Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 246 (2024) 5244–5253

Procedia Computer Science

www.elsevier.com/locate/procedia

28th International Conference on Knowledge-Based and Intelligent Information & Engineering Systems (KES 2024)

Diff-Props: is Semantics Preserved within a Diffusion Model?

Simone Bonechi[a,b,*], Paolo Andreini[b], Barbara Toniella Corradini[b,c], Franco Scarselli[b]

[a]*Department of Social, Political and Cognitive Science, University of Siena, Siena, Italy*
[b]*Department of Information Engineering and Mathematics, University of Siena, Siena, Italy*
[c]*Department of Computer Engineering, University of Florence, Florence, Italy*

Abstract

The ambition to create increasingly realistic images has driven researchers to develop increasingly powerful models, capable of generalizing and generating high-resolution images, even in a multimodal setup (e.g., from textual input). Among the most recent generative networks, Stable Diffusion Models (SDMs) have achieved state-of-the-art showing great generative capabilities but also a high degree of complexity, both in terms of training and interpretability. Indeed, the impressive generalization capability of pre-trained SDMs has pushed researchers to exploit their internal representation to perform downstream tasks (e.g., classification and segmentation). *Understanding how well the model preserves semantic information is fundamental to improve its performance.* Our approach, namely Diff-Props, analyses the features extracted from the U-Net within Stable Diffusion Model to unveil how Stable Diffusion retains semantic information of an image in a pre-trained setup. Exploiting a set of different distance metrics, Diff-Props aims to analyse how features at different depths contribute to preserving the meaning of the objects in the image.

Keywords: Generative models; Diffusion models; Stable Diffusion; Zero-shot learning; Pre-trained models; Semantic preservation; Layer analysis; Multimodal systems

1. Introduction

In recent years, generative models have gained significant traction for their ability to create increasingly realistic images. Generative models aim to learn the underlying distribution of a dataset to generate fresh and realistic samples. The earliest models, like Generative Adversarial Networks (GANs), emerged as the state-of-the-art in image generation. The quality of generation increased over time, producing realistic images in different domains from natural [2] to medical images [4].

However, the quest for models capable of generating more authentic and varied data has led to the development of increasingly complex and resource-intensive architectures. Models like StyleGAN [11] and BigGAN [3], while capa-

[*] Corresponding author.
E-mail address: simone.bonechi@unisi.it

1877-0509 © 2024 The Authors. Published by Elsevier B.V.

10.1016/j.procs.2024.09.628

ble of producing high-resolution and remarkably realistic images, require training times that can stretch over weeks, making their training impractical in terms of computational resources and time. The emergence of multimodal models based on Large Language Models (LLMs) [27] and Diffusion Denoising Probabilistic Models (DDPMs) [10], with their expanding parameter count, has further exacerbated the challenges of training from scratch. DDPMs, for instance, are likelihood-based generative models that utilize variational inference to learn a denoising Markov chain. They have emerged as the new benchmark in image generation, surpassing BigGAN and VQVAE-2 [18] based on FID (Fréchet Inception Distance) metrics on ImageNet [5]. Recent advancements in this field have also demonstrated remarkable results in text-to-image generation with models such as DALL-E [23], Imagen [20], and Stable Diffusion [19]. Given the remarkable performance of these models and the impracticality of retraining generative multimodal models from scratch, new research avenues have emerged. These aim to leverage pre-trained models through techniques like zero-shot or few-shot learning to perform tasks beyond mere generation, such as image classification and segmentation. To tackle this challenge, novel research directions are focusing on extracting information from pre-trained models internal representations to execute downstream tasks. As a consequence, the new challenge in research is to comprehend the dynamics within a pre-trained model to maximize its potential for application not only in tasks for which a specific architecture was initially trained but also in orthogonal tasks (e.g., exploiting a generative model to perform semantic segmentation). Thus we wonder:

Is it possible to investigate the ability of layers of a U-Net within an SDM to represent the image semantic?

To the best of our knowledge, while many studies offer insights into the internal characteristics of a Stable Diffusion Model (SDM) [19], there is a lack of systematic contributions in this area. This paper aims to investigate the features of an SDM and offer insights into how different resolutions of the Diffusion U-Net contribute to semantics. We propose a distance-based approach, dubbed Diff-Props, to provide an overview of each level's capability in preserving object semantics within an image, as well as the relationship among the internal representation of various objects in the U-Net encoder and decoder. In a nutshell, as shown in Figure 1, we give SDM an image and we exploit the target segmentation mask to extract the internal features corresponding to different objects in the scene. Given the corresponding prototypes, we then infer about distances between objects (i.e., tensors) in the feature space.

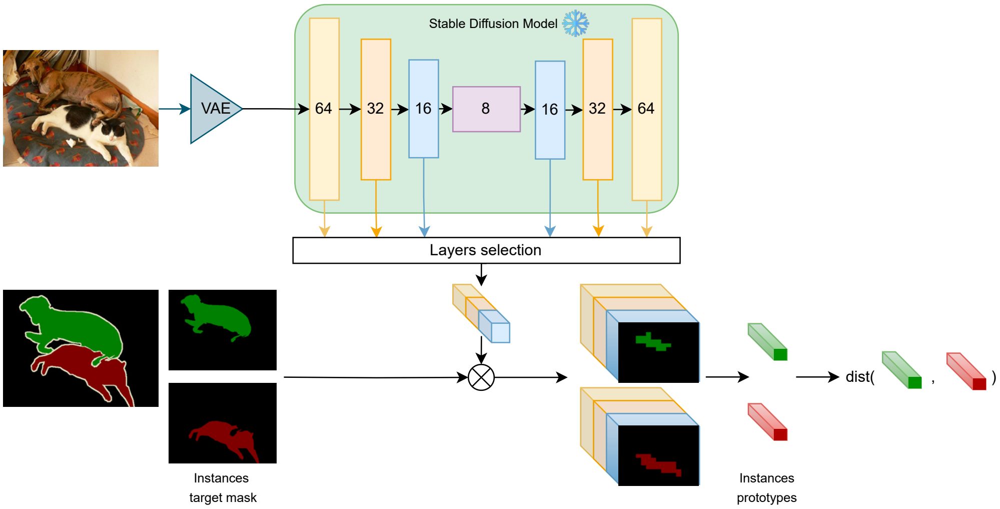

Fig. 1: Diff-Props overview. Given an image, the corresponding SDM internal features are extracted at various depths and resolutions, and filtered based on the target masks. The filtered features are averaged and the prototypes of the objects thereby obtained are compared in terms of distance within the feature space. *Our aim is to demonstrate the semantic proximity of classes within the feature space of an SDM.*

Our contribution can be summarized as follows:

- We provide insights for studying the contribution of various U-Net layers using different distance metrics, giving an idea of which measures are most effective for measurement.

- Diff-Props, an approach based on calculating distances in the feature space of a Diffusion U-Net, was introduced to this aim.
- We provide a greater understanding of complex models such as SDM, which could enable the enhancement of performance in downstream tasks.

The paper is organized as follows. Section 2 reviewed the literature related to the main aspects of diffusion models and their interpretability, while Section 3 described the datasets and metrics used in this study. Section 4 and Section 5 present the experimental setup and the obtained results respectively. Finally, Section 6 concludes and discusses possible future developments.

2. Related works

Diffusion Models (DMs). DMs constitute a family of generative models that garnered attention due to their ability to generate high-quality images by learning to reverse a diffusion process [24]. DMs are usually trained on huge datasets (LAION-5B [21]). During the forward diffusion process, the input image x undergoes incremental degradation by incorporating Gaussian noise across a predetermined number of time steps T. Conversely, in the reverse diffusion process, a neural network is trained to predict the amount of noise added to the image at time step t, denoted as $\epsilon_\theta(x_t, t)$, by minimizing the loss function:

$$\mathcal{L} = \mathbb{E}_{x_0,t,\bar{\epsilon}} \parallel \bar{\epsilon} - \epsilon_\theta(x_t, t) \parallel_2^2 \tag{1}$$

In the first implementation of DMs, DDPMs [10], the image is generated by a Markovian diffusion process. DDIMs (Denoising Diffusion Implicit Models) [25] generalize the diffusion process to a non-Markovian chain, allowing to generate higher-resolution images. Since both DDPMs and DDIMs generate images directly from their representation in the RGB pixel space, these approaches are significantly more computationally expensive compared to GANs. To address this issue, LDMs (Latent Diffusion Models) were introduced [19], which make use of a Variational Autoencoder to generate an encoded representation of the image in a lower-dimensional space compared to the high dimensional pixel space. The strength of LDMs is that, thanks to the encoding, they focus only on the important, semantic bits of the data, thereby significantly reducing the computational cost of these models. The neural backbone of LDMs is a time-conditional U-Net. In this work, we aim to explore the semantic properties of a pre-trained diffusion model known as Stable Diffusion [19], a latent text-to-image diffusion model with the ability to produce photo-realistic images based on any given text input. Like LDM, SDM also attempts to model a distribution $p(z|y)$ but it implements a conditional denoising autoencoder $\epsilon_\theta(z_t, t, y)$. This allows generation to be controlled through conditioning on y, which can be text or an image and is mapped to the intermediate layers of U-Net via a cross-attention layer.

SDMs internal features exploration. In [1], the authors explore a DDPM to identify which layers are most informative in terms of semantics, for the task of semantic segmentation. They demonstrate that it is possible to aggregate the internal features of a DDPM using K-Means and obtain a spatially coherent representation of the image. The analysis is conducted on various blocks of the decoder of the U-Net and diffusion timesteps t. Finally, an MLP is trained to predict the semantic label of a pixel based on its U-Net features. ODISE (Open-vocabulary DIffusion-based panoptic SEgmentation) [28], is a model for panoptic image segmentation, i.e., where each pixel of the image is labeled. ODISE relies on learning refined masks obtained from raw masks extracted from the layers of a U-Net within an SDM. They show they can outperform approaches based on clustering of the internal representation. In [6], the authors explore the feasibility of performing an image classification task using an SDM. The challenge lies in identifying which layers of the U-Net within a DM are most suitable for feature extraction. LD-ZNet [17] shows that the internal features of LDMs contain rich semantic information to perform text-based segmentation of synthetic images. In ASYRP paper [14], a special latent space h is introduced, which possesses significant properties – e.g., homogeneity and linearity. h comprises a 1×1 convolutional layer, which processes the concatenation sequence of bottleneck representations from the U-Net for each timestep. Despite the excellent properties of h enabling straightforward modification of image characteristics, it is unrelated to the feature space or latent space of the SDM. Adopting the Riemaniann geometry serves in [16] to understand the latent space of a diffusion model. The authors focus on finding a vector basis for the latent space by leveraging the pullback metric associated with their encoding feature maps. Their discoveries allow to move through the latent space and perform image editing via parallel transport of the vector basis. In this work, we systematically investigate the various layers of the U-Net using a feature distance-based approach to extract information about the semantics preserved by each layer.

3. Preliminary

Section 3.1 presents the two datasets used in this study, while 3.2 and 3.3 describe the similarity measures and the metrics employed to analyze the extracted features, respectively.

3.1. Datasets

3.1.1. Pascal-VOC 2012

The Pascal-VOC dataset [8] is a popular benchmark for image segmentation. Each image in the training and validation sets has pixel-level annotations for 20 object categories, a background class, and a "don't care" class for uncertain regions. In this study, we randomly selected 1000 Pascal-VOC images[1], ensuring that each of them contains at least one object with a size greater than 1% of the total image area. This criterion helps to focus on images with well-represented objects.

3.1.2. COCO 2017

The COCO dataset [15] is a large-scale image dataset containing more than 100,000 images designed for object detection, segmentation, and captioning tasks. Each image in the training and validation set comes with instance-level annotations for 80 object categories, along with background labels. Although COCO-2017 offers a wide range of object categories, in this study we extract a subset of 1000 images[1] tailored to our needs following these criteria:

- Object Size: Each image must contain at least one object with a size greater than 1% of the total image area. This ensures to focus on images with "well-represented" objects, avoiding images where the object size is too small to provide a meaningful representation in the SDM.
- Object Category: The object category should belong to one of the 20 Pascal-VOC classes. This ensures to focus on a common set of classes between the two datasets.

3.2. Distance Measures

This section describes the metrics used to compare feature vectors within the latent space of the U-Net in the SDM. These metrics allow us to quantify the relationships between features in the latent space. Given two n-dimensional vectors $\mathbf{X} = (x_1, x_2, ..., x_n)$ and $\mathbf{Y} = (y_1, y_2, ..., y_n)$ we can define the following measure:

Euclidean Distance. The Euclidean distance between two vectors, $\mathbf{X}$ and $\mathbf{Y}$ is defined as:

$$D_{\text{euc}}(\mathbf{X}, \mathbf{Y}) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2} \tag{2}$$

Distance Correlation. The Distance Correlation [26] is a statistical measure that captures both linear and nonlinear relationships between variables, offering a comprehensive view of their dependence beyond traditional correlation metrics.

$$D_{\text{corr}}(\mathbf{X}, \mathbf{Y}) = \sqrt{\frac{dCov^2(\mathbf{X}, \mathbf{Y})}{\sqrt{dVar^2(\mathbf{X})dVar^2(\mathbf{Y})}}} \tag{3}$$

where the distance covariance $dCov^2(\mathbf{X}, \mathbf{Y})$ and the distance variance $dVar^2(\mathbf{X})$ between $\mathbf{X}$ and $\mathbf{Y}$ are:

$$dVar^2(\mathbf{X}) = dCov^2(\mathbf{X}, \mathbf{X}) \tag{4}$$

[1] The list of the 1000 images can be downloaded at: https://github.com/bcorrad/Diff-Props

$$dCov^2(\mathbf{X},\mathbf{Y}) = \frac{1}{n^2}\sum_{i=1}^{n}\sum_{j=1}^{n}(a_{ij} - \bar{a}_{i\cdot} - \bar{a}_{\cdot j} + \bar{a})(b_{ij} - \bar{b}_{i\cdot} - \bar{b}_{\cdot j} + \bar{b}) \tag{5}$$

if x_i, x_j are two samples of $\mathbf{X}$ and y_k, y_l are two samples of $\mathbf{Y}$, it is possible to define:
$a_{ij} = D_{euc}(x_i, x_j)$, $b_{kl} = D_{euc}(y_k, y_l)$ and $\bar{a}_{i\cdot} = \frac{1}{n}\sum_{j=1}^{n} a_{ij}$, $\quad \bar{a}_{\cdot j} = \frac{1}{n}\sum_{i=1}^{n} a_{ij}$, $\quad \bar{b}_{i\cdot} = \frac{1}{n}\sum_{j=1}^{n} b_{ij}$, $\quad \bar{b}_{\cdot j} = \frac{1}{n}\sum_{i=1}^{n} b_{ij}$

Manhattan Distance. The Manhattan distance [12] in an arbitrary space is defined as the sum of the absolute differences between the corresponding coordinates along each dimension:

$$D_{\text{man}}(\mathbf{X},\mathbf{Y}) = \sum_{i=1}^{n}|x_i - y_i| \tag{6}$$

Cosine Distance. The cosine distance between two non-zero vectors, $\mathbf{X}$ and $\mathbf{Y}$, is defined as:

$$D_{\cos}(\mathbf{X},\mathbf{Y}) = 1 - \frac{\mathbf{X}\cdot\mathbf{Y}}{||\mathbf{X}||||\mathbf{Y}||} \tag{7}$$

where $\cdot$ denotes the dot product of the vectors, and $||\mathbf{X}||$ and $||\mathbf{Y}||$ are the Euclidean norms of the vectors. The $\frac{\mathbf{X}\cdot\mathbf{Y}}{||\mathbf{X}||||\mathbf{Y}||}$ is the cosine similarity between $\mathbf{X}$ and $\mathbf{Y}$ and quantifies the dissimilarity between vectors based on the angle between them. The cosine distance is simply the complement of the cosine similarity.

3.3. Metrics and Reliability Measures

Dunn index. The Dunn index [7] measures the compactness (intra-cluster similarity) and the separation between clusters (inter-cluster dissimilarity). It can be defined as the ratio of the smallest inter-cluster distance to the largest intra-cluster distance:

$$\text{Dunn Index} = \frac{\min_{i\neq j} d_{min}(C_i, C_j)}{\max_k d_{max}(x \in C_k)} \tag{8}$$

where $d_{min}(C_i, C_j)$ represents the minimum pairwise distance between any two clusters C_i, C_j and $d_{max}(x \in C_k)$ calculates the maximum distance between any two points within a single cluster. The higher the Dunn index, the better.

Kruskal-Wallis. The Kruskal-Wallis test [13] is a non-parametric method for assessing the equality of medians across multiple groups and serves as an alternative to one-way ANOVA [9] when its assumptions, such as normality and homogeneity of variances, are not satisfied. Indeed, the Kruskal-Wallis test does not require the data to follow a specific distribution. In this study, we apply the Kruskal-Wallis test to assess whether there are statistically significant differences between the groups of interest.

4. Experimental Setup

This work aims to investigate whether the U-Net architecture, inside the SDM, preserves object semantics within its feature space. In this section, we present the common experimental setup used throughout this study. We leverage an existing implementation of SDM available on GitHub[2] rescaling the input image to 512×512 and setting $t = 0$ (one inference step in U-Net), i.e. last denoising step. The following processing pipeline (Figure 1) is used:

1. **Image Encoding:** Images are first encoded through the VAE.
2. **U-Net feature extraction:** The images are passed through the Diffusion U-Net, and their internal representation is extracted from various residual blocks of the architecture.
3. **Feature Concatenation (Optional)**: Once the features are extracted, they are bilinearly resized to match the highest spatial dimension (height and width) across all selected layers. Features are then concatenated along the channel dimension, creating a single feature tensor that combines information from all chosen layers.

[2] https://github.com/hkproj/pytorch-stable-diffusion

4. **Mask each object in the feature space**: To analyze object-specific features, we leverage a technique that relies on the target segmentation instance maps. We first bilinearly rescale the target mask to the resolution of the feature maps, to overlay the instance mask on the tensor and isolate specific object instances within the feature space. This allows the creation of new filtered feature maps with information about the masked objects only.
5. **Prototype extraction**: For each object instance in the feature space, we then compute an average, yielding a prototype vector for each object.

This procedure is applied to all 1,000 images from both the Pascal-VOC and COCO datasets to generate a prototype representing each object present in the images. To assess the extent to which object semantics are retained in the latent space the prototype vectors are evaluated through pairwise comparison utilizing the vector distance metrics described in Section 3.2. The distances between each vector pair are included in one of four sets, based on their characteristics. The sets are defined as $S = \{D(I_{i,c}, J_{j,k})\}$ where I an J indicate two different object instances, i and j are the images that contain the instances, c and k are the object classes, and D is the distance used to compare the two prototypes.

Same class - Same image (S_{scsi})*.* The set contains the comparison between instances belonging to the same class within the same image.

$$S_{\text{scsi}} = \{D(I_{i,c}, J_{j,k}) | i = j, c = k\} \tag{9}$$

Same class - Different image (S_{scdi})*.* In this set, each instance from an image is compared with all other instances of the same class located in different images.

$$S_{\text{scdi}} = \{D(I_{i,c}, J_{j,k}) | i \neq j, c = k\} \tag{10}$$

Different class - Same image (S_{dcsi})*.* The set contains the comparison between instances of different classes within the same image.

$$S_{\text{dcsi}} = \{D(I_{i,c}, J_{j,k}) | i = j, c \neq k\} \tag{11}$$

Different class - Different image (S_{dcdi})*.* In this set, each instance from an image is compared with all instances of different classes in different images.

$$S_{\text{dcdi}} = \{D(I_{i,c}, J_{j,k}) | i \neq j, c \neq k\} \tag{12}$$

Figure 2 presents an example of the prototype extraction procedure from two images.

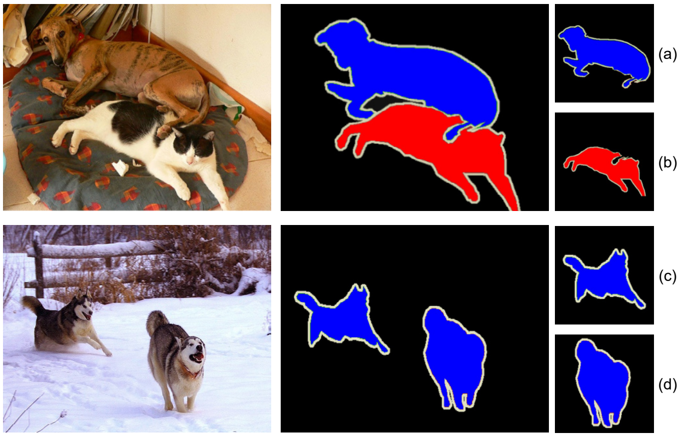

Fig. 2: The feature prototypes are extracted for each object ((a), (b), (c), and (d)) based on each instance map.

Using prototypes a, b, c, and d (in Figure 2), the four distance sets are defined as follows:

$$S_{\text{scsi}} = \{D(c, d)\}, \qquad S_{\text{scdi}} = \{D(a, c), D(a, d)\}, \qquad S_{\text{dcsi}} = \{D(a, b)\}, \qquad S_{\text{dcdi}} = \{D(b, c), D(b, d)\}.$$

Firstly, these sets are used to evaluate the most effective distance measure that could be used, in this scenario, to distinguish objects of different classes. We compare the measures introduced in Section 3.2 by analyzing feature distances extracted from specific U-Net layers. This comparison aims to identify the measure that best captures the separation between classes within the latent space. Then, using the selected measure, our goal is to assess the preservation of semantic information (object properties) within the latent space. Ideally, objects belonging to the same class should be closer together than those from different classes. This translates to maximizing the differentiation between sets containing distances from instances of the same class compared to sets with instances from different classes. To achieve this, we characterize each set by calculating its average and standard deviation. We then perform pairwise comparisons between sets. First, we employ the Kruskal-Wallis test to determine if the sets are statistically distinct. We chose this test after assessing the non-normal distribution using the Shapiro-Wilk test [22]. Subsequently, we treat the sets as clusters and calculate the Dunn Index, which measures the degree of separation between them, providing a quantitative assessment of their distinctness.

5. Results

The results of the comparison between different vector distance measures are presented in Section 5.1. While, in Section 5.2, we conduct an ablation study to investigate whether the U-Net layers within the SDM effectively preserve semantic information.

5.1. Distance Measure Selection

We evaluated the different distance measures to compare U-Net features, considering all the layers in the encoder and the decoder to avoid preferences toward specific layers (spatial resolution: 64×64, 32×32, and 16×16), excluding the 8 × 8 resolution bottleneck. Following the experimental setup described in Section 4, we calculated four sets of distances (S_{scsi}, S_{scdi}, S_{dcsi} and S_{dcdi}) using all the metric distances described in Section 3.2. The distances were computed on 2000 images collected from the Pascal-VOC and COCO datasets (1000 for each dataset), following the extraction procedure detailed in Section 3.1. Our goal is to identify the distance measure that best separates features between classes (S_{scsi} vs S_{dcsi}, S_{scsi} vs S_{dcdi}, S_{scdi} vs S_{dcsi}, S_{scdi} vs S_{dcdi} and S_{dcsi} vs S_{dcdi}) while minimizing the distances within the same class (S_{scsi} vs S_{scdi}). The results reported in Table 1 indicate that the correlation distance consistently leads to a higher Dunn Index in most cases. While other metrics may occasionally outperform it by a small margin, the correlation distance generally achieves comparable performance. The Kruskal-Wallis test assesses the statistical significance separation (p-value < 0.05). Therefore, in all the following experiments we employed the correlation distance.

		Pascal-VOC				COCO 2017			
Sets	Metric	D_{euc}	D_{cor}	D_{man}	D_{cos}	D_{euc}	D_{cor}	D_{man}	D_{cos}
S_{scsi} vs S_{scdi}	p-value Dunn Index	3.1×10^{-32} 0.30312	1.3×10^{-33} 0.33343	$\mathbf{1.2\times10^{-38}}$ **0.41411**	3.7×10^{-33} 0.33027	1.4×10^{-45} 0.35941	$\mathbf{2.3\times10^{-53}}$ **0.50012**	9.6×10^{-51} 0.39991	2.3×10^{-52} 0.49589
S_{scsi} vs S_{dcsi}	p-value Dunn Index	1.3×10^{-20} 0.65856	9.2×10^{-22} 0.80092	$\mathbf{2.2\times10^{-23}}$ **0.80684**	9.5×10^{-22} 0.78081	5.6×10^{-17} 0.37309	4.0×10^{-21} 0.59327	7.0×10^{-20} 0.42288	$\mathbf{4.5\times10^{-21}}$ **0.59863**
S_{scsi} vs S_{dcdi}	p-value Dunn Index	1.9×10^{-52} 0.60352	$\mathbf{2.0\times10^{-54}}$ 0.79657	4.4×10^{-54} 0.66042	3.3×10^{-54} **0.79998**	2.3×10^{-60} 0.48857	$\mathbf{2.8\times10^{-70}}$ **0.67989**	1.6×10^{-64} 0.56369	7.4×10^{-70} 0.67637
S_{scdi} vs S_{dcsi}	p-value Dunn Index	0.02742 0.04510	0.00028 0.08312	0.01164 0.06120	**0.00012** **0.08946**	0.00516 0.07081	**0.00195** **0.08545**	0.01620 0.05874	0.00816 0.07451
S_{scdi} vs S_{dcdi}	p-value Dunn Index	0 0.23696	0 **0.29406**	0 0.27972	0 0.29364	0 0.11407	0 0.19403	0 0.13810	0 **0.19816**
S_{dcsi} vs S_{dcdi}	p-value Dunn Index	9.1×10^{-18} 0.21439	1.5×10^{-18} **0.26777**	1.0×10^{-19} 0.22345	$\mathbf{2.0\times10^{-17}}$ 0.26179	3.8×10^{-15} 0.19077	$\mathbf{5.4\times10^{-27}}$ **0.27705**	5.6×10^{-16} 0.20624	1.4×10^{-25} 0.27002

Table 1: Comparison between distance sets using the two datasets subsets.

5.2. U-Net Features Selection

To investigate the capability of the SDM to preserve the image semantics, we employed the experimental setup described in Section 4. More specifically, the effect of extracting features from different layers of the U-Net is evaluated as follows. We compare the features extracted: at different spatial resolutions, from the encoder and the decoder and from specific layers.

Selection of the spatial resolution. We compare features extracted from various spatial resolutions in both the encoder and decoder of the U-Net to identify the optimal location for feature extraction. Four scenarios are compared:

- **64+32+16**: Use all the layers and concatenate the feature coming from all the resolutions w/o the bottleneck.
- **64**: Use only the features extracted from the layers with a spatial dimension of 64×64.
- **32**: Collect the features exclusively from the layers with a spatial dimension of 32×32.
- **16**: Use features from the 16×16 layers.

In Table 2 were reported the mean and standard deviation of the distances in the four sets (S_{scsi}, S_{scdi}, S_{dcsi} and S_{dcdi}) using the image subsets of Pascal-VOC and COCO as described above.

	S_{scsi}	S_{scdi}	S_{dcsi}	S_{dcdi}
Setup	Mean ± Std	Mean ± Std	Mean ± Std	Mean ± Std
13+32+64	0.29 ± 0.09	0.49 ± 0.14	0.54 ± 0.12	0.67 ± 0.10
64	0.06 ± 0.03	0.16 ± 0.06	0.18 ± 0.18	0.08 ± 0.21
32	0.31 ± 0.10	0.55 ± 0.16	0.60 ± 0.14	0.73 ± 0.12
16	0.40 ± 0.11	0.56 ± 0.16	0.66 ± 0.15	0.78 ± 0.09

(a) Pascal-VOC

	S_{scsi}	S_{scdi}	S_{dcsi}	S_{dcdi}
Setup	Mean ± Std	Mean ± Std	Mean ± Std	Mean ± Std
16+32+64	0.38 ± 0.11	0.62 ± 0.12	0.58 ± 0.12	0.71 ± 0.09
64	0.11 ± 0.05	0.19 ± 0.07	0.18 ± 0.08	0.22 ± 0.08
32	0.41 ± 0.12	0.69 ± 0.14	0.64 ± 0.14	0.80 ± 0.12
16	0.45 ± 0.14	0.70 ± 0.14	0.68 ± 0.12	0.82 ± 0.09

(b) COCO 2017

Table 2: Mean and standard deviation of the four sets of distances between object's features computed on the two datasets in the four setups.

The results show that these combinations do not completely capture object semantics: in the COCO dataset, all setups exhibit a higher mean distance between objects of the same class across different images (S_{scdi}), compared to objects of different classes within the same image (S_{dcsi}). This suggests that the image context can influence some feature layers bringing the representation of objects of different classes closer. However, a positive trend emerges: layers with a spatial resolution of 16×16 exhibit the lowest difference between S_{scdi} and S_{dcsi} in the COCO dataset. This finding encourages us to delve deeper into the analysis of these specific layers.

Encoder Features vs Decoder Features. We collect features from the layers with a spatial size of 16×16 comparing the features extracted from all the layers of the **Encoder** and the **Decoder**, separately. In **??** deviation of the distances obtained in the four sets (S_{scsi}, S_{scdi}, S_{dcsi} and S_{dcdi}), using the image subsets of Pascal-VOC and COCO, in the two setups.

	S_{scsi}	S_{scdi}	S_{dcsi}	S_{dcdi}
Setup	Mean ± Std	Mean ± Std	Mean ± Std	Mean ± Std
Encoder	0.42 ± 0.10	0.61 ± 0.13	0.74 ± 0.14	0.81 ± 0.11
Decoder	0.39 ± 0.12	0.56 ± 0.16	0.66 ± 0.15	0.78 ± 0.09

(a) Pascal-VOC

	S_{scsi}	S_{scdi}	S_{dcsi}	S_{dcdi}
Setup	Mean ± Std	Mean ± Std	Mean ± Std	Mean ± Std
Encoder	0.52 ± 0.13	0.72 ± 0.11	0.77 ± 0.13	0.84 ± 0.10
Decoder	0.45 ± 0.15	0.70 ± 0.14	0.67 ± 0.12	0.82 ± 0.09

(b) COCO 2017

Table 3: Mean and standard deviation of the four sets of distances between the object's features computed on the two datasets in the two setups.

Interestingly, the encoder layers at resolution 16×16 seem to preserve object semantic information: in both Pascal-VOC and COCO datasets we observe as desired that $S_{scsi} < S_{scdi} < S_{dcsi} < S_{dcdi}$. Although – as desired – S_{scdi} is lower than S_{dcsi}, we deeper investigate if the gap between S_{scsi} and S_{scdi} can be further reduced analysing the individual 16×16 layers of the encoder.

Single Encoder Layer Evaluation. Hence, we compare the features collected from the **1st Layer** and the **2nd Layer** of the encoder having spatial resolution of 16×16, separately. In Table 4 were reported the mean and standard deviation of the distances in the four sets (S_{scsi}, S_{scdi}, S_{dcsi} and S_{dcdi}) respectively using the image subset of Pascal-VOC and COCO in the two setups described above. Our analysis of the 16×16 encoder layers reveals a significant difference between the features extracted from the first and second layers. The first layer reliably captures the desired trend in

	S_{scsi}	S_{scdi}	S_{dcsi}	S_{dcdi}
Setup	Mean ± Std	Mean ± Std	Mean ± Std	Mean ± Std
1st Layer	0.42 ± 0.10	0.59 ± 0.13	0.72 ± 0.15	0.77 ± 0.12
2nd Layer	0.42 ± 0.11	0.63 ± 0.15	0.75 ± 0.14	0.85 ± 0.10

(a) Pascal-VOC

	S_{scsi}	S_{scdi}	S_{dcsi}	S_{dcdi}
Setup	Mean ± Std	Mean ± Std	Mean ± Std	Mean ± Std
1st Layer	0.50 ± 0.13	0.67 ± 0.12	0.75 ± 0.15	0.80 ± 0.12
2nd Layer	0.54 ± 0.12	0.78 ± 0.10	0.79 ± 0.12	0.89 ± 0.08

(b) COCO 2017

Table 4: Mean and standard deviation of the four sets of distances between the object's features computed on the two datasets in the two setups.

the data: $S_{scsi} < S_{scdi} < S_{dcsi} < S_{dcdi}$. Notably, the gap between S_{dcsi} and S_{dcdi} is smaller than the gap between S_{scdi} and S_{dcsi}. Additionally, S_{scsi} and S_{scdi} are closer than using all encoder features, indicating a better preservation of intra-class similarity.

To further evaluate the ability of these features to retain semantic information, we compare the results where the trend $S_{scsi} < S_{scdi} < S_{dcsi} < S_{dcdi}$ is respected using the Dunn Index. These tests will quantitatively evaluate the separation between the distributions of these distances, providing a more rigorous measure of semantic preservation. Table 5 presents the results of the Kruskal-Wallis test along with the Dunn Index obtained using only the layer of the encoder with spatial size 16×16 on Pascal-VOC and COCO 2017.

		Pascal-VOC			COCO 2017		
Sets	Metric	Encoder	1st Layer	2nd Layer	Encoder	1st Layer	2nd Layer
S_{scsi} vs S_{scdi}	p-value	6.7×10^{-29}	4.9×10^{-25}	4.6×10^{-31}	1.3×10^{-34}	2.7×10^{-25}	3.0×10^{-43}
	Dunn Index	0.36537	0.29680	0.45131	0.37752	0.29486	0.41562
S_{scsi} vs S_{dcsi}	p-value	2.8×10^{-22}	1.0×10^{-20}	7.4×10^{-23}	1.1×10^{-19}	1.0×10^{-17}	1.0×10^{-20}
	Dunn Index	0.84087	0.79547	0.89908	0.59547	0.52241	0.69764
S_{scsi} vs S_{dcdi}	p-value	1.5×10^{-51}	9.1×10^{-49}	5.7×10^{-53}	3.2×10^{54}	1.0×10^{-47}	8.9×10^{-58}
	Dunn Index	0.69247	0.64157	0.71561	0.59327	0.54425	0.64197
S_{scdi} vs S_{dcsi}	p-value	4.5×10^{-13}	1.1×10^{-13}	3.9×10^{-11}	0.00029	2.2×10^{-7}	0.36489
	Dunn Index	0.25030	0.24274	0.25346	0.09334	0.14350	0.01823
S_{scdi} vs S_{dcdi}	p-value	0	0	0	0	0	
	Dunn Index	0.35424	0.32514	0.35593	0.22620	0.22631	0.20175
S_{dcsi} vs S_{dcdi}	p-value	0.00011	0.02609	8.2×10^{-9}	3.3×10^{-7}	0.00769	4.0×10^{-16}
	Dunn Index	0.12253	0.08500	0.15392	0.13545	0.08647	0.19083

Table 5: Comparison between the features extracted from the encoder layer on the two datasets.

These results, supported by both p-values and Dunn index, demonstrate that features extracted from the first layer of U-Net effectively distinguish between objects of different classes on both the PASCAL VOC and COCO datasets. Notably, we obtain statistically significant p-values ($p < 0.05$) for all dataset combinations in this layer. Furthermore, the Dunn index for the first level shows a clear separation between sets of characteristics corresponding to different classes (higher values). In contrast, features from the same class show lower Dunn Index values, indicating good intra-class similarity. This suggests that the first layer captures more discriminative features than other layers. Instead, for the second layer we obtain a non-significative p_palue for the S_{scdi} vs S_{dcsi} (0.36489) and a corresponding extremely low Dunn Index (0.01823). This indicates that the features extracted from the second layer do not allow to distinguish objects of the same class from objects of different classes. Note that, given this result, even though the separation in the other sets may be better, the second layer was found to be unable to preserve the semantics of the objects. A similar consideration can be made for the combination of both layers of the Encoder. Even if all p-values are valid, the Dunn Index for S_{scdi} vs S_{dcsi} is quite low (0.09334). This indicates poor class separability, meaning the model struggles to distinguish between objects of the same class in different images compared to objects of different classes. Consequently, this combination of layers is also not suitable for preserving image semantics.

6. Conclusions

This paper presents a novel and comprehensive investigation into the effectiveness of internal representations within SDMs. We leverage the model's ability to maintain locality in its internal features. This allows to generate prototypes for objects in a scene based on their corresponding segmentation masks. We then compare these prototypes by calculating their distances within the diffusion U-Net feature space. By analyzing different network layers, we aim to identify the layers that best preserve object semantics. Ideally, objects with similar semantics should have distances proportional to their similarity in the feature space. Our experiments provide insights into which layers prioritize semantic information, enabling informed choices based on the desired downstream application. This knowledge could expedite the development of zero-shot methods for tasks beyond generation, such as object re-identification and tracking within a scene. For instance, in video segmentation, the same object reappearing would exhibit minimal distance from its previous representation in the feature space.

References

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

[2] Beers, A., Brown, J., Chang, K., Campbell, J.P., Ostmo, S., Chiang, M.F., Kalpathy-Cramer, J., 2018. High-resolution medical image synthesis using progressively grown generative adversarial networks. arXiv preprint arXiv:1805.03144 .

[3] Brock, A., Donahue, J., Simonyan, K., 2018. Large scale gan training for high fidelity natural image synthesis, in: International Conference on Learning Representations.

[4] Ciano, G., Andreini, P., Mazzierli, T., Bianchini, M., Scarselli, F., 2021. A multi-stage gan for multi-organ chest x-ray image generation and segmentation. Mathematics 9.

[5] Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L., 2009. Imagenet: A large-scale hierarchical image database, in: 2009 IEEE Conference on Computer Vision and Pattern Recognition, pp. 248–255.

[6] Dhariwal, P., Nichol, A., 2021. Diffusion models beat gans on image synthesis. Advances in neural information processing systems 34, 8780–8794.

[7] Dunn, J.C., 1974. Well-separated clusters and optimal fuzzy partitions. Journal of cybernetics 4, 95–104.

[8] Everingham, M., Gool, L., Williams, C.K., Winn, J., Zisserman, A., 2010. The pascal visual object classes (voc) challenge. Int. J. Comput. Vision 88, 303–338.

[9] Girden, E.R., 1992. ANOVA: Repeated measures. 84, Sage.

[10] Ho, J., Jain, A., Abbeel, P., 2020. Denoising diffusion probabilistic models. Advances in neural information processing systems 33, 6840–6851.

[11] Karras, T., Laine, S., Aila, T., 2019. A style-based generator architecture for generative adversarial networks, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR).

[12] Krause, E.F., 1973. Taxicab geometry. The Mathematics Teacher 66, 695–706.

[13] Kruskal, W.H., Wallis, W.A., 1952. Use of ranks in one-criterion variance analysis. Journal of the American statistical Association 47, 583–621.

[14] Kwon, M., Jeong, J., Uh, Y., 2022. Diffusion models already have a semantic latent space. arXiv preprint arXiv:2210.10960 .

[15] Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L., 2014. Microsoft coco: Common objects in context, in: Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13, Springer. pp. 740–755.

[16] Park, Y.H., Kwon, M., Choi, J., Jo, J., Uh, Y., 2024. Understanding the latent space of diffusion models through the lens of riemannian geometry. Advances in Neural Information Processing Systems 36.

[17] Pnvr, K., Singh, B., Ghosh, P., Siddiquie, B., Jacobs, D., 2023. Ld-znet: A latent diffusion approach for text-based image segmentation, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 4157–4168.

[18] Razavi, A., Van den Oord, A., Vinyals, O., 2019. Generating diverse high-fidelity images with vq-vae-2. Advances in neural information processing systems 32.

[19] Rombach, R., Blattmann, A., Lorenz, D., Esser, P., Ommer, B., 2022. High-resolution image synthesis with latent diffusion models, in: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 10684–10695.

[20] Saharia, C., Chan, W., Saxena, S., Li, L., Whang, J., Denton, E.L., Ghasemipour, K., Gontijo Lopes, R., Karagol Ayan, B., Salimans, T., et al., 2022. Photorealistic text-to-image diffusion models with deep language understanding. Advances in neural information processing systems 35, 36479–36494.

[21] Schuhmann, C., Beaumont, R., Vencu, R., Gordon, C., Wightman, R., Cherti, M., Coombes, T., Katta, A., Mullis, C., Wortsman, M., et al., 2022. Laion-5b: An open large-scale dataset for training next generation image-text models. Advances in Neural Information Processing Systems 35, 25278–25294.

[22] SHAPIRO, S.S., WILK, M.B., 1965. An analysis of variance test for normality (complete samples). Biometrika 52, 591–611. URL: https://doi.org/10.1093/biomet/52.3-4.591, doi:10.1093/biomet/52.3-4.591.

[23] Shi, Z., Zhou, X., Qiu, X., Zhu, X., 2020. Improving image captioning with better use of captions. arXiv preprint arXiv:2006.11807 .

[24] Sohl-Dickstein, J., Weiss, E., Maheswaranathan, N., Ganguli, S., 2015. Deep unsupervised learning using nonequilibrium thermodynamics, in: International conference on machine learning, PMLR. pp. 2256–2265.

[25] Song, J., Meng, C., Ermon, S., 2020. Denoising diffusion implicit models. arXiv preprint arXiv:2010.02502 .

[26] Székely, G.J., Rizzo, M.L., 2007. Distance correlation: a measure for dependence between multivariate random variables. Annals of Statistics 35, 2769–2792.

[27] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I., 2017. Attention is all you need. Advances in neural information processing systems 30.

[28] Xu, J., Liu, S., Vahdat, A., Byeon, W., Wang, X., De Mello, S., 2023. Open-vocabulary panoptic segmentation with text-to-image diffusion models, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 2955–2966.