and produces clean images as output.

*Equal contribution.

Preprint. Under review.

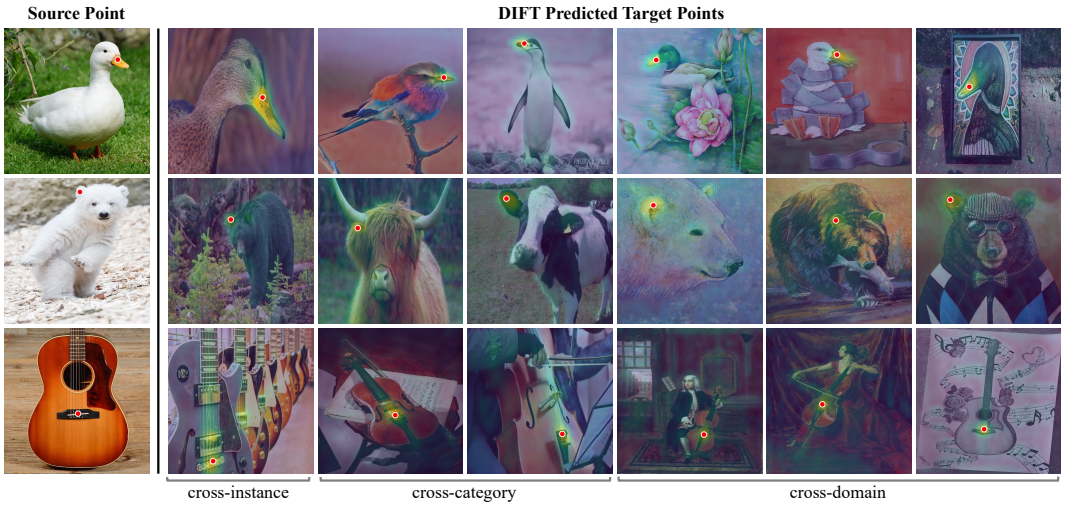


Figure 1: Given a red source point in an image (far left), we would like to develop a model that automatically find the corresponding point in the images on the right. Without any fine-tuning or correspondence supervision, our proposed diffusion features (DIFT) could establish semantic correspondence across instances, categories and even domains, e.g., from a duck to a penguin, from a photo to an oil-painting. More results are in Figs. 9 and 10 of Appendix B.

As such they already extract features from the input image that can be used for correspondence. Unfortunately, the U-Net is trained to *de-noise*, and so has been trained on *noisy* images. Our strategy for handling this issue is simple but effective: we *add noise* to the input image (thus simulating the forward diffusion process) before passing it into the U-Net to extract feature maps. We call these feature maps (and through a slight abuse of notation, our approach) **DI**ffusion **F**ea**T**ures (**DIFT**). DIFT can then be used to find matching pixel locations in the two images by doing simple nearest neighbor lookup using cosine distance. We find the resulting correspondences are surprisingly robust and accurate (Fig. 1), even across multiple categories and image modalities.

A little blocker for the experiment, and the proposed solution

We evaluate DIFT with two different types of diffusion models, on three groups of visual correspondence tasks including semantic correspondence, geometric correspondence, and temporal correspondence. We compare DIFT with other baselines, including task-specific methods, and other self-supervised models trained with similar datasets and similar amount of supervision (DINO [10] and OpenCLIP [38]). Although simple, DIFT demonstrates strong performance on all tasks without any additional fine-tuning or supervision, outperforms both weakly-supervised methods and other self-supervised features, and even remains on par with the state-of-the-art supervised methods on semantic correspondence.

2 Related Work

Visual Correspondence. Establishing visual correspondences between different images is crucial for various computer vision tasks such as Structure-from-Motion / 3D reconstruction [2, 74, 61, 75], object tracking [23, 93], image recognition [64, 81, 9] and segmentation [51, 48, 73, 30]. Traditionally, correspondences are established using hand-designed features, such as SIFT [52] and SURF [6]. With the advent of deep learning, methods that learn to find correspondences in a supervised-learning regime have shown promising results [47, 15, 43, 37]. However, these approaches are difficult to scale due to the reliance on ground-truth correspondence annotations. To overcome difficulties in collecting a large number of image pairs with annotated correspondences, recent works have started looking into how to build visual correspondence models with weak supervision [87] or self-supervision [88, 39]. Meanwhile, recent works on self-supervised representation learning [10] has yielded strong per-pixel features that could be used to identify visual correspondence [82, 3, 10, 30]. In particular, recent work has also found that the internal representation of Generative Adversarial Networks (GAN) [24] could be used for identifying visual correspondence [94, 63, 58] within certain image categories. Our work shares similar spirits with these works: we show that diffusion models could generate

Why is it hard to learn correspondence via supervised learning?

features that are useful for identifying visual correspondence on general images. In addition, we show that features generated at different timesteps and different layers of the de-noising process encode different information that could be used for determining correspondences needed for different downstream tasks.

Diffusion Model [79, 34, 80, 42] is a powerful family of generative models. Ablated Diffusion Model [18] first showed that diffusion could surpass GAN's image generation quality on ImageNet [16]. Subsequently, the introduction of classifier-free guidance [35] and latent diffusion model [71] made it scale up to billions of text-image pairs [76], leading to the popular open-sourced text-to-image diffusion model, i.e., Stable Diffusion. With its superior generation ability, recently people also start investigating the internal representation of diffusion models. For example, previous works [83, 33] found that the intermediate-layer features and attention maps of diffusion models are crucial for controllable generations; other works [5, 90, 96] explored adapting pre-trained diffusion models for various downstream visual recognition tasks. Different from these works, we are the first to directly evaluate the efficacy of features inherent to pre-trained diffusion models on various visual correspondence tasks.

Alignment with the other research in a similar direction

3 Problem Setup

Given two images I_1, I_2 and a pixel location p_1 in I_1, we are interested in finding its corresponding pixel location p_2 in I_2. Relationships between p_1 and p_2 could be semantic correspondence (i.e., pixels of different objects that share similar semantic meanings), geometric correspondence (i.e., pixels of the same object captured from different viewpoints), or temporal correspondence (i.e., pixels of the same object in a video that may deform over time).

The most straightforward approach to obtaining pixel correspondences to is to first extract dense image features in both images and then match them. Specifically, we denote the dense feature map of I_i as F_i, and the pixel-level feature at location p as $F_i(p)$, which is extracted through bilinear interpolation on F_i. Then we can obtain the pixel correspondence for p_1 as:

$$p_2 = \arg\min_p d(F_1(p_1), F_2(p)) \tag{1}$$

p = pixel in ith image
F = Feature map
I = Image
d = cosine distance

where d is a distance metric and we use cosine distance by default in this work.

4 Diffusion Features (DIFT)

In this section, we first review what diffusion models are and then explain how we extract dense features on real images using pre-trained diffusion models.

4.1 Image Diffusion Model

Diffusion models [34, 80] are generative models that transform a Normal distribution to an arbitrary data distribution. In our case, we use image diffusion models, thus the data distribution and the Gaussian prior are both over the space of 2D images.

During training, Gaussian noise of different magnitudes is added to clean data points to obtain noisy data points. This is typically thought of as a "diffusion" process, where the starting point of the diffusion x_0 is a clean image from the training dataset and x_t is a noisy image obtained by "mixing" x_0 with noise:

$$x_t = \sqrt{\alpha_t} x_0 + (\sqrt{1-\alpha_t})\epsilon \tag{2}$$

where $\epsilon \sim \mathcal{N}(0, \mathbf{I})$ is the randomly-sampled noise, and $t \in [0, T]$ indexes "time" in the diffusion process with larger time steps involving more noise. The amount of noise is determined by $\{\alpha_t\}_1^T$, which is a pre-defined noise schedule. We call this the diffusion *forward* process.

A neural network f_θ is trained to take x_t and time step t as input and predict the input noise ϵ. For image generation, f_θ is usually parametrized as a U-Net [72, 18, 71]. Once trained, f_θ can be used to "reverse" the diffusion process. Starting from pure noise x_T sampled from a Normal distribution, f_θ can be iteratively used to estimate noise ϵ from the noisy data x_t and remove this noise to get a cleaner data x_{t-1}, eventually leading to a sample x_0 from the original data distribution. We call this the diffusion *backward* process.

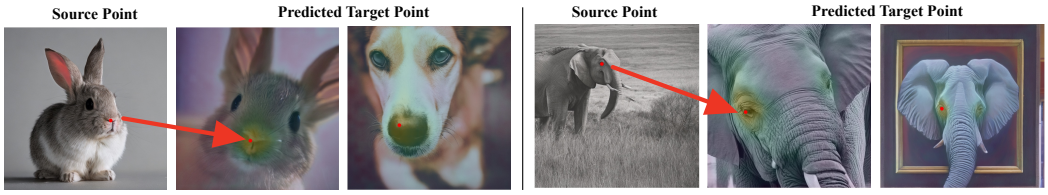


Figure 2: Given a Stable Diffusion generated image, we extract its intermediate layer activations at a certain time step t during its backward process, and use them as the feature map to predict the corresponding points. Although simple, this method produces correct correspondences on generated images already not only within category, but also cross-category, even in cross-domain situations, e.g., from a photo to an oil painting.

4.2 Extract Diffusion Features on Real Images

We hypothesize that diffusion models learn correspondence implicitly [83, 62] in Sec. 1. However, to verify this claim and extract this knowledge from a black box neural network, we need to devise a methodology. To begin our exploration, we focus on *generated* images, where we have access to the complete internal state of the network throughout the entire backward process. To conduct our experiments, we utilize the publicly available state-of-the-art diffusion model, i.e., Stable Diffusion [71]. Given each generated image, we extract the feature maps of its intermediate layers at a specific time step t during the backward process, which we then utilize to establish correspondences between two different generated images as described in Sec. 3. As illustrated in Fig. 2, this straightforward approach allows us to find correct correspondences between generated images, even when they belong to different categories or domains.

Why generated images first?

Given effective correspondences on *generated* images, a natural question arises: how can we obtain similar features for *real* images? The challenge lies in the fact that the real image itself does not belong to the training distribution of the U-Net (which was trained on noisy images), and we do not have access to the intermediate noisy images that would have been produced during the generation of this image. Fortunately, we found a simple approximation using the forward diffusion process to be effective enough. Specifically, we first add *noise* of time step t to the real image (Eq. (2)) to move it to the x_t distribution, and then feed it to network f_θ together with t to extract the intermediate layer activations as our DIffusion FeaTures, namely DIFT. As shown in Figs. 1 and 3, this approach yields surprisingly good correspondences for real images.

The challenge in repeating the same exercise for real images

Proposed solution

Moving forward, a crucial consideration is the selection of the time step t and the network layer from which we extract features. Intuitively we find that a larger t and an earlier network layer tend to yield more semantically-aware features, while a smaller t and a later layer focus more on low-level details. The optimal choice of t and layer depend on the specific correspondence task at hand, as different tasks may require varying trade-offs between semantic and low-level features. For example, semantic correspondence likely benefits from more semantic-level features, whereas geometric correspondence between two views of the same instance may perform well with low-level features. We therefore use a 2D grid search to determine these two hyper-parameters for each correspondence task. For a comprehensive list of the hyper-parameter values used in this paper, please refer to Appendix A.

The importance of time step t and the feature layer

Lastly, to enhance the stability of the representation in the presence of random noise added to the input image, we extract features from multiple noisy versions with different samples of noise, and average them to form the final representation.

5 Semantic Correspondence

In this section, we investigate how to use the proposed DIFT to identify pixels that share similar semantic meanings across images (e.g., the eyes of two different cats in two different images).

5.1 Model Variants and Baselines

We extract DIFT from two commonly used, open-sourced image diffusion models: Stable Diffusion 2-1 (SD) [71] and Ablated Diffusion Model (ADM) [18]. SD is trained on the LAION [76] whereas

Figure 3: Visualization of semantic correspondence prediction on SPair-71k using different features. The leftmost image is the source image with a set of keypoints; the rightmost image contains the ground-truth correspondence for a target image whereas any images in between contain keypoints found using feature matching with various features. Different colors indicate different keypoints. We use circles to indicate correctly-predicted points under the threshold $\alpha_{bbox} = 0.1$ and crosses for incorrect matches. DIFT is able to establish correct correspondences under clustered scenes (row 3), viewpoint change (row 2 and 4), and occlusions (row 5). See Fig. 11 in Appendix B for more results.

ADM is trained on ImageNet [16] without labels. We call these two features DIFT$_{sd}$ and DIFT$_{adm}$ respectively.

To separate the impact of training data on the performance of DIFT, we also evaluate two other commonly used self-supervised features as baselines that share basically the same training data: OpenCLIP [38] with ViT-L/14 [19] trained on LAION, as well as DINO [10] with ViT-B/8 trained on ImageNet [16] without labels. Note that for both DIFT and other self-supervised features, we do not fine-tune or re-train the models with any additional data or supervision.

5.2 Benchmark Evaluation

Datasets. We conduct evaluation on three popular benchmarks: SPair-71k [56], PF-WILLOW [29] and CUB-200-2011 [86]. SPair-71k is the most challenging semantic correspondence dataset, containing diverse variations in viewpoint and scale with 12,234 image pairs on 18 categories for testing. PF-Willow is a subset of PASCAL VOC dataset [21] with 900 image pairs for testing. For CUB, following [59], we evaluate 14 different splits of CUB (each containing 25 images) and report the average performance across all splits.

Evaluation Metric. Following prior work, we report the percentage of correct keypoints (PCK). The predicted keypoint is considered to be correct if they lie within $\alpha \cdot \max(h, w)$ pixels from the ground-truth keypoint for $\alpha \in [0, 1]$, where h and w are the height and width of either the image (α_{img}) or the bounding box (α_{bbox}). To find a suitable time step and layer feature to use for DIFT

Table 1: PCK($\alpha_{bbox} = 0.1$) `per image` of various methods on SPair-71k. All the DIFT results have gray background for better reference. Based on the supervision used, methods are grouped into 3 groups: (a) fully supervised with correspondence annotations, (b) weakly supervised with in-domain image pairs or collections, (c) no supervision. Best numbers inside group (a) are **bolded**. Colors of numbers indicate the **best**, second-best results among group (b) and (c). Without any supervision (either correspondence or in-domain image collection), both DIFT$_{sd}$ and DIFT$_{adm}$ outperform previous weakly-supervised methods with a large margin; DIFT$_{sd}$ even outperforms the best supervised methods on 9 out of 18 categories, and only have 2 points behind on overall PCK. Compared to other off-the-shelf features such as OpenCLIP and DINO, DIFT is able to outperform its contrastive-learning counterparts by over 12 points.

Sup.	Method	SPair-71K Category																	
		Aero	Bike	Bird	Boat	Bottle	Bus	Car	Cat	Chair	Cow	Dog	Horse	Motor	Person	Plant	Sheep	Train	TV	All
(a)	CATs [15]	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
	MMNet [95]	55.9	37.0	65.0	35.4	50.0	63.9	45.7	62.8	**28.7**	65.0	54.7	51.6	38.5	34.6	41.7	36.3	77.7	62.5	50.4
	TransforMatcher [43]	**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 [37]	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**
(b)	NCNet [70]	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
	CNNGeo [68]	23.4	16.7	40.2	14.3	36.4	27.7	26.0	32.7	12.7	27.4	22.8	13.7	20.9	21.0	17.5	10.2	30.8	34.1	20.6
	WeakAlign [69]	22.2	17.6	41.9	15.1	38.1	27.4	27.2	31.8	12.8	26.8	22.6	14.2	20.0	22.2	17.9	10.4	32.2	35.1	20.9
	A2Net [77]	22.6	18.5	42.0	16.4	37.9	30.8	26.5	35.6	13.3	29.6	24.3	16.0	21.6	22.8	20.5	13.5	31.4	36.5	22.3
	SFNet [46]	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 [49]	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
	PSCNet [40]	28.3	17.7	45.1	15.1	37.5	30.1	27.5	47.4	14.6	32.5	26.4	17.7	24.9	24.5	19.9	16.9	34.2	37.9	27.0
(c)	DINO [10]	43.6	27.2	64.9	24.0	30.5	31.4	28.3	55.2	16.8	40.2	37.1	32.9	29.1	41.1	22.0	26.8	36.4	26.9	33.9
	DIFT$_{adm}$ (ours)	49.7	39.2	77.5	29.3	40.9	36.1	30.5	75.5	23.7	63.7	**52.8**	49.3	34.1	**52.3**	39.3	37.3	59.6	45.4	46.3
	OpenCLIP [38]	51.7	31.4	68.7	28.4	31.5	34.9	**36.1**	56.4	21.1	44.5	41.5	41.2	41.2	51.8	21.7	28.6	46.3	20.7	38.4
	DIFT$_{sd}$ (ours)	**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**

Table 2: PCK($\alpha_{bbox} = 0.1$) `per point` of various methods on SPair-71k. The groups and colors follow Tab. 1. "Mean" denotes the PCK averaged over categories. Same as in Tab. 1, without any supervision, both DIFT$_{sd}$ and DIFT$_{adm}$ outperform previous weakly-supervised methods with a large margin, and also outperform their contrastive-learning counterparts by over 14 points.

Sup.	Method	SPair-71K Category																		
		Aero	Bike	Bird	Boat	Bottle	Bus	Car	Cat	Chair	Cow	Dog	Horse	Motor	Person	Plant	Sheep	Train	TV	Mean	All
(b)	NBB [1, 28]	29.5	22.7	61.9	26.5	20.6	25.4	14.1	23.7	14.2	27.6	30.0	29.1	24.7	27.4	19.1	19.3	24.4	22.6	27.4	-
	GANgealing [63]	-	37.5	-	-	-	-	-	67.0	-	-	23.1	-	-	-	-	-	-	57.9	-	-
	NeuCongeal [59]	-	29.1	-	-	-	-	-	53.3	-	-	35.2	-	-	-	-	-	-	-	-	-
	ASIC [28]	57.9	25.2	68.1	24.7	35.4	28.4	30.9	54.8	21.6	45.0	47.2	39.9	26.2	48.8	14.5	24.5	49.0	24.6	36.9	-
(c)	DINO [10]	45.0	29.5	66.3	22.8	32.1	36.3	31.7	54.8	18.7	43.1	39.2	34.9	31.0	44.3	23.1	29.4	38.4	27.1	36.0	36.7
	DIFT$_{adm}$ (ours)	51.6	40.4	77.6	30.7	43.0	47.2	42.1	74.9	26.6	67.3	**55.8**	52.7	36.0	**55.9**	46.3	45.7	62.7	47.4	50.2	52.0
	OpenCLIP [38]	53.2	33.4	69.4	28.0	33.3	41.0	41.8	55.8	23.3	47.0	43.9	44.1	43.5	55.1	23.6	31.7	47.8	21.8	41.0	41.4
	DIFT$_{sd}$ (ours)	**63.5**	**54.5**	**80.8**	**34.5**	**46.2**	**52.7**	**48.3**	**77.7**	**39.0**	**76.0**	54.9	**61.3**	**53.3**	46.0	**57.8**	**57.1**	**71.1**	**63.4**	**57.7**	**59.5**

and other self-supervised features, we grid search the hyper-parameters using SPair-71k and use the same hyper-parameter settings for PF-WILLOW and CUB.

We notice there are discrepancies in the measurement of PCK in previous literature. Some works [37, 43, 15] [1] use the total number of correctly-predicted points in the whole dataset (or each category split) divided by the total number of predicted points as the final PCK, while some works [63, 59, 28] [2] first calculate a PCK value for each image and then average it across the dataset (or each category split). We denote the first number as PCK `per point` and the second as PCK `per image`. We calculate both metrics for DIFT and self-supervised features, and compare them to methods using that metric respectively.

proof that evaluation in academic benchmarks is still a mess 😓

Quantitative Results. We report our results in Tabs. 1 to 3. In addition to feature matching using DINO and OpenCLIP, we also report state-of-the-art fully-supervised and weakly-supervised methods in the respective tables for completeness. Across the three datasets, we observe that features learned via diffusion are much more suitable for establishing semantic correspondence compared to features learned using contrastive approaches (DIFT$_{sd}$ vs. OpenCLIP, DIFT$_{adm}$ vs. DINO).

Even without any supervision (be it explicit correspondence or in-domain data), DIFT outperforms all the weakly-supervised baselines on all benchmarks by a large margin. It even outperforms the state-of-the-art supervised methods on PF-WILLOW, and for 9 out of 18 categories on SPair-71k.

Qualitative Results. To get a better understanding of DIFT's performance, we visualize a few correspondences on SPair-71k using various off-the-shelf features in Fig. 3. We observe that DIFT is

[1] ScorrSAN [37]'s evaluation code snippet, which calculates PCK `per image`.

[2] GANgealing [63]'s evaluation code snippet, which calculates PCK `per point`.

Table 3: Comparison with state-of-the-art methods on PF-WILLOW PCK `per image` (left) and CUB `per point` (right). Colors of numbers indicate the **best**, second-best results. All the DIFT results have gray background for better reference. DIFT$_{sd}$ achieves the best results without any fine-tuning or supervision with in-domain annotations or data.

Sup.	Method	PCK@α_{bbox}	
		$\alpha = 0.05$	$\alpha = 0.10$
(a)	SCNet [31]	38.6	70.4
	DHPF [57]	49.5	77.6
	PMD [49]	-	75.6
	CHM [55]	52.7	79.4
	CATs [15]	50.3	79.2
	TransforMatcher [43]	-	76.0
	SCorrSAN [37]	54.1	80.0
(c)	DINO [10]	30.8	51.1
	DIFT$_{adm}$ (ours)	46.9	67.0
	OpenCLIP [38]	34.4	61.3
	DIFT$_{sd}$ (ours)	**58.1**	**81.2**

Sup.	Method	PCK@$\alpha_{img} = 0.1$
(b)	GANgealing [63]	56.8
	NeuCongeal [59]	65.6
(c)	DINO [10]	66.4
	DIFT$_{adm}$ (ours)	78.0
	OpenCLIP [38]	67.5
	DIFT$_{sd}$ (ours)	**83.5**

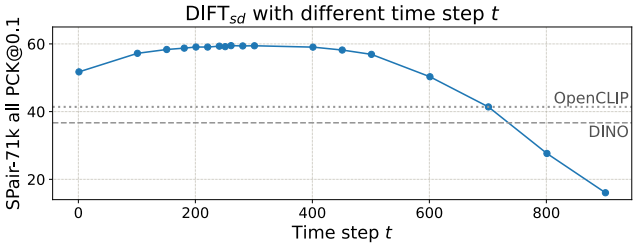
Figure 4: Given image patch specified in the leftmost image (red rectangle), we use DIFT$_{sd}$ to retrieve the top-5 nearest patches in images from different categories in the SPair-71k test set. DIFT is able to find correct correspondence for different objects sharing similar semantic parts, e.g., the wheel of an airplane vs. the wheel of a bus. More results are in Fig. 12 of Appendix B.

able to identify correct correspondences under cluttered scenes, viewpoint changes, and instance-level appearance changes. More results are in Fig. 11 of Appendix B.

In addition to visualizing correspondence within the same categories in SPair-71k, we also visualize the correspondence established using DIFT$_{sd}$ across various categories in Fig. 4. Specifically, we select an image patch from a random image and query the image patches with the nearest DIFT embedding in the rest of the test split but from different categories. DIFT is able to identify correct correspondence across various categories. More results are in Fig. 12 of Appendix B.

Sensitivity to the choice of time step t. For DIFT$_{sd}$, we plot how its PCK `per point` varies with different choices of t on SPair-71k in Fig. 5. DIFT is robust to the choice of t on semantic correspondence, as a wide range of t outperforms the other off-the-shelf self-supervised features.

The robustness of correspondence is amazing TBH!



Wow!! 😮

Figure 5: The PCK `per point` of DIFT$_{sd}$ on SPair-71k. DIFT is robust to the choice of time step t. It yields high PCA scores with a wide range of t, outperforming other off-the-shelf self-supervised features.

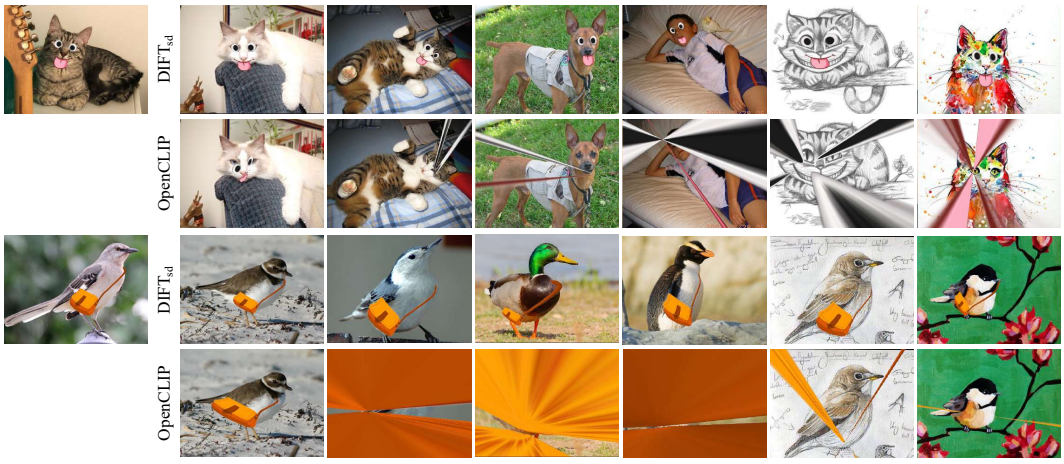


Figure 6: Edit propagation. The first column shows the source image with edits, and the rest columns are the propagated results on new images from different instances, categories, and domains, respectively. Compared to OpenCLIP, DIFT$_{sd}$ propagates edits much more accurately. More results are in Fig. 13 of Appendix B.

5.3 Application: Edit Propagation

One application of DIFT is image editing: we can propagate edits in one image to others that share semantic correspondences. This capability is demonstrated in Fig. 6, where we showcase DIFT's ability to reliably propagate edits across different instances, categories, and domains, without any correspondence supervision. More results are in Appendix B, Fig. 13.

Simple application of semantic correspondence

To achieve this propagation, we simply compute a homography transformation between the source and target images using only matches found in the regions of the intended edits. By applying this transformation to the source image edits, we can integrate them into the corresponding regions of the target image. Figure 6 shows the results for both OpenCLIP and DIFT$_{sd}$ using the same propagation techniques. OpenCLIP fails to compute reasonable transformation due to the lack of reliable correspondences. In contrast, DIFT$_{sd}$ achieves much better results, further justifying the effectiveness of DIFT in finding semantic correspondences.

6 Other Correspondence Tasks

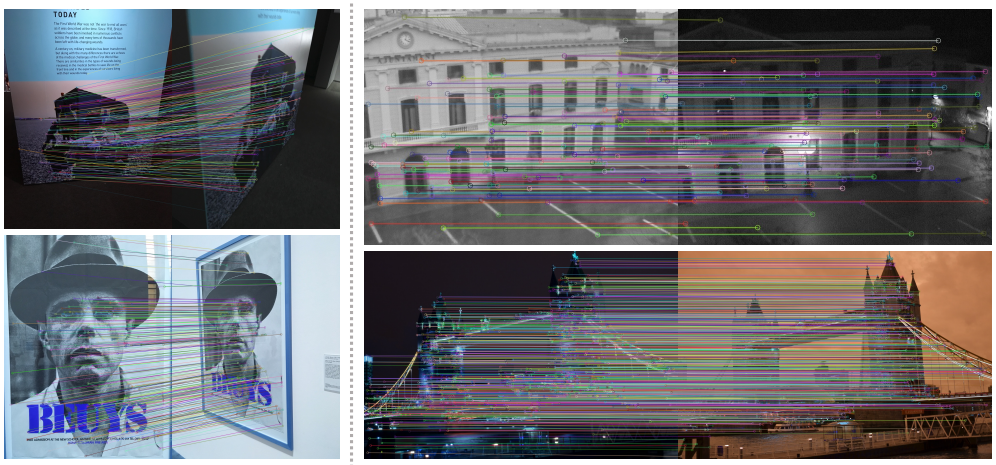
We also evaluate DIFT on geometric correspondence and temporal correspondence. Same as in Sec. 5, we compare DIFT to its other off-the-shelf self-supervised features as well as task-specific methods.

6.1 Geometric Correspondence

Intuitively, we find when t is small, DIFT focuses more on low-level details, which makes it useful as a geometric feature descriptor.

Table 4: Homography estimation accuracy [%] at 1, 3, 5 pixels on HPatches. Colors of numbers indicate the **best**, second-best results. All the DIFT results have gray background for better reference. DIFT with SuperPoint keypoints achieves competitive performance.

Method	Geometric Supervision	$\epsilon = 1$	$\epsilon = 3$	$\epsilon = 5$
SIFT [52]	None	40.5	68.1	77.6
LF-Net [60]	Strong	34.8	62.9	73.8
SuperPoint [17]	Strong	37.4	73.1	82.8
D2-Net [20]	Strong	16.7	61.0	75.9
ContextDesc [53]	Strong	41.0	73.1	82.2
R2D2 [67]	Strong	40.0	75.0	84.7
w/ SuperPoint kp.				
CAPS [87]	Weak	44.8	74.5	**85.7**
DINO [10]	None	37.0	69.1	82.0
DIFT$_{adm}$ (ours)	None	44.6	73.3	83.3
OpenCLIP [38]	None	29.1	63.5	74.3
DIFT$_{sd}$ (ours)	None	**45.9**	**77.8**	84.6



Viewpoint Change | Illumination Change

Figure 7: Sparse feature matching using DIFT$_{sd}$ on HPatches after removing outliers. Left are image pairs under viewpoint change, and right are ones under illumination change. Although never trained with correspondence labels, it works well under both challenging changes. More results are in Fig. 14 of Appendix B.

Setup. We evaluate DIFT for homography estimation using the HPatches benchmark [4]. It contains 116 sequences, where 57 sequences have illumination changes and 59 have viewpoint changes. We follow the corner correctness metric used in SuperPoint [17], and transform the four corners of one image into the other image. The four corners transformed using estimated homography are then compared with those computed using the ground-truth homography. We deem the estimated homography correct if the average error of the four corners is less than ϵ pixels.

Results. Following SuperPoint [17] and CAPS [87], we extract a maximum of 1,000 keypoints from each image, and use RANSAC to estimate the homography from mutual nearest neighbor matches. We report the comparison of homography accuracy between DIFT and other methods in Tab. 4. Visualization of the matched points can be found in Fig. 7. Though not trained using any explicit geometry supervision, DIFT is still on par with the methods that utilize explicit geometric supervision signals designed specifically for this task, such as correspondences obtained from Structure-from-Motion [74] pipelines. This shows that not only semantic-level correspondence, but also geometric correspondence emerges from image diffusion models. More results are in Fig. 14 of Appendix B.

6.2 Temporal Correspondence

DIFT also demonstrates strong performance on temporal correspondence tasks, including video object segmentation and pose tracking, although never trained or fine-tuned on such video data.

Table 5: Video label propagation results on DAVIS-2017 and JHMDB. Colors of numbers indicate the **best**, second-best results. All the DIFT results have gray background for better reference. DIFT even outperforms other self-supervised learning methods specifically trained with video data.

(pre-)Trained on Videos	Method	Dataset	DAVIS $\mathcal{J}\&\mathcal{F}_m$	DAVIS $\mathcal{J}_m$	DAVIS $\mathcal{F}_m$	JHMDB PCK@0.1	JHMDB PCK@0.2
✗	InstDis[89]	ImageNet [16] w/o labels	66.4	63.9	68.9	58.5	80.2
	MoCo [32]		65.9	63.4	68.4	59.4	80.9
	SimCLR [12]		66.9	64.4	69.4	59.0	80.8
	BYOL [26]		66.5	64.0	69.0	58.8	80.9
	SimSiam [13]		67.2	64.8	68.8	59.9	81.6
	DINO [10]		71.4	67.9	74.9	57.2	81.2
	DIFT$_{adm}$ (ours)		**75.7**	**72.7**	**78.6**	**63.4**	**84.3**
	OpenCLIP [38]	LAION [76]	62.5	60.6	64.4	41.7	71.7
	DIFT$_{sd}$ (ours)		70.0	67.4	72.5	61.1	81.8
✓	VINCE [25]	Kinetic [11]	65.2	62.5	67.8	58.8	80.4
	VFS [91]		68.9	66.5	71.3	60.9	80.7
	UVC [50]		60.9	59.3	62.7	58.6	79.6
	CRW [39]		67.6	64.8	70.2	58.8	80.3
	Colorization [85]		34.0	34.6	32.7	45.2	69.6
	CorrFlow [45]	OxUvA [84]	50.3	48.4	52.2	58.5	78.8
	TimeCycle [88]	VLOG [22]	48.7	46.4	50.0	57.3	78.1
	MAST [44]	YT-VOS [92]	65.5	63.3	67.6	-	-
	SFC [36]		71.2	68.3	74.0	61.9	83.0

Setup. We evaluate the learned representation on two challenging video tasks: (1) DAVIS-2017 video instance segmentation benchmark [66], (2) Joints for the HMDB (JHMDB) benchmark [41], which involves tracking 15 human pose keypoints.

Following evaluation setups in [50, 39, 10, 91], representations are used as a similarity function: we segment scenes with nearest neighbors between consecutive video frames. Note that there is no training involved in this label propagation process. We report region-based similarity $\mathcal{J}$ and contour-based accuracy $\mathcal{F}$ [65] for DAVIS, PCK for JHMDB.

Results. Table 5 reports the experimental results, comparing DIFT with a wide range of self-supervised features (pre-)trained with or without video data. DIFT$_{adm}$ outperforms all the other self-supervised learning methods, on both benchmarks, even surpassing models specifically trained on video data by a significant margin. DIFT also yields the best results within the same pre-training dataset. We also show qualitative results in Fig. 8, presenting examples of video instance segmentation results, comparing DIFT$_{adm}$ with DINO. DIFT$_{adm}$ produces masks with clearer boundaries when single or multiple objects are presented in the scene. DIFT$_{adm}$ also attends well to objects in the presence of occlusion (see bottom example). Figure 15 of Appendix B includes more visualizations.

7 Discussion and Conclusion

Would diffusion inversion help? Another way to get x_t from a real input image is diffusion inversion. We tried using DDIM inversion [78] to recover input image's corresponding x_t, then feeding into f_θ to get diffusion feature. However, we don't see much difference in performance on SPair-71k. Meanwhile inversion make the inference process several times slower. We'll leave how to utilize diffusion inversion to get better correspondence to future work.

Does correspondence also exist in SD's encoder? We also evaluated SD's VAE encoder's performance on all benchmarks and find its performance was lower by an order of magnitude. So DIFT$_{sd}$'s correspondence only emerges inside its U-Net and requires diffusion-based training.

Would task-specific adaptation lead DIFT to better results? More sophisticated mechanisms could be applied to further enhance the diffusion features, e.g., concatenating and re-weighting features from different time step t and different network layers, or even fine-tuning the network with task-specific supervision. Some recent works [5, 90, 96] fine-tune either the U-Net or the attached head for dense prediction tasks and yield better performance. However, task-specific adaptation entangles the quality

Figure 8: Video label propagation results on DAVIS-2017. Colors indicate segmentation masks for different instances. Blue rectangles show the first frames. Compared to DINO, DIFT$_{adm}$ produces masks with more accurate and sharper boundaries. More results are in Fig. 15 of Appendix B.

of the features themselves with the efficacy of the fine-tuning procedure. To keep the focus on the representation, we chose to avoid any fine-tuning to demonstrate the quality of the off-the-shelf DIFT. Nevertheless, our preliminary experiments suggest that such fine-tuning would indeed further improve performance on correspondence. We'll leave how to better adapt DIFT to downstream tasks to future works.

Ethical Considerations. Although DIFT can be used with any diffusion model parameterized with a U-Net, the dominant publicly available model is the one trained on LAION [76]. The LAION dataset has been identified as having several issues including racial bias, stereotypes and pornography [7]. Diffusion models trained on these datasets inherit these issues. While these issues may a priori seem less important for estimating correspondences, it might lead to differing accuracies for different kinds of images. One could obtain the benefit of good correspondences without the associated issues if one could trained a diffusion model on a curated dataset. Unfortunately, the huge computational cost also prohibits the training of diffusion models in academic settings on cleaner datasets. We hope that our results encourage efforts to build more carefully trained diffusion models.

It's high time that we debias all our datasets! Arghhh this sucks! 😤

Conclusion. This paper demonstrates that correspondence emerges from image diffusion models without explicit supervision. We propose a simple technique to extract this implicit knowledge out of deep neural nets as a feature map named DIFT, and make it help us do a variety of correspondence tasks on real images. With extensive experiments, we show that although without any explicit supervision, DIFT outperforms both weakly-supervised methods and other off-the-shelf self-supervised features in identifying semantic, geometric and temporal correspondences, and it even remains on par with the state-of-the-art supervised methods on semantic correspondence. We hope our work would inspire future research on how to better utilize these emergent correspondence from image diffusion, as well as rethinking diffusion models as self-supervised models.

Acknowledgement. This work was partially funded by NSF 2144117 and the DARPA Learning with Less Labels program (HR001118S0044). We would like to thank Zeya Peng for her help on the edit propagation section and the project page, thank Kamal Gupta for sharing the evaluation details in the ASIC paper, and thank Aaron Gokaslan, Utkarsh Mall, Jonathan Moon, Boyang Deng for valuable discussion and feedback.

A Implementation Details

The total time step T for both diffusion models (ADM and SD) is 1000. U-Net consists of downsampling blocks, middle blocks and upsampling blocks. We only extract features from the upsampling blocks. ADM's U-Net has 18 upsampling blocks and SD's U-Net has 4 upsampling blocks (the definition of blocks are different between these two models). Feature maps from the n-th upsampling block output are used as the final diffusion feature. As mentioned in the last paragraph of Sec. 4.2, when extracting features for one single image, we use a batch of random noise to get an averaged feature map. The batch size is 8 by default. Sometimes we shrink it to 4 due to the GPU memory constraints. This section lists the time step t and upsampling block index n (n starts from 0) we used for each DIFT variant on different tasks.

Semantic Correspondence. We use $t = 101$ and $n = 4$ for DIFT$_{adm}$, $t = 261$ and $n = 1$ for DIFT$_{sd}$. These hyper-parameters are shared on all semantic correspondence tasks including SPair-71k, PF-WILLOW, and CUB, as well as the visualizations in Figs. 1, 9 and 10. We don't use image-specific prompts for DIFT$_{sd}$. Instead, we use a general prompt "a photo of a `[class]`" where `[class]` denotes the string of the input images' category, which is given by the dataset. For example, for the images of SPair-71k under cat class, the prompt would be "a photo of a cat".

Geometric Correspondence. On HPatches, we use $t = 26$, $n = 11$ for DIFT$_{adm}$ and $t = 0$, $n = 2$ for DIFT$_{sd}$. In addition, for DIFT$_{sd}$, each image's prompt is a null prompt, i.e., an empty string `""`.

Temporal Correspondence. The configurations we use for DIFT$_{adm}$ and DIFT$_{sd}$ are:

dataset	method	time step t	block index n	task-specific hyper-params
DAVIS-2017	DIFT$_{adm}$	51	7	0.1 / 15 / 10 / 28
DAVIS-2017	DIFT$_{sd}$	51	2	0.2 / 15 / 15 / 28
JHMDB	DIFT$_{adm}$	101	5	0.2 / 5 / 15 / 28
JHMDB	DIFT$_{sd}$	51	2	0.1 / 5 / 15 / 14

Last column follows the format of temperature for softmax / propagation radius / top-k similar labels / number of preceding frames. In addition, for DIFT$_{sd}$, each image's prompt is a null prompt, i.e., an empty string `""`.

B Additional Qualitative Results

Correspondence on diverse internet images. Same as Fig. 1, in Figs. 9 and 10 we show more correspondence prediction on various image groups that share similar semantics. For each target image, the DIFT$_{sd}$ predicted point will be displayed as a red circle, together with a heatmap showing the per-pixel cosine distance calculated using DIFT$_{sd}$. We can see it works well across instances, categories, and even image domains, e.g., from an umbrella photo to an umbrella logo.

Semantic correspondence comparison among off-the-shelf features on SPair-71k. Same as Fig. 3, we show more comparison in Fig. 11, where we can see DIFT works well under challenging occlusion, viewpoint change and intra-class appearance variation.

Cross-category semantic correspondence. Same as Fig. 4, in Fig. 12 we select an interesting image patch from a random source image and query the image patches with the nearest DIFT$_{sd}$ features in the rest of the test split but with different categories. We see that DIFT is able to identify reasonable correspondence across various categories.

Image editing propagation. Similar to Fig. 6, Fig. 13 shows more examples on edit propagation using our proposed DIFT$_{sd}$. It further demonstrates the effectiveness of DIFT on finding semantic correspondence, even when source image and target image are from different categories or domains.

Geometric correspondence. Same as Fig. 7, in Fig. 14 we show the sparse feature matching results using DIFT$_{sd}$ on HPatches. Though not trained using any explicit geometry supervision, DIFT still works well under large viewpoint change and challenging illumination change.

Temporal correspondence. Similar to Fig. 8, Fig. 15 presents additional examples of video instance segmentation results on DAVIS-2017, comparing DINO, DIFT$_{adm}$ and Ground-truth (GT). We can see DIFT$_{adm}$ could create instance masks that closely follow the silhouette of instances (see the car on the first row of Fig. 15 as an example).

Summary

Page 1

- Drawing correspondence between images is a fundamental problem in computer vision. Good correspondences are necessary for many applications like object tracking, video segmentation, image editing, etc.
- Drawing correspondences is easy for humans. We can do it across drastically different categories or even different modalities. On the other hand, it's a hard problem to solve in computer vision.
- Given the recent advancements, the authors seek an answer to the question: Can computers learn accurate correspondences without any labeled data?
- Diffusion models produce exceptional results for image-to-image translation tasks. They can seamlessly convert an object of one category to an object of another category without changing the context or pose. It suggests that diffusion-based models might learn correspondences implicitly. But how do we formalize it?

Page 2, 3

- Existing deep learning models have shown great success in drawing correspondences. The problem with these models is that most SOTA models rely on labeled data, and scaling these datasets for better results is hard because of the limited number of ground truth samples.
- Given the success of diffusion-based models on tasks like image-to-image translation, the authors try to find out if diffusion models learn correspondences implicitly.
- To the same end, the authors leverage pre-trained UNet-based diffusion models. They use the Stable Diffusion model for performing all experiments in this paper.

Problem Setup

- Given two images I1, I2, and a pixel location p1 in I1, we are interested in finding its corresponding pixel location p2 in I2. The relationship between pixels p1 and p2 can be semantic, geometric, or temporal.
- For obtaining pixel correspondence, we can extract image features in both the images, and match them. The feature map for the ith image is denoted by Fi. Check the equation on Page 3

Diffusion Features (DIFT) extraction

- To begin with, the authors focus on the images generated by the Stable Diffusion model. Why generated images first? Because this gives access to the complete internal state of the network throughout the entire backward process.
- Given each generated image, we extract the feature maps of its intermediate layers at a specific time step t during the backward process, which we then utilize to establish correspondences between two different generated images as described in the previous section.
- Now that the correspondence between generated images has been established, how do we repeat the same exercise for real images? The challenge lies in the fact that the real image itself does not belong to the training distribution of the U-Net (which was trained on noisy images), and we do not have access to the intermediate noisy images that would have been produced during the generation of this image.

- To address the above challenge, the authors came up with a simple yet effective solution. They first add noise of time step `t` to the real image using the forward process to move it to the xt distribution. Then they feed this noisy image to the network along with timestep `t` to extract the intermediate layer activations/feature maps named DIffusion Features (DIFT).
- The authors found that this approach yields equally good correspondence for visual images.
- One crucial question remains: What are the optimal timestep and the feature layer for finding different kinds of correspondence? The authors find that a larger t and an earlier network layer tend to yield more semantically-aware features, while a smaller t and a later layer focus more on low-level details. The optimal choice of t and layer depends on the specific correspondence task at hand, as different tasks may require varying trade-offs between semantic and low-level features. Hence they use a 2D grid search to determine these two hyper-parameters for each correspondence task.

Semantic Correspondence

- The task is to identify pixels that share similar semantic meanings across images.
- Percentage of correct key points (PCK) used as the evaluation metric. The predicted key point is considered to be correct if they lie within $\alpha \cdot \max(h, w)$ pixels from the ground-truth keypoint for $\alpha \in [0, 1]$, where h and w are the height and width of either the image (aimg) or the bounding box (abbox).
- The authors observe that features learned via diffusion are much more suitable for establishing semantic correspondence compared to features learned using contrastive approaches (DIFT(SD) vs. OpenCLIP, DIFT(ADM) vs. DINO). DIFT outperforms all the weakly-supervised baselines on all benchmarks by a large margin.
- Even for a randomly selected patch in an image, DIFT can successfully identify correct correspondence across various categories.

Geometric Correspondence

- When the time step `t` is small, DIFT focuses more on low-level details making it useful as a geometric feature descriptor.
- DIFT is evaluated for homography estimation using the HPatches benchmark with corner correctness metric.
- Though not trained using any explicit geometry supervision, DIFT is still on par with the methods that utilize explicit geometric supervision signals designed specifically for this task.

Temporal Correspondence

- The authors evaluate the learned representation on two challenging video tasks: (1) DAVIS-2017 video instance segmentation benchmark, and (2) Joints for the HMDB (JHMDB) benchmark, which involves tracking 15 human pose key points.
- The authors report region-based similarity and contour-based accuracy for DAVIS, and PCK for JHMDB.
- DIFT(adm) outperforms all the other self-supervised learning methods, on both benchmarks, even surpassing models specifically trained on video data by a significant margin. DIFT also yields the best results within the same pre-training dataset

Figure 9: DIFT can find correspondence on real images across instances, categories, and even domains, e.g., from a photo of statue of liberty to a logo.

Figure 10: DIFT can find correspondence on real images across instances, categories, and even domains, e.g., from a photo of an aeroplane to a sketch.

Figure 11: Semantic correspondence using various off-the-shelf features on SPair-71k. Circles indicates correct predictions while crosses for incorrect ones.

Figure 12: Given the image patch specified in the leftmost image (red dot), we use DIFT$_{sd}$ to query the top-5 nearest image patches from different categories in the SPair-71k test set. DIFT is still able to find correct correspondence for object parts with different overall appearance but sharing the same semantic meaning, e.g., the leg of a bird vs. the leg of a dog.

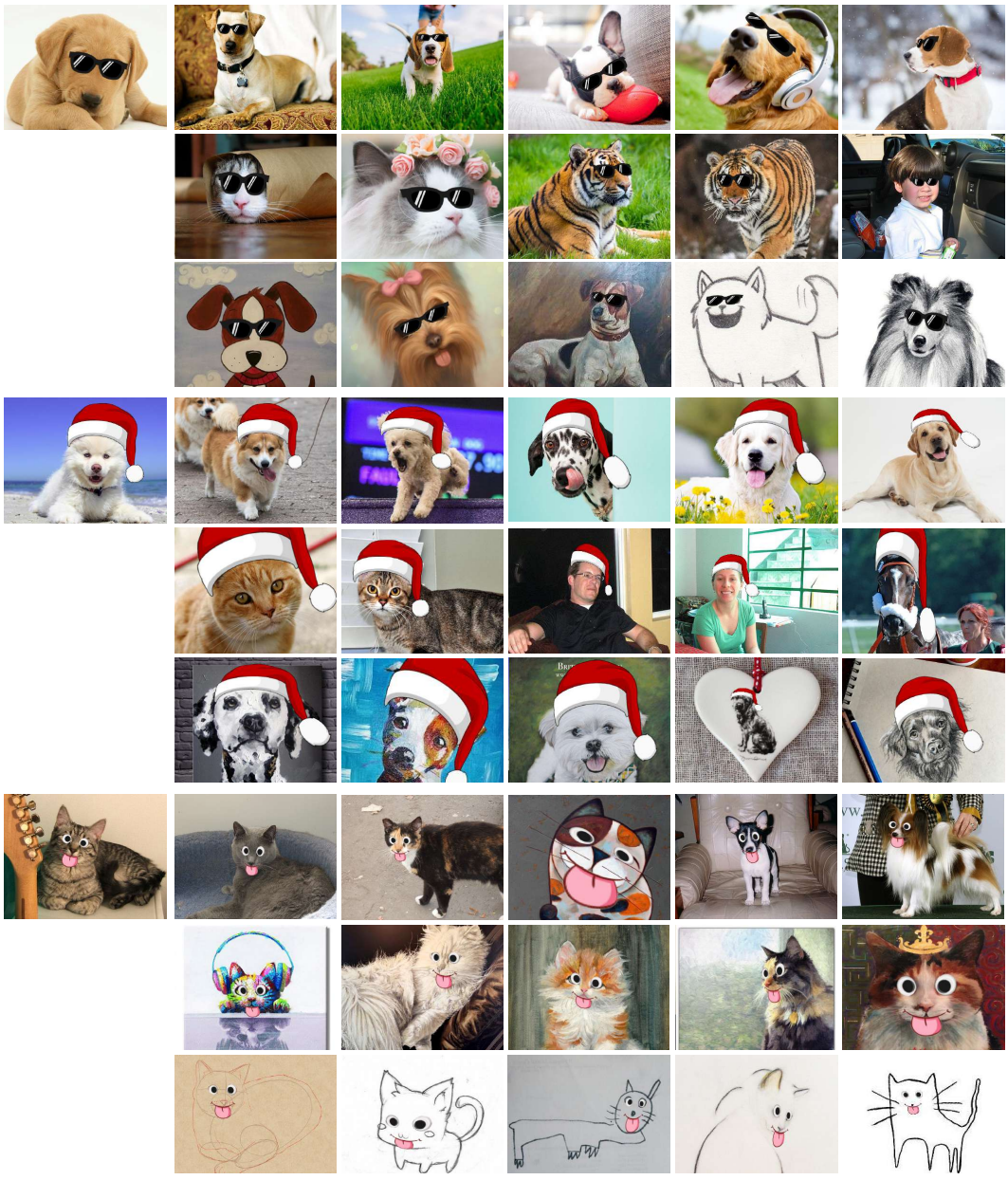


Figure 13: Edit propagation using DIFT$_{sd}$. Far left column: edited source images. Right columns: target images with the propagated edits. Note that despite the large domain gap in the last row, DIFT$_{sd}$ still manages to establish reliable correspondences for correct propagation.

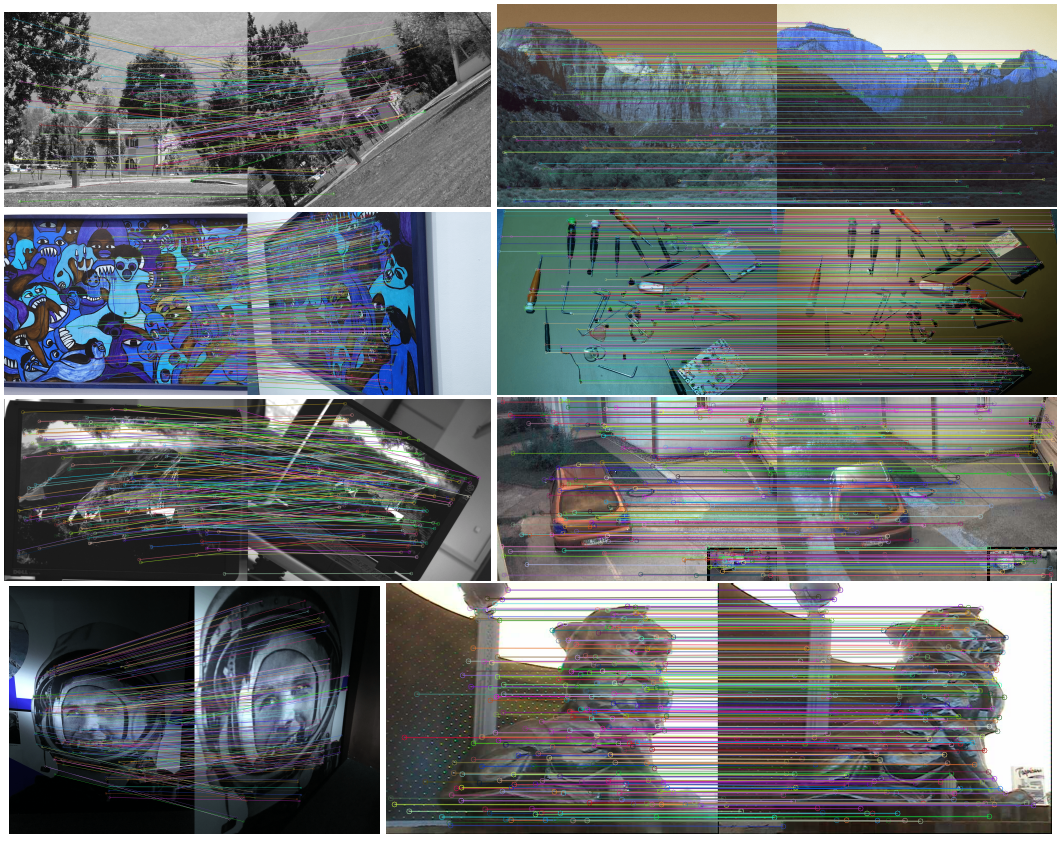


Figure 14: Sparse feature matching using DIFT$_{sd}$ on HPatches after removing outliers. Left are image pairs under viewpoint change, and right are ones under illumination change. Although never trained with geometric correspondence labels, it works well under both challenging changes.

Figure 15: Additional video label propagation results on DAVIS-2017. Colors indicate segmentation masks for different instances. Blue rectangles show the first frames. GT is short for "Ground-Truth".

References

[1] K. Aberman, J. Liao, M. Shi, D. Lischinski, B. Chen, and D. Cohen-Or. Neural best-buddies: Sparse cross-domain correspondence. *ACM Transactions on Graphics (TOG)*, 37(4):1–14, 2018. 6

[2] S. Agarwal, Y. Furukawa, N. Snavely, I. Simon, B. Curless, S. M. Seitz, and R. Szeliski. Building rome in a day. *Communications of the ACM*, 54(10):105–112, 2011. 2

[3] S. Amir, Y. Gandelsman, S. Bagon, and T. Dekel. Deep vit features as dense visual descriptors. *arXiv preprint arXiv:2112.05814*, 2(3):4, 2021. 2

[4] V. Balntas, K. Lenc, A. Vedaldi, and K. Mikolajczyk. Hpatches: A benchmark and evaluation of handcrafted and learned local descriptors. In *CVPR*, 2017. 9

[5] D. Baranchuk, I. Rubachev, A. Voynov, V. Khrulkov, and A. Babenko. Label-efficient semantic segmentation with diffusion models. *arXiv preprint arXiv:2112.03126*, 2021. 3, 10

[6] H. Bay, T. Tuytelaars, and L. Van Gool. Surf: Speeded up robust features. *Lecture notes in computer science*, 3951:404–417, 2006. 2

[7] A. Birhane, V. U. Prabhu, and E. Kahembwe. Multimodal datasets: misogyny, pornography, and malignant stereotypes. *arXiv preprint arXiv:2110.01963*, 2021. 11

[8] T. Brooks, A. Holynski, and A. A. Efros. Instructpix2pix: Learning to follow image editing instructions. *arXiv preprint arXiv:2211.09800*, 2022. 1

[9] K. Cao, M. Brbic, and J. Leskovec. Concept learners for few-shot learning. *arXiv preprint arXiv:2007.07375*, 2020. 2

[10] M. Caron, H. Touvron, I. Misra, H. Jégou, J. Mairal, P. Bojanowski, and A. Joulin. Emerging properties in self-supervised vision transformers. In *ICCV*, pages 9650–9660, 2021. 1, 2, 5, 6, 7, 9, 10

[11] J. Carreira and A. Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *CVPR*, pages 6299–6308, 2017. 10

[12] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR, 2020. 10

[13] X. Chen and K. He. Exploring simple siamese representation learning. In *CVPR*, pages 15750–15758, 2021. 10

[14] Y. Chen, M. Mancini, X. Zhu, and Z. Akata. Semi-supervised and unsupervised deep visual learning: A survey. *IEEE TPAMI*, 2022. 1

[15] S. Cho, S. Hong, S. Jeon, Y. Lee, K. Sohn, and S. Kim. Cats: Cost aggregation transformers for visual correspondence. *NeurIPS*, 34:9011–9023, 2021. 2, 6, 7

[16] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, pages 248–255. Ieee, 2009. 3, 5, 10

[17] D. DeTone, T. Malisiewicz, and A. Rabinovich. Superpoint: Self-supervised interest point detection and description. In *CVPRW*, pages 224–236, 2018. 9

[18] P. Dhariwal and A. Nichol. Diffusion models beat gans on image synthesis. *NeurIPS*, 34:8780–8794, 2021. 1, 3, 4

[19] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 5

[20] M. Dusmanu, I. Rocco, T. Pajdla, M. Pollefeys, J. Sivic, A. Torii, and T. Sattler. D2-net: A trainable cnn for joint detection and description of local features. *arXiv preprint arXiv:1905.03561*, 2019. 9

[21] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman. The pascal visual object classes (voc) challenge. *IJCV*, 88:303–338, 2010. 5

[22] D. F. Fouhey, W.-c. Kuo, A. A. Efros, and J. Malik. From lifestyle vlogs to everyday interactions. In *CVPR*, pages 4991–5000, 2018. 10

[23] S. Gao, C. Zhou, C. Ma, X. Wang, and J. Yuan. Aiatrack: Attention in attention for transformer visual tracking. In *ECCV*, pages 146–164. Springer, 2022. 1, 2

[24] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial networks. *Communications of the ACM*, 63(11):139–144, 2020. 2

[25] D. Gordon, K. Ehsani, D. Fox, and A. Farhadi. Watching the world go by: Representation learning from unlabeled videos. *arXiv preprint arXiv:2003.07990*, 2020. 10

[26] J.-B. Grill, F. Strub, F. Altché, C. Tallec, P. H. Richemond, E. Buchatskaya, C. Doersch, B. A. Pires, Z. D. Guo, M. G. Azar, et al. Bootstrap your own latent: A new approach to self-supervised learning. *arXiv preprint arXiv:2006.07733*, 2020. 10

[27] J. Gui, T. Chen, Q. Cao, Z. Sun, H. Luo, and D. Tao. A survey of self-supervised learning from multiple perspectives: Algorithms, theory, applications and future trends. *arXiv preprint arXiv:2301.05712*, 2023. 1

[28] K. Gupta, V. Jampani, C. Esteves, A. Shrivastava, A. Makadia, N. Snavely, and A. Kar. Asic: Aligning sparse in-the-wild image collections. *arXiv preprint arXiv:2303.16201*, 2023. 6

[29] B. Ham, M. Cho, C. Schmid, and J. Ponce. Proposal flow. In *CVPR*, 2016. 5

[30] M. Hamilton, Z. Zhang, B. Hariharan, N. Snavely, and W. T. Freeman. Unsupervised semantic segmentation by distilling feature correspondences. In *ICLR*, 2022. 1, 2

[31] K. Han, R. S. Rezende, B. Ham, K.-Y. K. Wong, M. Cho, C. Schmid, and J. Ponce. Scnet: Learning semantic correspondence. In *ICCV*, 2017. 7

[32] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick. Momentum contrast for unsupervised visual representation learning. In *CVPR*, pages 9729–9738, 2020. 10

[33] A. Hertz, R. Mokady, J. Tenenbaum, K. Aberman, Y. Pritch, and D. Cohen-Or. Prompt-to-prompt image editing with cross attention control. *arXiv preprint arXiv:2208.01626*, 2022. 3

[34] J. Ho, A. Jain, and P. Abbeel. Denoising diffusion probabilistic models. *NeurIPS*, 33:6840–6851, 2020. 1, 3

[35] J. Ho and T. Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022. 3

[36] Y. Hu, R. Wang, K. Zhang, and Y. Gao. Semantic-aware fine-grained correspondence. In *ECCV*, pages 97–115. Springer, 2022. 10

[37] S. Huang, L. Yang, B. He, S. Zhang, X. He, and A. Shrivastava. Learning semantic correspondence with sparse annotations. *arXiv preprint arXiv:2208.06974*, 2022. 2, 6, 7

[38] G. Ilharco, M. Wortsman, R. Wightman, C. Gordon, N. Carlini, R. Taori, A. Dave, V. Shankar, H. Namkoong, J. Miller, H. Hajishirzi, A. Farhadi, and L. Schmidt. Openclip, July 2021. If you use this software, please cite it as below. 2, 5, 6, 7, 9, 10

[39] A. Jabri, A. Owens, and A. A. Efros. Space-time correspondence as a contrastive random walk. *NeurIPS*, 2020. 2, 10

[40] S. Jeon, S. Kim, D. Min, and K. Sohn. Pyramidal semantic correspondence networks. *IEEE TPAMI*, 44(12):9102–9118, 2021. 6

[41] H. Jhuang, J. Gall, S. Zuffi, C. Schmid, and M. J. Black. Towards understanding action recognition. In *ICCV*, pages 3192–3199, 2013. 10

[42] T. Karras, M. Aittala, T. Aila, and S. Laine. Elucidating the design space of diffusion-based generative models. *arXiv preprint arXiv:2206.00364*, 2022. 1, 3

[43] S. Kim, J. Min, and M. Cho. Transformatcher: Match-to-match attention for semantic correspondence. In *CVPR*, pages 8697–8707, 2022. 2, 6, 7

[44] Z. Lai, E. Lu, and W. Xie. Mast: A memory-augmented self-supervised tracker. In *CVPR*, pages 6479–6488, 2020. 10

[45] Z. Lai and W. Xie. Self-supervised learning for video correspondence flow. *arXiv preprint arXiv:1905.00875*, 2019. 10

[46] J. Lee, D. Kim, J. Ponce, and B. Ham. Sfnet: Learning object-aware semantic correspondence. In *CVPR*, pages 2278–2287, 2019. 6

[47] J. Y. Lee, J. DeGol, V. Fragoso, and S. N. Sinha. Patchmatch-based neighborhood consensus for semantic correspondence. In *CVPR*, pages 13153–13163, 2021. 2

[48] W. Li, O. Hosseini Jafari, and C. Rother. Deep object co-segmentation. In *ACCV*, pages 638–653. Springer, 2019. 2

[49] X. Li, D.-P. Fan, F. Yang, A. Luo, H. Cheng, and Z. Liu. Probabilistic model distillation for semantic correspondence. In *CVPR*, pages 7505–7514, 2021. 6, 7

[50] X. Li, S. Liu, S. De Mello, X. Wang, J. Kautz, and M.-H. Yang. Joint-task self-supervised learning for temporal correspondence. *NeurIPS*, 32, 2019. 10

[51] S. Liu, L. Zhang, X. Yang, H. Su, and J. Zhu. Unsupervised part segmentation through disentangling appearance and shape. In *CVPR*, pages 8355–8364, 2021. 2

[52] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 60:91–110, 2004. 2, 9

[53] Z. Luo, T. Shen, L. Zhou, J. Zhang, Y. Yao, S. Li, T. Fang, and L. Quan. Contextdesc: Local descriptor augmentation with cross-modality context. In *CVPR*, pages 2527–2536, 2019. 9

[54] C. Meng, Y. He, Y. Song, J. Song, J. Wu, J.-Y. Zhu, and S. Ermon. Sdedit: Guided image synthesis and editing with stochastic differential equations. In *ICLR*, 2021. 1

[55] J. Min and M. Cho. Convolutional hough matching networks. In *CVPR*, 2021. 7

[56] J. Min, J. Lee, J. Ponce, and M. Cho. Spair-71k: A large-scale benchmark for semantic correspondence. *arXiv prepreint arXiv:1908.10543*, 2019. 5

[57] J. Min, J. Lee, J. Ponce, and M. Cho. Learning to compose hypercolumns for visual correspondence. In *ECCV*, 2020. 7

[58] J. Mu, S. De Mello, Z. Yu, N. Vasconcelos, X. Wang, J. Kautz, and S. Liu. Coordgan: Self-supervised dense correspondences emerge from gans. In *CVPR*, pages 10011–10020, 2022. 2

[59] D. Ofri-Amar, M. Geyer, Y. Kasten, and T. Dekel. Neural congealing: Aligning images to a joint semantic atlas. In *CVPR*, 2023. 1, 5, 6, 7

[60] Y. Ono, E. Trulls, P. Fua, and K. M. Yi. Lf-net: Learning local features from images. *NeurIPS*, 31, 2018. 9

[61] O. Özyeşil, V. Voroninski, R. Basri, and A. Singer. A survey of structure from motion*. *Acta Numerica*, 26:305–364, 2017. 2

[62] G. Parmar, K. K. Singh, R. Zhang, Y. Li, J. Lu, and J.-Y. Zhu. Zero-shot image-to-image translation. *arXiv preprint arXiv:2302.03027*, 2023. 1, 4

[63] W. Peebles, J.-Y. Zhu, R. Zhang, A. Torralba, A. Efros, and E. Shechtman. Gan-supervised dense visual alignment. In *CVPR*, 2022. 2, 6, 7

[64] Y. Peng, X. He, and J. Zhao. Object-part attention model for fine-grained image classification. *IEEE TIP*, 27(3):1487–1500, 2017. 2

[65] F. Perazzi, J. Pont-Tuset, B. McWilliams, L. Van Gool, M. Gross, and A. Sorkine-Hornung. A benchmark dataset and evaluation methodology for video object segmentation. In *CVPR*, pages 724–732, 2016. 10

[66] J. Pont-Tuset, F. Perazzi, S. Caelles, P. Arbeláez, A. Sorkine-Hornung, and L. Van Gool. The 2017 davis challenge on video object segmentation. *arXiv preprint arXiv:1704.00675*, 2017. 10

[67] J. Revaud, P. Weinzaepfel, C. De Souza, N. Pion, G. Csurka, Y. Cabon, and M. Humenberger. R2d2: repeatable and reliable detector and descriptor. *arXiv preprint arXiv:1906.06195*, 2019. 9

[68] I. Rocco, R. Arandjelovic, and J. Sivic. Convolutional neural network architecture for geometric matching. In *CVPR*, pages 6148–6157, 2017. 6

[69] I. Rocco, R. Arandjelović, and J. Sivic. End-to-end weakly-supervised semantic alignment. In *CVPR*, pages 6917–6925, 2018. 6

[70] I. Rocco, M. Cimpoi, R. Arandjelović, A. Torii, T. Pajdla, and J. Sivic. Neighbourhood consensus networks. *NeurIPS*, 31, 2018. 6

[71] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer. High-resolution image synthesis with latent diffusion models, 2021. 1, 3, 4

[72] O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18*, pages 234–241. Springer, 2015. 1, 3

[73] J. C. Rubio, J. Serrat, A. López, and N. Paragios. Unsupervised co-segmentation through region matching. In *CVPR*, pages 749–756. IEEE, 2012. 2

[74] J. L. Schönberger and J.-M. Frahm. Structure-from-motion revisited. In *CVPR*, 2016. 1, 2, 9

[75] J. L. Schönberger, E. Zheng, M. Pollefeys, and J.-M. Frahm. Pixelwise view selection for unstructured multi-view stereo. In *ECCV*, 2016. 2

[76] C. Schuhmann, R. Beaumont, R. Vencu, C. Gordon, R. Wightman, M. Cherti, T. Coombes, A. Katta, C. Mullis, M. Wortsman, et al. Laion-5b: An open large-scale dataset for training next generation image-text models. *arXiv preprint arXiv:2210.08402*, 2022. 3, 4, 10, 11

[77] P. H. Seo, J. Lee, D. Jung, B. Han, and M. Cho. Attentive semantic alignment with offset-aware correlation kernels. In *ECCV*, pages 349–364, 2018. 6

[78] J. Song, C. Meng, and S. Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020. 10

[79] Y. Song and S. Ermon. Generative modeling by estimating gradients of the data distribution. *NeurIPS*, 32, 2019. 3

[80] Y. Song, J. Sohl-Dickstein, D. P. Kingma, A. Kumar, S. Ermon, and B. Poole. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020. 1, 3

[81] L. Tang, D. Wertheimer, and B. Hariharan. Revisiting pose-normalization for fine-grained few-shot recognition. In *CVPR*, pages 14352–14361, 2020. 2

[82] N. Tumanyan, O. Bar-Tal, S. Bagon, and T. Dekel. Splicing vit features for semantic appearance transfer. In *CVPR*, pages 10748–10757, 2022. 1, 2

[83] N. Tumanyan, M. Geyer, S. Bagon, and T. Dekel. Plug-and-play diffusion features for text-driven image-to-image translation. *arXiv preprint arXiv:2211.12572*, 2022. 1, 3, 4

[84] J. Valmadre, L. Bertinetto, J. F. Henriques, R. Tao, A. Vedaldi, A. W. Smeulders, P. H. Torr, and E. Gavves. Long-term tracking in the wild: A benchmark. In *ECCV*, pages 670–685, 2018. 10

[85] C. Vondrick, A. Shrivastava, A. Fathi, S. Guadarrama, and K. Murphy. Tracking emerges by colorizing videos. In *ECCV*, pages 391–408, 2018. 10

[86] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie. The caltech-ucsd birds-200-2011 dataset. 2011. 5

[87] Q. Wang, X. Zhou, B. Hariharan, and N. Snavely. Learning feature descriptors using camera pose supervision. In *ECCV*, pages 757–774. Springer, 2020. 2, 9

[88] X. Wang, A. Jabri, and A. A. Efros. Learning correspondence from the cycle-consistency of time. In *CVPR*, pages 2566–2576, 2019. 1, 2, 10

[89] Z. Wu, Y. Xiong, S. Yu, and D. Lin. Unsupervised feature learning via non-parametric instance-level discrimination. *arXiv preprint arXiv:1805.01978*, 2018. 10

[90] J. Xu, S. Liu, A. Vahdat, W. Byeon, X. Wang, and S. De Mello. Open-vocabulary panoptic segmentation with text-to-image diffusion models. *arXiv preprint arXiv:2303.04803*, 2023. 3, 10

[91] J. Xu and X. Wang. Rethinking self-supervised correspondence learning: A video frame-level similarity perspective. *arXiv preprint arXiv:2103.17263*, 2021. 10

[92] N. Xu, L. Yang, Y. Fan, D. Yue, Y. Liang, J. Yang, and T. Huang. Youtube-vos: A large-scale video object segmentation benchmark. *arXiv preprint arXiv:1809.03327*, 2018. 10

[93] B. Yan, Y. Jiang, P. Sun, D. Wang, Z. Yuan, P. Luo, and H. Lu. Towards grand unification of object tracking. In *ECCV*, pages 733–751. Springer, 2022. 2

[94] Y. Zhang, H. Ling, J. Gao, K. Yin, J.-F. Lafleche, A. Barriuso, A. Torralba, and S. Fidler. Datasetgan: Efficient labeled data factory with minimal human effort. In *CVPR*, 2021. 2

[95] D. Zhao, Z. Song, Z. Ji, G. Zhao, W. Ge, and Y. Yu. Multi-scale matching networks for semantic correspondence. In *ICCV*, pages 3354–3364, 2021. 6

[96] W. Zhao, Y. Rao, Z. Liu, B. Liu, J. Zhou, and J. Lu. Unleashing text-to-image diffusion models for visual perception. *arXiv preprint arXiv:2303.02153*, 2023. 3, 10