CleanDIFT: Diffusion Features without Noise

Nick Stracke* Stefan Andreas Baumann* Kolja Bauer* Frank Fundel Björn Ommer
CompVis @ LMU Munich, MCML
{nick.stracke,b.ommer}@lmu.de
compvis.github.io/cleandift

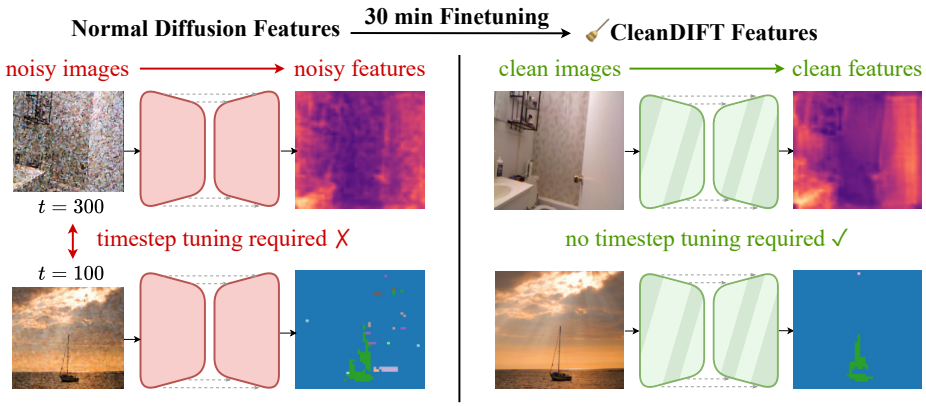

Figure 1. Our proposed CleanDIFT feature extraction method yields noise-free, timestep-independent, general-purpose features that significantly outperform standard diffusion features. CleanDIFT operates on clean images, while extracting diffusion features with existing approaches requires adding noise to an image before passing it through the model. Adding noise reduces the information present in the image and requires tuning a timestep per downstream task.

Abstract

Internal features from large-scale pre-trained diffusion models have recently been established as powerful semantic descriptors for a wide range of downstream tasks. Works that use these features generally need to add noise to images before passing them through the model to obtain the semantic features, as the models do not offer the most useful features when given images with little to no noise. We show that this noise has a critical impact on the usefulness of these features that cannot be remedied by ensembling with different random noises. We address this issue by introducing a lightweight, unsupervised fine-tuning method that enables diffusion backbones to provide high-quality, noise-free semantic features. We show that these features readily outperform previous diffusion features by a wide margin in a wide variety of extraction setups and downstream tasks, offering better performance than even ensemble-based methods at a fraction of the cost.

1. Introduction

Learning meaningful visual representations that capture a vast amount of world knowledge remains a key problem in the field of computer vision. Diffusion models can be trained at scale in a self-supervised manner and have rapidly advanced the state of the art in image [12, 41, 51] and video generation [13, 25, 58], making them a good candidate to learn visual representations. Many early works have already achieved impressive results using internal features from large-scale pretrained diffusion models for a wide variety of tasks, such as semantic correspondence detection [62, 66, 67], semantic segmentation [2, 39, 63], panoptic segmentation [64], object detection [8], and classification [31]. However, the optimal approach to extract this world knowledge from a diffusion model remains uncertain.

To understand why that is the case, we take a look at how diffusion models are trained: a varying amount of noise is added to a clean input image (forward process) and the model is tasked to remove the noise from the image (backward process). The amount of added noise is dependent on the diffusion *timestep*. As a result, the model learns to operate on noisy images and also becomes *dependent* on the noise timestep as different noise levels require the model to

*Equal Contribution

perform different tasks [1, 4]. Since noisy images inherently contain less information than clean images (cf. Figure 2), we hypothesize that this harms the internal feature representation of diffusion models [9] and, thus, the extractable world knowledge. Furthermore, the timestep acts as a hyperparameter that influences the internal feature representation and needs to be picked independently for every down-stream application (cf. Figure 14).

We propose a novel feature extraction method that (1) eliminates the need to destroy information by adding noise to the input; and (2) produces timestep-independent generic diffusion features useful for a wide range of down-stream tasks, alleviating the need to tune a noising timestep per down-stream task. We show how to adapt an off-the-shelf large-scale pre-trained diffusion backbone to provide these features at minimal cost (approximately 30 minutes of fine-tuning on a single A100 GPU) and demonstrate improved performance across a wide range of downstream tasks.

We achieve this by viewing a diffusion model as a family of T feature extractors that operate on images with different noise levels and provide features with different characteristics. We consolidate all T feature extraction functions in our feature extractor by aligning their internal representations. Specifically, we initialize our feature extractor as a trainable copy of the diffusion model; fine-tune it with clean images and no timestep input; and align its features with all T time-dependent feature extractors of the diffusion model.

We evaluate our improved features across a wide variety of downstream tasks, such as semantic correspondence matching, monocular depth estimation, semantic segmentation, and classification, and find that they consistently improve upon approaches based on standard diffusion features. These improvements are most evident for dense visual tasks such as semantic correspondence matching, where our features show substantial performance gains across a wide variety of setups [62, 66, 67] and set a new state-of-the-art for unsupervised semantic correspondence matching. Additionally, our proposed method eliminates the need for noise or timestep ensembling [62], offering substantial speed gains (e.g., $8\times$ over DIFT [62]), on top of improved quality. Our method is generic and integrates easily with established methods, such as fusing diffusion and DINOv2 features [66, 67].

Our main contributions are as follows:

1. We propose CleanDIFT, a finetuning approach for diffusion models that enables them to operate on clean images and makes the inherent world knowledge of these models more accessible.
2. We show how to consolidate information from all diffusion timesteps into a single feature prediction, removing the need for task-specific timestep tuning.
3. We demonstrate significant performance gains of our diffusion feature extraction technique across a wide range of down-stream tasks, notably surpassing the current state of the art in zero-shot unsupervised semantic correspondence detection. We further demonstrate the generality of our enhanced features by showing that these performance gains transfer to advanced methods that fuse diffusion features or operate in a supervised setting.
4. Our proposed approach is significantly more efficient than previous methods that tried to address this problem by noise ensembling or supervised training.

(a) Reconstruction without Noise

(b) Diffusion Model Reconstruction with added noise ($t = 261$ [62])

Figure 2. **Deterioration of Diffusion Features**. As current methods *need* to pass noisy images to the model to obtain useful features, they significantly reduce the information available. We alleviate this problem by obtaining useful features without noise, improving the performance of downstream tasks.

2. Related Work

Self-Supervised Representation Learning Features from large, pre-trained foundation models have been shown to yield competitive performance to supervised models for a variety of downstream tasks, both in zero-shot and fine-tuning settings [20, 43, 48]. These foundation models are trained on different pre-text tasks like inpainting [20], predicting transformations [19], patch reordering [38, 42], and discriminative tasks [6, 43]. DINOv2 [43] uses a discriminative objective combined with self-distillation to learn general-purpose visual features that have proven useful for a variety of downstream tasks [10]. CLIP [48] learns such features by employing a contrastive objective on text-image pairs. Masked Autoencoders [20] (MAEs) are trained to reconstruct masked out patches of the input, also resulting in general-purpose visual features.

Diffusion Models as Self-Supervised Learners Diffusion models [24, 60, 61] are generative models that have defined the state-of-the-art in image generation [12, 14, 41, 46, 51, 53], video generation [13, 47], and audio generation [15, 30] in recent years. Their primary purpose is to generate high-quality samples (images, videos, etc.). However, generation can also be interpreted as a pretext task for

learning expressive features, since the model has to build up comprehensive world knowledge in order to generate plausible samples [9, 17, 27, 32, 62]. Features from diffusion models (typically referred to as *diffusion features*) are obtained by passing a noised image through the diffusion model and extracting intermediate feature representations. They have been shown to be useful for a variety of tasks such as finding semantic correspondences [21, 35, 62], semantic and panoptic segmentation [2, 39, 63, 64], classification [31], and object detection [8].

For semantic correspondence matching, features are leveraged to identify semantically matching regions across images. Existing approaches utilize diffusion features either in a zero-shot setting [21, 62] or fine-tune them for the semantic correspondence task [33, 35, 67]. Some zero-shot approaches do not fine-tune on semantic correspondence but still require tuning a prompt to activate attention maps at the query location of the correspondence [21]. In contrast, our approach aims to provide universal features usable for various down-stream tasks in a true zero-shot manner.

Further, diffusion features have been shown to complement features from other self-supervised learning methods such as DINOv2 [18, 43, 66]. DINOv2 features provide sparse but accurate semantic information, while Diffusion features provide dense spatial information albeit with sometimes inaccurate semantics. State-of-the-art approaches for semantic correspondence detection exploit this and fuse features [66, 67]. Compared to DINOv2 features, diffusion features yield smoother, spatially more coherent correspondences [66].

Distillation for Diffusion Models Knowledge Distillation [22] is a technique used to distill knowledge from a teacher model into a student model. In the context of diffusion models, this is typically applied either to reduce the required denoising steps [54] or to distill a classifier-free guided [23] model into one without CFG [36]. While our approach is inspired by distillation, we consolidate features from T different models, with T being the number of discrete diffusion timesteps, because the features are different at every timestep.

3. Method

3.1. Preliminaries

Diffusion Models Diffusion models are trained to predict a clean image $\mathbf{x}_0$ given a noisy image $\mathbf{x}_t$, either explicitly or implicitly. The noisy image $\mathbf{x}_t$ is a weighted sum with random Gaussian noise $\mathbf{x}_t = \sqrt{\alpha_t}\mathbf{x}_0 + \sqrt{1-\alpha_t}\boldsymbol{\epsilon}$ with timestep-dependent coefficients α_t and noise $\boldsymbol{\epsilon} \sim \mathcal{N}(0, \mathbf{I})$. $t \in [0, T]$ denotes the time step of the diffusion process, with $t = 0$ corresponding to the clean image and $t = T$ corresponding to pure noise.

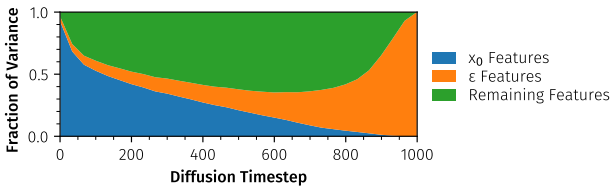

Figure 3. Fraction of variance of diffusion features explained by 1) encoding the clean image at $t = 0$ (no additive noise), and 2) encoding just the added noise $\boldsymbol{\epsilon}$ at $t = 999$. Even at relatively low timesteps such as $t = 261$ as used by DIFT [62], a substantial part of the features directly depends only on the added noise.

Intuitively, the model faces different objectives for different noise levels [1, 4]: for very high noise, there is little information in the input, and the model is first generating the coarse structure of the image [50]. At lower noise levels, more high- and medium-frequency information is available and the task shifts to generating finer details and intricate structures. This multi-objective nature intuitively explains why previous methods found diffusion features extracted from different timesteps to provide information with differing semantics.

Diffusion Feature Extraction Typically, diffusion feature extraction happens after first adding noise to an image and passing the resulting $\mathbf{x}_t$ to a U-Net [52] denoiser. Features are then extracted at multiple hand-picked locations of the U-Net decoder [2, 39, 62, 66, 67]. Different levels of noise added to the input image result in features beneficial for different downstream applications. Typical diffusion timesteps are $t = 261$ [62], $t = 100$ [66] or $t = 50$ [67]. By adding noise to the input image, these methods bottleneck the perceptual information the model can extract. To illustrate this, we show Stable Diffusion 2.1's reconstruction of an image noised at $t = 261$ [62] in Figure 2.

Diffusion Features Encode Noise We hypothesize that diffusion features extracted from noisy images $\mathbf{x}_t$ encode the noise $\boldsymbol{\epsilon}$ in addition to information from the image. We investigate this hypothesis in a very simple setting by examining how well features $\text{feat}(\boldsymbol{\epsilon}; T)$ extracted from only the noise $\boldsymbol{\epsilon}$ approximate the features $\text{feat}(\mathbf{x}_t = \sqrt{\alpha_t}\mathbf{x}_0 + \sqrt{1-\alpha_t}\boldsymbol{\epsilon}; t)$. Using least squares, we fit a single scalar approximation coefficient to obtain the optimal reconstruction. We then quantify how much of the variance of the overall features is explained by this approximation (cf. Figure 3). Even at relatively low timesteps, such as $t = 261$ used by DIFT [62], encoding pure noise explains a substantial fraction of the features' variance. Current diffusion feature methods extract this information jointly with the image information. Our proposed method addresses this issue by eliminating the noise from the feature extraction process.

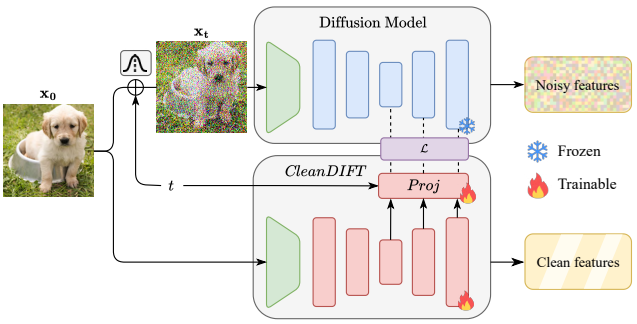

Figure 4. Our training setup. We train our model to predict features from a clean input image, while the frozen diffusion model is fed the noisy image. The projection heads project our model's features onto the noisy diffusion model features, given the noising timestep t. For downstream tasks, we discard the projection heads and directly use our model's internal representations as features.

We further analyze the residual and similarly decompose it via the features predicted for the clean image $\text{feat}(x_t = x_0; t = 0)$. We find that they do not fully explain the remainder of the features either. Instead, a substantial part of the feature variance at medium noise timesteps is timestep dependent and cannot be attributed to components present at $t = 0$ or $t = T$. This matches observations by previous works [62] that found diffusion features at higher timesteps offer better semantics, despite the added noise at the image input.

3.2. ✓CleanDIFT: Noise-Free Diffusion Features

We present CleanDIFT, our method to address the problem of noisy and time-dependent diffusion features. *CleanDIFT* extracts clean DIffusion FeaTures from a pretrained diffusion backbone through a lightweight fine-tuning process. An overview of our setup is shown in Figure 4.

Extraction Setup We train our feature extraction model to match the diffusion model's internal representations. We initialize the feature extraction model as a trainable copy of the diffusion model. Crucially, the feature extraction model is given the clean input image, while the diffusion model receives the noisy image and the corresponding timestep as input. Our goal is to obtain a single, noise-free feature map from the feature extraction model that consolidates the information of the diffusion model's timestep-dependent internal representations into a single one. To align our model's representations with the timestep-dependent diffusion model features during training, we introduce pointwise timestep-conditioned feature projection heads. The feature maps predicted by these projection heads are then aligned to the diffusion model's features. For feature extraction at inference time, we usually discard the projection heads and directly use the feature extraction model's internal representations. However, the projection heads can also be used to efficiently obtain feature maps for specific timesteps by reusing the feature extraction model's internal representations and passing them through the projection heads for different t values.

Training Objective We regard the diffusion model as a family of feature extraction functions $\text{feat}(\cdot, \boldsymbol{\epsilon}, t)$ for timestep $t \in [1, 999]$ and noise $\boldsymbol{\epsilon} \sim \mathcal{N}(0, \mathbf{I})$. Each of these functions maps an image $\mathbf{x}$ to a feature vector $\text{feat}(\mathbf{x}, \boldsymbol{\epsilon}, t)$. We aim to consolidate the information provided by all feature extraction functions into a single joint function $\text{feat}_c(\cdot)$ with the same dimensionality:

$$\underline{\text{Stable Diffusion}} \qquad \underline{\checkmark\text{CleanDIFT}}$$

$$\left.\begin{array}{l}\text{feat}(\mathbf{x}, \boldsymbol{\epsilon}, t = 1) \\ \text{feat}(\mathbf{x}, \boldsymbol{\epsilon}, t = 2) \\ \vdots \\ \text{feat}(\mathbf{x}, \boldsymbol{\epsilon}, t = 999)\end{array}\right\} \xrightarrow[\text{information}]{\text{consolidate}} \text{feat}_c(\mathbf{x})$$

To align our model's features with the diffusion model's features, we maximize the similarity between the diffusion model's features and the projected features of our feature extraction model:

$$\text{feat}_c(\mathbf{x}) \rightarrow \begin{cases} \text{proj}(\cdot, t = 1) \longrightarrow \mathcal{L} \longleftarrow \text{feat}(\mathbf{x}, \boldsymbol{\epsilon}, t = 1) \\ \text{proj}(\cdot, t = 2) \longrightarrow \mathcal{L} \longleftarrow \text{feat}(\mathbf{x}, \boldsymbol{\epsilon}, t = 2) \\ \qquad\qquad \vdots \\ \text{proj}(\cdot, t = 999) \longrightarrow \mathcal{L} \longleftarrow \text{feat}(\mathbf{x}, \boldsymbol{\epsilon}, t = 999) \end{cases}$$

Specifically, we minimize the negative cosine similarity between the diffusion model's features and our model's features extracted at stages $k = \{1, ..., K\}$ in the network. Given a clean image $\mathbf{x}_0$, the feature extraction model's output for feature map k is denoted as $\text{feat}_c^{(k)}(\mathbf{x}_0)$. Our CleanDIFT feature map is then adapted by the learned projection heads $\text{proj}^{(k)}(\text{feat}_c^{(k)}(\mathbf{x}_0), t)$, where $\text{proj}^{(k)}(\cdot, \cdot)$ is the projection head for feature map k. The diffusion model receives the noisy image $\mathbf{x}_t$ corresponding to the same $\mathbf{x}_0$ and timestep t. The projection head then learns a timestep-dependent alignment from CleanDIFT features to the diffusion model's features $\text{feat}^{(k)}(\mathbf{x}_t; t)$. Putting it all together, our loss function is defined as:

$$\mathcal{L} = -\sum_{k=1}^{K} \text{sim}(\text{proj}^{(k)}(\text{feat}_c^{(k)}(\mathbf{x}_0); t), \text{feat}^{(k)}(\mathbf{x}_t; t)). \quad (1)$$

For each training image $\mathbf{x}_0$, we sample I different noising timesteps t_i in a stratified manner, with each timestep $t_i \sim \mathcal{U}(\frac{i}{I}T, \frac{i+1}{I}T)$, where T is the maximum timestep. By sampling multiple timesteps per image we incentivize the feature extraction model to match the diffusion model's features across the entire noise spectrum.

4. Experiments

We test our hypothesis that the proposed extraction setup enables us to leverage more of the world knowledge inherent in diffusion models compared to existing diffusion feature extraction methods while being task-agnostic and timestep-independent. To that end, we evaluate our features on a wide range of downstream tasks: unsupervised zero-shot semantic correspondence, monocular depth estimation, semantic segmentation, and classification. We compare our features against standard diffusion features, methods that combine diffusion features with additional features, and non-diffusion-based approaches.

4.1. Experimental Setup

Implementation Details Following previous works [62, 66, 67], we evaluate our method on a Stable Diffusion (SD) backbone [51]. We apply our method to SD 1.5 and SD 2.1 to enable fair comparisons with existing methods that use either. We fully fine-tune our feature extraction model on image-caption pairs for only 400 steps, taking 30 minutes on a single A100 GPU, which was sufficient for our strong performance and further training did not yield any significant gains. We extract features after the U-Net's middle block and after each of the U-Net's decoder blocks, except the two final blocks. A detailed visualization of where we extract features is provided in Figure 11. This yields a total of $K = 11$ feature maps that we align between the diffusion model and the feature extraction model. Our point-wise feature projection heads consist of three stacked Feed Forward Networks (FFNs) that are zero-initialized such that initially they act as identity mappings due to their residual connections. Since every aligned feature map has its own projection head, this results in 45M additional trainable parameters for SD 2.1. We study the effect of different projection head architectures in Sec. E. We train using Adam [29] with a batch size of 8 and a learning rate of 2e-6 with a linear warmup. For stratified timestep sampling, we utilize $I = 3$ stratification bins across all our experiments, i.e. three different noise levels per training image.

Datasets We fine-tune our feature extraction model on a random subset of COYO-700M [5], which is similar to the LAION [56], the dataset that Stable Diffusion 1.5 and 2.1 were trained on originally. That way, we ensure that all performance improvements originate from the feature extraction model consolidating the diffusion model's internal feature representations over time, not from choosing a different dataset that matches the test dataset distribution more closely. The subset selects images with a minimum size of 512^2. We crop and resize them to match the corresponding input resolution of the underlying diffusion model. We analyze the effect of using different datasets in Sec. G.

Figure 5. Semantic correspondence results using DIFT [62] features with the standard SD 2.1 ($t = 261$) and our CleanDIFT features. Our clean features show significantly less incorrect matches than the base diffusion model.

4.2. Unsupervised Semantic Correspondence

As many previous diffusion feature methods focus on (unsupervised) semantic correspondence matching [35, 62, 66, 67], we perform an extensive evaluation of our method on this task. Following previous works on semantic correspondence matching [62, 66, 67], we measure our performance in Percentage of Correct Keypoints (PCK). We average PCK directly across all keypoints, not over images. We use $\alpha = 0.1$ as a threshold and report both PCK values with error margins relative to the image size and to the bounding box size, denoted as PCK_{img} and PCK_{bbox} respectively. We evaluate the performance on the test split of the SPair-71k dataset, which consists of approximately 12k image pairs from 18 categories. Some existing works [62, 66, 67] evaluate on additional datasets but find SPair-71k to be the most challenging and therefore the most informative benchmark. For the text prompt we use "A photo of a `category`.", with `category` being the corresponding category of the SPair image. We experiment with distilling the text conditioning in Sec. B.

Results We first compare our extracted features to DIFT [62], an approach that detects semantic correspondences using standard diffusion features. Substituting these with our CleanDIFT features yields a performance increase of 1.79 absolute percentage points for PCK_{img} and 1.86 percentage points for PCK_{bbox}. Notably, DIFT averages the extracted feature maps across 8 different noise samples. Without this averaging over noise samples, our performance gain is even larger (2.81 PCK@α_{img} gain). This indicates that our feature extraction model learns more than a mere averaging over the noise in the diffusion model's feature maps (see Figure 5 and Sec. C for examples). We present an extended version of the time-step dependent performance analysis conducted by [62] in Figure 6: We evaluate the

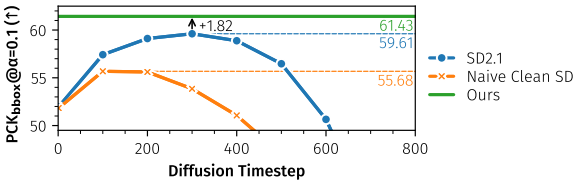

Figure 6. Following [62], we evaluate semantic correspondence matching accuracy for different noise levels. Our feature extractor outperforms the standard noisy diffusion features across all timesteps t. We additionally demonstrate that simply providing the diffusion model with a clean image and a non-zero timestep does not result in improved performance.

diffusion model's performance for different timesteps t in two settings. In the first setting, we provide the diffusion model with a noisy input image $\mathbf{x}_t$ as usual. In the second setting, we demonstrate that feeding the clean image along with a non-zero timestep is not a viable solution to obtain meaningful features: We provide the diffusion model with the clean input image $\mathbf{x}_0$ for all timesteps t. We observe that the model's performance for the clean input image degrades faster and has a lower peak than for the noisy input. This is to be expected, as the diffusion model was trained on noisy images, not clean images. Importantly, our CleanDIFT features are timestep-independent and consistently outperform standard diffusion features, even when the latter are optimized for the best-performing timestep. We further observe that this advantage generalizes effectively to other backbones, such as DiTs [44] (see Tab. 7).

A Tale of Two Features [66] extends the approach of DIFT by combining diffusion features with DINOv2 [43] features. Again, we replace the standard diffusion features with our CleanDIFT features and observe that the performance gain transfers when combining our features with DINOv2 features. Telling Left from Right [67] further improves upon the results of A Tale of Two Features by introducing a test-time adaptive pose alignment strategy. We observe that the performance gain transfers to this setting as well. To the best of our knowledge, Telling Left from Right combined with our CleanDIFT features sets a new state-of-the-art in unsupervised zero-shot semantic correspondence matching. In summary, replacing standard diffusion features with our CleanDIFT features consistently results in a significant performance improvement across all three methods. We show an overview of the results in Tab. 1 and a more extensive evaluation per category in Sec. B.

We also investigate the performance of our features in a supervised fine-tuning setting for semantic correspondence matching. Following [35], we train an aggregation network that uses all extracted feature maps and learns to aggregate them into a single task-specific feature map for semantic correspondence matching. In contrast to [35], we do not have to perform costly DDIM inversion [59] to obtain a matching noisy image for every timestep. Instead,

Method	Our Features	PCK@α (↑)	
		$\alpha_{\text{img}} = 0.1$	$\alpha_{\text{bbox}} = 0.1$
General Approaches			
DINOv2+NN	-	-	55.6
Diff. Feat.-based Approaches			
DIFT [62]	✗	66.53	59.57
	✓	68.32 ▲1.79	61.43 ▲1.86
A Tale of Two Features [66]	✗	72.31	63.73
	✓	73.35 ▲1.04	64.81 ▲1.08
Telling Left from Right [67]	✗	77.07	68.64
	✓	**78.40** ▲1.33	**69.99** ▲1.35

Table 1. Zero-shot unsupervised semantic correspondence matching performance comparison on SPair71k [37]. Our improved features consistently lead to substantial improvements in matching performance. We report PCK on the test split of SPair71k, aggregated per point. Numbers are reproduced, for a discussion and comparison to reported numbers view Tab. 5.

we directly feed the clean image to our feature extraction model. Therefore, extracting features with our CleanDIFT approach is 50x faster, since we perform a single denoiser forward pass while [35] perform 50 for the inversion. Our model achieves a PCK_{img} value of 72.48 vs their 72.75 and a PCK_{bbox} value of 64.37 vs their 64.53. We observe a slight performance regression compared to their approach, however, at a speedup of $50\times$. Luo et al. [35] also present a single-step ablation of their full method that only requires a single forward pass which makes it more comparable to ours. We outperform this single-step version by a wide margin of 9.0 percentage points for PCK_{img} and 9.1 percentage points for PCK_{bbox}.

4.3. Depth Estimation

We also investigate monocular depth estimation on NYUv2 [40]. Similar to [43], we follow the evaluation protocol from [34]. We use SD 2.1 as the base model and resize the input to the model's native resolution of 768^2. We extract features from the same location as [62] and obtain a feature map of dimension 48^2. Unlike [43], we do not upsample the features and directly apply the linear probe. The probe predicts depth in 256 uniform bins which we combine with a classification loss after a linear normalization following [3]. We train one probe for our CleanDIFT features and one for standard diffusion features at $t = 299$, as that timestep minimizes the error in our settings. Our qualitative results (see Figure 7) show a substantial fidelity gap in the estimated depth maps between the features from the standard SD 2.1 backbone and the features from our feature extraction model. This is reflected in a substantial improvement in quantitative metrics over the baseline as seen in Tab. 2. Lastly, we reuse the probe trained on standard diffusion features and apply it on the CleanDIFT features. While this does not match the performance of the CleanDIFT probe, it still achieves significantly better re-

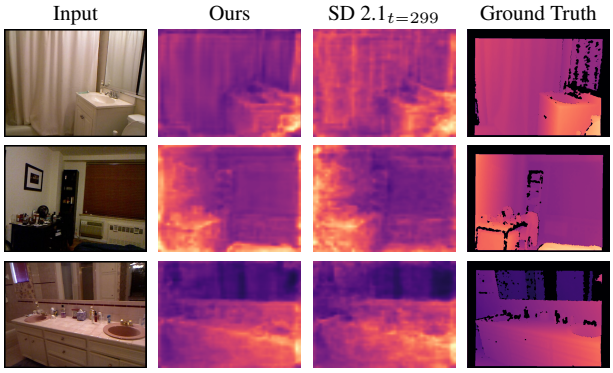

Figure 7. Qualitative results for depth estimation using a linear probe on diffusion features on NYUv2 [40]. Our CleanDIFT features enable substantially better depth estimation than standard diffusion features. Note how the CleanDIFT features are far less noisy when compared to the standard diffusion features.

Method	Backbone	RMSE (↓)
Self-Supervised Methods		
OpenCLIP [28]	ViT-G/14	0.541
MAE [20]	ViT-H/14	0.517
DINO [6]	ViT-B/8	0.555
iBOT [68]	ViT-L/16	0.417
DINOv2 [43]	ViT-g/14	**0.344**
Diffusion Features		
DIFT-like [62]	SD 2.1 [51]	0.469
	Ours	**0.444**$_{\blacktriangledown 0.025}$
	+ Probes from noisy features	0.453$_{\blacktriangledown 0.016}$

Table 2. Monocular Depth Estimation. Following [43], we evaluate metric depth prediction on NYUv2 [40] using a linear probe. Our clean features outperform the noisy features by a significant margin. Probes trained on the noisy features can be reused for the clean features, but incur a smaller performance gain.

sults when compared to using standard diffusion features. This indicates that our features can be used as a drop-in replacement for the original diffusion features and offer improved performance on downstream applications.

4.4. Semantic Segmentation

To further investigate the difference between standard noisy diffusion features and our CleanDIFT features, we evaluate on the semantic segmentation task by training linear probes on our CleanDIFT features and on standard diffusion features. We utilize SD2.1 as the diffusion backbone and extract features at the same location as [62]. This procedure yields feature maps of size 48^2. We train our linear probe on the 48^2 feature maps and upscale the obtained segmentation masks using nearest neighbor upsampling. We train and evaluate on the PASCAL VOC dataset [16]. Following common practice [39], we use mean Intersection over Union (mIOU) as the evaluation metric. Qualitative results are shown in Figure 8. Using our features, we observe significantly less noisy segmentations than with standard diffusion features. We show a quantitative comparison of our CleanDIFT feature's performance against standard diffusion features across timesteps in Figure 9. Notably, the optimal timestep appears to be around $t = 100$, in contrast to the optimal timestep for semantic correspondences, which [62] found to be $t = 261$. This highlights the need for tuning a timestep individually per downstream task. Our method both alleviates the need for such a timestep tuning and outperforms the standard diffusion features for the optimal timestep.

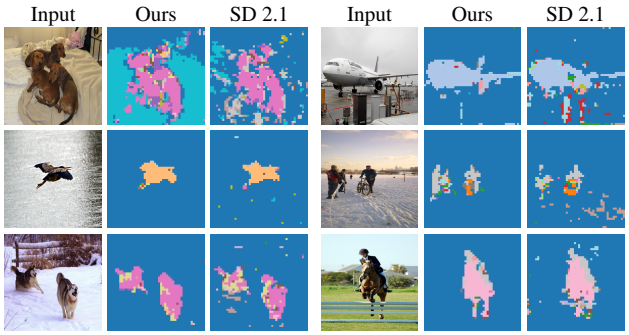

Figure 8. Qualitative results for semantic segmentation from diffusion features on Pascal VOC [16]. Standard SD features use $t = 100$ as the timestep, which we found to perform best quantitatively (cf. Figure 9). Note how the CleanDIFT segmentation maps are far less noisy than those of the standard diffusion features.

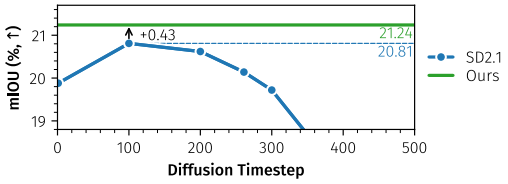

Figure 9. Performance on semantic segmentation using linear probes. Our clean features outperform the noisy diffusion features for the best noising timestep t. Semantic segmentation performance of a standard diffusion model heavily depends on the used noising timestep. Unlike for semantic correspondence matching, the optimal t value appears to be around $t = 100$.

4.5. Classification

To assess the impact of our method on non-spatial tasks, we evaluate classification performance using pooled features. Pooling mitigates the influence of localized noise, so we anticipate classification performance to remain on par with standard diffusion features unless our setup introduces detrimental effects. We perform k-Nearest Neighbor (kNN) classification with $k = 10$ on ImageNet1k [11], using SD 1.5 as the diffusion backbone. We sweep across feature maps and timesteps t for the base model, with results presented in Figure 10. Our analysis shows that the fea-

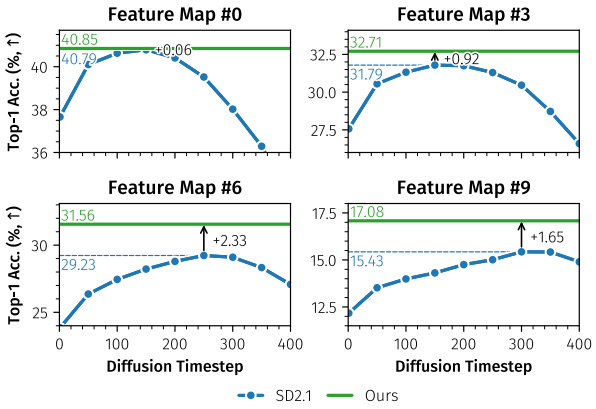

Figure 10. Classification performance on ImageNet1k [11], using kNN classifier with $k = 10$ and cosine similarity as the distance metric. We sweep over different timesteps and feature maps. We find that the feature map with the lowest spatial resolution (feature map #0) yields the highest classification accuracy.

ture map with the lowest spatial resolution, i.e., feature map #0 (see Figure 11), achieves the highest classification accuracy. Furthermore, the optimal timestep t for the base model varies between feature maps. For the best-performing feature map, $t = 100$ yields the highest classification accuracy.

Importantly, CleanDIFT features slightly outperform the standard diffusion features even when using an optimal timestep t for the base model showing that it does not introduce any detrimental effects.

4.6. Ablation Studies

For simplicity, we perform our ablation studies using DIFT [62] and evaluate the performance for unsupervised zero-shot semantic correspondence matching on a subset of the SPair71k [37] test split.

Training Objective During training, we maximize the cosine similarity between projected outputs of our feature extraction model and standard diffusion features to align them. To investigate the influence of the employed similarity metric, we compare feature extraction models trained on three different alignment objectives commonly used in similar contexts: mean absolute error (L_1), mean squared error (L_2), and cosine similarity. Quantitative results are provided in Tab. 3. While all objectives result in feature extraction models that outperform standard diffusion features, cosine similarity consistently performs the best across the alignment objectives by a significant margin, confirming our choice of similarity metric.

Projection Heads We investigate the influence of our proposed projection heads that are used to project CleanDIFT features onto standard diffusion features. The alignment of feature extraction model and diffusion model is determined by the utilized similarity metric and the projection heads. Therefore, we evaluate our feature extraction model's performance with and without the projection heads in combination with all three similarity metrics. An overview of the comparison is given in Tab. 3. In our main configuration that uses cosine similarity, using the projection heads yields slight performance improvements of 0.24 percentage points for PCK_{img} and 0.06 percentage points for PCK_{bbox} compared to fine-tuning without the projection heads. As the projection heads are typically not used for inference, they add computational overhead only during the lightweight fine-tuning. Therefore, we argue that it is worthwhile to include them and leverage the small performance gain. Additionally, they can be reused to efficiently obtain feature maps for specific timesteps.

Objective	Projection Heads	PCK@α (↑)	
		$\alpha_{\text{img}} = 0.1$	$\alpha_{\text{bbox}} = 0.1$
Cosine Sim.	✓	**68.32**	**61.43**
	✗	<u>68.16</u>	<u>61.29</u>
L_2	✓	66.23	59.13
	✗	66.49	59.43
L_1	✓	66.91	60.00
	✗	66.87	59.91
SD 2.1	-	63.41	55.92

Table 3. Ablation Study Results. We evaluate the feature extraction models' performance for zero-shot semantic correspondence matching on the SPair71k test split. PCK is aggregated per point.

5. Conclusion

In this paper, we introduced CleanDIFT, a novel approach for extracting diffusion features. CleanDIFT produces noise-free, timestep-independent, general-purpose diffusion features by consolidating timestep-dependent representations from a pre-trained diffusion backbone into a unified feature representation. We achieve this alignment between our feature extraction model and the pre-trained diffusion backbone through a lightweight fine-tuning procedure that takes approximately 30 minutes on a single A100 GPU. Operating directly on clean images, our method eliminates the information loss associated with adding noise to input images. Furthermore, CleanDIFT removes the requirement for tuning timesteps for each downstream task and avoids the computational overhead of ensembling over noise levels or timesteps. Instead, our method efficiently extracts features with just a single forward pass at inference time, substantially reducing inference costs compared to methods relying on ensembling or inversion. Extensive evaluations of CleanDIFT across diverse downstream tasks demonstrate significant performance improvements over conventional diffusion features.

Acknowledgments

This project has been supported by the German Federal Ministry for Economic Affairs and Climate Action within the project "NXT GEN AI METHODS – Generative Methoden für Perzeption, Prädiktion und Planung", the project "GeniusRobot" (01IS24083), funded by the Federal Ministry of Education and Research (BMBF), the bidt project KLIMA-MEMES, and the German Research Foundation (DFG) project 421703927. The authors gratefully acknowledge the Gauss Center for Supercomputing for providing compute through the NIC on JUWELS at JSC and the HPC resources supplied by the Erlangen National High Performance Computing Center (NHR@FAU funded by DFG project 440719683) under the NHR project JA-22883.

Further, we would like to thank Owen Vincent for continuous technical support.

References

[1] Yogesh Balaji, Seungjun Nah, Xun Huang, Arash Vahdat, Jiaming Song, Qinsheng Zhang, Karsten Kreis, Miika Aittala, Timo Aila, Samuli Laine, et al. ediff-i: Text-to-image diffusion models with an ensemble of expert denoisers. *arXiv preprint arXiv:2211.01324*, 2022. 2, 3

[2] Dmitry Baranchuk, Andrey Voynov, Ivan Rubachev, Valentin Khrulkov, and Artem Babenko. Label-efficient semantic segmentation with diffusion models. In *International Conference on Learning Representations*, 2022. 1, 3

[3] Shariq Farooq Bhat, Ibraheem Alhashim, and Peter Wonka. Adabins: Depth estimation using adaptive bins. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4009–4018, 2021. 6

[4] Giulio Biroli, Tony Bonnaire, Valentin De Bortoli, and Marc Mézard. Dynamical regimes of diffusion models. *arXiv preprint arXiv:2402.18491*, 2024. 2, 3

[5] Minwoo Byeon, Beomhee Park, Haecheon Kim, Sungjun Lee, Woonhyuk Baek, and Saehoon Kim. Coyo-700m: Image-text pair dataset. https://github.com/kakaobrain/coyo-dataset, 2022. 5

[6] 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, 7

[7] Junsong Chen, Jincheng YU, Chongjian GE, Lewei Yao, Enze Xie, Zhongdao Wang, James Kwok, Ping Luo, Huchuan Lu, and Zhenguo Li. Pixart-α: Fast training of diffusion transformer for photorealistic text-to-image synthesis. In *The Twelfth International Conference on Learning Representations*, 2024. 14

[8] Shoufa Chen, Peize Sun, Yibing Song, and Ping Luo. Diffusiondet: Diffusion model for object detection. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 19830–19843, 2023. 1, 3

[9] Xinlei Chen, Zhuang Liu, Saining Xie, and Kaiming He. Deconstructing denoising diffusion models for self-supervised learning. *arXiv preprint arXiv:2401.14404*, 2024. 2, 3

[10] Timothée Darcet, Maxime Oquab, Julien Mairal, and Piotr Bojanowski. Vision transformers need registers. In *The Twelfth International Conference on Learning Representations*, 2024. 2

[11] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009. 7, 8

[12] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34:8780–8794, 2021. 1, 2

[13] Patrick Esser, Johnathan Chiu, Parmida Atighehchian, Jonathan Granskog, and Anastasis Germanidis. Structure and content-guided video synthesis with diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7346–7356, 2023. 1, 2

[14] Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, et al. Scaling rectified flow transformers for high-resolution image synthesis. In *Forty-first International Conference on Machine Learning*, 2024. 2, 14

[15] Zach Evans, CJ Carr, Josiah Taylor, Scott H Hawley, and Jordi Pons. Fast timing-conditioned latent audio diffusion. In *Forty-first International Conference on Machine Learning*, 2024. 2

[16] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The pascal visual object classes (voc) challenge. *International Journal of Computer Vision*, 88(2): 303–338, 2010. 7, 13

[17] Michael Fuest, Pingchuan Ma, Ming Gui, Johannes S. Fischer, Vincent Tao Hu, and Björn Ommer. Diffusion models and representation learning: A survey. *CoRR*, abs/2407.00783, 2024. 3

[18] Frank Fundel, Johannes Schusterbauer, Vincent Tao Hu, and Björn Ommer. Distillation of diffusion features for semantic correspondence. *WACV*, 2025. 3

[19] Spyros Gidaris, Praveer Singh, and Nikos Komodakis. Unsupervised representation learning by predicting image rotations. In *International Conference on Learning Representations*, 2018. 2

[20] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16000–16009, 2022. 2, 7

[21] Eric Hedlin, Gopal Sharma, Shweta Mahajan, Hossam Isack, Abhishek Kar, Andrea Tagliasacchi, and Kwang Moo Yi. Unsupervised semantic correspondence using stable diffusion. *Advances in Neural Information Processing Systems*, 36, 2023. 3

[22] Geoffrey Hinton, Oriol Vinyals, and Jeffrey Dean. Distilling the knowledge in a neural network. In *NIPS Deep Learning and Representation Learning Workshop*, 2015. 3

[23] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022. 3

[24] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020. 2

[25] Jonathan Ho, Tim Salimans, Alexey Gritsenko, William Chan, Mohammad Norouzi, and David J Fleet. Video diffusion models. *Advances in Neural Information Processing Systems*, 35:8633–8646, 2022. 1

[26] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. Lora: Low-rank adaptation of large language models. *ICLR*, 1(2):3, 2022. 14

[27] Drew A Hudson, Daniel Zoran, Mateusz Malinowski, Andrew K Lampinen, Andrew Jaegle, James L McClelland, Loic Matthey, Felix Hill, and Alexander Lerchner. Soda: Bottleneck diffusion models for representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 23115–23127, 2024. 3

[28] Gabriel Ilharco, Mitchell Wortsman, Ross Wightman, Cade Gordon, Nicholas Carlini, Rohan Taori, Achal Dave, Vaishaal Shankar, Hongseok Namkoong, John Miller, Hannaneh Hajishirzi, Ali Farhadi, and Ludwig Schmidt. Openclip, 2021. 7

[29] Diederik P. Kingma and Jimmy Lei Ba. Adam: A method for stochastic optimization. In *Proceedings of the 3rd International Conference on Learning Representations (ICLR)*, 2015. 5

[30] Zhifeng Kong, Wei Ping, Jiaji Huang, Kexin Zhao, and Bryan Catanzaro. Diffwave: A versatile diffusion model for audio synthesis. In *International Conference on Learning Representations*, 2021. 2

[31] Alexander C Li, Mihir Prabhudesai, Shivam Duggal, Ellis Brown, and Deepak Pathak. Your diffusion model is secretly a zero-shot classifier. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2206–2217, 2023. 1, 3

[32] Daiqing Li, Huan Ling, Amlan Kar, David Acuna, Seung Wook Kim, Karsten Kreis, Antonio Torralba, and Sanja Fidler. Dreamteacher: Pretraining image backbones with deep generative models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 16698–16708, 2023. 3

[33] Xinghui Li, Jingyi Lu, Kai Han, and Victor Adrian Prisacariu. Sd4match: Learning to prompt stable diffusion model for semantic matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 27558–27568, 2024. 3

[34] Zhenyu Li, Xuyang Wang, Xianming Liu, and Junjun Jiang. Binsformer: Revisiting adaptive bins for monocular depth estimation. *IEEE Transactions on Image Processing*, 2024. 6

[35] 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*, pages 47500–47510. Curran Associates, Inc., 2023. 3, 5, 6, 13

[36] Chenlin Meng, Robin Rombach, Ruiqi Gao, Diederik Kingma, Stefano Ermon, Jonathan Ho, and Tim Salimans. On distillation of guided diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14297–14306, 2023. 3

[37] 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, 8, 12, 13

[38] Ishan Misra and Laurens van der Maaten. Self-supervised learning of pretext-invariant representations. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6707–6717, 2020. 2

[39] Koichi Namekata, Amirmojtaba Sabour, Sanja Fidler, and Seung Wook Kim. Emerdiff: Emerging pixel-level semantic knowledge in diffusion models. In *The Twelfth International Conference on Learning Representations*, 2023. 1, 3, 7

[40] Pushmeet Kohli Nathan Silberman, Derek Hoiem and Rob Fergus. Indoor segmentation and support inference from rgbd images. In *ECCV*, 2012. 6, 7, 14

[41] Alex Nichol, Prafulla Dhariwal, Aditya Ramesh, Pranav Shyam, Pamela Mishkin, Bob McGrew, Ilya Sutskever, and Mark Chen. Glide: Towards photorealistic image generation and editing with text-guided diffusion models. *arXiv preprint arXiv:2112.10741*, 2021. 1, 2

[42] Mehdi Noroozi and Paolo Favaro. Unsupervised learning of visual representations by solving jigsaw puzzles. In *European conference on computer vision*, pages 69–84. Springer, 2016. 2

[43] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel HAZIZA, Francisco Massa, Alaaeldin El-Nouby, Mido Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Herve Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. DINOv2: Learning robust visual features without supervision. *Transactions on Machine Learning Research*, 2024. 2, 3, 6, 7

[44] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4195–4205, 2023. 6

[45] Ethan Perez, Florian Strub, Harm De Vries, Vincent Dumoulin, and Aaron Courville. Film: Visual reasoning with a general conditioning layer. In *Proceedings of the AAAI conference on artificial intelligence*, 2018. 14

[46] Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. Sdxl: Improving latent diffusion models for high-resolution image synthesis. In *The Twelfth International Conference on Learning Representations*, 2024. 2, 14

[47] Adam Polyak, Amit Zohar, Andrew Brown, Andros Tjandra, Animesh Sinha, Ann Lee, Apoorv Vyas, Bowen Shi, Chih-Yao Ma, Ching-Yao Chuang, et al. Movie gen: A cast of media foundation models. *arXiv preprint arXiv:2410.13720*, 2024. 2

[48] 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. 2

[49] Prajit Ramachandran, Barret Zoph, and Quoc V Le. Searching for activation functions. *arXiv preprint arXiv:1710.05941*, 2017. 14

[50] Severi Rissanen, Markus Heinonen, and Arno Solin. Generative modelling with inverse heat dissipation. *arXiv preprint arXiv:2206.13397*, 2022. 3

[51] 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, 5, 7

[52] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, pages 234–241, Cham, 2015. Springer International Publishing. 3

[53] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in neural information processing systems*, 35:36479–36494, 2022. 2

[54] Tim Salimans and Jonathan Ho. Progressive distillation for fast sampling of diffusion models. *arXiv preprint arXiv:2202.00512*, 2022. 3

[55] Axel Sauer, Dominik Lorenz, Andreas Blattmann, and Robin Rombach. Adversarial diffusion distillation. In *European Conference on Computer Vision*, pages 87–103. Springer, 2024. 14

[56] 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. *Advances in Neural Information Processing Systems*, 35:25278–25294, 2022. 5

[57] Noam Shazeer. Glu variants improve transformer. *arXiv preprint arXiv:2002.05202*, 2020. 14

[58] Uriel Singer, Adam Polyak, Thomas Hayes, Xi Yin, Jie An, Songyang Zhang, Qiyuan Hu, Harry Yang, Oron Ashual, Oran Gafni, et al. Make-a-video: Text-to-video generation without text-video data. In *The Eleventh International Conference on Learning Representations*, 2023. 1

[59] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *International Conference on Learning Representations*, 2021. 6

[60] Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. *Advances in neural information processing systems*, 32, 2019. 2

[61] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020. 2

[62] Luming Tang, Menglin Jia, Qianqian Wang, Cheng Perng Phoo, and Bharath Hariharan. Emergent correspondence from image diffusion. In *Advances in Neural Information Processing Systems*, pages 1363–1389. Curran Associates, Inc., 2023. 1, 2, 3, 4, 5, 6, 7, 8, 12, 13, 14, 15

[63] Weijia Wu, Yuzhong Zhao, Mike Zheng Shou, Hong Zhou, and Chunhua Shen. Diffumask: Synthesizing images with pixel-level annotations for semantic segmentation using diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1206–1217, 2023. 1, 3

[64] 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, 13, 15

[65] Biao Zhang and Rico Sennrich. Root mean square layer normalization. *Advances in Neural Information Processing Systems*, 32, 2019. 14

[66] 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*, pages 45533–45547. Curran Associates, Inc., 2023. 1, 2, 3, 5, 6, 12, 13, 15

[67] 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 the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3076–3085, 2024. 1, 2, 3, 5, 6, 12, 13, 15

[68] Jinghao Zhou, Chen Wei, Huiyu Wang, Wei Shen, Cihang Xie, Alan Yuille, and Tao Kong. Image BERT pre-training with online tokenizer. In *International Conference on Learning Representations*, 2022. 7