UNIVERSIDAD AUTÓNOMA DE MADRID

ESCUELA POLITÉCNICA SUPERIOR

Master in Deep Learning for
Audio and Video Signal Processing

MASTER THESIS

SYNTHETIC DATA GENERATION USING LATENT DIFFUSION MODELS FOR SEMANTIC SEGMENTATION OF URBAN SCENES

Pablo Marcos Manchón
Advisor: Juan Carlos San Miguel Avedillo

June 2023

SYNTHETIC DATA GENERATION USING LATENT DIFFUSION MODELS FOR SEMANTIC SEGMENTATION OF URBAN SCENES

Pablo Marcos Manchón
Advisor: Juan Carlos San Miguel Avedillo

Dpto. Tecnología Electrónica y de las Comunicaciones
Escuela Politécnica Superior
Universidad Autónoma de Madrid
June 2023

Trabajo parcialmente financiado por el Gobierno de España bajo el proyecto PID2021-125051OB-I00 (HVD): Recolección de datos visuales: permitiendo la visión por computador en escenarios con datos desfavorables (2022-2025)

Resumen

Este estudio investiga la utilización de modelos latentes de difusión texto-imagen (LDM) para generar conjuntos de datos sintéticos en tareas de segmentación semántica. Se enfoca específicamente en su aplicación en escenarios urbanos, donde la escasez de datos anotados motiva el uso de datos sintéticos.

La investigación se centra en los mapas de atribución con atención difusiva (DAAM, por sus siglas en inglés), una técnica existente de explicabilidad que asigna la influencia de cada parte de un texto a las regiones de una imagen generada por un LDM.

Se proponen dos extensiones de DAAM. En primer lugar, se introduce "Open Vocabulary DAAM", que permite la construcción de mapas de atribución para textos arbitrarios, independientemente de si se utilizaron como texto de entrada para la generación de las imágenes sintéticas. En segundo lugar, se propone "Linear DAAM", una versión simplificada que facilita la generación de mapas de atribución para palabras individuales. Estas modificaciones permiten utilizar este método para la segmentación de objetos basada en descripciones semánticas.

Para abordar el desafío de seleccionar la palabra más apropiada para describir semánticamente un objeto, se propone una estrategia de optimización en el espacio de los textos. Este enfoque tiene como objetivo identificar las palabras más precisas para describir las regiones objetivo, mejorando así la precisión de las máscaras de segmentación.

Para validar la metodología propuesta, se realizaron una serie de experimentos en un conjunto de datos generado mediante el modelo *Stable Diffusion*. Los resultados obtenidos corroboran la efectividad de las palabras optimizadas en la segmentación de objetos en diversas imágenes.

Este trabajo contribuye al problema de investigación en dos aspectos principales. En primer lugar, en el ámbito de la explicabilidad, mediante el desarrollo de "Open Vocabulary DAAM", una herramienta con potencial para analizar las relaciones semánticas aprendidas en estos modelos, así como los posibles sesgos y los mecanismos de síntesis involucrados. En segundo lugar, avanza en la investigación sobre modelos de segmentación basados en vocabulario abierto al proponer una estrategia para buscar palabras descriptivas de objetos, lo que mejora las máscaras de segmentación sin necesidad de volver a entrenar los modelos.

Aunque los hallazgos presentados en este estudio son preliminares, resaltan el potencial del uso de mapas de atención en la segmentación de objetos. En conjunto, este trabajo sienta los cimientos para futuras investigaciones en este campo.

Palabras clave

Mapas de Atribución con Atención Difusiva (DAAM), Stable Diffusion, Modelos de Difusión, Generatión de datos sintéticos, Texto-Imagen, Modelos Generativos, Segmentación Semántica, Escenas Urbanas, Visión Artificial

Abstract

This master's thesis investigates the use of text-to-image Latent Diffusion Models (LDM) for generating synthetic datasets in semantic segmentation tasks. Specifically, it focuses on their application in urban scenarios, where the scarcity of annotated data motivates the use of synthetic data.

The research centers around Diffusion Attentive Attribution Maps (DAAM), an existing explainability method used to attribute the influence of each part of a text prompt to regions in a generated image produced by an LDM.

Two extensions of DAAM are proposed. Firstly, "Open Vocabulary DAAM" is introduced, enabling the construction of attribution maps for arbitrary texts, regardless of whether they were used as prompts for generating the synthetic images. Secondly, "Linear DAAM" is presented as a simplified version that facilitates the generation of attribution maps for individual words. These modifications facilitate the use of this method for object segmentation based on semantic descriptions.

To address the challenge of selecting the most appropriate word to semantically describe an object, an optimization strategy in the text-embedding space is proposed. This approach aims to identify the most accurate words for describing target regions, thereby enhancing the precision of segmentation masks.

To validate the proposed methodology, a series of experiments were conducted on a dataset generated using Stable Diffusion. The results confirm the effectiveness of optimized tokens in segmenting objects across diverse images, thereby emphasizing the valuable semantic information contained within these tokens.

This work contributes to the research problem in two main aspects. Firstly, it deepens the explainability of LDMs through the development of "Open Vocabulary DAAM," a tool with the potential to analyze learned semantic relationships, potential biases, and synthesis mechanisms. Secondly, it advances research on Open Vocabulary-based segmentation models by proposing a strategy for searching descriptive words for an object, resulting in improved segmentation masks without the need for model retraining.

Although these findings are preliminary, they strongly highlight the potential of attention maps in object segmentation. Moreover, they provide a solid foundation for future research in this field.

Keywords

Diffusion Attentive Attribution Maps (DAAM), Stable Diffusion, Latent Diffusion Models, Synthetic data generation, Text-to-Image, Generative models, Semantic Segmentation, Urban Scenes, Computer Vision

Agradecimientos

En primer lugar, me gustaría agradecer a Juan Carlos por guiarme semana tras semana consiguiendo que no me dispersase demasiado y pudiera convertir todo este esfuerzo en algo tangible. Me has enseñado que aún quedan profesores con vocación y la motivación suficiente para escuchar lo que sus alumnos tienen que decir. A Roberto, por tus sabios consejos con los que has evitado que me frustara en exceso y tirara la toalla. Aunque en la portada no hubiera hueco para un segundo tutor, realmente has sido uno muy bueno. Sin vosotros todo este trabajo no habría podido acabar en algo por escrito.

A ChatGPT, por ser el asistente que hubiera soñado tener hace unos meses.

A Marina, por ser el pilar más fundamental en mi vida. Siempre tienes las palabras para alegrame cualquier día. En poco estaremos celebrando que nos volverá a dar la luz del sol. Te quiero.

A mi madre, Mar. Por haberme enseñado a no olvidar nunca el lado humano de las cosas. Todos los días me sigues enseñando pequeñas lecciones de vida. A mi padre, Juan Ángel. Por apoyarme de la mejor forma que sabes y cuidar de las gatas.

Por último, me gustaría agradecer a toda mi familia y amigos por haber soportado mi desaparición estos meses. Estoy bien, no me habían secuestrado, estaba terminando el máster.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Deep Learning has revolutionized the field of Computer Vision over the recent years, mainly due to the availability of large-scale datasets and computational resources [1, 2]. This evolution has accelerated the interest in the field and its use is widespread in many applications, including autonomous driving, surveillance, human-computer interaction, and medicine [3].

Alongside this, generative models have seen remarkable advancements, with diffusion models gaining popularity due to their prowess in generating high-quality images, audios, and videos from textual descriptions [4, 5, 6, 7, 8]. However, there is a growing need for methods that can explain how these models operate, facilitating their transparency, reliability, and adoption in practical applications [9]. Despite significant progress in explaining discriminative models, explaining generative ones, especially diffusion models, remains relatively new and exploratory due to their recent emergence [10].

Although diffusion models have made strides in artistic image generation from textual cues, the potential of using these models for synthetic data generation, particularly for semantic segmentation tasks, remains an open question [11, 12, 13, 14].

1.1 Motivation

Semantic segmentation, a fundamental task in Computer Vision, involves assigning semantic labels to every pixel in an image. The availability of large-scale labeled datasets is critical for training deep learning models for this task, but is often constrained due to the high cost of annotation [15]. Existing solutions, such as coarse annotations, community annotated datasets, and training with synthetic data, have been employed to address this data scarcity [16, 17, 18]. However, these methods often grapple with domain adaptation issues and a lack of diversity in training data, which may limit their effectiveness in real-world scenarios [19, 18].

Therefore, there is a strong motivation to develop diverse and adaptable synthetic datasets, which could potentially leverage generative models [20]. This research aims to bridge this gap and enhance the reliability and adaptability of deep learning models for tasks like semantic segmentation, making them more accessible for a wide range of use cases.

Semantic segmentation of urban scenes, taken from the perspective of a vehicle in an urban environment, has a high application potential in autonomous driving for tasks like obstacle detection and lane recognition. However, the availability of realistic

synthetic datasets with ground truth for semantic segmentation is limited, despite the growing interest from both research and industry [21]. While there are existing synthetic datasets and simulators for semantic segmentation, they often lack realism and variability.

1.2 Objectives

This master's thesis aims to address the open question: "Is it possible to use diffusion models to generate synthetic datasets for training semantic segmentation models in urban scenes?" Given the exploratory nature of this question, the work serves as an exploratory analysis to assess the potential of diffusion models, particularly the recent Stable Diffusion architecture [6], which will be the foundation of this work.

Specifically, the objectives of this project can be divided into the following milestones:

- Conduct a comprehensive study of the state of the art in the use of synthetic data for semantic segmentation to understand the overall problem to be addressed.
- Explore diffusion model research and examine existing explainability techniques for analyzing these models to identify suitable approaches for addressing the problem.
- Study the Stable Diffusion architecture and its internal processes to identify key elements for generating accurate ground truth for semantic segmentation. Propose a method to address the challenge.
- Perform practical experiments using the proposed methods to evaluate their limitations and potential.
- Identify potential applications and future research directions.

1.3 Report structure

This work is structured into five chapters, which are as follows:

- Chapter 1 provides an introduction to the research problem, outlining the objectives, research questions, hypotheses, and the significance and scope of the study.
- Chapter 2 presents a comprehensive review of the existing literature related to the research problem, focusing on three key areas: semantic segmentation, generative models, and explainability.
- Chapter 3 details the proposed methods for addressing the research problem, providing a theoretical framework underlying the proposed solution.
- Chapter 4 presents the experiments conducted based on the proposed methods and evaluates their results.
- Finally, Chapter 5 presents the conclusions drawn from this work, analyzing its potential and limitations, and reflects on possible avenues for future research.

Chapter 2

Related work

This chapter overviews the state of the art in three key areas of Computer Vision relevant to this master thesis: semantic segmentation, generative models, and explainability. This review aims to establish a comprehensive understanding of the methods explored in this thesis, as well as to contextualize the motivation of synthetic data generation to develop semantic segmentation methods.

The chapter is divided into three sections, each focusing on the aforementioned areas. We start by exploring the challenges of semantic segmentation and the various approaches taken to overcome the lack of annotated data. Next, we delve into the literature on generative models, with a specific emphasis on the recent Stable Diffusion architecture that is used to generate synthetic images in this work. Finally, we discuss the issue of explainability in Computer Vision, concentrating notably on attention-based methodologies. We specifically examine the ongoing research that employs attention maps for achieving explainability in Latent Diffusion Models. These maps serve as a pivotal approach in text-to-image models, attributing the influence of input prompts on the images generated.

2.1 Semantic Segmentation

Semantic segmentation is a Computer Vision task that aims to assign a semantic label to each pixel of an input image. More formally, the objective is to learn an injective function that maps an image to its corresponding segmentation map, as the one illustrated in Figure 2.1.

The task is closely related to other Computer Vision tasks, such as depth estimation, which estimates the depth of each pixel in an image, or lane detection, which predicts the geometry of a road from the perspective of a vehicle. In the domain of urban scenes, solutions to these problems are core components for the development of applications in autonomous driving [22]. The accuracy and robustness of algorithms for these tasks are critical for developing safe and reliable autonomous vehicles.

Despite significant progress in recent years, semantic segmentation remains a challenging task due to the high variability of natural scenes, as well as the limited availability of annotated data. To address the lack of annotated data, researchers have explored various approaches, such as transfer learning, domain adaptation, and synthetic data generation. In the following subsections, we discuss these approaches in detail.

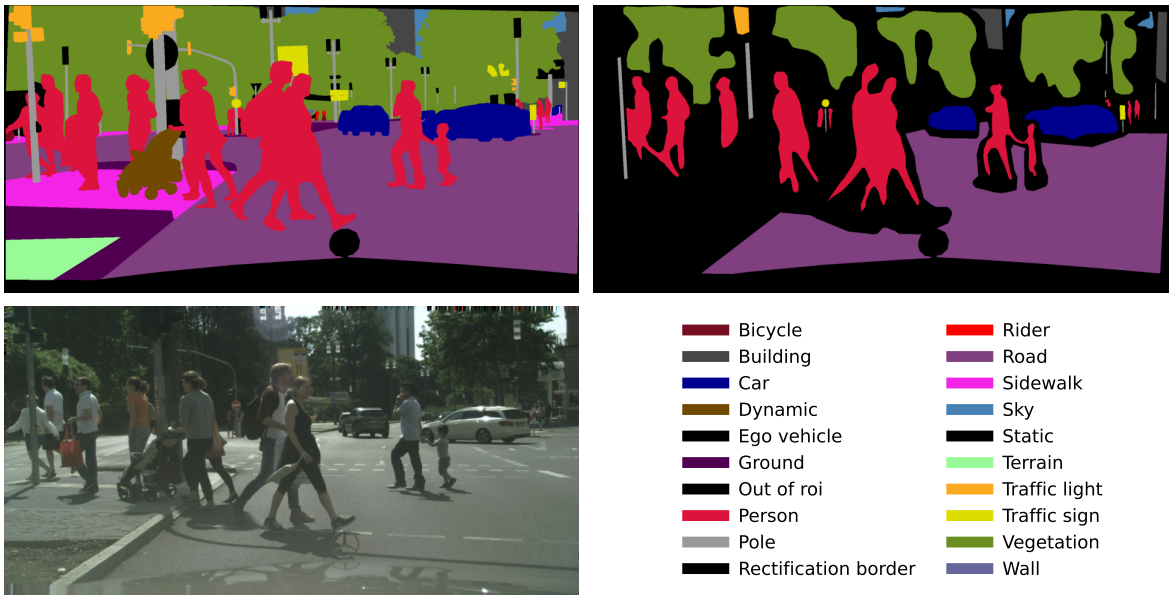


Figure 2.1: Example of an urban scene with semantic segmentation annotations from the Cityscapes Dataset [16]. Top-left: Fine-grained ground-truth annotation. Top-right: coarse annotation. Bottom-left: image. Bottom-right: semantic classes.

2.1.1 Architectures

Deep convolutional neural networks have emerged as the leading approach to address semantic segmentation [1]. Unlike other tasks such as classification, where the output of the models is a vector with one score per class, semantic segmentation requires the prediction of a score for each pixel of the input image and its corresponding semantic class, making it a computationally demanding operation.

Although researchers have proposed successful architectures that process images while maintaining the spatial dimension internally in all layers [23], the use of such architectures comes with a high computational cost. For this reason, the most common architectures downsample the images with convolutional layers to reduce computational requirements and upsample the compact representation to produce an output of the same size as the input. This takes advantage of spatial locality, making the process more computationally efficient [24].

One of the earliest architectures that proposes the aforementioned approach is the Fully Convolutional Network (FCN) [25]. FCN uses several convolutional layers to reduce the spatial dimension of the input image while increasing the number of channels to maintain local semantic information. A final pixel-wise layer is used to generate a detailed segmentation, as illustrated in Figure 2.2.

To prevent the potential loss of spatial information due to the downsampling layers, researchers proposed the U-NET architecture [26], a symmetric encoder-decoder design (shown in Figure 2.3) that incorporates connections between the downsampling and upsampling stages. These connections merge the semantic information from the higher layers with the spatial information from the lower layers obtaining fine-grained predictions.

Many contemporary semantic segmentation architectures are being proposed in this direction, aiming to compactly encode semantic information while minimizing spatial information loss. As an example, two popular approaches are the use of atrous convo-

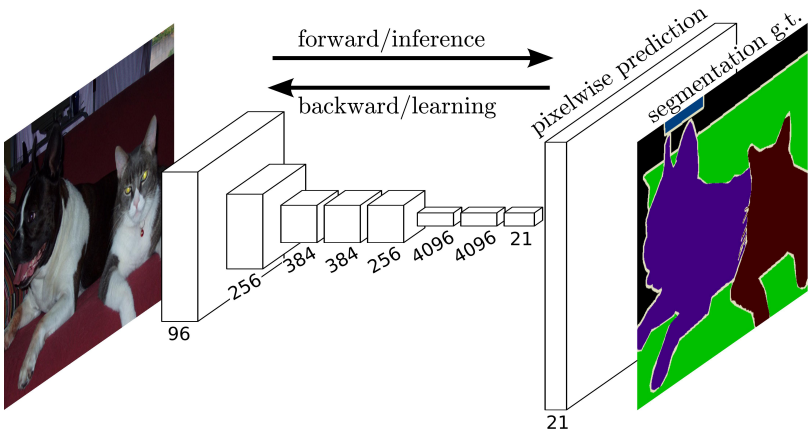


Figure 2.2: Fully convolutional network for semantic segmentation [25]

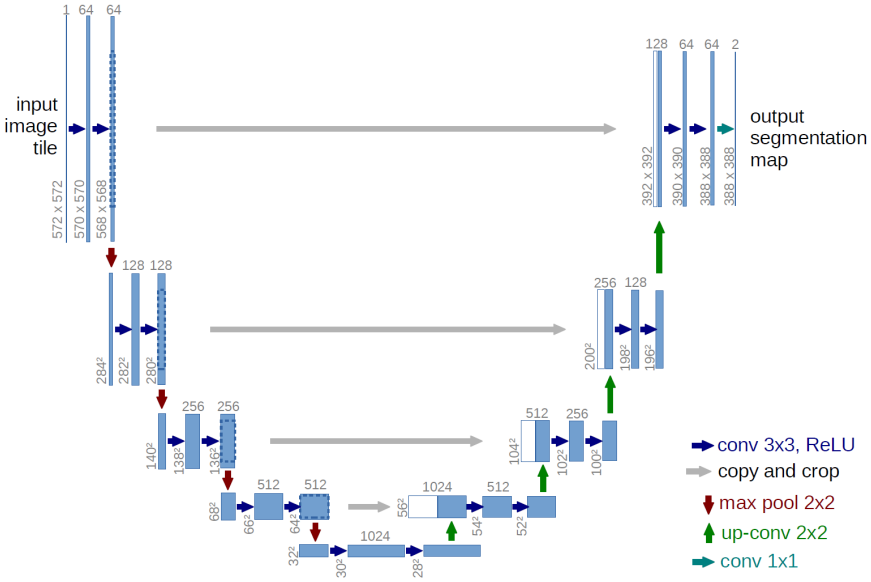
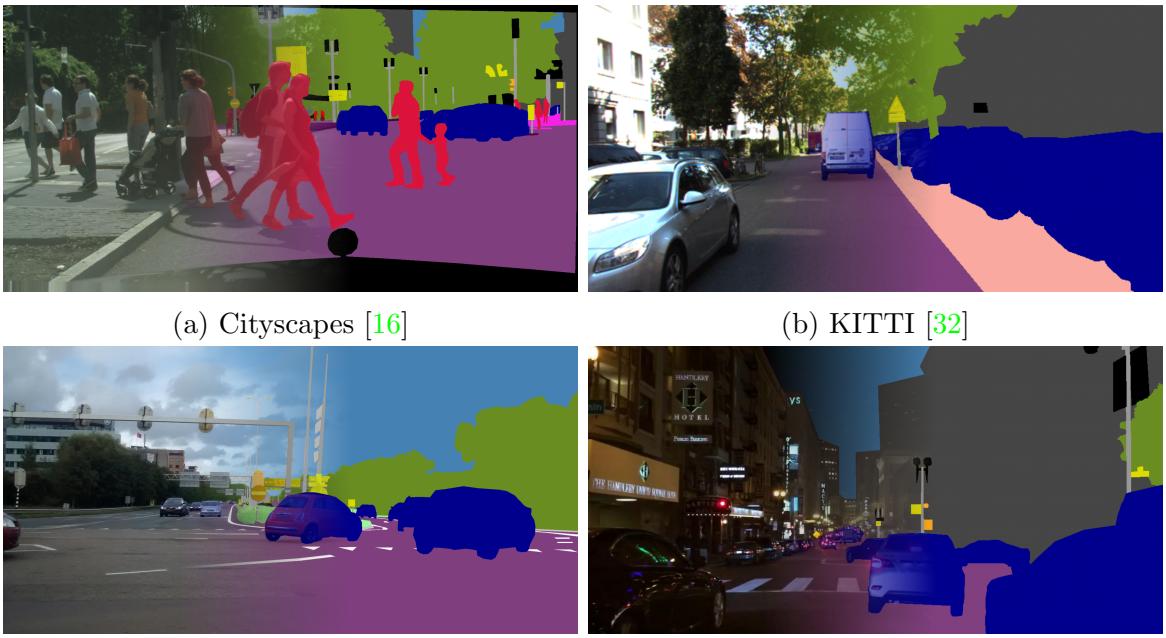


Figure 2.3: U-NET architecture. Source [26].

lutions and multi-resolution features. Atrous convolutions [27], also known as dilated convolutions, increase the receptive field by introducing gaps between kernel elements, resulting in the efficient processing of high-resolution images without increasing the number of parameters. On the other hand, multi-resolution features, used in HRNet [28], involves using convolutions of different sizes in parallel to extract information at multiple levels of resolution and combine it for a more robust representation. This strategy allows the use of features with complementary information, improving accuracy and enabling fine-grained predictions.

Since these models are complex and require large amounts of data, training strategies such as curriculum learning [29], knowledge distillation [30], and adversarial losses [31] are commonly used to improve their performance. For this reason, much of the research efforts are based on exploring new training strategies and creating datasets with increased variability, particularly in domains such as urban scenes. In the following subsection, we introduce some of the most widely used datasets and benchmarks for training and compare the performance of these models.



(a) Cityscapes [16] (b) KITTI [32]

(c) Mapillary Vistas [17] (d) Berkeley Deepdrive Dataset [33]

Figure 2.4: Visual examples of real datasets for urban scenes

2.1.2 Datasets and benchmarks

One of the main factors contributing to the success of semantic segmentation models is the availability of large datasets with high-quality annotations for training and evaluation. To address this issue, the research community has created several public datasets and benchmarks to train these models and unify comparisons. In this section, we review the most commonly used real datasets for semantic segmentation of urban scenes.

Urban scenes datasets are composed of images taken from cameras on moving vehicles or captured by drones, and commonly they show the urban environment from the driver's point of view. These images are annotated with pixel-level labels with different semantic categories, such as roads, buildings, pedestrians, and vehicles. The annotations are used as ground truth segmentation maps to train and evaluate semantic segmentation models. Some of the most widely used datasets in this area, illustrated in Figure 2.4, are Cityscapes [16], KITTI [32], Mapillary [17], and BDD100K [33].

Cityscapes [16] contains high-quality images of street scenes from 50 different cities across Europe, with pixel-level annotations of 30 different semantic classes. KITTI [32] contains images of urban scenes captured by a camera mounted on a moving vehicle and annotated with semantic labels. Mapillary [17] is a crowd-sourced dataset that contains images taken from street-level perspectives by a large community of users with global coverage. BDD100K [33] contains diverse and challenging images with different weather conditions in urban and suburban areas. Table 2.1 contains information of these and other of the most widely used datasets of urban scenes.

As a way to increase the variability and the number of annotated images, some datasets provide low-quality annotations that can be used for pre-training. For example, Cityscapes provides 20,000 extra coarse annotations (as shown in Figure 2.1). However, due to the high cost of manually annotating images with high detail [15], these datasets consist of only a few thousand images. Despite the effort to create more

Dataset	Geographic coverage	Classes	Public images
Cityscapes [16]	50 cities (Germany)	30	5,000 (+20,000)
KITTI [32]	1 city (Germany)	30	200
Mapillary Vistas [17]	Global coverage	124	20,000
Berkeley DeepDrive [33]	4 cities (USA)	40	10,000
Wilddash2 [34]	Global coverage	30	4,256
Apollo Scape [35]	4 regions in China	25	146,997
India Driving Dataset [36]	182 scenes (India)	34	10,000
Audi A2D2 [21]	South of Germany	38	41,277

Table 2.1: Summary of widely used datasets in semantic segmentation of urban scenes

diverse and high-quality datasets, a still unsolved problem is the high bias between them. Models trained on a dataset do not generalize correctly to other ones due to differences in image quality, lighting conditions, and annotation criteria [17]. This issue is known as domain shift and is one of the main challenges in semantic segmentation research.

To enable the comparison of different models under the same conditions, it is necessary to define a set of evaluation metrics. To address this problem, the research community has created benchmarks associated with each of the main datasets to evaluate models under the same conditions. The most common metric for semantic segmentation is Intersection over Union (IoU), which measures the overlap between the predicted and ground truth segmentation maps. Although other metrics, such as precision and recall, are also used to evaluate model performance, and the choice of new metrics is still an open discussion in the research community [37, 38, 39].

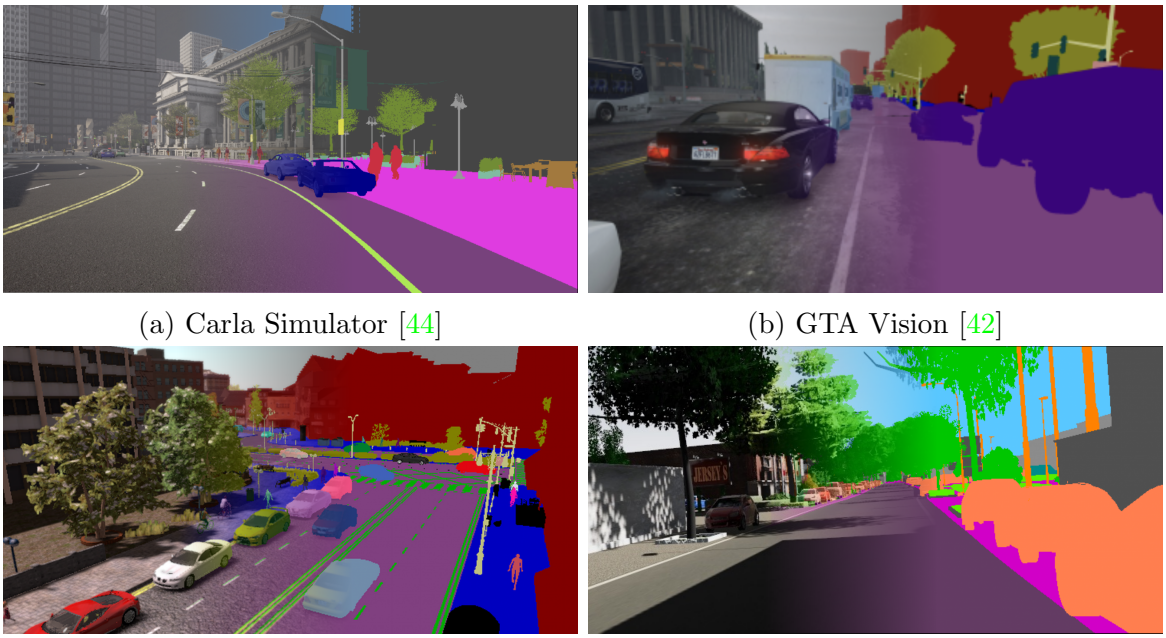
2.1.3 Synthetic data

Since the first attempts to develop autonomous driving systems, a major challenge has been the availability of datasets that can capture the variability of driving environments [40]. To address this issue, researchers have employed synthetic data generated by simulators to increase this variability.

In recent years, simulators based on game engines, such as Unity [32, 41] or GTA V [42], have become popular. Simulators allow the extraction of complete information from the environment, including semantic classes, 3D boxes, multiple perspectives, or depth maps in a pixel-accurate way, as in the examples illustrated in Figure 2.5. Other techniques, such as the use of Generative Adversarial Networks [43], have also been employed to increase the amount of data and variability from pre-existing datasets. These methods are typically employed to alter weather or light conditions of scenes, or even to introduce new elements into images.

However, training or pre-training on synthetic data presents challenges due to domain adaptation problems. The domain adaptation problems arise because of the difference in image distributions between synthetic data used for training models and real-world images. Due to this difference, systems lose efficiency when deployed in real-life situations. To mitigate this issue, two primary strategies are used: input and output adaptation.

Input adaptation involves adjusting the synthetic images used during the training step to make them more photorealistic. This can be achieved by using increasingly



(a) Carla Simulator [44] (b) GTA Vision [42]

(c) SYNTHIA [41] (d) Virtual KITTI [32]

Figure 2.5: Visual examples of synthetic datasets for urban scenes

realistic simulators or by employing generative models to adapt the style of simulator images to make them more photorealistic. However, there is no general consensus on whether photorealism is a necessary aspect that can be solved by increasing the variability in the datasets.

Output adaptation involves incorporating strategies during the training of semantic segmentation models to increase their generalization and reduce the impact of the difference between synthetic and real distributions. This group of strategies includes the use of transfer learning, curriculum learning [29], knowledge distillation [30], or adversarial losses [31].

In conclusion, the generation of scenarios capable of including as much variability as possible is fundamental for the development of semantic segmentation systems. Synthetic datasets can include environmental situations that are difficult to simulate and can reduce the need for large real-world datasets, which is crucial for the development of more robust autonomous driving systems. For these reasons, both the scientific community and the industry are putting great efforts into developing new methods for the creation of synthetic data and techniques to reduce the problems of domain shift and domain adaptation.

2.2 Generative models

Generative models, particularly those based on neural networks for the generation of new images, have gained significant popularity recently. Unlike discriminative models, which focus on learning the conditional distribution $P(Y|X)$ of output data Y given input data X, generative models aim to learn the joint distribution $P(X, Y)$.

For instance, consider a style transfer problem such as the one proposed in [45], where an image from an urban environment simulator is enhanced to improve its photorealism. A generative model attempts to learn the probability distribution of (X_i, Y_i)

pairs, where X_i is an input image and Y_i is an improved version of the image. By sampling an element from the learned distribution of Y, a generative model can produce an image that fits within the given domain. Using this approach, generative models can be used to solve tasks such as style transfer, data augmentation, or synthetic data generation.

In contrast, discriminative models concentrate on learning the conditional distribution $P(Y|X)$ to make predictions based on input data X. In a classification-like problem, this is done by learning the boundary decision that separates different classes of Y based on the input X. For example, in semantic segmentation, a discriminative model predicts the probability of semantic classes based on an input image X_i. Although discriminative approaches are very effective for tasks such as classification or regression, they are not suitable for generating new samples by sampling the learned distribution.

In Computer Vision, generative models have advanced significantly due to the availability of large computational resources and datasets with billions of images [46], which have enabled the training of deep generative models. The design of models with controllable latent spaces has been a key factor in this evolution [47]. Latent spaces are low-dimensional representations learned from the input data that capture their underlying structure. By manipulating the values of these latent variables, the generative model can control specific aspects of the output or generate mixtures of output samples [48].

Another significant development has been the creation of conditional generative models, which introduce a third element such as a text prompt to guide the generation of samples in a more direct way. This has led to the development of generative models that generate images [6, 5], audio [49], or video [50] based on natural language descriptions.

In the following subsections, we review the main architectures that have contributed to the development of generative models and have led to the design of diffusion models[51]. We pay special attention to the Stable Diffusion architecture [6], which is the focus of this work.

2.2.1 Generative architectures

Deep generative models have achieved impressive results by learning a compact, expressive latent space that enables better synthesis or manipulation of high-dimensional data. However, developing generative architectures has been more challenging than their discriminative counterparts, given the complexity of the task. Nevertheless, machine learning's historical roots in classical architectures like Boltzmann machines [52] or autoencoders [53, 54] provided a foundation for the evolution of these models.

Autoencoders

Autoencoders, which date back to the 1980s [53, 54], are neural architectures in which a bottleneck layer with a small number of neurons is introduced. In their original design, the network is trained to predict the same output example as the one given as input, but due to the bottleneck layer, a compact representation is forced to be learned. Figure 2.6 shows an example of a Convolutional Autoencoder [55] following this scheme, which is trained to generate the same digit given as input but with a layer in the middle.

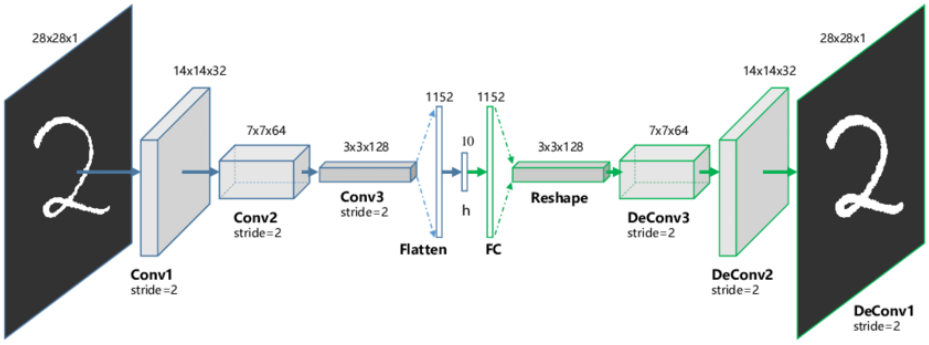


Figure 2.6: Convolutional Autoencoder example. Source [55].

To increase the robustness of the learned representation, researchers have proposed different strategies. For example, Denoising Autoencoders [56] introduce noise in the input samples, Contractive Autoencoders [57] include penalty terms, or other works proposed the learning of more complex unsupervised tasks [58, 59, 60].

Autoencoders have been successfully applied to reconstruct noisy images [56] and perform style transfer tasks such as image colorization [61]. However, when used for generating new synthetic samples using new values of the learned latent space representation in the bottleneck layer, the generated samples may lack variability. Additionally, the learned space may not exhibit desirable properties, as interpolation in the space does not always produce high-quality samples. As a result, other architectures such as Generative Adversarial Networks [43] or Latent Diffusion models have become more popular in these types of applications.

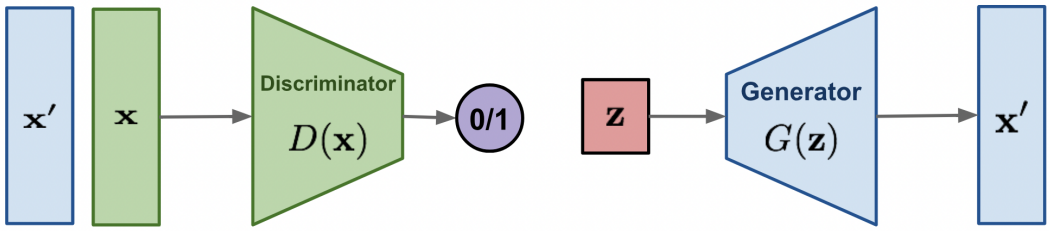
Generative Adversarial Networks (GANs)

Generative Adversarial Networks (GANs) [43] were proposed in 2014 as a revolutionary idea in deep learning, inspiring research in the field ever since. GANs are composed of two neural networks - a generator network G and a discriminator network D. The generator network is trained to generate synthetic samples x' from a latent vector z from a known distribution, such as a Gaussian noise distribution. Meanwhile, the discriminator network is trained to classify synthetic and real samples, distinguishing between them. Figure 2.7a provides an illustration of this architecture.

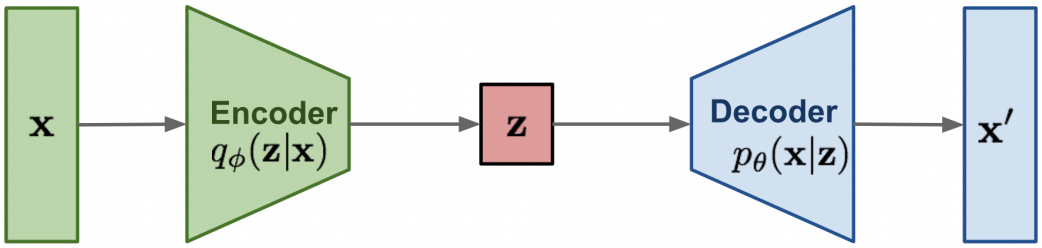
The training of GANs is formulated as a search for a Nash equilibrium in a zero-sum game, in which the discriminator seeks to minimize its cost function $J^{(D)}$, while the generator tries to maximize it. The original formulation of $J^{(D)}$ is defined as

$$J^{(D)}(\theta^{(D)}, \theta^{(G)}) = -\frac{1}{2}\mathbb{E}_{x\sim X} \log D(x) - \frac{1}{2}\mathbb{E}_z \log(1 - D(G(z))). \tag{2.1}$$

However, this theoretical definition is not very practical for training, so variations of the cost functions were proposed to make the training process more stable, but with the same underlying idea [62]. Even with these modifications, GAN training is dynamic and sensitive to nearly every aspect of its setup, so additional training strategies have been proposed to make them more stable. For example, Progressive GANs [63] have been proposed to improve the stability of GAN training, increasing iteratively the number of layers of the networks during the training epochs. Additionally, strategies to condition generation and gain control in the synthetic samples created have been proposed [64, 65, 66].



(a) GAN Architecture



(b) VAE Architecture

Figure 2.7: VAE and GAN architecture. Source [71].

Despite their difficult training process, GANs are one of the most widely used architectures for generative tasks today. One of the reasons for their popularity is the desirable properties of their latent space, which allows for the varied generation of new samples and their control [48, 64]. GANs have been successfully applied to a wide range of applications in Computer Vision, including image synthesis [65], image-to-image translation [67], and generation of images from segmentation masks [68, 69]. They have also been applied to video synthesis [70] and other applications in natural language processing and signal processing.

Variational Autoencoders (VAEs)

The Variational Autoencoder (VAE) is a generative model introduced in 2014 [72]. Traditional autoencoders can encode input samples into a latent representation in the bottleneck, which can then be used to reconstruct the original input. However, they lack variability in their generated samples, making them less suitable for content generation. The VAE is an extension of traditional autoencoders with a similar encoder-decoder structure, but it adds a probabilistic layer to the encoder. The encoder transforms the input samples into the parameters of a variational distribution, allowing for the generation of new content by sampling from the learned distribution in the latent space. The decoder takes a latent value sampled from the learned distribution and transforms it into the output space.

To train the VAE, the model maximizes the Evidence Lower Bound (ELBO) of the data given the model parameters, which involves computing the reconstruction loss and the Kullback-Leibler divergence between the learned distribution and a prior distribution [73]. Thus, the training problem is defined as the maximization of the ELBO, which is formulated as:

$$L_{\theta,\phi}(x) = \ln p_\phi(x) - D_{KL}(p_\phi(\cdot \mid x) \,\|\, q_\theta(\cdot \mid x)) \tag{2.2}$$

The VAE has played an important role in the current revolution of synthetic image generation, with important works such as the VQ-VAE [74], the basis of DALL·E [75],

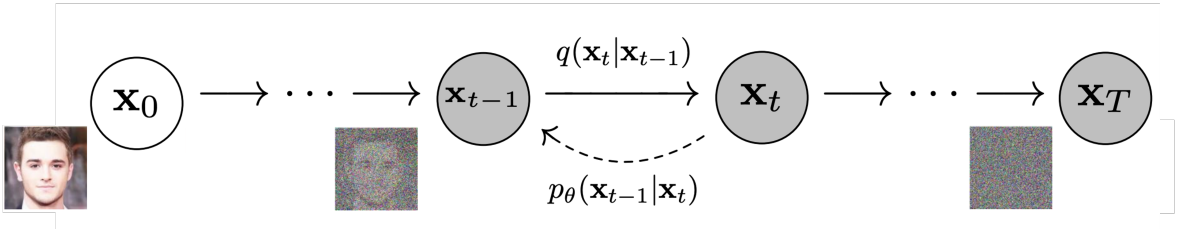


Figure 2.8: Forward diffusion process of a DDPM [51].

one of the text-to-image models that preceded the most popular architectures currently employed based on diffusion models [5, 6], in which is also a key component.

2.2.2 Diffusion Models

Diffusion Models, specifically Denoising Diffusion Probabilistic Models (DDPMs), are a type of latent generative model proposed in 2020 [51]. Their main purpose is to generate high-quality samples from a given distribution. DDPMs are based on an iterative process inspired by non-equilibrium thermodynamics, which involves applying a chain of diffusion steps that gradually introduce random noise to the input data. Although diffusion models based on this concept have been proposed before [76], the approach of the DDPMs proposed in [51] enabled the subsequent development of the recently popular text-to-image architectures.

Fordward diffusion process

Figure 2.8 illustrates the forward diffusion process of a DDPM, in which a data point $\mathbf{x}0 \sim q(x)$ from a distribution of input images is gradually transformed through a sequence of T diffusion steps until the image become indistinguishable due to the noise. In its original formulation [51], the diffusion process uses Gaussian noise with a predefined variance schedule $\{\beta_t \in (0,1)\}_{t=1}^{T}$. When $T \to \infty$, the result of the process, $\mathbf{x}_T$, is distributed as an isotropic Gaussian variable. With this formulation, for an arbitrary timestep t, the process q is distributed as

$$q(\mathbf{x}_t|\mathbf{x}_{t-1}) \sim \mathcal{N}\left(\sqrt{1-\beta_t}\mathbf{x}_{t-1}, \beta_t\mathbf{I}\right). \tag{2.3}$$

Employing a cumulative product of the timestep variances $\tilde{\alpha}_t = 1 - \prod_{i=1}^{t}(1-\beta_t)$, the distribution can be reparameterized in terms of the original input $\mathbf{x}_0$ as

$$q(\mathbf{x}_t|\mathbf{x}_{t-1}) \sim \mathcal{N}\left(\sqrt{1-\tilde{\alpha}_t}\mathbf{x}_0, \tilde{\alpha}_t\mathbf{I}\right). \tag{2.4}$$

Reverse diffusion process

If we could reverse the diffusion process and sample from $q(x_{t-1}|x_t)$, we could reconstruct the original sample from a noise input $\mathbf{x}_T$. However, this inverse process is intractable, thus it is not possible to estimate it directly. Instead, we can train a model p_θ to approximately perform the inverse process using the Algorithm 1 described in [51], in which the function ϵ_θ learns to predict the noise ϵ added in a timestep from $\mathbf{x}_t$.

Algorithm 1 DDPM training

repeat
  $\mathbf{x}_0 \sim q(\mathbf{x}_0)$
  $t \sim \text{Uniform}(\{1, \dots, T\})$
  $\epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
  Take gradient descent step on
    $\nabla_\theta \left\| \epsilon - \epsilon_\theta \left(\sqrt{1-\tilde{\alpha}_t}\mathbf{x}_0 + \sqrt{\tilde{\alpha}_t}\epsilon, t \right) \right\|^2$
until converged

Algorithm 2 DDPM sampling

$\mathbf{x}_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
for $t = T, \dots, 1$ **do**
  $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ if $t > 1$, else $\mathbf{z} = \mathbf{0}$
  $\mathbf{x}_{t-1} = \frac{1}{\sqrt{1-\beta_t}} \left(\mathbf{x}_t - \frac{\beta_t}{\sqrt{\tilde{\alpha}_t}} \epsilon_\theta(\mathbf{x}_t, t) \right) + \sqrt{\beta_t}\mathbf{z}$
end for
return x_0

The training algorithm, samples an initial data point x_0 from the input distribution $q(x_0)$ and a random timestep t between 1 and T. Then, it samples a random vector ϵ from the noise distribution. It takes a gradient descent step on the difference between ϵ and ϵ_θ, evaluated at $(\sqrt{1-\tilde{\alpha}_t}\mathbf{x}_0 + \sqrt{\tilde{\alpha}_t}\epsilon_t, t)$. The sampling process (Algorithm 2), generates a sample $\mathbf{x}_T$ from the noise distribution and performs a reverse diffusion process over T timesteps using ϵ_θ to obtain the reconstructed image.

In practice, several variations of the original proposal have been developed to address specific challenges and use cases. For instance, there have been introduced mechanisms for conditional generation [4] and the use of text as input [5, 6]. In addition, other classes of noises, schedules, and losses are employed to improve the convergence of the inverse process modeling the noise as an ODE [77]. However, diffusion-based models that have recently gained popularity, such as DALLE·2 [5] or Stable Diffusion [6], are based on the same underlying idea as above.

Stable Diffusion

Stable Diffusion [6] is a widely used latent diffusion model for image generation. Its open-source nature has allowed for the emergence of many research works exploring the architecture and a wide range of applications. The model has gained significant traction since its launch, thanks to the release of several pre-trained versions. In this work, we focus on the Stable Diffusion 2.0 version[1], a 1.1 billion parameters version pre-trained on the LAION 5-billion image dataset [46]. The architecture is highly versatile, with variations that allow it to perform image manipulation tasks such as image inpainting, outpainting, and super-resolution. However, in this work we focus on the original text-to-image sampling mode.

The architecture comprises three main components: a deep language model to generate word embeddings, a VAE that encodes and decodes latent vectors into images, and a time-conditional denoising network responsible for denoising the latent models in the diffusion process. Additionally, some versions of the model include a final classifier to filter out Not Safe for Work (NSFW) content. A diagram illustrating the components of the model can be seen in Figure 2.9.

The deep language model τ_θ, generally a CLIP model [78], is responsible for transforming the input text prompts in natural language into word embeddings. These embeddings are vectors of fixed size in a more appropriate space in which there are preserved structures that maintain semantic relationships between words. These vectors are then used in the diffusion process to condition the generation by guiding the generated images by means of attention mechanisms. Generally, this language compo-

[1]Stable Diffusion 2.0: huggingface.co/stabilityai/stable-diffusion-2 (Accessed June 2023).

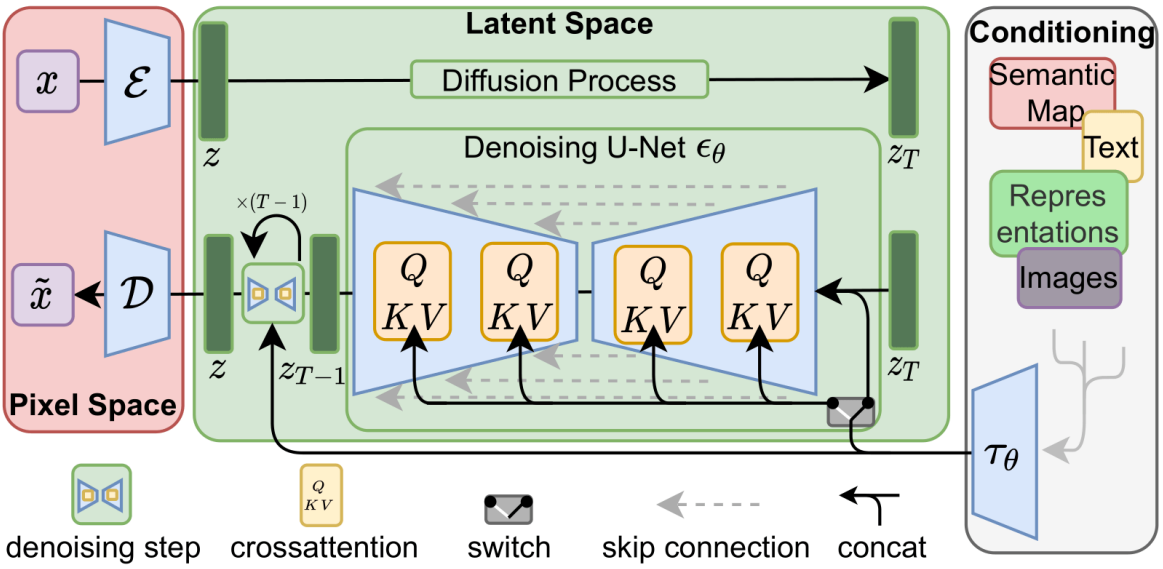


Figure 2.9: Stable Diffusion architecture. Source [6].

nent is pre-trained separately and its weights are not modified during the training of the diffusion model [79].

The conditional denoiser network ϵ_θ, generally a conditional U-NET as in Stable Diffusion 2, is in charge of performing the iterative process of inverse diffusion. Unlike the original U-NET [26] (see Figure 2.3), a variation is used in which cross-attention layers are introduced, in charge of introducing the information of the text embeddings to guide the denoising process. Later, in Section 2.3.2, we dwell on these attention mechanisms, since we extract from them the information of the semantic segmentation masks during the generation of synthetic images.

The VAE [72] is formed by an encoder $\mathcal{E}$ and a decoder $\mathcal{D}$ (see Figure 2.7b). The encoder $\mathcal{E}$ is in charge of transforming images to the latent space of the network where the diffusion process is applied. Although the encoder is mainly used for training the architecture, it can be used in the sampling process to initialize the diffusion process with an image. The VAE decoder $\mathcal{D}$ transforms the U-NET output after applying the inverse diffusion process to convert the denoised latent variable z into an image.

Challenges in Synthetic Data Generation using Stable Diffusion

Despite the progress made in text-to-image generation, there are still several challenges that need to be addressed. One major challenge is the difficulty of generating images that accurately match the input text. This is particularly challenging when dealing with too specific concepts or complex scenes that require a high level of visual understanding.

Several methods have been proposed to address this challenge, including the use of additional constraints on the architecture to respect a predefined structure, such as a given segmentation map, during image generation [80]. Other works focus on adjusting the attention generated on the cross-attention layers during image generation to mitigate the model from discarding part of the text information given as input [81].

Improving the accuracy of text prompts is also an important issue that has been addressed in recent works. The term *prompt engineering* has become popular to describe the optimization of queries to improve image generation [82]. Among the lines

of research focused on automating this process, we can highlight works focused on obtaining queries that generate images as similar as possible to a given image [83] or the use of reinforcement learning to train systems capable of generating better text prompts [84]. These methods can significantly improve the generation of high-quality images from text without modifying the text-to-image architecture.

All these challenges motivate the development of methods for the explainability of latent diffusion image generation. A better understanding of the underlying processes allows for the creation of text prompts that are better suited to generate images, as well as the development of more sophisticated techniques to interact with these models. These topics are explored in more detail in Section 2.3.

2.3 Explainability

The rise of Deep Learning has paved the way for remarkable advances in Computer Vision, enabling us to solve tasks that would have been difficult to tackle in the past [1]. However, the increasing complexity of these models and their widespread use across various domains have brought new challenges. One such challenge is the lack of transparency and interpretability in these models, which can hinder their potential applications in real-world scenarios.

Traditionally, these models have been treated as a black box, with little attention paid to their internal workings. However, this approach presents significant limitations, as it becomes difficult to explain how the model arrived at a particular decision. Furthermore, the lack of transparency in these models poses legal and ethical risks, as accountability becomes challenging when the decision-making process is not transparent [9].

To address these issues, the field of explainable Artificial Intelligence (*XAI*) has emerged, aiming to develop methods that make these models more transparent and interpretable [10]. This is particularly relevant in Computer Vision [85], where interpretability and explainability can be fundamental for understanding the inner workings of models and addressing issues such as bias and fairness [86].

While the terms explainability and interpretability are often used interchangeably due to their close relation, they are distinct tasks. Interpretability measures the extent to which a model can associate a cause with an effect, based on its input and output variables. For instance, *Shapley values* can be used to assign a numeric weight to each input variable, reflecting its influence on an output prediction [87]. On the other hand, explainability is a broader task that pertains to the ability of a model's decisions to be understood by humans, including the ability to describe the model's behavior and the factors that influence it [9]. For example, an explanation of a model's behavior might involve identifying specific features in an input image that caused the model to produce a particular output prediction [88]. Alternatively, it could entail identifying which features activate a certain layer or node of a neural network, providing insights into the hierarchy of features used by the network [89], and identifying potential biases in the model.

In this section, we explore the issue of explainability in Computer Vision, focusing specifically on diffusion models. We begin by reviewing the concept of explainability in Computer Vision and the main methods currently employed. Next, we delve into the specific challenges related to the explainability of diffusion models, discussing recent developments and identifying potential avenues for future research. Finally, we discuss

(a) Original (b) CAM of “Nurse” (c) CAM of “Doctor”

Figure 2.10: Class activation map (Grad-CAM) of a biased model. (a) A stock photo image labeled as “Doctor”. (b) Areas on which a biased model relies to predict the “Nurse” class (c) and “Doctor”. Source [90].

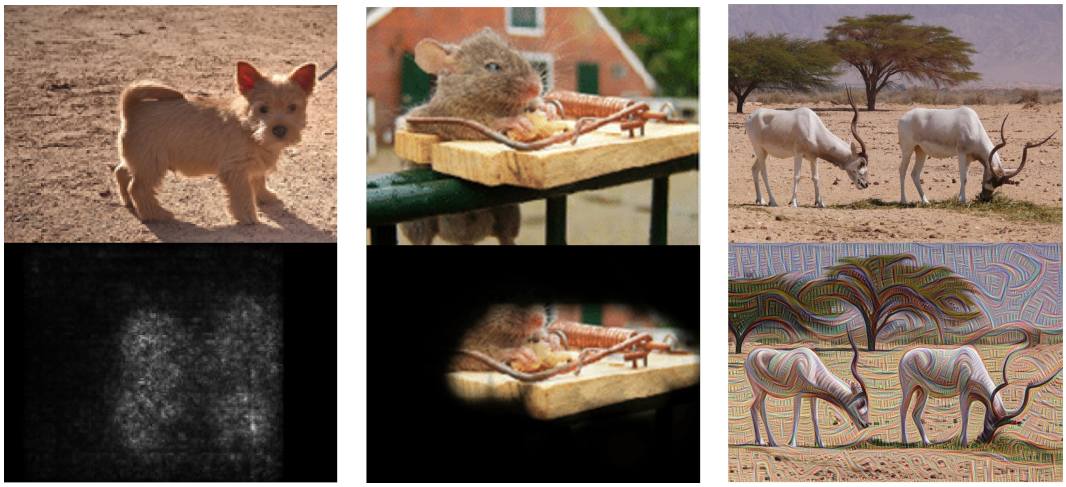
the relevance of explainability in the scenario of synthetic image generation for semantic segmentation, highlighting the significance of transparent and interpretable models in real-world applications.

2.3.1 Explainability in Computer Vision

Explainability methods for Computer Vision are becoming increasingly important due to regulatory requirements and the need for increased reliability [9]. Computer vision models are particularly susceptible to acquiring biases present in the datasets on which they are trained [86, 94]. Figure 2.10 provides a striking example of this phenomenon, where an image labeled as “Doctor” (2.10a) is accompanied by the regions on which a biased model relies to classify the image as ”Nurse” (2.10b) and “Doctor” (2.10c), both sex-agnostic classes [90]. Notably, the biased model places a disproportionate weight on the long hair region to classify the image as “Nurse”, and on the stethoscope to classify it as ”Doctor“. Such biases can result in harmful outcomes in various scenarios and may adversely impact the model’s performance. However, explainability methods can help to detect and mitigate the impact of these biases.

To achieve explainability in Computer Vision, researchers use established methods to interpret convolutional models. These methods can be classified into three main approaches: gradient-based attribution methods, perturbation-based methods, and techniques for feature visualization using optimization techniques.

Gradient-based attribution methods are commonly used in Computer Vision to determine the contribution of each input feature to the final model prediction through the backpropagation of gradients. One such method is saliency maps [91], which calculate the gradient of a class score with respect to the input image, highlighting the aspects that most influence the image’s classification as that class (Figure 2.11a). Other popular methods, such as Class Activation Maps (CAM) [95] and Gradient-weighted Class Activation Mapping (GradCAM) [90], use global average pooling in conjunction with gradient backpropagation to create class activation maps. These maps are heatmaps that identify the most relevant areas for classifying an image, as illustrated in Figure 2.10. These attribution maps can be used to detect potential biases in classification models and can even be used to perform object location and semantic segmentation of the object being classified.



(a) (b) (c)

Figure 2.11: Examples of explainability methods for Computer Vision. (a) Image of a dog and the corresponding Saliency Map [91] of the class "Dog". (b) Image of a mouse in a trap and a masked version using Extremal perturbations [92] with the most discriminative part to classify the image as "mouse trap". (c) Feature optimization to maximize a convolutional layer response [93] using an image of a savanna as prior. The processed image contains more highlighted edges, indicating that the layer is activated in the presence of this type of low-level feature. Source [91, 92, 93].

Perturbation-based methods involve masking or altering input features to observe the effect on the model output. Examples of perturbation-based methods are Minimal Image Representation [96] and Extremal Perturbations [92], which attempt to learn a mask that removes parts of an image that have minimal impact on the output of a model, as illustrated in Figure 2.11b. In contrast, other methods, such as Meaningful Perturbations [97], try to blur or mask the parts of an image that most influence the model prediction.

Feature visualization methods are focused on understanding the behavior of neural networks by finding inputs that activate certain parts of the network through optimization. One commonly used technique is activation maximization, which involves maximizing the activation of a saliency map [91]. However, to make the obtained inputs more interpretable, additional methods like Feature Inversion [98] or Inceptionism [93] utilize images as priors to generate more meaningful representations. By optimizing an input image that amplifies the activation of specific neurons in a neural network, we can obtain images that highlight the features that those neurons are encoding. This enables us to gain insights into whether a layer is filtering a specific texture or a semantic concept, as shown in Figure 2.11c.

Explainability research in Computer Vision is currently a highly active field, with new techniques used in other domains still being explored, such as the use of counterfactual explanations [99], adversarial attacks [100], or causal reasoning [101]. While current methods have shown promise in detecting and mitigating biases in Computer Vision models, their application still poses several challenges. For example, there is a lack of consistency across techniques, which makes it difficult to compare and evaluate their interpretations. Measuring the effectiveness of these methods in practice is also complicated, especially when considering their impact on downstream tasks. Addition-

ally, most established techniques are based on image classification, and extending them into generative tasks is still an emerging issue.

In recent years, explainability research has extended to other domains, such as natural language processing, where models such as transformers have been shown to achieve state-of-the-art performance on a variety of tasks but can be difficult to interpret [102, 103]. One approach to interpretability in language models is to use attention mechanisms to visualize the words or phrases that the model is focusing on during prediction [102].

Overall, explainability methods are critical in ensuring the reliability and trustworthiness of machine learning models. As machine learning continues to be deployed in increasingly complex and sensitive applications, the development of effective and practical explainability methods will be essential for achieving transparency and accountability.

2.3.2 Explainability of Diffusion Models

The emergence of diffusion models for image generation is a relatively recent development, and research on methods for their explainability is still an emerging topic. Current research in this area focuses on text-attribution methods, which attempt to answer the question of which parts of an image are influenced most by each word in the text prompt. These methods aim to create heat maps of the influence of each word in the input text on the resulting image.

Unfortunately, early investigations in this direction found that methods based on gradients and perturbations cannot be adapted to diffusion models due to the iterative process involved in the diffusion process. Gradient methods require a backpropagation pass for every pixel for all T time steps, which makes computation intractable and the process highly unstable. Perturbation-based methods result in significantly different images even with minor perturbations, making them unsuitable for diffusion models [20].

As a result, the main methods being developed for explainability in diffusion models are based on ideas from natural language processing, where attention to words indicates lexical attribution [103]. These methods exploit the conditional mechanisms based on cross-attention, where the text guides the image generation. One such method is the Diffusion Attentive Attribution Maps (DAAMs) [20], whose maps are based on these attention mechanisms.

Other methods being explored for explainability in diffusion models involve modifying the attention process to highlight the attention of the tokens that need to be explained [81], or by perturbing the attention [104].

However, since this is still an emerging topic, many aspects remain to be investigated, and there are many potential issues that may expand the applications of these methods. In this work, we focus on exploring DAAMs for the extraction of semantic segmentation masks during synthetic image generation. As the DAAM method is important for our work, we provide a more detailed explanation of the process in Chapter 3.

Chapter 3

Proposed methods

This chapter introduces the theoretical framework that forms the core of this work, focused on the use of the attentions within LDMs for explainability purposes and their alignment with different objects in synthetically produced urban scenes.

Initially, in Section 3.1, we present the theory behind the DAAM explainability method [20]. This method allows the creation of attribution maps that relate the input text to the images generated by an LDM, further enabling the generation of segmentation masks. By attributing the influence of input tokens in the generated images, DAAM provides a valuable tool for understanding the connection between textual prompts and visual output.

Proceeding further, in Section 3.2, we propose an extension of this explainability method, referred to as Open Vocabulary DAAM. This extension overcomes the limitation of the original DAAM by enabling the construction of attribution maps for any given text, not restricted to the tokens used during image generation. Open Vocabulary DAAM enhances the flexibility and applicability of the method, allowing for the exploration of diverse textual prompts and their influence on the generated images.

Finally, in Section 3.3, we propose a methodology for optimizing an input text embedding to maximize the Intersection over Union (IoU) of DAAM-generated masks. This optimization process aims to align the generated masks more closely with the desired semantic segmentation masks, enhancing the fidelity and accuracy of synthetically generated ground truth.

Through this examination, we aim to cover the DAAM theoretical framework, its potential pitfalls, and paths for advancement. This foundation sets the stage for the empirical explorations presented in Chapter 4, where we evaluate and analyze the practical implications of these methods on the task of semantic segmentation in urban scenes.

3.1 Diffusion Attentive Attribution Maps

The Diffusion Attentive Attribution Maps (DAAM) method represents a pioneering work in enhancing the explainability of text-to-image diffusion models [20]. This technique facilitates the attribution of individual input words' influence on the synthetic images generated by the model.

By leveraging the attention mechanisms activated during the diffusion process, DAAM constructs heatmaps corresponding to each token used as input. These attention-based heatmaps align with semantically significant image areas, capturing not only pri-

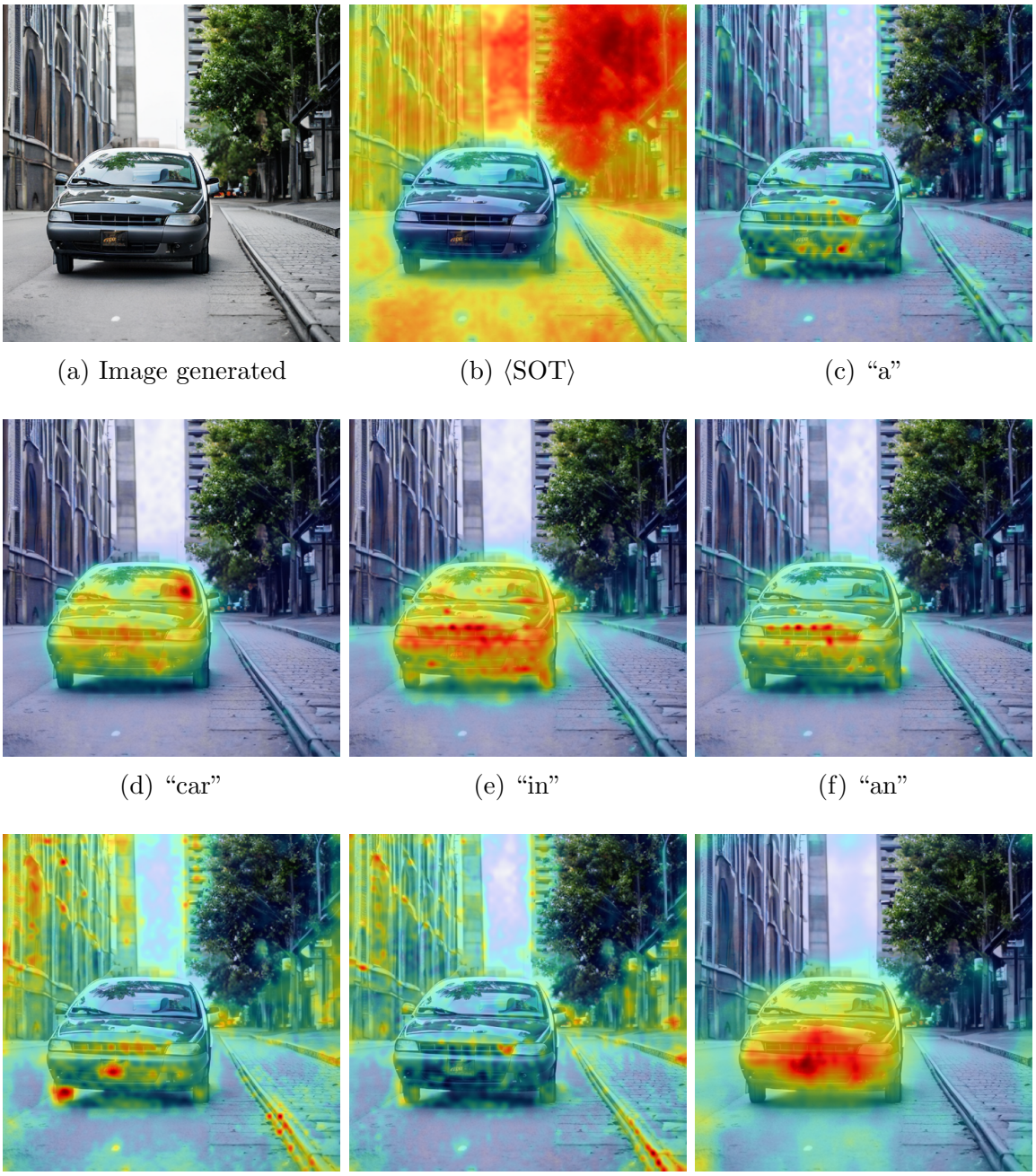
mary objects but also abstract concepts embedded within the image, such as adjectives, verbs, or semantic relationships between words [20].

For example, Figure 3.1 presents an image generated with Stable Diffusion [6] using the text prompt "A car in an urban environment" (Figure 3.1a), accompanied by DAAMs for each token in the prompt. This prompt was chosen for this example after practical tests with various prompts, where it generated examples with a main object in the image within urban scenes, which aligns with the research context of this work. Each heatmap corresponds to a specific word in the input text, including the starting and ending special tokens, providing insights into how each word influences different regions of the generated image. The following observations can be made from the heatmaps:

- ⟨SOT⟩ (3.1b): The heatmap primarily accumulates information from the image background, such as the city street and building. This attention pattern is common for the onset token, which tends to gather attention from regions not influenced by other tokens [20].
- "a" (3.1c): The heatmap exhibits a dispersed pattern across the entire image, suggesting that the word "a" influences various regions without a specific focus.
- "car" (3.1d): The heatmap precisely outlines the shape of the car in the center of the image, highlighting the strong influence of the word "car" on that object.
- "in" (3.1e): The heatmap focuses on the region of the car, indicating an semantic association between the word "in" and "car".
- "an" (3.1f): The heatmap centers on the shape of the car but contains more noise, suggesting some ambiguity or uncertainty in the attribution.
- "urban" (3.1g) and "environment" (3.1h): The heatmaps draw attention from various regions of the image, particularly the background building, indicating a strong association between these words and the elements of the environment.
- ⟨EOT⟩ (3.1i): The heatmap complements the onset token's heatmap (⟨SOT⟩), with a specific focus on the car's location during the image generation process.

This example illustrates the information provided by DAAM, revealing how each word influences distinct regions of the generated image. These heatmaps exhibit comparable performance to unsupervised learning techniques in segmenting primary objects within the image [20]. By analyzing these heatmaps, we gain a deeper understanding of the text-to-image generation process and establish a foundation for future analysis and enhancements of diffusion models through the examination of their denoising subnetwork attentions.

When considering the denoising subnetwork ϵ_θ within a Latent Diffusion Model (LDM), there are various architectural choices available. However, U-Nets [26] have emerged as a popular option, due to their strong capabilities in image segmentation, and is the subnetwork used in Stable Diffusion (refer to Section 2.2.2). U-Nets consist of downsampling convolution blocks that preserve local context and upsampling deconvolutional blocks responsible for restoring the output to its original size [20]. This iterative transformation removes noise from a latent vector, ultimately producing an output image through the utilization of a Variational Autoencoder (VAE) [6].



(a) Image generated (b) ⟨SOT⟩ (c) “a”

(d) “car” (e) “in” (f) “an”

(g) “urban” (h) “environment” (i) ⟨EOT⟩

Figure 3.1: An image generated using Stable Diffusion with the text “A car in an urban environment” along with overlayed DAAMs for each token.

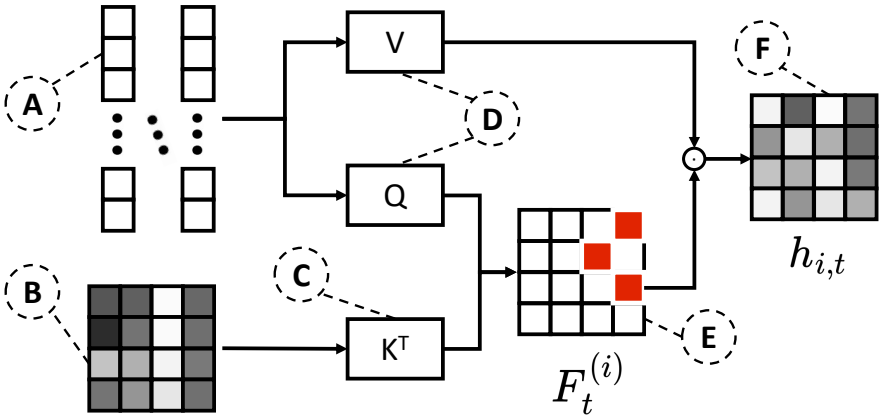


Figure 3.2: Illustration of the cross-attention mechanism in an U-NET block: (A) the text embedding X; (B) the unconditioned hidden state $\hat{h}_{i,t}$;(C) the attention key $W_k^{(i)}\hat{h}_{i,t}$; (D) the attention query $W_q^{(i)}X$ and value $W_v^{(i)}X$; (E) the attention matrix $F_t^{(i)}$ (represented as red squares the activated attentions); (F) and the U-NET block output $h_{i,t}$ conditioned by the text embedding.

The denoising process operates in a lower-dimensional latent space, represented by 2D vectors denoted as $\ell_t \in \mathbb{R}^{w\times h}$ at a specific time-step t. In each iteration of this process, the U-NET's downsampling blocks generate a series of K intermediate states $\{h_{i,t}^{\downarrow}\}_{i=1}^{K}$. These hidden states, with dimensions $\lceil \frac{w}{c^i} \rceil \times \lceil \frac{h}{c^i} \rceil$, gradually decrease in size as they pass through the downsampling blocks. The reduction factors c^i are typically set to 1, 2, 4, and 8. Following the downsampling, the upsampling blocks scale up the downsampled hidden state $h_{K,t}^{\downarrow}$ to $\{h_{i,t}^{\downarrow}\}_{i=1}^{K} \subset \mathbb{R}^{\lceil \frac{w}{c^i} \rceil \times \lceil \frac{h}{c^i} \rceil}$, effectively restoring the original dimension of the vector ℓ_t at the network's input. By iteratively applying this process, an approximation of the reverse diffusion process is achieved, enabling the reconstruction of an image from a noisy vector.

To incorporate textual information into the process, U-Net blocks from Stable Diffusion utilize multi-headed cross-attention layers that follow the attention mechanism proposed in the transformers architecture [6, 105]. This mechanism is illustrated in Figure 3.2. Specifically, in the case of a downsampling block, the U-Net takes a text embedding $X := [x_1; \cdots ; x_{l_W}] \in \mathbb{R}^{l_C \times l_W}$ consisting of l_W tokens, along with the output of the fully convolutional layers of the block before conditioning, denoted as $\hat{h}_{i,t}^{\downarrow}$. The conditioned output $h_{i,t}^{\downarrow}$ of the block is then calculated as follows:

$$\begin{aligned} h_{i,t}^{\downarrow} &= F_t^{(i)}\left(\hat{h}_{i,t}^{\downarrow}, X\right) \cdot \left(W_v^{(i)\downarrow} X\right), \\ F_t^{(i)}\left(\hat{h}_{i,t}^{\downarrow}, X\right) &= \text{softmax}\left((W_q^{(i)\downarrow}\hat{h}_{i,t}^{\downarrow})(W_k^{(i)\downarrow}X)^T/\sqrt{d}\right). \end{aligned} \tag{3.1}$$

In Equation 3.1, the matrices $W_k^{(i)\downarrow}$, $W_q^{(i)\downarrow}$, and $W_v^{(i)\downarrow}$ are projection matrices with $l_H^{(i)}$ attention heads. These matrices transform the text embedding X and $\hat{h}_{i,t}^{\downarrow}$ into vectors with dimensions $\lceil \frac{w}{c^i} \rceil \times \lceil \frac{h}{c^i} \rceil \times l_H^{(i)}$. Furthermore, the attention scores $F_t^{(i)\downarrow}$ have dimensions $\lceil \frac{w}{c^i} \rceil \times \lceil \frac{h}{c^i} \rceil \times l_H^{(i)} \times l_W$, which are obtained by concatenating the attention generated by each of the l_W tokens.

To clarify the notation, we use $F_{t,k,h}^{(i)\downarrow}[x, y]$ to denote the attention array generated by the k-th token of the h-th multi-attention head in the i-th downsampling block at time $t \in [1, T]$. Similarly, in the case of upsampling blocks, we refer to their outputs

as $h^{\uparrow}_{i,t}$ and their respective attention arrays as $F^{(i)\uparrow}_{t,k,h}[x,y]$. For brevity, when discussing an attention array that can be either from a downsampling or an upsampling block, we omit the arrow index $\downarrow$ or $\uparrow$.

In Equation 3.1, it is important to note that the softmax function is applied token-wise, independently for each attention head. Specifically, for each spatial coordinate (x, y) and each head h, we have the following constraint:

$$\sum_{k=1}^{l_W} F^{(i)}_{t,k,h}[x,y] = 1. \tag{3.2}$$

This constraint ensures that, for each head and spatial point (x, y), the attention is distributed among the different tokens.

To combine the attention arrays $F^{(i)}_{t,k,h} \in \mathbb{R}^{\lceil \frac{w}{c^i} \rceil \times \lceil \frac{h}{c^i} \rceil}$ from different blocks, interpolation is performed to unify them to the dimension of the original latent space $w \times h$. Due to the fully convolutional nature of the network, these arrays retain the same spatial distribution as the generated images. Therefore, scaling the attentions to different resolutions allows for their aggregation. We denote these scaled arrays as $\tilde{F}^{(i)}_{t,k,h} \in \mathbb{R}^{w \times h}$. In the original formulation of DAAM [20], the authors propose the use of bicubic interpolation for this scaling. However, in our preliminary experiments, we found that similar results in terms of IoU can be achieved using bilinear interpolation.

To create a heatmap from the scaled attention arrays, they are aggregated across all timesteps and heads using the following equation:

$$D^{\mathbb{R}}_k[x,y] := \sum_{t,i,l} \tilde{F}^{(i)\downarrow}_{t,k,l}[x,y] + \tilde{F}^{(i)\uparrow}_{t,k,l}[x,y]. \tag{3.3}$$

The resulting $D^{\mathbb{R}}_k$ represents a soft heat map, where higher values indicate a stronger attribution to the token k (see Fig. 3.1). To convert this soft heat map into a binary mask, a threshold τ is applied relative to the maximum value of the attention map:

$$D^{\mathbb{I}_\tau}_k[x,y] := \mathbb{I}\left(D^{\mathbb{R}}_k \;\geq\; \tau \cdot \max_{x,y} D^{\mathbb{R}}_k \right). \tag{3.4}$$

Here, $\mathbb{I}(\cdot)$ is the indicator function, and $\tau \in [0, 1]$. The resulting binary mask highlights regions where the token k is attributed. Figure 3.3 provides an illustration of the DAAM construction process for a token.

DAAM serves as a powerful explainability technique for LDM, providing insights into how textual input information is synthesized and enabling the construction of semantic segmentation mechanisms. However, there are limitations to using these attention maps as ground truth for semantic segmentation tasks. For instance, binary masks depend on the threshold τ, which requires further study to determine optimal values. Another limitation is that the original formulation of DAAM [20] does not consider generating heatmaps for tokens that are not present. This limitation prevents us from extracting masks for objects in the image, such as a tree in an urban environment, if it was not mentioned in the initial text.

Furthermore, we found that the influence area of a token does not always align with the annotation criteria of the desired semantic segmentation masks. This discrepancy arises from the semantic relationships learned by the network and the influence of the word on other parts of the scene.

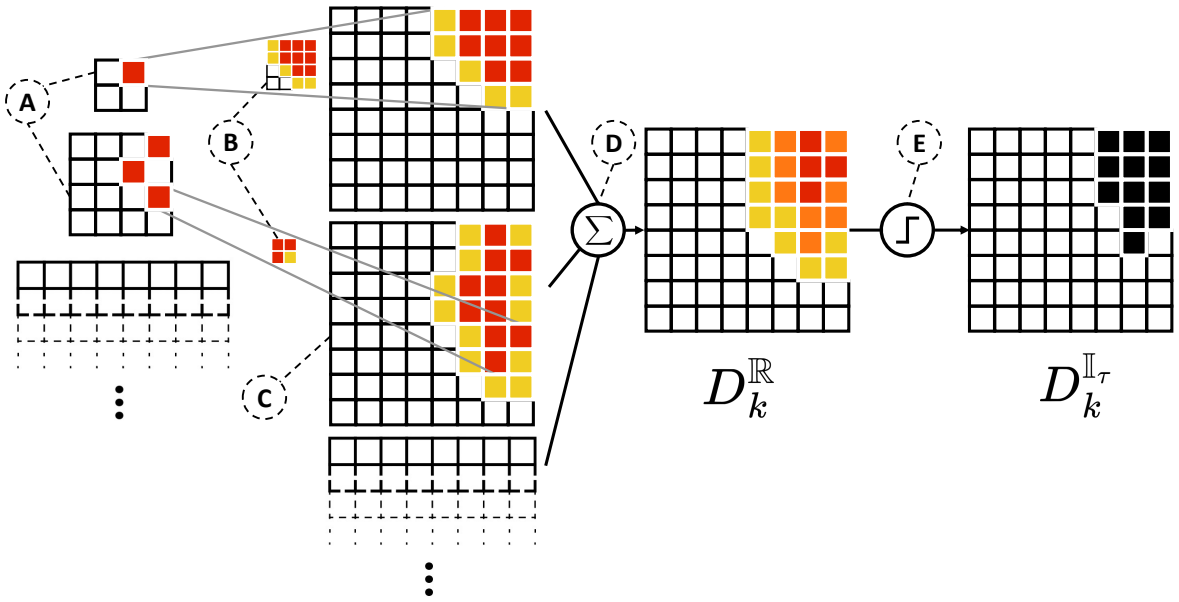


Figure 3.3: Illustration of computing DAAM for some word: (A) the multiscale attention arrays from Eqn. 3.1; (B) the bicubic interpolation (C) resulting in expanded maps; (D) summing the heatmaps across the layers, as in Eqn. 3.3; (E) and the thresholded maps. Source [20].

These limitations make it challenging to use DAAM beyond its role as an explanation tool, especially for semantic segmentation tasks. In this work, we propose several modifications to extend the method and overcome these limitations in Sections 3.2 and 3.3.

3.2 Open Vocabulary-based DAAM

One of the main limitations of DAAM [20], beyond its use as an explainability technique, is its restriction to generating attribution maps only for tokens present in the text prompt used to generate a sentence. This limitation becomes particularly significant when applying the method to generate semantic segmentation masks, as it would require all words to be present in the text that generated the image. In the example shown in Figure 3.1, where an image is generated with the prompt "A car in an urban environment," it can be observed that besides the car, the urban scene also contains a sidewalk, a tree, and a building, for which the original method cannot generate attribution maps.

Although these tokens have not explicitly influenced the generation process, they are semantically related to elements in the generated scene. Hence, it raises the question of whether the attention they would have generated aligns with the regions corresponding to the respective objects in the image. The use of DAAM in semantic segmentation motivates the extension of the method in this regard. Furthermore, it allows for a deeper exploration of the model's internal workings, uncovering learned semantic relationships and biases.

In this section, we propose two modifications to the original formulation of DAAM [20], which expand its capabilities by incorporating attention maps for out-of-vocabulary tokens.

First, we introduce the "Open Vocabulary DAAM" modification. In this approach, the original DAAM process is divided into two steps: generating the image and capturing the hidden states, followed by generating the attention matrices using an arbitrary text embedding. This modification enables the generation of heatmaps for arbitrary prompts and their tokens. However, it is important to note that the attention captured by each token is relative to the attention captured by other tokens due to the token-wise softmax operation. One limitation of this first approach is that to generate an attention heatmap for a specific word, it will be necessary to include it in a context sentence with other tokens that capture the attention of the non-interested regions.

Furthermore, we present the "Linear Open Vocabulary DAAM" modification. In this variant, we remove the softmax operation and directly aggregate the attention linear projections. This modification aims to explore the impact of removing the softmax operation on attention patterns and the resulting heatmaps.

We describe the procedures for generating attention maps from arbitrary sentences, addressing the challenges associated with tokens not present in the prompt.

3.2.1 Open Vocabulary DAAM

In this subsection, we propose a modification to the construction of DAAM[20] to enable the evaluation of attention heatmaps for text prompts different from those used to generate the image, which guided the reverse diffusion process. To achieve this, we introduce a two-phase approach for constructing the attention maps.

In the first phase, referred to as the "collection" phase, a text prompt $X := [x_1; \cdots ; x_{l_W}]$, referred to as the generator text embedding, is provided. An image is generated using the LDM, and during this process, the outputs of the downsampling and upsampling blocks of the U-NET prior to conditioning are saved. These outputs, denoted as $\{\hat{h}^{\downarrow}_{i,t}\}_{i=1}^{K}$ and $\{\hat{h}^{\uparrow}_{i,t}\}_{i=1}^{K}$ in equation 3.1, contain spatial information about the generated image. In the original DAAM formulation [20], these hidden states are used as input to the cross-attention layers of the U-NET, which incorporate the textual information and generate the attention arrays $F_t^{(i)}$ for constructing the DAAMs.

In the second phase, utilizing the collected hidden states $\hat{h}_{i,t}$, we construct the attention arrays based on a new text prompt. This text prompt can have different lengths and contain different tokens. Given the new text embedding $X' := [x_1; \cdots ; x_{l_{W'}}] \in \mathbb{R}^{l_C \times l_{W'}}$, referred to as the contextual text embedding, we generate the attribution arrays as follows:

$$F^{(i)}_{X,t}(X') = \text{softmax}\left((W_q^{(i)\downarrow}\hat{h}^{\downarrow}_{i,t})(W_k^{(i)\downarrow}X')^T/\sqrt{d}\right). \tag{3.5}$$

Here, $\hat{h}^{\downarrow}_{i,t}$ represents the unconditioned hidden states collected during the previous step. This computation of the attention arrays is essentially the same as the one used in the original formulation of DAAM [20], utilizing the hidden states from the generator text embedding and the contextual embedding as keys in the cross-attention. The notation $F^{(i)}_{X,t}$ is used to refer to this attention array, indicating that the hidden states $\hat{h}_{i,t}$ were generated using the text embedding X. Similar to Equation 3.1, the softmax function is applied token-wise, independently for each attention head (see the constraint in Equation 3.2).

Once the new attention arrays are constructed, we can obtain the DAAMs for the text prompt X' following the same methodology as in the original method [20].

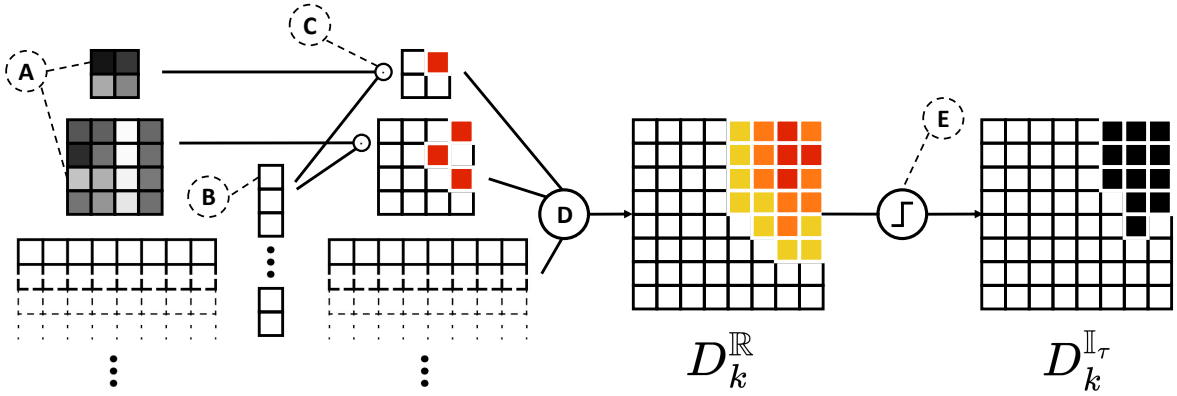


Figure 3.4: Illustration of computing DAAM for open vocabulary: the hidden states $W_q^{(i)}\hat{h}_{i,t}^{\downarrow}$ stored during the reverse diffusion process (A); the text embedding (B); the cross-attention matrices (C) used to compute the DAAM (D); and the thresholded maps (E). Modification of figure from [20].

Firstly, we rescale the spatial dimensions of the attention arrays, denoted as $\tilde{F}_{X,t}^{(i)}$, to have dimensions $w \times h \times l_H^{(i)} \times l_{W'}$.

After constructing the rescaled attention arrays, we can generate the new heatmap based on attention by aggregating the heads from all blocks and timesteps of the process. Specifically, we have:

$$D_{X,k}^{\mathbb{R}}[x,y]\,(X') := \sum_{t,i,l} \tilde{F}_{X,t,k,l}^{(i)\downarrow}[x,y]\,(X') + \tilde{F}_{X,t,k,l}^{(i)\uparrow}[x,y]\,(X')\,. \tag{3.6}$$

The entire process is illustrated in Figure 3.4. Similar to the original version of DAAM, these modifications allow for the generation of attention heatmaps for arbitrary text prompts, providing insights into the attention patterns in the image.

Figure 3.5 illustrates the results of generating attention maps on the image with the prompt "a car in an urban environment" (see Figure 3.1a) for the tokens "tree," "building," and "sidewalk." As a context phrase X' for extracting the attention, the phrase "a car and a ⟨token⟩ in an urban environment" has been used, where ⟨token⟩ represents each of the three words depicted in Figures 3.5a to 3.5c. The figure shows that despite not being present in the generator text prompt, the tokens are indeed related to semantically relevant areas. The attribution of the "tree" token is primarily focused on the tree's leaves, "building" is attributed to the buildings in the background, and "sidewalk" is attributed to the ground in the image. Therefore, it appears that the attention information successfully extends to tokens that were not present during the generation process.

This modification has significant potential as it effectively extends the attribution of different words in an image generated by an LDM to any arbitrary text prompt. However, since the attention is relative to the other tokens in the context phrase, studying the attribution of a single token becomes complex. This limitation arises when generating segmentation masks for objects in the scene. Nonetheless, it can still be valuable for other applications, such as understanding how modifications to a prompt influence image generation, enabling the construction of better prompts for LDMs.

(a) "tree" (b) "building" (c) "sidewalk"

Figure 3.5: Example of Open Vocabulary DAAMs: An image generated with the prompt "A car in an urban environment" with overlaid attention maps for the tokens "tree" (3.5a), "building" (3.5b), and "sidewalk" (3.5c). The attention maps were extracted using the contextual prompt "A car and a ⟨token⟩ in an urban environment," where ⟨token⟩ represents each of the depicted words.

3.2.2 Linear Open Vocabulary DAAM

One of the main limitations of the first proposed modification, the "Open Vocabulary DAAM," is the requirement of defining a context phrase to extract attention maps for tokens that are not present in the prompt used to generate the image. For example, in Figure 3.5, attention masks for "tree," "building," and "sidewalk" were generated using the context phrase "A car and a ⟨token⟩ in an urban environment". This dependence on a context phrase makes it challenging to use this extension as an explainability method for understanding the relation of standalone words in the image, as well as for generating attention masks for individual objects in synthetic images. The attribution generated by this method strongly relies on how the contextual phrase is constructed. Thus, there is a need to find a way to extract attention without considering the attention of other tokens.

After conducting several preliminary experiments to study the role of the softmax function in the attribution of different tokens in the phrase, it was found that aggregating the attentions of a single token without applying softmax still resulted in attention heatmaps focused on the same areas as when using a tailored context phrase. Therefore, the second proposed variation, called "Linear Open Vocabulary DAAM," is a modification of the previous method without the application of softmax.

To generate a Linear DAAM, the first step is to perform a collection phase. This is done by generating an image with a text prompt X and capturing the hidden states $\hat{h}_{i,t}$. This collection phase is identical to the one proposed in the previous extension (see section 3.2.1).

Once the hidden states have been captured, we proceed to compute the attention of an individual token without applying the softmax function, as defined in Equation 3.5. Specifically, for a given token embedding x for which we aim to generate an attribution map, the attention is calculated as:

$$L_{X,t}^{(i)}(x) = (W_q^{(i)\downarrow}\hat{h}_{i,t}^{\downarrow})(W_k^{(i)\downarrow}x)^T . \tag{3.7}$$

In this context, the attention array $L_{X,t}^{(i)} \in \mathbb{R}^{\lceil \frac{w}{c^i} \rceil \times \lceil \frac{h}{c^i} \rceil \times l_H^{(i)}}$ corresponds to the result

(a) "tree" (b) "building" (c) "sidewalk"

Figure 3.6: Example of Linear Open Vocabulary DAAMs: An image generated with the prompt "A car in an urban environment" with overlaid linear attention maps for the tokens "tree" (3.6a), "building" (3.6b), and "sidewalk" (3.6c).

of computing the dot product between the query projection $W_q^{(i)\downarrow}\hat{h}_{i,t}^{\downarrow}$ and the key projection $W_k^{(i)\downarrow}x$. This computation enables us to capture the token's relevance within the image, without the incorporation of the softmax normalization utilized in Equation 3.5. Subsequently, the attention arrays are rescaled to dimensions $w \times h \times l_H^{(i)}$, denoted as $\tilde{L}_{X,t}^{(i)}$, to facilitate their aggregation. Finally, all the attention arrays are combined into a unified heatmap by summing them together.

$$LD_X^{\mathbb{R}}[x,y] := \sum_{t,i,l} \tilde{L}_{X,t,l}^{(i)\downarrow}[x,y] + \tilde{L}_{X,t,l}^{(i)\uparrow}[x,y]. \tag{3.8}$$

In Figure 3.6, the results of three linear attention maps, denoted as $LD_X^{\mathbb{R}}$ in eq. 3.8, are presented for the tokens "tree," "building," and "sidewalk" on the example image generated with the text prompt "A car in an urban environment." The example demonstrates how these heatmaps generate attention in the same regions as their softmax counterparts, without the need for a context phrase that interferes with their attention. This characteristic enhances their utility as an explainability method, simplifying the examination of biases acquired by a model or the generation of semantic segmentation masks based on a single word.

However, despite the potential of this method as an explainability technique for LDMs, it is still limited as a standalone approach for extracting ground truth in semantic segmentation tasks. A key limitation shared with the original DAAM approach [20] is that the regions attributed to a word may not align with the semantic segmentation mask assigned to the class represented by that word. For instance, in Figure 3.6c, it is evident that the attribution of the word "sidewalk" also extends attention to the road section, which is typically labeled as a distinct class in urban scene segmentation tasks. In other cases, the word may generate attention in other areas due to different learned semantic relationships within the network. To rigorously investigate this issue, an optimization approach for searching the text embedding that maximizes the Intersection over Union (IoU) with the target region is proposed in Section 3.3.

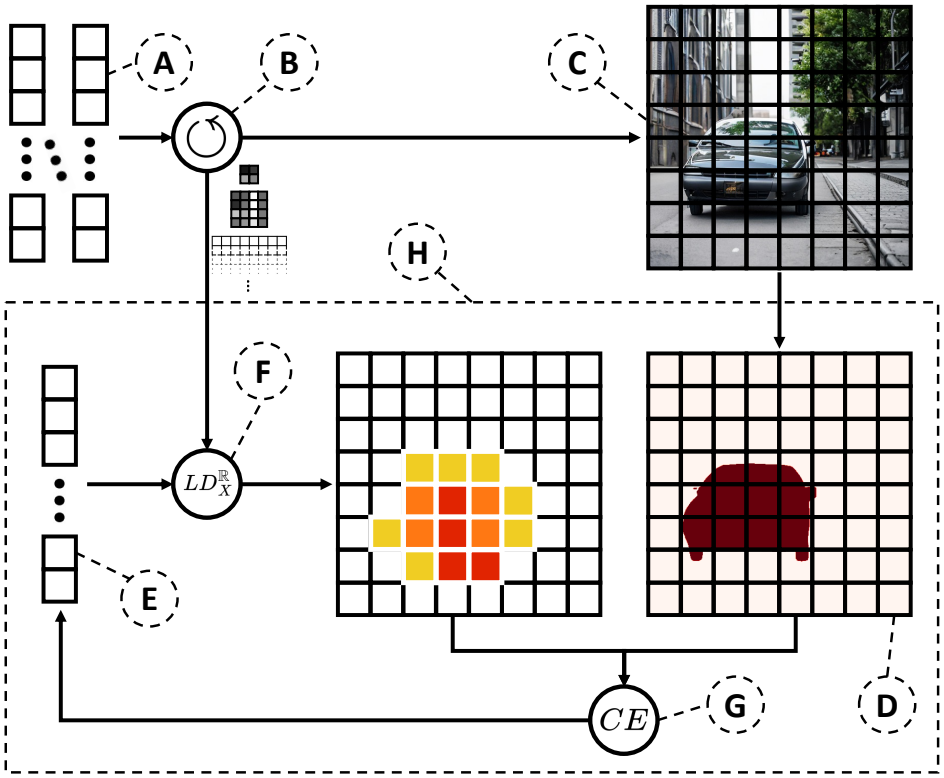


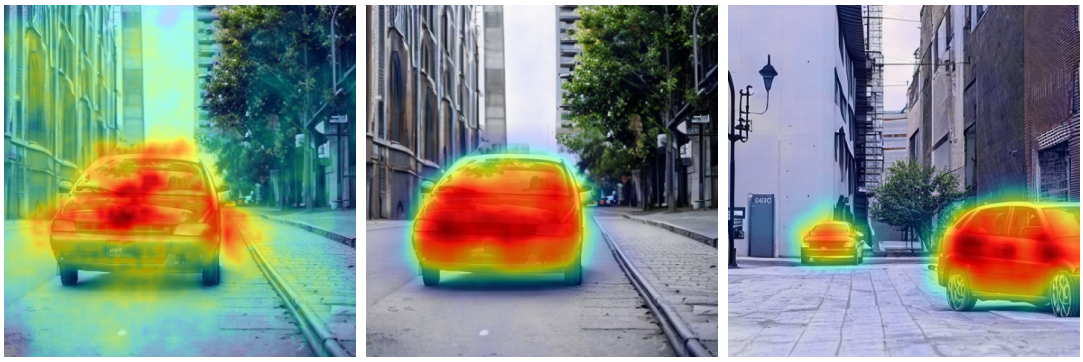
Figure 3.7: Prompt Optimization Process Diagram. The diagram illustrates the following steps: (A) Input of a text embedding X into a LDM (B), resulting in the generation of a synthetic image (C). (D) The target mask is annotated. A token embedding x (E) is utilized to generate the Open Vocabulary DAAM (F). The error between the target mask and the attention heatmap is calculated, and gradients are backpropagated (G) and x is updated. (H) This process is repeated within an optimization loop.

3.3 Prompt optimization via DAAM

The proposed extensions of DAAM for open vocabulary usage provide increased flexibility as an explainability tool. They allow us to highlight the semantic relationships learned by a Latent Diffusion Model (LDM), such as Stable Diffusion, by examining the influence of words on its internal attentions. However, initial tests showed that the attribution maps extracted for object nouns in generated images often do not align with the corresponding ground truth semantic segmentation masks. Although these maps demonstrate semantic coherence, they also include unwanted regions, which presents challenges for their application beyond explainability.

For instance, the attribution map for the word "building" (Figure 3.6b) extended its influence to the sidewalk, likely due to their semantic association within the urban context. In another case (Figure 3.6c), the "sidewalk" map also encompassed the road area, deviating from the desired segmentation. Such irregularities impede the effective use of attention maps in semantic segmentation tasks.

To address these limitations, we propose approaching the problem as a minimization problem solved through gradient descent on the extension of DAAM for Open Vocabulary proposed in this work (Fig. 3.7 illustrates this process). We will focus on formulating the method in its simplest variant: the search for a single token, x, that maximizes a specific region based on its Linear DAAM.



(a) Initial heatmap (b) Optimized heatmap (c) Transferred heatmap

Figure 3.8: Example of Optimization via DAAM. The figure showcases the optimization process applied to the example image (Fig 3.1a). It includes the initial Linear DAAM of the token "car" overlaid on the image (Fig. 3.8a), the optimized heatmap generated by $\hat{x}$ (Fig. 3.8b), and the Linear DAAM of $\hat{x}$ on a different image (Fig. 3.8c).

A Linear DAAM, denoted as $LD_X^{\mathbb{R}}$, can be understood as a function that takes a token embedding, $x \in \mathbb{R}^{l_C}$ and a set of hidden states, $LD^{\mathbb{R}}(X'; \hat{h}_{1,1}^{\downarrow}, \cdots, \hat{h}_{T,K}^{\uparrow})$, and generates an attention map. In this optimization, we keep the hidden states fixed and consider the DAAM solely as a function of x, indicated by the subscript X to represent the text embedding that generated these states. In other words, $LD_X^{\mathbb{R}}(\cdot) : \mathbb{R}^{l_C} \to \mathbb{R}^{w \times h}$. To formulate the optimization problem, we introduce a binary mask, $G_X \in \mathbb{R}^{w \times h}$, representing the ground truth of the target influential area. Our objective is to minimize a cost function C, which should be differentiable, in charge of measure the divergence between the output of the Linear DAAM and the ground truth G_X:

$$\hat{x} = \arg\min_{x} C\left(LD_X^{\mathbb{R}}(x), G_X\right). \tag{3.9}$$

This objective function (Eq. 3.9) is differentiable with respect to x, as $LD_X^{\mathbb{R}}$ is the sum of linear projections of the input token (Eq. 3.7) and spatial interpolations of these projections (Eq. 3.8). Therefore, we can optimize the objective function using gradient descent. Figure 3.7 illustrates this iterative optimization process, where the token embedding is updated based on the gradients of the objective function.

We found that the cross-entropy loss between the soft heatmap and the ground truth worked effectively for the optimization. The cross-entropy loss is defined as:

$$CE_{loss}\left(LD_X^{\mathbb{R}}, G_X\right) = -\sum_{x,y} G_X[x,y] \cdot \log\left(LD_X^{\mathbb{R}}[x,y]\right). \tag{3.10}$$

To compute the cross-entropy loss, the heatmap needs to be normalized in the range of $[0, 1]$. Since the linear DAAM is a sum of linear projections and is not inherently normalized like softmax-based heatmaps. We employed a linear scaling, also known as min-max scaling, to resize the heatmap values to the desired range of $[0, 1]$. This normalization effectively scaled the heatmap values, ensuring consistent visualization of the heatmaps throughout the optimization process. For clarity, the normalization step has not been included in the notation of Equation 3.10.

Figure 3.8 visually presents the optimization results conducted on the example image discussed throughout this chapter (Fig. 3.1a). In Fig. 3.8a, we observe the Linear DAAM generated for the token "car." A comparison with the DAAM generated

using the original non-linear method (Fig. 3.1d) reveals notable differences in behavior. While the resulting mask maintains its focus on the car, it exhibits a broader influence that extends beyond the immediate region. This is because the absence of softmax normalization between tokens in a text embedding allows the attention to encompass surrounding background areas.

To tackle this issue, we utilize "car" as the initial token x for the optimization process. The resulting heatmap generated by $\hat{x}$ after optimization is depicted in Figure 3.8b. It can be observed that the resulting area more precisely delineates the region surrounding the car, without attracting attention from the urban environment.

To assess the semantic information encapsulated within $\hat{x}$ and its generalizability, we evaluate its performance on a different image generated by the LDM. As shown in Figure 3.8c, we observe a scene featuring two cars on a street. Notably, the optimized vector effectively generates attention around both car instances, indicating the transferability of learned semantic knowledge to influence attentions in diverse image contexts.

Furthermore, this approach is also valid for optimizing Open Vocabulary DAAMs, in which we would need to jointly optimize a text embedding $X' = [x_1, ...; x_{l_{W'}}]$ with a segmentation mask for each token in the phrase. In this case $D_X^{\mathbb{R}}(\cdot) : \mathbb{R}^{l_C \times l_{W'}} \rightarrow \mathbb{R}^{w \times h \times l_{W'}}$.

In the general case, where X' is optimized across multiple images generated with text prompts $\{X_j\}_{j=1}^{N}$, where the length of the tokens X_j can vary, the objective function is defined as follows:

$$\hat{X'} = \arg\min_{X'} \sum_{j=1}^{N} \sum_{k=1}^{l_{W'}} C\left(D_{X_j,k}^{\mathbb{R}}(X'), G_{X_j,k}\right). \tag{3.11}$$

Where for practical experiments we use as loss C the cross entropy CE_{loss}. This objective function (Eq. 3.11) shares similarities with the loss function employed in training a semantic segmentation model. However, unlike traditional training with images, we utilize the attentions generated during their generation in the LDM (indexed by j). Instead of working with multiple semantic classes as the output objective, we focus on optimizing the activations of multiple tokens (indexed by k). Moreover, instead of updating the weights of the model, our optimization process revolves around refining the input text embedding ($X' \in \mathbb{R}^{l_C \times l_{W'}}$) . It is important to emphasize that, similar to a semantic segmentation model, we utilize binary masks as ground truth for each image and class/token in the training dataset (G_{X_j}, k). Figure 3.9 illustrates the procedure proposed to optimize this function.

In the following chapter, we conduct practical experiments aimed at analyzing the potential of prompt optimization via DAAM method in generating accurate segmentation masks. These experiments explore the effectiveness of the optimization process by assessing the quality and alignment of the generated masks in comparison to the ground truth annotations. Moreover, we investigate the generalization capabilities of the optimized text embeddings across a diverse range of images generated by the LDM.

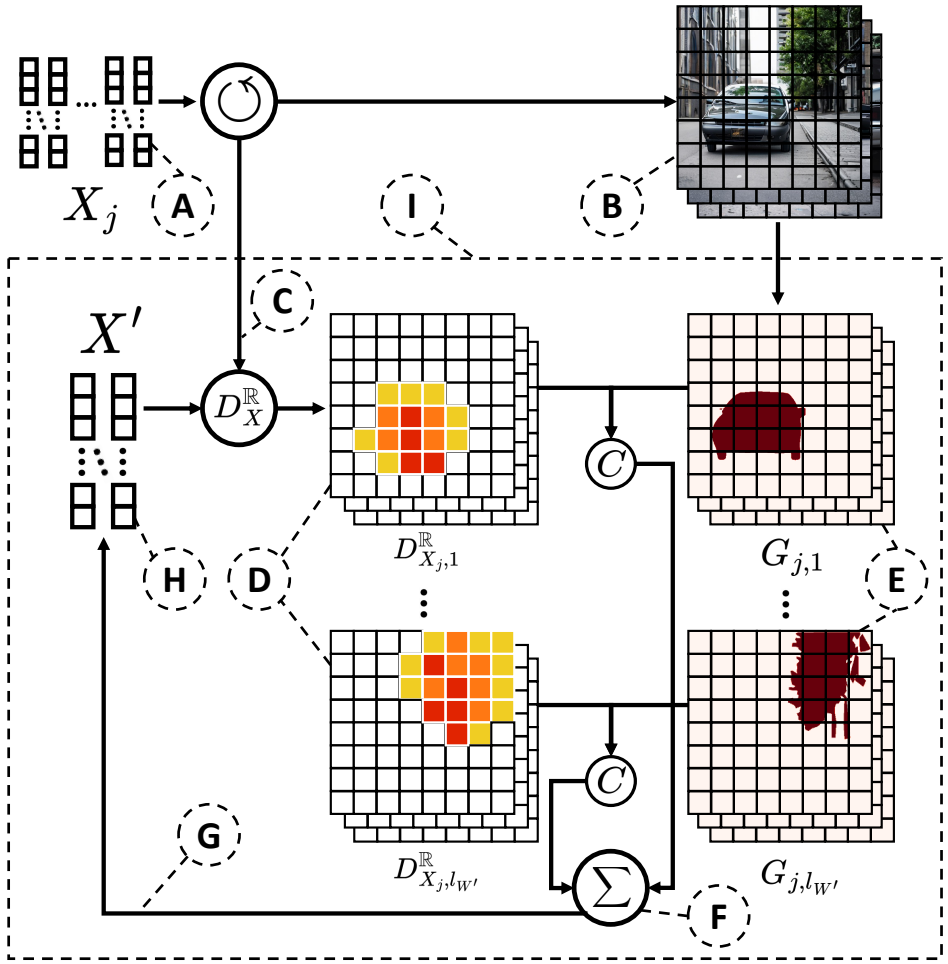


Figure 3.9: Optimization process of DAAM in the general case. The diagram illustrates the following steps: (A) Inputting the set of text embeddings X_j into a LDM, resulting in the generation of synthetic images (B). The attentions generated by the LDM are fixed in the DAAM function (C). DAAM generates heatmaps (D) corresponding to the target masks (E), which are used to compute the loss for each image and token (F). The gradients of the loss are propagated (G) and employed to update the text embedding X' (H). This iterative process occurs within an optimization loop (I).

Chapter 4

Experiments and Results

This chapter presents a series of practical experiments aimed at exploring the potential of utilizing DAAM and the proposed extensions for object segmentation in synthetically generated images, particularly focusing on urban scenes. To conduct these experiments, a synthetic dataset was created using Stable Diffusion as the foundational model (Sec. 4.1). The experiments are organized into three sections, mirroring the structure of the proposed methods proposed in Chapter 3, in order to assess the impact of each modification.

The first section, 4.2, evaluates the masks generated by DAAM in its original formulation [20] to establish a baseline for comparing the proposed modifications. Next, in section 4.3, we measure the effects of linearizing the attention maps using Linear DAAMs (Sec. 3.2). In section 4.4, we optimize text prompts for each class in the dataset and repeat the evaluation. Finally, in Section 4.5, we provide a summary of the experiment and a comparison of the results.

The goal of these experiments is to analyze the limitations and potential benefits of utilizing internally generated attentions from an LDM within a simplified urban scene scenario. Additionally, we aim to identify possible directions for future research, expanding the application of DAAM beyond its initial framework.

4.1 Experiment framework

This section overviews the experiment framework, including the architecture used and the details of the generated dataset.

To conduct the experiments, a synthetic dataset of simple urban scene scenarios was created. The dataset was intentionally designed to emphasize simplicity, with each image featuring a single primary object class. Four commonly encountered classes [16] in urban scene semantic segmentation tasks were chosen: "car," "person," "traffic light," and "rider".

The images were generated using the Stable Diffusion (2-base) architecture, which was also employed in the original DAAM paper [20]. However, the proposed methods and implementation are compatible with other versions of Stable Diffusion [6]. Initial tests conducted with other versions have yielded similar results.

A dataset consisting of 200 images was created, with 50 images per class. The text prompt "A ⟨token⟩ in an urban environment" was utilized to generate the images, where the token corresponds to each class name. In order to generate diverse images with the same text prompt, the random seed used to generate the initial noisy latent

(a) Car (b) Person (c) Traffic Light (d) Rider

(e) Car mask (f) Person mask (g) Traffic light mask (h) Rider mask

Figure 4.1: Visual examples of images and corresponding segmentation masks in the dataset. The first row showcases four distinct scenes, each representing one of the classes: "car", "person", "traffic light", and "rider". In the second row, the same images are presented with their respective ground truths overlayed.

vector was changed for each image. For more detailed information about the dataset and to access the images, please refer to Appendix C.

The LDM noise modeling process employs the PNDM scheduler [106], which utilizes a multi-step Runge-Kutta method to model the diffusion process as an ODE and accelerate the convergence of the diffusion process. The reverse diffusion model uses 30 timesteps to generate the images. To ensure reproducibility, the random seeds used in the process were stored, allowing for the replication of the diffusion process and generation of the same images and internal attentions.

Figure 4.1 showcases an example image for each of the four classes in the dataset (Figs. 4.1a to 4.1d). The images illustrate the simplicity of the scenes, with a clear focus on a single primary object from each class.

Ground truth annotations were created for each of the 200 images based on their corresponding primary class, following the standards used in Cityscapes [16]. The SAM (Segment Anything Model) [107] was employed to generate accurate segmentation masks, with manual supervision to ensure precise annotations. These segmentation masks serve as the reference for measuring the Intersection over Union (IoU) between the binary heatmaps generated by DAAM and its proposed extensions.

The latent space dimensions of Stable Diffusion 2-base (referred to as $w \times h$ in Chapter 3), and therefore the dimensions of the generated attention maps, are 64 x 64. However, the images are generated at a resolution of 512 x 512, due to the VAE final step [6]. To measure the IoU, the heatmaps are rescaled using bicubic interpolation to match the size of the images.

Examples of the binary masks can be seen in Figs. 4.1e to 4.1h. Notably, for the traffic light class (4.1g), the post is excluded, as it is typically labeled as class "pole" in

(a) Car (b) Person (c) Traffic light (d) Rider

Figure 4.2: Examples of DAAM-generated soft heatmaps. Each subfigure displays an image for each class with the overlayed soft heatmap generated using DAAM.

urban scene segmentation datasets. Similarly, the rider class includes only the person riding the motorcycle, without including the motorcycle itself (4.1h).

4.2 DAAM baseline

In this initial experiment, we evaluate the masks generated by the original DAAM method [20]. These findings establish a baseline for understanding the performance of DAAM's attention in semantic segmentation tasks and identifying the key challenges that arise when using this explainability technique in such contexts.

To evaluate the masks, we generated soft heatmaps, $D_k^{\mathbb{R}}$, following Eq. 3.3, using the original implementation of DAAM. The token corresponding to the class name was extracted from the text prompt "A ⟨token⟩ in an urban environment" for each class, except for the "traffic light" class, which consisted of two tokens. To prevent attention from being dispersed throughout the entire scene due to the semantic association with the word "traffic," we specifically considered the token "light" for this class.

Figure 4.2 displays the soft heatmaps for each example image of the four classes. We observe distinct issues in each class, which can be explained from a semantic perspective of the generated attention:

- In the example of the "car" class (Fig. 4.2a), the attention attributed to the word "car" correctly focuses on the cars. However, the attention within these cars is irregular, and in this specific example, there is minimal attention given to the cars in the background of the scene.
- For the "person" class (Fig. 4.2b), the attention is dispersed throughout the image, as the semantic relationship learned by the network for the word "person" influences the entire scene.
- In the "traffic light" class (Fig. 4.2c), where we considered the attention generated by the word "light," the attention correctly focuses on the red light of the traffic signal (see Fig. 4.1c). However, the segmentation does not cover the entire expected area according to the annotation criteria (4.1g).
- For the "rider" class (Fig. 4.2d), the attention is primarily concentrated on the motorcycle. This could be interpreted as the network learning that the presence of a motorcycle gives semantic meaning to the concept rider. However, our goal is to obtain a mask that specifically segments the person riding the motorcycle.

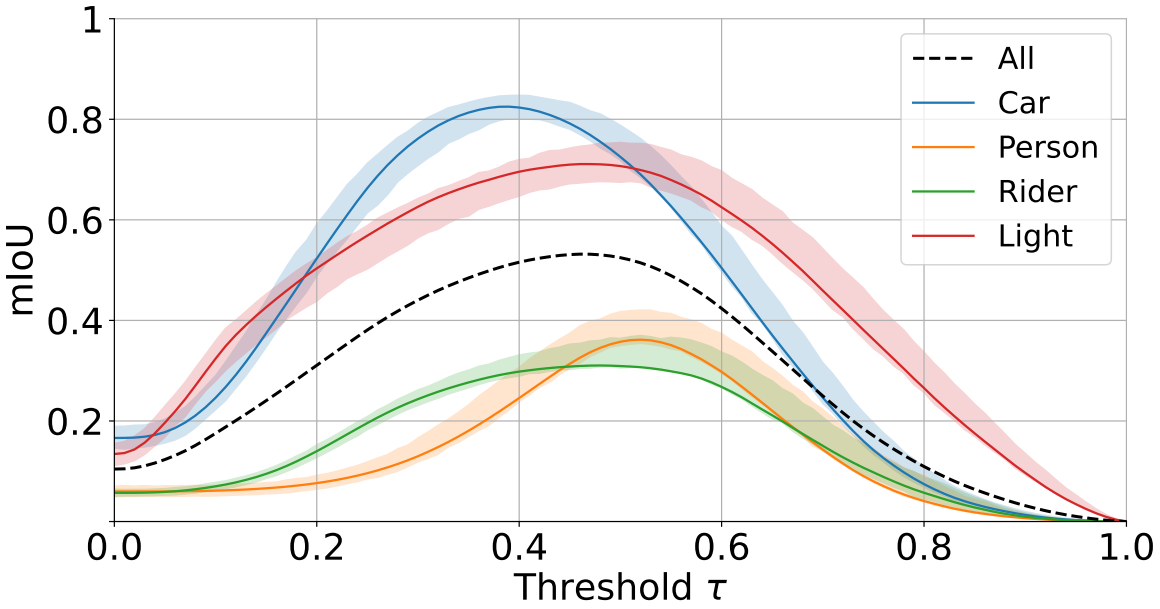


Figure 4.3: Comparison of mean Intersection over Union (mIoU) for each class at different threshold values. The graph illustrates the mIoU values for the "car," "light," "person," and "rider" classes, along with the 5th and 95th percentile values. The curves demonstrate distinct profiles for each class, highlighting the varying performance of DAAM in segmenting different object classes.

To assess the performance in terms of mIoU, we employed a fixed threshold to binarize the heatmaps (as per Eq. 3.4). Considering that mIoU is dependent on the threshold, we varied it from 0 to 1 to examine the curves' behavior. The results of the experiment are shown in Figure 4.3, revealing four distinct patterns for each class. The figure showcases the mIoU (per-pixel) for each class and includes the 5th and 95th percentile mIoU values for individual class examples. It is noteworthy that the behavior of mIoU across different thresholds remains consistent within each class. The curves' profiles can be explained by the challenges illustrated in the previous examples, which are prevalent throughout the dataset:

- The "car" class is segmented correctly, reaching a maximun mIoU of 82.5.
- The "light" class achieves an mIoU of 71.1, as the attention generated only relates to the light part of the traffic light.
- The "person" class reaches an mIoU of 36.2, as the attention is centered on the persons' silhouettes but generates dispersed attention throughout the scenes.
- The "rider" class has a maximum IoU of 31.0 because the network's concept aligns with the object "motorcycle" rather than the rider.

These results underscore the limitations of utilizing DAAM as a direct means of extracting ground truth, even in a straightforward scenario. The challenge lies in effectively controlling the attention's focus, which is constrained by the choice of words in the prompt used to generate the image. In the subsequent experiments, we examine this same scenario using the proposed modifications of the method, aiming to extend its applicability beyond its original purpose of explainability.

(a) Car (b) Person (c) Traffic light (d) Rider

Figure 4.4: Examples of Linear DAAM-generated soft heatmaps. Each subfigure displays an image for each class with the overlayed soft heatmap generated using Linear DAAM.

4.3 Linear Open Vocabulary DAAM

In this experiment, our objective is to assess the impact of removing the softmax nonlinearities in the original DAAM method [20] and instead utilizing Linear DAAMs (Section 3.2.2). It is worth noting that this experiment is not performed for the Open Vocabulary DAAM version (Section 3.2), as evaluating it with tokens present in both the generator and the query sentence is equivalent to the original DAAM method, and thus equivalent to the previous experiment.

The softmax normalization in DAAM ensures that the extracted attention is relative to the importance among different tokens in a prompt. However, by eliminating this normalization, we can construct heatmaps using a single token without the need to extract the attentions from an entire sentence, which can interfere with the attention of the target token. This provides us with more flexibility in generating ground truth based on a single word, such as the name of the object we want to segment. However, it also changes the nature of the generated heatmap as it is no longer relative to a contextual sentence.

To assess the impact of this change, we repeat the mIoU measurements on the same set of 200 dataset images, following the same threshold variation τ as in the previous experiment (Section 4.2). During image generation using the original text prompt "A ⟨token⟩ in an urban environment," we store the attentions $\hat{h}_{t,i}$, which are then used to construct the linear soft heatmaps $LD_X^{\mathbb{R}}$ (see Eq. 3.8). Although the proposed extension for generating Linear DAAMs allows for the evaluation of any arbitrary token, we extract the maps for the same tokens present in the prompt that generated the image to enable a meaningful comparison with the previous experiment.

Figure 4.4 illustrates the resulting heatmaps for each example image. It is evident that there is a decline in performance as the generated maps exhibit more generalized attribution throughout the surroundings of the objects. For example, in Figure 4.4a depicting an example from the "car" class, attention is now directed towards the road due to the semantic relationship between the car and that particular area of the scene. In contrast, the original method (Fig. 4.2) produced masks that were more aligned with the actual cars. This change can be attributed to the removal of the softmax with the ⟨SOT⟩ token, which typically absorbs the influence of the background areas not influenced by the primary object [20]. This behavior is exemplified in Figure 3.1b, where the attention map of the sentence start token captures attention from the entire image background.

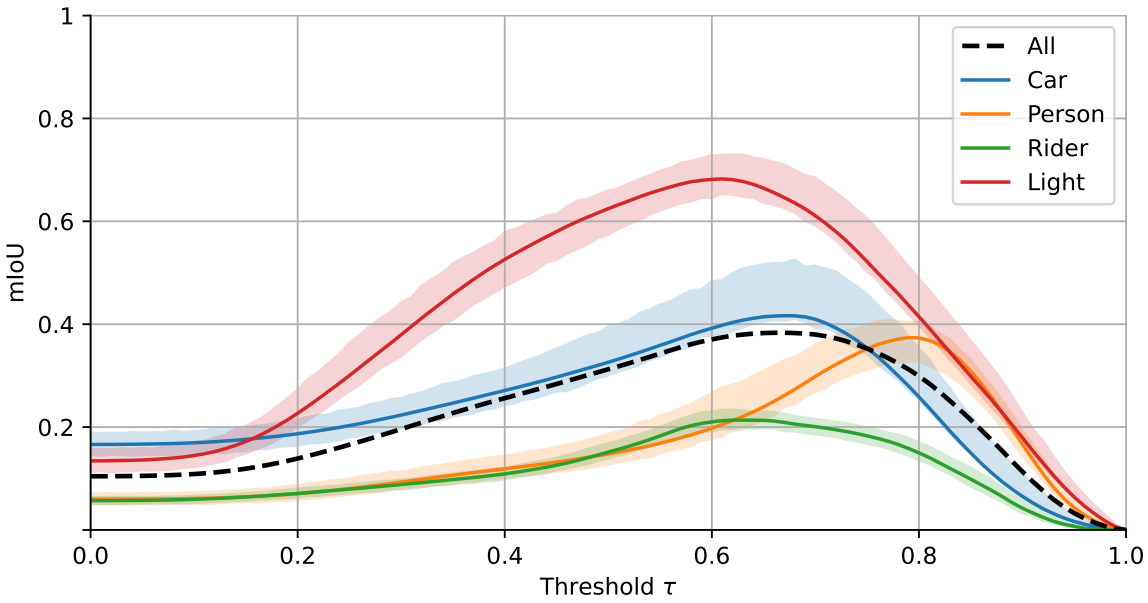


Figure 4.5: Mean Intersection over Union (mIoU) between ground truth masks and Linear DAAM heatmaps for each class at varying threshold values. The graph depicts the mIoU values for the "car," "light," "person," and "rider" classes, providing insights into the pixel-wise segmentation performance. The shaded regions represent the 5th and 95th percentile values, offering a range of mIoU performance across different examples within each class.

Analyzing Figure 4.5, which illustrates the threshold vs. mIoU curves, a significant reduction in performance can be observed compared to the results obtained with the original method (Fig. 4.3). This can be attributed to the aforementioned behavior (Figure 3.1b). Especially in the "Car" class, the maximum mIoU reached is 41.7, whereas with the original method we obtained 82.5.

Despite this decline in performance, these findings serve as a valuable baseline for the subsequent experiment. In the next phase, we optimize the tokens to identify the most suitable word for accurately segmenting the main object while minimizing the influence on other parts of the image. By searching for a token that precisely aligns with the desired region, we aim to mitigate this issue. Additionally, we aim to analyze how the semantic information from these optimized tokens transfers when evaluating the masks on other images that have not been specifically optimized.

4.4 Text Prompt optimization

In this third experiment, we measure the impact of using an optimized token via DAAM for segmenting the classes. To optimize these tokens, we follow the method proposed in Section 3.3. This optimization can be understood as "finding the most suitable word for segmenting an object," for example, representing the concept of a car $\hat{x}_{\text{car}}$ instead of using the generated attentions for the token "car".

Firstly, we conduct the optimization of the Linear DAAMs. To do this, we follow the experimental design illustrated in Figure 4.6. We create a training set for optimization, and the remaining images are used as a test set to measure the performance curves.

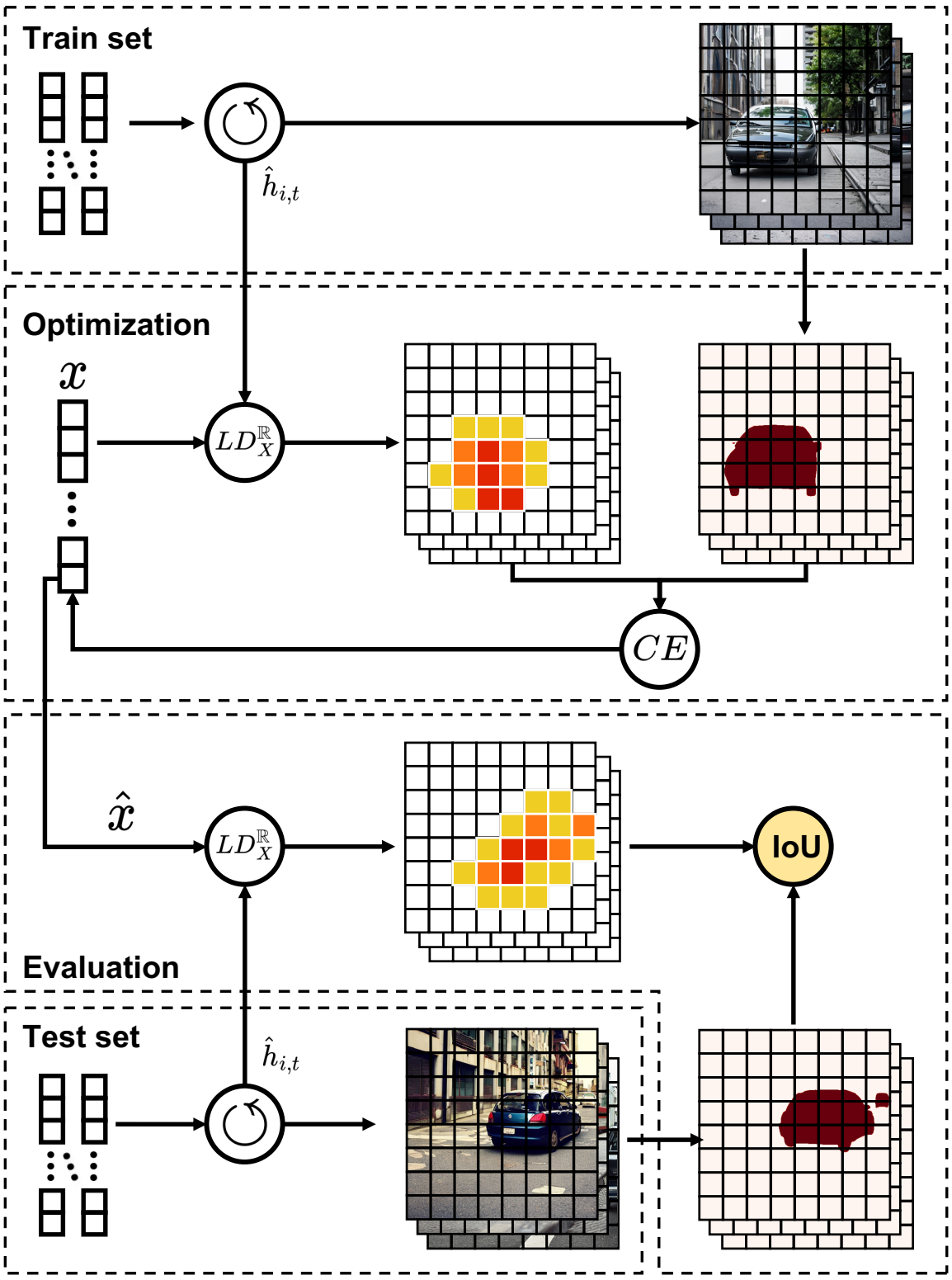


Figure 4.6: Experiment Design: Text Prompt Optimization for Linear DAAMs. The diagram depicts the workflow of the experiment, comprising four main steps: Train set, Optimization, Test set, and Evaluation. In the Train set phase, synthetic images are generated and manually annotated. The Optimization phase iteratively optimizes a token "x" to improve alignment. The Test set consists of a separate set of images. Finally, in the Evaluation phase, the optimized token is used with test attentions, and the IoU values between the test heatmaps and segmentation masks are measured.

(a) Car (b) Person (c) Traffic light (d) Rider

Figure 4.7: Examples of Linear DAAM-generated soft heatmaps. Each subfigure displays an image for each class with the overlayed soft heatmap generated using DAAM.

To evaluate the importance of the number of images used to optimize the query, we perform the experiment with 1, 2, and 5 images in the training set. For more detailed information on this optimization process, including the loss vs. epoch curves (Figure B.2), please refer to Appendix B.

Figure 4.7 illustrates the resulting masks obtained by using the optimized tokens $\hat{x}$ with two images as the training set (different from the examples provided). The generated masks show a higher level of alignment with the targeted objects for segmentation. This is in contrast to the non-optimized Linear DAAMs (Fig. 4.4), which exhibited more dispersed attention around the objects. From a semantic perspective, this finding suggests that the optimized tokens $\hat{x}$ capture information about the objects' silhouettes while disregarding information about their surroundings.

The trend of improvement is consistently observed in the mIoU vs. threshold curves (Figure 4.8) when using two images to optimize the tokens.

The optimization of these tokens shows promising results in terms of their effectiveness, even with a limited number of training samples. The maximum mIoU values achieved by the curves for different training set sizes are summarized in Table 4.1 (Figs. B.5). Notably, the experiment utilizing only 2 test images per class demonstrates the best performance, and in some cases, the test sets outperform the training sets. These findings suggest that the tokens $\hat{x}$ may possess semantic information that allows them to represent objects independently of their surroundings, facilitating knowledge transfer across images.

	1 train sample		2 train samples		5 train samples	
	Train	Test	Train	Test	Train	Test
Car	88.4	85.9	86.1	86.0	86.3	**86.6**
Person	65.1	70.0	65.5	**71.8**	65.7	70.9
Traffic Light	88.3	69.7	70.3	**72.2**	70.3	70.3
Rider	41.1	38.6	50.7	**47.8**	47.4	38.7
All	69.8	63.1	69.8	**67.3**	63.6	65.4

Table 4.1: Linear DAAM optimization. Comparison train and test.

Finally, we repeated this experiment by optimizing the non-linear version of DAAM for open vocabulary (Sec. 3.2). Unlike the linear version, in this variant, the attention is normalized across the tokens in the phrase. To perform this optimization, we followed the procedure illustrated in Figure 3.9. We used an embedding $X' = [x_1, x_2]$ with two

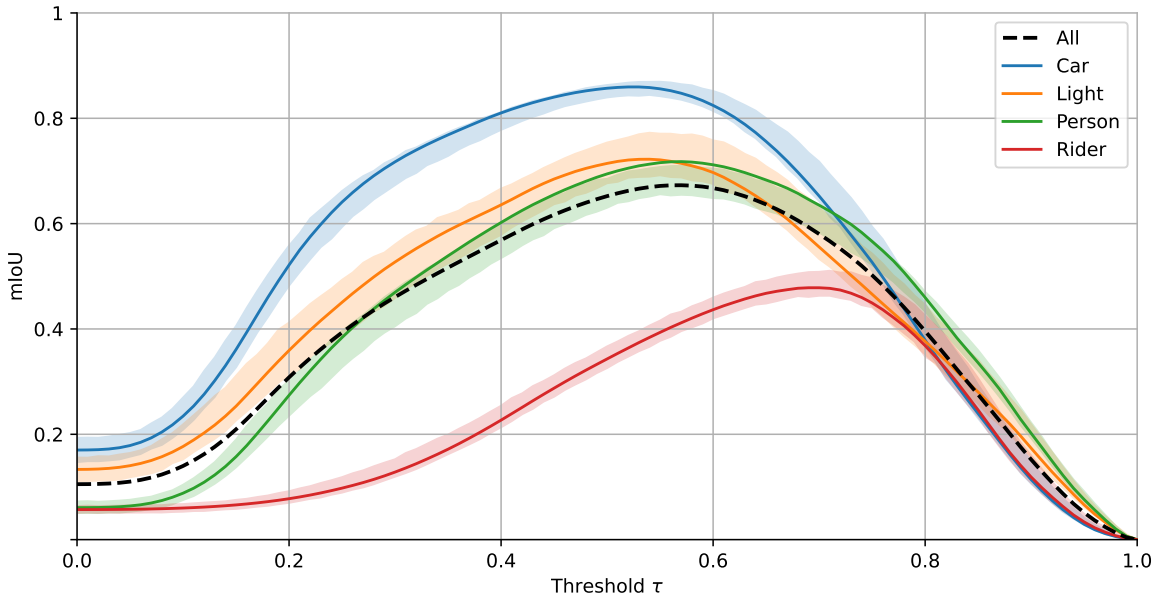


Figure 4.8: Optimized Linear DAAM, mIoU vs threshold curves. Performance of the test set optimized with 2 images per class as train set.

tokens. The first token was used to segment the area of the main object (e.g., car), and the second token was used to segment the background (complementary mask). The details of the optimization process can be found in Appendix B, including the loss vs. epoch curves (Fig. B.3). The curves of mIoU vs. thresholds (Fig. B.4) and examples of the masks (Fig. B.1) are also included in the appendix because they visually resemble the results obtained with the optimization of the linear DAAMs. This outcome indicates that optimizing a token separately or using a phrase where one token represents the background and another token represents the object produce similar results, reaffirming our hypothesis that optimizing the tokens separately converges to a token that represents the object in the text space without information about its surroundings.

In the next section, we summarize the results of the four measurements performed in the experiments: DAAM and Linear DAAM with and without optimization. This summary allows for a better understanding of the results without having to interpret the mIoU curves.

4.5 Experiment Summary

In summary, this chapter presents a comparison of the four measurements conducted: DAAM and Linear DAAM with and without optimization, supplemented by visual examples for qualitative comparison. To ensure an objective evaluation of segmentation performance, we utilize two metrics derived from the mIoU vs. threshold curves: the maximum IoU attained and the AUC (Area Under the Curve). The evaluations were conducted on a test set comprising 48 images per class, with the two remaining images dedicated to the prompt optimization.

The comparison of maximum mIoU values is presented in Table 4.2. The results clearly indicate that the experiments using optimized tokens outperform their non-

optimized counterparts for all classes. Among the optimized query versions, the non-linear variant achieves the highest maximum mIoU of 67.3 across the entire dataset, surpassing the linear version, which achieves a maximum mIoU of 66.6 for the entire dataset.

To evaluate the performance of the different versions without being affected by the chosen threshold, Table 4.3 showcases the Area Under the Curve (AUC) instead of the maximum value. Consistently, the experiments with optimized queries outperform their non-optimized counterparts. Notably, the optimized Linear DAAM experiment achieves the highest AUC of 39.5 for the entire dataset, primarily driven by its strong performance in the "Person" class.

For a qualitative comparison of these results, Figure 4.9 presents several examples from each class, illustrating a consistent pattern observed in the tables. In the "traffic light" class, all versions exhibit similar performance, with masks centered around the traffic lights. However, in the "Person" and "Car" classes, the non-optimized Linear DAAMs amplify the scattered attention observed in the original DAAM version. On the other hand, in the "Rider" case, where the attention naturally aligns with the bicycle, the optimization successfully focuses the attention on the rider, aligning with the intended ground truth.

These preliminary findings provide support for the hypothesis that by "searching" for a more suitable word to describe an object, it is feasible to guide the attention of a text-to-image LDM and extract segmentation masks for specific object classes. In the next chapter, we present our conclusions and initiate a discussion on potential directions for further investigation in this study, as well as explore potential applications in various tasks.

	Non-Optimized		Optimized	
	DAAM	LDAAM	DAAM	LDAAM
Car	82.5	41.7	**86.0**	85.8
Person	36.2	37.4	**71.6**	71.3
Traffic Light	71.1	68.2	**72.5**	72.3
Rider	31.0	21.3	**47.6**	43.1
All	53.2	38.3	**67.3**	66.6

Table 4.2: Summary of experiments: maximun mIoU

	Non-Optimized		Optimized	
	DAAM	LDAAM	DAAM	LDAAM
Car	39.7	23.7	**51.6**	51.5
Person	13.8	15.4	36.6	**41.3**
Traffic Light	43,3	38.3	**42.8**	42.7
Rider	15.6	11.0	**23.5**	22.5
All	28.1	22.1	38.6	**39.5**

Table 4.3: Summary of experiments: AUC

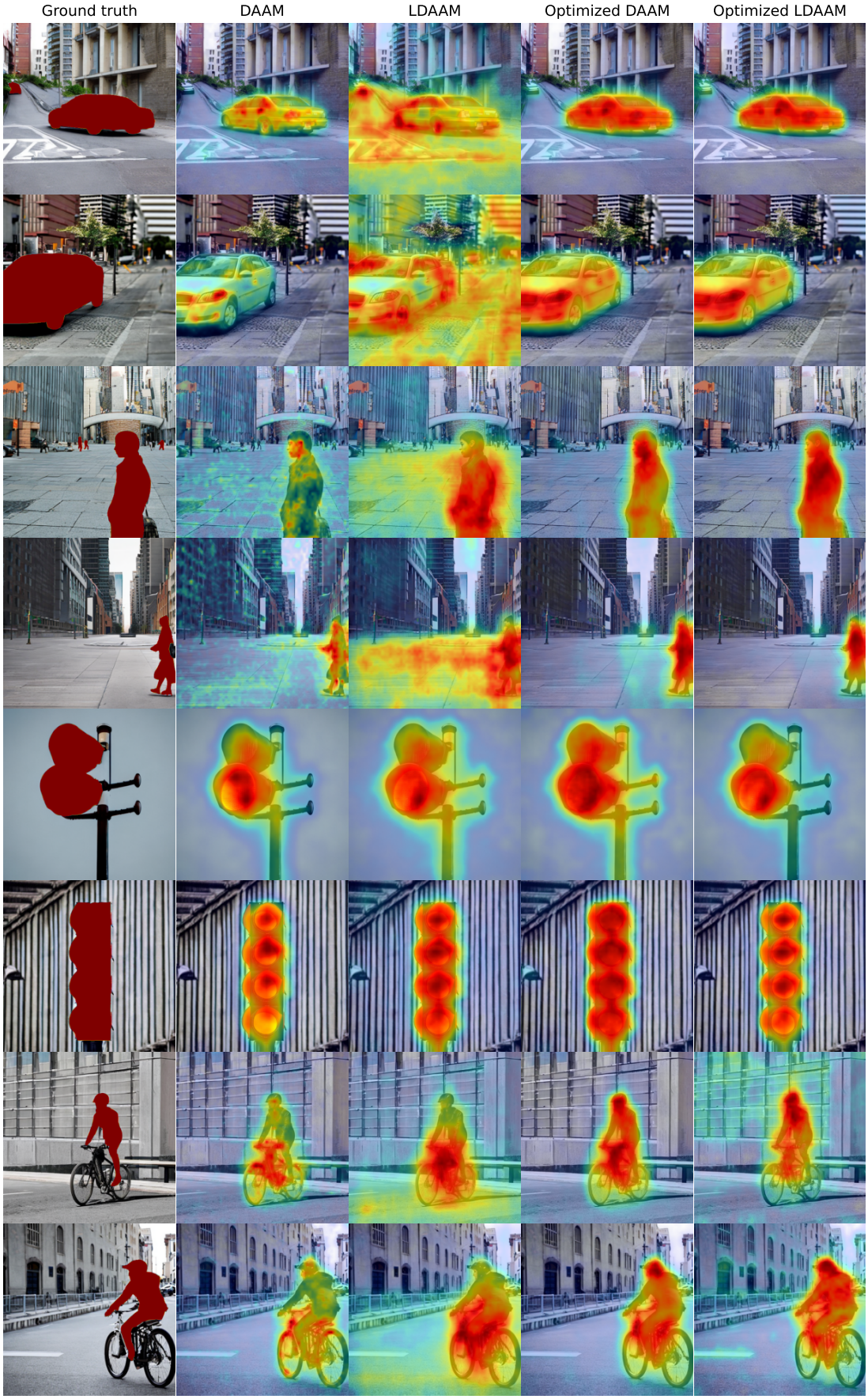


Figure 4.9: Experiments examples. By columns: Ground truth, DAAM, Linear DAAM, DAAM with prompt optimized, Linear DAAM Optimized.

Chapter 5

Conclusions and Future Work

5.1 Conclusions

This master's thesis aimed to explore the question: "Is it possible to use recent text-to-image Latent Diffusion Models (LDMs) to generate synthetic datasets for semantic segmentation?" Specifically, we focused on the context of urban scenes for training Computer Vision applications related to driving.

To tackle this question, we proposed the use of Diffusion Attentive Attribution Maps (DAAM) as base for our problem. DAAM [20] is an explainability method that leverages the attention mechanisms of LDMs to establish the relationship between tokens in the input text and the corresponding areas in the generated images.

In order to extend the usage of DAAM beyond its explainability formulation, two extensions were proposed. First, "Open Vocabulary DAAM" was introduced to evaluate text prompts different from those used to generate the image. Second, a simplified version called "Linear DAAM" was developed, which allows the evaluation of individual words without the need for attention to be relative to other tokens in the prompt. This simplification facilitates the use of the method to generate masks for specific objects based on a semantically representative word.

During preliminary experiments with these methods, one of the initial limitations observed was that words used to describe an object did not consistently result in aligned masks for the target area. To address this challenge, an optimization problem in the text-embedding space was formulated to identify the most suitable word that accurately describes the target region, thereby improving the segmentation masks. This optimization was approached as a gradient descent problem on the text embedding, leveraging the differentiability of Open Vocabulary DAAM. By adopting this approach, the potential and limitations of DAAM could be assessed, mitigating the impact of word selection on the segmentation results.

In summary, the experiments conducted on a simple dataset generated with Stable Diffusion demonstrated the effectiveness of optimizing text embeddings and utilizing DAAM for semantic segmentation. The results showed that optimizing tokens for segmenting objects in one or two images led to the transfer of this information to other images. This finding supports our hypothesis that the discovered tokens contain semantic information about the objects to be segmented. For instance, while the word "car" can describe the general concept of vehicle, the optimized tokens provide a more precise description of the object without considering its surroundings.

In conclusion, this research contributes to the field in two significant ways. Firstly, it

advances the field of LDM explainability through the development of Open Vocabulary DAAMs. This tool holds great potential for enhancing our understanding of LDMs, unraveling learned semantic relationships, and exploring issues such as acquired biases or the mechanisms responsible for synthesizing different parts of generated images. Secondly, it contributes to the progress of Open Vocabulary-based segmentation models. The proposed methodology enables the search for words that accurately describe target objects, resulting in improved generated masks without requiring model retraining. Although this work represents an early exploration of using DAAM for segmentation in a simplified scenario, it unveils the potential of attention maps in segmenting generated objects.

To finalize, in the following section, we discuss potential future directions for this research, exploring the possibilities for further exploration and applications of the proposed methods in various tasks.

5.2 Future work

Given the broad scope of this work and the exploratory nature of the research question, it has opened up a multitude of avenues for future investigation. To provide a comprehensive discussion of potential directions, we have structured this section into three parts. In Section 5.2.1, we present several potential lines of investigation to extend the proposed methods for semantic segmentation in more complex scenarios. In Section 5.2.2, we explore potential applications for enhancing the explainability of text-to-image LDMs, leveraging the methodologies introduced in this work. Lastly, Section 5.2.3 presents potential applications of the proposed methods in LDMs that integrate diverse signal modalities.

5.2.1 Enhancing Semantic Segmentation with DAAM

While the main focus of this work has been the study of DAAM applied to semantic segmentation, the exploratory nature of the research and the experimental setup have primarily addressed a simplified scenario. However, this scenario serves as a means to validate the proposed methods, assess their potential and limitations, and establish a baseline for future investigations.

Improving Heatmaps Generated by DAAM

While the proposed method for optimizing text tokens has shown effectiveness in controlling the generated segmentation masks, its performance in terms of mIoU is limited compared to state-of-the-art algorithms for urban scene semantic segmentation. One of the sources of this limitation stems from the small size of the latent space in LDMs. In the case of Stable Diffusion 2, the attention maps are generated in a 64x64-dimensional space, while the generated images have a resolution of 512x512. To perform segmentation, the attention maps are scaled to match the image size, which compromises their ability to capture fine details.

One potential approach to address this limitation is to scale the attention maps using the VAE of Stable Diffusion [6]. The VAE transforms the latent variable resulting from the reverse diffusion process (64x64) into a larger image size, such as 512x512.

Exploring the use of this mechanism to scale the attention maps could enhance their granularity and precision.

The precision of the masks is also significantly influenced by the aggregation of attention maps from different layers. Due to the downsampling performed in the U-Net of Stable Diffusion, not all attention arrays have a dimension of 64x64 (as shown in Fig. 3.3). Deeper blocks generate attention maps at resolutions of 32x32 or even 16x16, which are then scaled up to 512x512. An analysis in Appendix A reveals that this directly impacts the generated masks, as low-resolution attention maps exhibit lower precision and alignment with the objects. Investigating the selection of the most suitable layers for extracting segmentation masks and exploring different aggregation strategies can directly improve the results obtained. The impact of these layers is clearly observed in the examples presented in Fig. 4.9, where the heatmaps of the optimized masks exhibit a halo effect around the segmented object silhouettes due to the contribution of low-resolution attention maps.

Lastly, to evaluate the contribution of synthetic data generated by LDMs and DAAM to semantic segmentation models, it is necessary to assess their impact on the performance of models trained with these datasets. Evaluating the performance of DAAM in more complex scenarios and measuring the improvement achieved by the models would be a valuable direction for future research within the scope of the initial research question.

Open Vocabulary-based Semantic Segmentation

Beyond using the heatmaps directly as segmentation masks, the proposed methods hold great potential as features for segmentation models. An example of a successful approach in this regard is presented in [108] for open-vocabulary panoptic and semantic segmentation tasks. Similar to DAAM, the model described in that work extracts attention maps generated by the cross-attention mechanisms of Stable Diffusion. However, instead of utilizing these maps as masks, they are employed as features within a segmentation model. The methodology proposed in this master's thesis, which focuses on identifying tokens that align with specific concepts, could have a significant impact on open-vocabulary segmentation models [108] or object tracking [109], enabling finer control over open-vocabulary labels for describing specific objects, without the need of retraining these models.

5.2.2 Advancing the Explainability of Text-to-Image LDMs

While this work is framed within the context of urban scene segmentation, a significant portion of the contributions made focus on extending methods for the explainability of text-to-image LDMs. Therefore, several potential research directions can be pursued in this field.

Mask-To-Text Explainability method

The methodology proposed for finding tokens that align with a given target region in an image (Section 3.3) can be leveraged as a method to transform a mask image into a text embedding. This is precisely the inverse problem of open-vocabulary object segmentation. From an explainability standpoint, this method can be highly valuable

for studying the textual space used as input in text-to-image LDMs, enabling the analysis of the semantic relationships associated by the network with each object.

One potential research direction would involve studying optimization mechanisms for finding these tokens that optimize a mask. It would be necessary to investigate the convergence behavior of these tokens and analyze whether they converge to semantically similar points in the text embedding regardless of the initialization of the optimization. For example, starting from different object masks representing a car in different images, understanding if they all converge to a point in the text embedding related to the concept of a vehicle. Furthermore, a deeper exploration of how these discovered points generalize can be conducted by directly using them to generate new images and studying how the network represents them.

Further Study of LDM mechanisms

Expanding on the mask-to-text methods can also serve as valuable tools for studying the internal mechanisms of LDMs. For example, they can be applied to investigate the biases acquired by the models. To illustrate the potential of continuing this work in studying biases, synthetic images with people in various situations can be generated. Experiments can then be conducted to analyze the tokens attributed to the silhouettes and examine their relationship with different semantic concepts in the text embedding.

Additionally, to gain a better understanding of the synthesis mechanisms in LDMs, the proposed optimization methodology can be employed on a subset of the attention blocks and heads of the U-Net in Stable Diffusion (see Eq. 3.6). This approach would enable a more detailed examination of the semantic information synthesized at different layers. Preliminary analysis presented in Appendix A suggests that shallower layers contain coarse-grained semantic information, while deeper layers capture fine-grained semantic details. This line of research can contribute to a deeper understanding of diffusion architectures and provide valuable insights for designing more efficient architectures in the future.

5.2.3 Exploring Multi-Modal Extensions of DAAM

While DAAM [20] and this work focus on text-to-image models, the theoretical framework presented in Chapter 3 is applicable to LDMs that employ cross-attention mechanisms to guide the internal denoising process. To conclude this discussion on potential future research directions, we can explore how this work could be extended to LDMs based on modalities other than text and image.

Text-to-$\mathcal{S}$ LDMs

For models that take text as input, the application of Open Vocabulary DAAM could be extended to open-vocabulary tasks in Text-to-$\mathcal{S}$ LDMs. One notable advantage of this approach is the intuitive nature of the text space as an input, which facilitates the study of attribution in the $\mathcal{S}$ space. While there are already models capable of achieving high precision in image segmentation, applying this approach to other modalities where such models are not available presents significant potential.

This extension can be particularly relevant for recently developed text-to-audio LDMs like AudioLDM [49]. AudioLDM proposes a Stable Diffusion-like architecture that employs a U-Net with similar attention mechanisms to generate high-fidelity audio

from text descriptions. By expanding Open Vocabulary DAAM to incorporate this type of model, it would be possible to investigate how the model attributes different aspects of the generated audio. Many of the research directions discussed in this section can be effectively extended to this context.

$\mathcal{S}$-to-Image LDMs

Lastly, in the case of $\mathcal{S}$-to-Image LDMs, the intuitive nature of the image space can be leveraged. By extending the proposed optimization methodology in Section 3.3 to this type of LDMs, it becomes possible to study the space $\mathcal{S}$ by analyzing how object masks project onto it from the image space.

For example, this approach could be particularly interesting when applied to fMRI-to-image models, such as the one presented in [110]. This work introduces a variation of Stable Diffusion for image reconstruction from fMRI scans that capture the brain activity of a person viewing an image. They propose replacing the text encoder in Stable Diffusion [6, 78] with a module that takes fMRI signals as input. In this architecture, the input space, fMRI, is highly complex and challenging to study. Therefore, exploring how objects in an image are influenced by the input fMRIs could be a valuable research direction. Understanding these mechanisms could shed light on whether there is any correlation between the patterns used by the network for image reconstruction from this signal and the biological function of brain activity.

Bibliography

[1] J. Chai, H. Zeng, A. Li, and E. W. Ngai, "Deep learning in computer vision: A critical review of emerging techniques and application scenarios," *Machine Learning with Applications*, vol. 6, p. 100134, 2021. 1, 4, 15

[2] C. Sun, A. Shrivastava, S. Singh, and A. Gupta, "Revisiting unreasonable effectiveness of data in deep learning era," in *2017 IEEE International Conference on Computer Vision (ICCV)*, pp. 843–852, 2017. 1

[3] S. Dong, P. Wang, and K. Abbas, "A survey on deep learning and its applications," *Comput. Sci. Rev.*, vol. 40, may 2021. 1

[4] P. Dhariwal and A. Nichol, "Diffusion models beat gans on image synthesis," in *Advances in Neural Information Processing Systems* (M. Ranzato, A. Beygelzimer, Y. Dauphin, P. Liang, and J. W. Vaughan, eds.), vol. 34, pp. 8780–8794, Curran Associates, Inc., 2021. 1, 13

[5] A. Ramesh, P. Dhariwal, A. Nichol, C. Chu, and M. Chen, "Hierarchical text-conditional image generation with clip latents," *ArXiv*, vol. abs/2204.06125, 2022. 1, 9, 12, 13

[6] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, "High-resolution image synthesis with latent diffusion models," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10684–10695, 2022. 1, 2, 9, 12, 13, 14, 20, 22, 33, 34, 46, 49

[7] F.-A. Croitoru, V. Hondru, R. T. Ionescu, and M. Shah, "Diffusion models in vision: A survey," *ArXiv*, vol. abs/2209.04747, 2022. 1

[8] H. Cao, C. Tan, Z. Gao, G. Chen, P.-A. Heng, and S. Z. Li, "A survey on generative diffusion model," *ArXiv*, vol. abs/2209.02646, 2022. 1

[9] J. Gerlings, A. Shollo, and I. D. Constantiou, "Reviewing the need for explainable artificial intelligence (xai)," in *Hawaii International Conference on System Sciences*, 2020. 1, 15, 16

[10] R. Saleem, B. Yuan, F. Kurugollu, A. Anjum, and L. Liu, "Explaining deep neural networks: A survey on the global interpretation methods," *Neurocomputing*, vol. 513, pp. 165–180, 2022. 1, 15

[11] K. Man and J. Chahl, "A review of synthetic image data and its use in computer vision," *Journal of Imaging*, vol. 8, no. 11, 2022. 1

[12] A. Kazerouni, E. K. Aghdam, M. Heidari, R. Azad, M. Fayyaz, I. Hacihaliloglu, and D. Merhof, "Diffusion models for medical image analysis: A comprehensive survey," *ArXiv*, vol. abs/2211.07804, 2022. 1

[13] B. Trabucco, K. Doherty, M. Gurinas, and R. Salakhutdinov, "Effective data augmentation with diffusion models," *ArXiv*, vol. abs/2302.07944, 2023. 1

[14] A. Kotelnikov, D. Baranchuk, I. Rubachev, and A. Babenko, "Tabddpm: Modelling tabular data with diffusion models," *ArXiv*, vol. abs/2209.15421, 2022. 1

[15] H. Lin, P. Upchurch, and K. Bala, "Block annotation: Better image annotation with sub-image decomposition," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019. 1, 6

[16] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele, "The cityscapes dataset for semantic urban scene understanding," in *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 1, 4, 6, 7, 33, 34

[17] G. Neuhold, T. Ollmann, S. Rota Bulo, and P. Kontschieder, "The mapillary vistas dataset for semantic understanding of street scenes," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Oct 2017. 1, 6, 7

[18] R. Alcover Couso, "On exploring the use of synthetic data for semantic segmentation in videos," Master's thesis, Universidad Autónoma de Madrid, 2021. 1

[19] F. E. Nowruzi, P. Kapoor, D. Kolhatkar, F. A. Hassanat, R. Laganière, and J. Rebut, "How much real data do we actually need: Analyzing object detection performance using synthetic and real data," *ArXiv*, vol. abs/1907.07061, 2019. 1

[20] R. Tang, L. Liu, A. Pandey, Z. Jiang, G. Yang, K. Kumar, P. Stenetorp, J. Lin, and F. Ture, "What the daam: Interpreting stable diffusion using cross attention," *ArXiv*, vol. abs/2210.04885, 2022. 1, 18, 19, 20, 23, 24, 25, 26, 28, 33, 35, 37, 45, 48, 63, 65

[21] J. Geyer, Y. Kassahun, M. Mahmudi, X. Ricou, R. Durgesh, A. S. Chung, L. Hauswald, V. H. Pham, M. Mühlegg, S. Dorn, T. Fernandez, M. Jänicke, S. G. Mirashi, C. Savani, M. Sturm, O. Vorobiov, M. Oelker, S. Garreis, and P. Schuberth, "A2d2: Audi autonomous driving dataset," *ArXiv*, vol. abs/2004.06320, 2020. 2, 7

[22] S. M. Grigorescu, B. Trasnea, T. T. Cocias, and G. Macesanu, "A survey of deep learning techniques for autonomous driving," *Journal of Field Robotics*, vol. 37, pp. 362 – 386, 2019. 3

[23] K. Sun, Y. Zhao, B. Jiang, T. Cheng, B. Xiao, D. Liu, Y. Mu, X. Wang, W. Liu, and J. Wang, "High-resolution representations for labeling pixels and regions," *ArXiv*, vol. abs/1904.04514, 2019. 4

[24] S. Minaee, Y. Boykov, F. Porikli, A. Plaza, N. Kehtarnavaz, and D. Terzopoulos, "Image segmentation using deep learning: A survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 7, pp. 3523–3542, 2022. 4

[25] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3431–3440, 2015. 4, 5

[26] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *2015 Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, pp. 234–241, 2015. 4, 5, 14, 20

[27] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder-decoder with atrous separable convolution for semantic image segmentation," in *2018 European Conference Computer Vision (ECCV)*, p. 833–851, 2018. 5

[28] J. Wang, K. Sun, T. Cheng, B. Jiang, C. Deng, Y. Zhao, D. Liu, Y. Mu, M. Tan, X. Wang, W. Liu, and B. Xiao, "Deep high-resolution representation learning for visual recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, pp. 3349–3364, 2019. 5

[29] Y. Zhang, P. David, and B. Gong, "Curriculum domain adaptation for semantic segmentation of urban scenes," in *2017 IEEE International Conference on Computer Vision (ICCV)*, pp. 2039–2049, 2017. 5, 8

[30] F. Tung and G. Mori, "Similarity-preserving knowledge distillation," *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 1365–1374, 2019. 5, 8

[31] Y.-H. Tsai, W.-C. Hung, S. Schulter, K. Sohn, M.-H. Yang, and M. Chandraker, "Learning to adapt structured output space for semantic segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 5, 8

[32] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the kitti vision benchmark suite," in *2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3354–3361, 2012. 6, 7, 8

[33] F. Yu, H. Chen, X. Wang, W. Xian, Y. Chen, F. Liu, V. Madhavan, and T. Darrell, "Bdd100k: A diverse driving dataset for heterogeneous multitask learning," *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2633–2642, 2018. 6, 7

[34] O. Zendel, M. Schörghuber, B. Rainer, M. Murschitz, and C. Beleznai, "Unifying panoptic segmentation for autonomous driving," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 21351–21360, June 2022. 7

[35] X. Huang, P. Wang, X. Cheng, D. Zhou, Q. Geng, and R. Yang, "The apolloscape open dataset for autonomous driving and its application," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 42, no. 10, pp. 2702–2719, 2020. 7

[36] G. Varma, A. Subramanian, A. Namboodiri, M. Chandraker, and C. Jawahar, "IDD: A dataset for exploring problems of autonomous navigation in unconstrained environments," in *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, IEEE, jan 2019. 7

[37] E. Fernandez-Moral, R. Martins, D. Wolf, and P. Rives, "A new metric for evaluating semantic segmentation: Leveraging global and contour accuracy," in *2018 IEEE Intelligent Vehicles Symposium (IV)*, pp. 1051–1056, 2018. 7

[38] Y. Zhang, S. Mehta, and A. Caspi, "Rethinking semantic segmentation evaluation for explainability and model selection," *ArXiv*, vol. abs/2101.08418, 2021. 7

[39] Y.-J. Cho, "Weighted intersection over union (wiou): A new evaluation metric for image segmentation," *ArXiv*, vol. abs/2107.09858, 2021. 7

[40] D. A. Pomerleau, "Alvinn: An autonomous land vehicle in a neural network," *Advances in Neural Information Processing Systems 1*, p. 305–313, 1989. 7

[41] G. Ros, L. Sellart, J. Materzynska, D. Vazquez, and A. M. Lopez, "The synthia dataset: A large collection of synthetic images for semantic segmentation of urban scenes," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3234–3243, 2016. 7, 8

[42] S. R. Richter, V. Vineet, S. Roth, and V. Koltun, "Playing for data: Ground truth from computer games," in *2016 European Conference Computer Vision (ECCV)*, pp. 102–118, 2016. 7, 8

[43] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in Neural Information Processing Systems*, pp. 2672–2680, 2014. 7, 10

[44] A. Dosovitskiy, G. Ros, F. Codevilla, A. M. López, and V. Koltun, "Carla: An open urban driving simulator.," in *CoRL*, vol. 78 of *Proceedings of Machine Learning Research*, pp. 1–16, 2017. 8

[45] X. Guo, Z. Wang, Q. Yang, W. Lv, X. Liu, Q. Wu, and J. Huang, "Gan-based virtual-to-real image translation for urban scene semantic segmentation," *Neurocomputing*, vol. 394, pp. 127–135, 2020. 8

[46] C. Schuhmann, R. Beaumont, R. Vencu, C. W. Gordon, R. Wightman, M. Cherti, T. Coombes, A. Katta, C. Mullis, M. Wortsman, P. Schramowski, S. R. Kundurthy, K. Crowson, L. Schmidt, R. Kaczmarczyk, and J. Jitsev, "LAION-5b: An open large-scale dataset for training next generation image-text models," in *Thirty-sixth Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2022. 9, 13

[47] A. Asperti and V. Tonelli, "Comparing the latent space of generative models," *Neural Computing and Applications*, vol. 35, 10 2022. 9

[48] Y. Shen, J. Gu, X. Tang, and B. Zhou, "Interpreting the latent space of gans for semantic face editing," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9240–9249, 2020. 9, 11

[49] H. Liu, Z. Chen, Y. Yuan, X. Mei, X. Liu, D. Mandic, W. Wang, and M. D. Plumbley, "Audioldm: Text-to-audio generation with latent diffusion models," *arXiv*, vol. abs/2301.12503, 2023. 9, 48

[50] J. Ho, W. Chan, C. Saharia, J. Whang, R. Gao, A. Gritsenko, D. P. Kingma, B. Poole, M. Norouzi, D. J. Fleet, and T. Salimans, "Imagen video: High definition video generation with diffusion models," *ArXiv*, vol. abs/2210.02303, 2022. 9

[51] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," in *Advances in Neural Information Processing Systems*, vol. 33, pp. 6840–6851, 2020. 9, 12

[52] D. H. Ackley, G. E. Hinton, and T. J. Sejnowski, "A learning algorithm for Boltzmann machines," *Cognitive Science*, vol. 9, pp. 147–169, 1985. 9

[53] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning internal representations by error propagation," in *Parallel Distributed Processing: Explorations in the Microstructure of Cognition, Volume 1: Foundations*, pp. 318–362, MIT Press, 1986. 9

[54] D. H. Ballard, "Modular learning in neural networks," in *Sixth National Conference on Artificial Intelligence*, pp. 279–284, 1987. 9

[55] X. Guo, X. Liu, E. Zhu, and J. Yin, "Deep clustering with convolutional autoencoders," in *Neural Information Processing*, pp. 373–382, 2017. 9, 10

[56] P. Vincent, H. Larochelle, Y. Bengio, and P.-A. Manzagol, "Extracting and composing robust features with denoising autoencoders.," in *2008 International Conference on Machine Learning (ICML)*, vol. 307, pp. 1096–1103, 2008. 10

[57] S. Rifai, P. Vincent, X. Muller, X. Glorot, and Y. Bengio, "Contractive autoencoders: Explicit invariance during feature extraction.," in *2011 International Conference on Machine Learning (ICML)*, pp. 833–840, 2011. 10

[58] D. Pathak, P. Krähenbühl, J. Donahue, T. Darrell, and A. A. Efros, "Context encoders: Feature learning by inpainting," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2536–2544, 2016. 10

[59] R. Zhang, P. Isola, and A. A. Efros, "Split-brain autoencoders: Unsupervised learning by cross-channel prediction," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 645–654, July 2017. 10

[60] Z. Shu, M. Sahasrabudhe, R. A. Guler, D. Samaras, N. Paragios, and I. Kokkinos, "Deforming autoencoders: Unsupervised disentangling of shape and appearance," in *2018 European Conference on Computer Vision (ECCV)*, 2018. 10

[61] R. Zhang, P. Isola, and A. A. Efros, "Colorful image colorization," in *2016 European Conference on Computer Vision (ECCV)*, 2016. 10

[62] M. Lucic, K. Kurach, M. Michalski, S. Gelly, and O. Bousquet, "Are gans created equal? a large-scale study.," *CoRR*, vol. abs/1711.10337, 2017. 10

[63] T. Karras, T. Aila, S. Laine, and J. Lehtinen, "Progressive growing of GANs for improved quality, stability, and variation," in *2018 International Conference on Learning Representations (ICLR)*, 2018. 10

[64] A. Radford, L. Metz, and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," in *2016 International Conference on Learning Representations (ICLR)*, 2016. 10, 11

[65] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, X. Chen, and X. Chen, "Improved techniques for training gans," in *2016 Advances in Neural Information Processing Systems*, vol. 29, 2016. 10, 11

[66] X. Chen, Y. Duan, R. Houthooft, J. Schulman, I. Sutskever, and P. Abbeel, "Infogan: Interpretable representation learning by information maximizing generative adversarial nets.," in *2016 Neural Information Processing Systems (NIPS)*, pp. 2172–2180, 2016. 10

[67] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5967–5976, 2017. 11

[68] T.-C. Wang, M.-Y. Liu, J.-Y. Zhu, A. Tao, J. Kautz, and B. Catanzaro, "High-resolution image synthesis and semantic manipulation with conditional gans.," *CoRR*, vol. abs/1711.11585, 2017. 11

[69] T. Park, M.-Y. Liu, T.-C. Wang, and J.-Y. Zhu, "Semantic image synthesis with spatially-adaptive normalization," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2332–2341, 2019. 11

[70] T.-C. Wang, M.-Y. Liu, J.-Y. Zhu, G. Liu, A. Tao, J. Kautz, and B. Catanzaro, "Video-to-video synthesis," in *2018 Neural Information Processing Systems (NIPS)*, p. 1152–1164, 2018. 11

[71] L. Weng, "What are diffusion models?." https://lilianweng.github.io/posts/2021-07-11-diffusion-models/, 2021. Accessed: 2023-03-16. 11

[72] D. P. Kingma and M. Welling, "Auto-Encoding Variational Bayes," in *2014 International Conference on Learning Representations (ICLR)*, 2014. 11, 14

[73] A. Asperti, D. Evangelista, and E. Piccolomini, "A survey on variational autoencoders from a green ai perspective," *SN Computer Science*, vol. 2, 07 2021. 11

[74] A. van den Oord, O. Vinyals, and k. kavukcuoglu, "Neural discrete representation learning," in *Advances in Neural Information Processing Systems*, vol. 30, 2017. 11

[75] A. Ramesh, M. Pavlov, G. Goh, S. Gray, C. Voss, A. Radford, M. Chen, and I. Sutskever, "Zero-shot text-to-image generation," in *2021 International Conference on Machine Learning (ICML)*, vol. 139, pp. 8821–8831, 2021. 11

[76] J. Sohl-Dickstein, E. Weiss, N. Maheswaranathan, and S. Ganguli, "Deep unsupervised learning using nonequilibrium thermodynamics," in *2015 International Conference on Machine Learning (ICLR)*, vol. 37, pp. 2256–2265, 2015. 12

[77] Y. Song, J. Sohl-Dickstein, D. P. Kingma, A. Kumar, S. Ermon, and B. Poole, "Score-based generative modeling through stochastic differential equations," in *2021 International Conference on Learning Representations (ICLR)*, 2021. 13

[78] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, and I. Sutskever, "Learning transferable visual models from natural language supervision," in *2021 International Conference on Machine Learning (ICML)*, 2021. 13, 49

[79] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An image is worth 16x16 words: Transformers for image recognition at scale," *2021 International Conference on Machine Learning (ICLR)*, 2021. 14

[80] L. Zhang and M. Agrawala, "Adding conditional control to text-to-image diffusion models," *arXiv*, vol. abs/2302.05543, 2023. 14

[81] H. Chefer, Y. Alaluf, Y. Vinker, L. Wolf, and D. Cohen-Or, "Attend-and-excite: Attention-based semantic guidance for text-to-image diffusion models," *ArXiv*, vol. abs/2301.13826, 2023. 14, 18

[82] K. Zhou, J. Yang, C. C. Loy, and Z. Liu, "Learning to prompt for vision-language models," *Int. J. Comput. Vision*, vol. 130, p. 2337–2348, sep 2022. 14

[83] Y. Wen, N. Jain, J. Kirchenbauer, M. Goldblum, J. Geiping, and T. Goldstein, "Hard prompts made easy: Gradient-based discrete optimization for prompt tuning and discovery," *ArXiv*, vol. abs/2302.03668, 2023. 15

[84] Y. Hao, Z. Chi, L. Dong, and F. Wei, "Optimizing prompts for text-to-image generation," *ArXiv*, vol. abs/2212.09611, 2022. 15

[85] S.-c. Z. Quan-shi ZHANG, "Visual interpretability for deep learning: a survey," *Frontiers of Information Technology & Electronic Engineering*, vol. 19, no. 1, p. 27, 2018. 15

[86] K. Sirotkin, P. Carballeira, and M. Escudero-Viñolo, "A study on the distribution of social biases in self-supervised learning visual models," in *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10432–10441, 2022. 15, 16

[87] S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," in *2017 Advances in Neural Information Processing Systems*, vol. 30, p. 4768–4777, 2017. 15

[88] R. Fong, M. Patrick, and A. Vedaldi, "Understanding deep networks via extremal perturbations and smooth masks," in *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 2950–2958, 2019. 15

[89] D. Erhan, Y. Bengio, A. Courville, and P. Vincent, "Visualizing higher-layer features of a deep network," *Technical Report, Univeristé de Montréal*, 01 2009. 15

[90] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-cam: Visual explanations from deep networks via gradient-based localization," in *2017 IEEE International Conference on Computer Vision (ICCV)*, pp. 618–626, 2017. 16

[91] K. Simonyan, A. Vedaldi, and A. Zisserman, "Deep inside convolutional networks: Visualising image classification models and saliency maps," *CoRR*, vol. abs/1312.6034, 2013. 16, 17

[92] R. Fong, M. Patrick, and A. Vedaldi, "Understanding deep networks via extremal perturbations and smooth masks," in *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 2950–2958, 2019. 17

[93] A. Mordvintsev, C. Olah, and M. Tyka, "Inceptionism: Going deeper into neural networks," *Google Res. Blog*, 2015. 17

[94] R. Geirhos, P. Rubisch, C. Michaelis, M. Bethge, F. A. Wichmann, and W. Brendel, "Imagenet-trained cnns are biased towards texture; increasing shape bias improves accuracy and robustness.," in *2019 International Conference on Learning Representations (ICLR)*, 2019. 16

[95] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba, "Learning deep features for discriminative localization," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2921–2929, 2016. 16

[96] B. Zhou, A. Khosla, À. Lapedriza, A. Oliva, and A. Torralba, "Object detectors emerge in deep scene cnns," *CoRR*, vol. abs/1412.6856, 2014. 17

[97] R. C. Fong and A. Vedaldi, "Interpretable explanations of black boxes by meaningful perturbation," *2017 IEEE International Conference on Computer Vision (ICCV)*, pp. 3449–3457, 2017. 17

[98] A. Mahendran and A. Vedaldi, "Understanding deep image representations by inverting them," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, (Los Alamitos, CA, USA), pp. 5188–5196, IEEE Computer Society, jun 2015. 17

[99] Y. Goyal, Z. Wu, J. Ernst, D. Batra, D. Parikh, and S. Lee, "Counterfactual visual explanations," in *2019 International Conference on Machine Learning (ICML)*, vol. 97, pp. 2376–2384, 2019. 17

[100] A. Ghorbani, A. Abid, and J. Zou, "Interpretation of neural networks is fragile," *2019 AAAI Conference on Artificial Intelligence*, vol. 33, pp. 3681–3688, Jul. 2019. 17

[101] Y. Liu, Y. Wei, H. Yan, G. Li, and L. Lin, "Causal reasoning meets visual representation learning: A prospective study," *Machine Intelligence Research*, vol. 19, pp. 485 – 511, 2022. 17

[102] S. Wiegreffe and Y. Pinter, "Attention is not not explanation," in *2019 Conference on Empirical Methods in Natural Language Processing and the International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pp. 11–20, 2019. 18

[103] K. Clark, U. Khandelwal, O. Levy, and C. D. Manning, "What does BERT look at? an analysis of BERT's attention," in *2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pp. 276–286, 2019. 18

[104] M. Deb, B. Deiseroth, S. Weinbach, P. Schramowski, and K. Kersting, "Atman: Understanding transformer predictions through memory efficient attention manipulation," *ArXiv*, vol. abs/2301.08110, 2023. 18

[105] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems*, pp. 5998–6008, 2017. 22

[106] L. Liu, Y. Ren, Z. Lin, and Z. Zhao, "Pseudo numerical methods for diffusion models on manifolds," in *2022 International Conference on Learning Representations (ICLR)*, 2022. 34

[107] A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, T. Xiao, S. Whitehead, A. C. Berg, W.-Y. Lo, P. Dollár, and R. Girshick, "Segment anything," *ArXiv*, vol. abs/2304.02643, 2023. 34

[108] J. Xu, S. Liu, A. Vahdat, W. Byeon, X. Wang, and S. De Mello, "ODISE: Open-Vocabulary Panoptic Segmentation with Text-to-Image Diffusion Models," *ArXiv*, vol. abs/2303.04803, 2023. 47

[109] S. Li, T. Fischer, L. Ke, H. Ding, M. Danelljan, and F. Yu, "Ovtrack: Open-vocabulary multiple object tracking," in *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 47

[110] Y. Takagi and S. Nishimoto, "High-resolution image reconstruction with latent diffusion models from human brain activity," in *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 49

Appendix

Appendix A

DAAM Layers Analysis

As part of the study on the attention arrays of DAAM, an exploration was conducted to observe how the attentions from different blocks and epochs aligned, in terms of mIoU, with the objects of the dataset generated. Due to the scope of the project, this line of investigation was left as a direction for future work. This appendix provides a summary of several results from the experiment conducted.

In the original version of DAAM [20], to construct the heatmap for a token, $D_k^{\mathbb{R}}$, all attention arrays from different blocks, heads, and timestamps are aggregated (3.3), meaning that they are summed along the i, l, and t dimensions before aggregation.

To study the variations in attention arrays between different blocks, the mIoU of the objects in the generated dataset was measured without aggregating these dimensions. Specifically, three variants were studied: aggregating only the multi-attention heads (l), aggregating the attention heads of each block across all epochs (l, t), and aggregating the attention heads and blocks for a single epoch (l, i). In other words:

$$\begin{aligned} D^{\mathbb{R}}_{X,k,t,i}[x,y] &:= \sum_{l} \tilde{F}^{(i)\downarrow}_{X,t,k,l}[x,y] + \tilde{F}^{(i)\uparrow}_{X,t,k,l}[x,y] \ , \\ D^{\mathbb{R}}_{X,k,i}[x,y] &:= \sum_{t,l} \tilde{F}^{(i)\downarrow}_{X,t,k,l}[x,y] + \tilde{F}^{(i)\uparrow}_{X,t,k,l}[x,y] \ , \text{ and} \\ D^{\mathbb{R}}_{X,k,t}[x,y] &:= \sum_{t,l} \tilde{F}^{(i)\downarrow}_{X,t,k,l}[x,y] + \tilde{F}^{(i)\uparrow}_{X,t,k,l}[x,y]. \end{aligned} \tag{A.1}$$

In Figure A.1, we present the preliminary results obtained from this experiment. The main matrix in the figure shows the mIoU scores when evaluating the heatmaps $D^{\mathbb{R}}_{X,k,t,i}$ on the entire generated dataset, including the classes "car," "person," "traffic light," and "rider."

The U-Net architecture of Stable Diffusion 2-base comprises 15 blocks with resolutions ranging from 16x16 to 64x64. In the figure, the blocks are arranged vertically, indicating whether they are upsample or downsample blocks. The images are generated through a process involving 31 iterations, represented along the horizontal axis.

A distinct pattern is observed in the blocks. The blocks with higher resolutions, such as 64x64, exhibit better alignment with the objects in the network, resulting in higher mIoU scores. In contrast, the deeper layers show mIoU scores close to 0. This discrepancy may be attributed to the Softmax operation with the other tokens in the text embedding (Eq. 3.1) not being effectively activated in lower layers. Further investigation is required, including a visual analysis of the heatmaps from these layers, to precisely determine the cause.

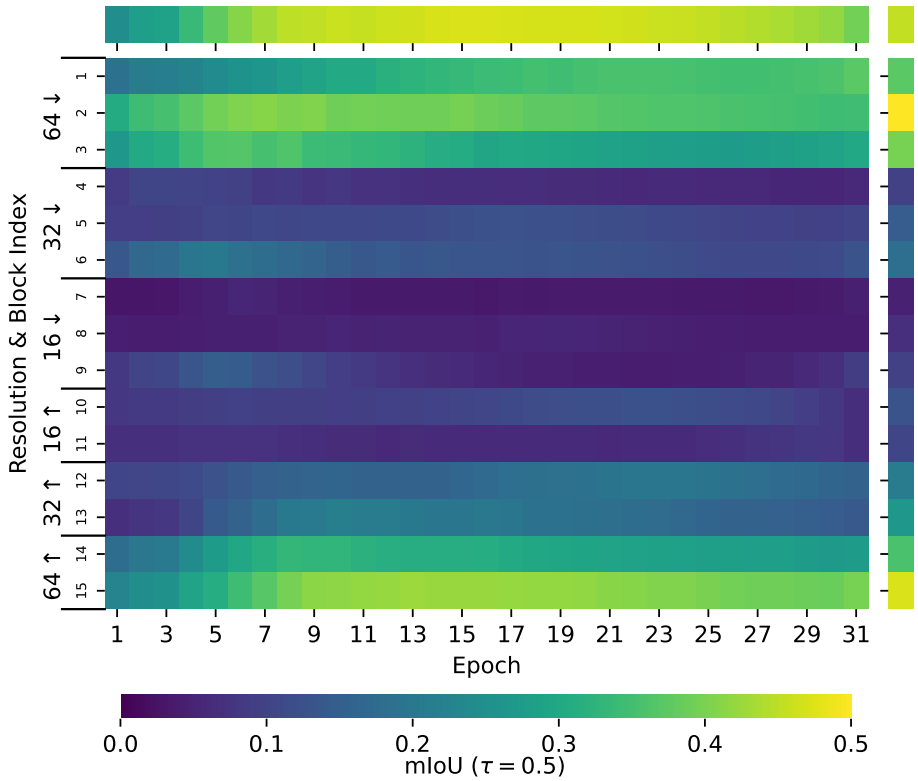
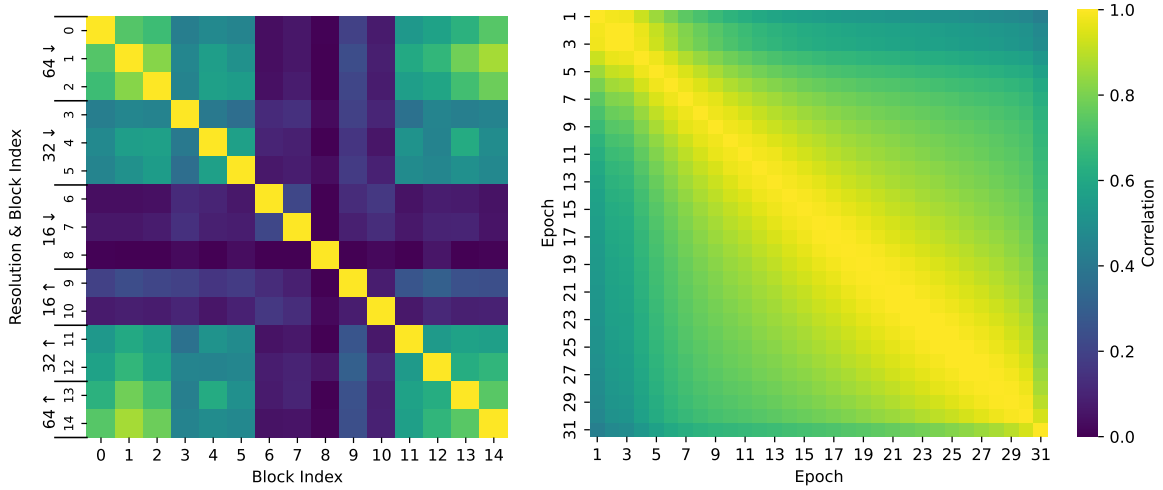


Figure A.1: Heatmap Analysis of DAAM Blocks and Epochs. The matrix illustrates the mIoU scores obtained from evaluating the heatmaps $D^{\mathbb{R}}_{X,k,t,i}$ on the generated dataset for different blocks and epochs. The vertical axis represents the U-Net blocks, ordered by their location in the U-net (downsample to upsample), while the horizontal axis corresponds to the epochs during the generation process. The additional top row shows the mIoU scores when aggregating all blocks in one epoch, and the additional right column displays the mIoU scores when aggregating a block across all epochs. The results reveal the variation in alignment between attention maps and ground truth objects across different blocks, epochs, and aggregation strategies.

Regarding the aggregation of a complete block within a single epoch (additional row on top of the matrix in Figure A.1), it is noticeable that the attention maps from the first and last epochs exhibit poorer alignment with the ground truth described by the token. This discrepancy may stem from the fact that, in the early epochs, the latent space is filled with noise, and the object has not yet fully formed. Similarly, in the last epochs, the attention maps may not prioritize the object as much, since the images are already close to their final state.

After examining the results disaggregated by layers in Figure A.1, it becomes evident that aggregating layers leads to higher mIoU scores for the resulting heatmaps. This observation suggests that different blocks contain complementary information, and choosing a single block would result in the loss of object-related information during segmentation. To assess the information shared among blocks, linear correlations were calculated between the aggregated heatmaps across all epochs ($D^{\mathbb{R}}_{X,k,i}$, subfigure A.2a) and across all blocks ($D^{\mathbb{R}}_{X,k,t}$, subfigure A.2a).

In the block correlation analysis (subfigure A.2a), it can be observed that blocks with higher resolutions (64x64 and 32x32) exhibit significant linear correlations. The



(a) Block correlations

(b) Epoch correlations

Figure A.2: Linear correlations among different elements in the DAAM heatmaps. Subfigure A.2a shows the linear correlations between the aggregated heatmaps across all epochs ($D^{\mathbb{R}}_{X,k,i}$) for each block, while Subfigure A.2b presents the linear correlations between the aggregated heatmaps across all blocks ($D^{\mathbb{R}}_{X,k,t}$) for each epoch. The color scale represents the correlation values, ranging from 0 to 1, where 0 indicates no correlation and 1 represents a perfect linear correlation.

strong correlation among the 64x64 blocks aligns with the high mIoU scores observed in the previous figure (A.1). If these blocks are activated in the regions where the objects are located (resulting in high mIoU scores), they will be activated in the same regions, thus explaining their linear correlation. However, the 32x32 blocks require further analysis, as they have low mIoU scores with the objects in the dataset but still exhibit moderate correlations (around 0.6) among themselves and with the 64x64 blocks. A more in-depth study is needed to determine the regions activated by these blocks.

Regarding the linear correlations between epochs (subfigure A.2b), a clear pattern emerges: heatmaps from nearby epochs are highly similar. This suggests that the focus of attention gradually shifts during the process, attending to different parts of the image. In the original work of DAAM [20], a brief ablation study was conducted, demonstrating that removing the attention maps from the first and second halves of the epochs resulted in lower mIoU scores. Although a more detailed analysis is required, this suggests that the attention maps across epochs are complementary, and aggregating all of them provides a more comprehensive mask for object segmentation.

It is worth noting that in the figure, the scale used to represent correlations ranges from 0 to 1 (rather than -1 to 1). This is because no results exhibited correlations below 0.

Appendix B

Text prompt optimization

This appendix includes additional figures related to the DAAM optimization experiment described in Section 4.4.

Specifically, it presents the loss vs epoch curves for the optimization of Linear DAAMs with different numbers of training samples (Figure B.2) and for the non-linear DAAM optimization (Figure B.3).

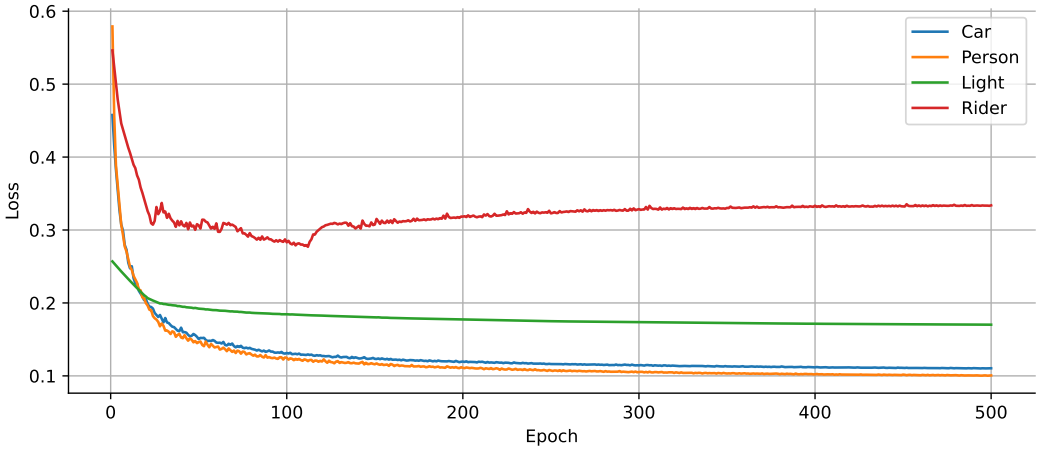
For the optimization process, we conducted 500 epochs, saving checkpoints of the embedding to be optimized every 30 steps. The final epoch, which yielded the best results in terms of mIoU on the test set, was selected. A learning rate of $lr = 3$ was utilized.

Additionally, IoU vs threshold curves for the training with different numbers of training samples are included (Figure B.2), as well as examples excluded from the main body of the work due to their similar results (Figure B.1).

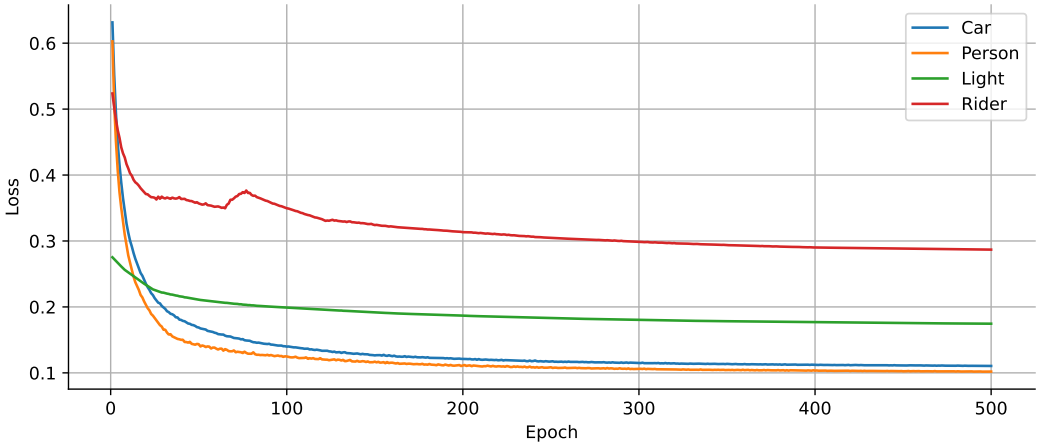
The optimization experiments were conducted on an Apple M1 Max laptop with 64GB of memory and a GPU with 32 cores. The optimization process required approximately 0.5 seconds per epoch per image.

(a) Car (b) Person (c) Traffic light (d) Rider

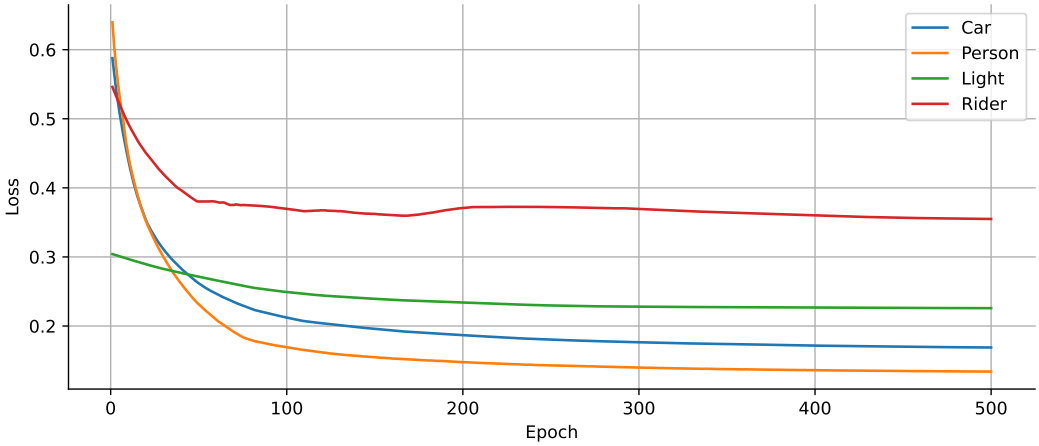
Figure B.1: Examples of DAAM-generated soft heatmaps with a optimized prompt. Each subfigure displays an image for each class with the overlayed soft heatmap generated using DAAM.



(a) Optimization with 1 image per class



(b) Optimization with 2 images per class



(c) Optimization with 5 images per class

Figure B.2: Linear DAAM optimization loss curves

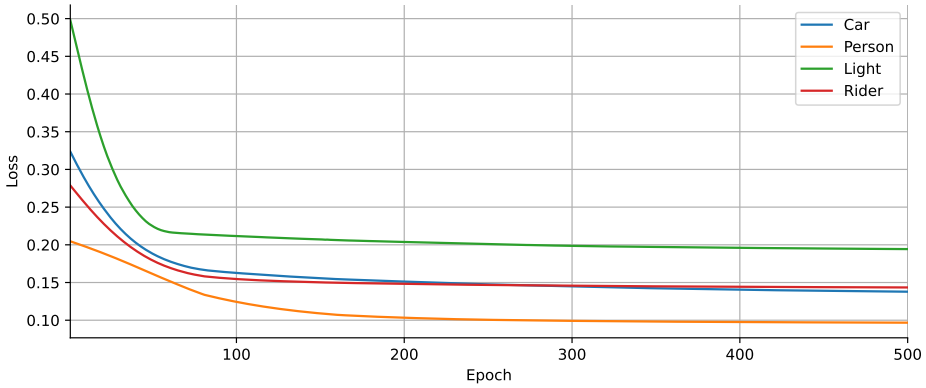


Figure B.3: DAAM optimization loss curves with 2 images as train. The token embedding optimized contains 2 words: The main token (car/Person/Light/Rider) and a background token (with the complementary target mask).

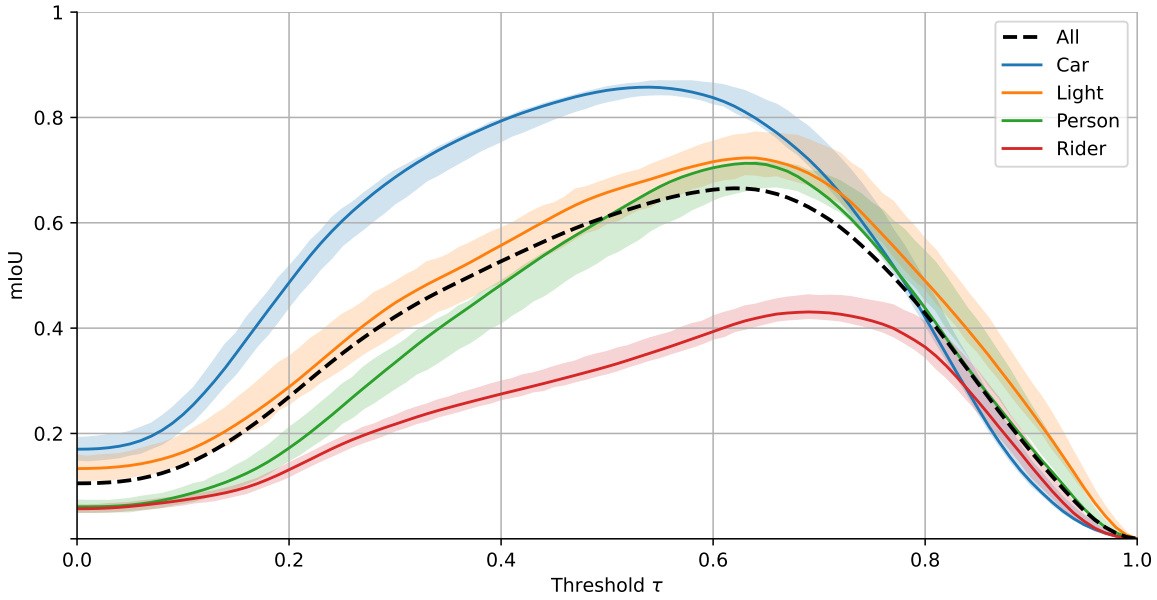
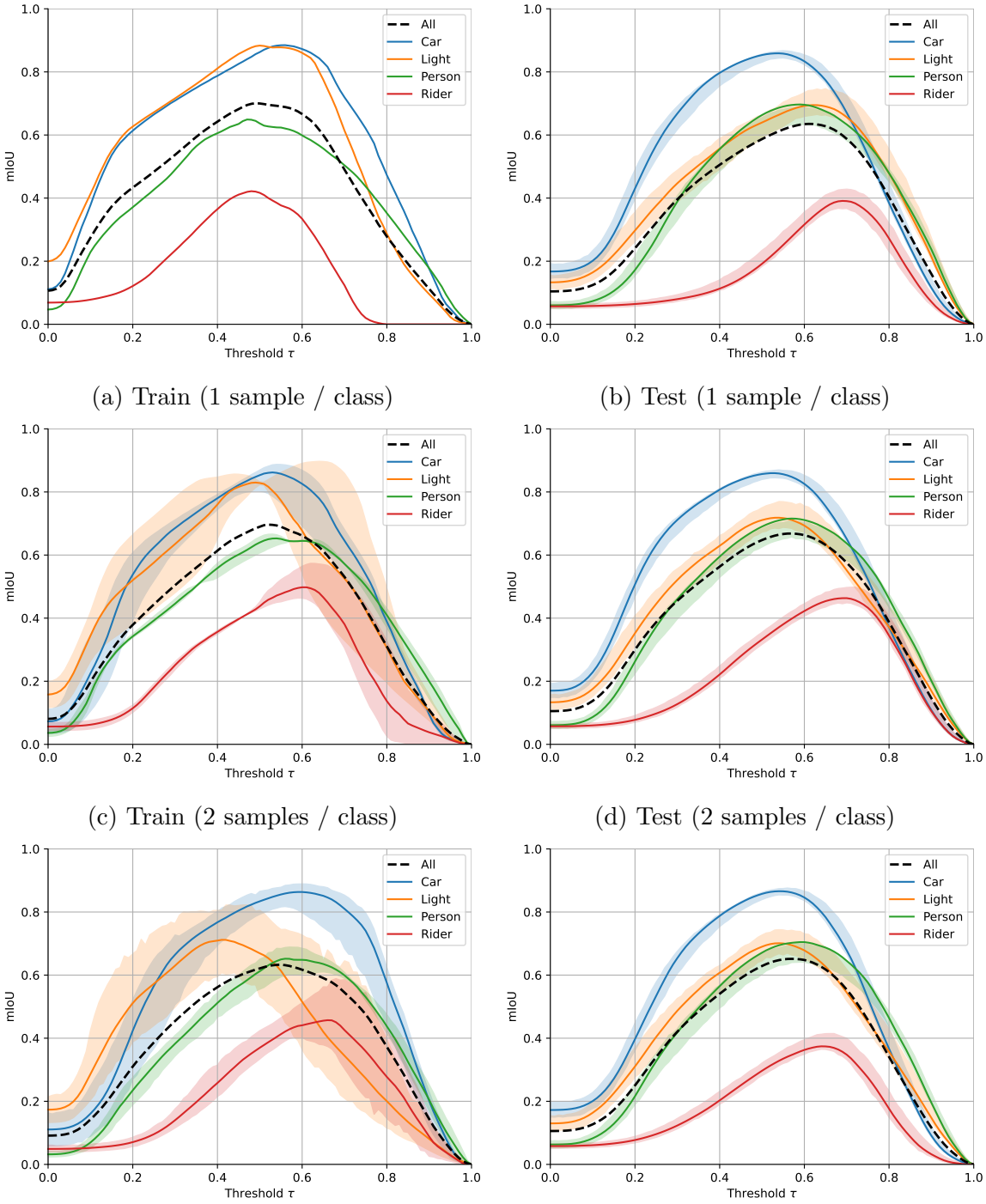


Figure B.4: Optimized DAAM (Non-linear Open Vocabulary DAAM), mIoU vs threshold curves. Performance of the test set



(a) Train (1 sample / class)

(b) Test (1 sample / class)

(c) Train (2 samples / class)

(d) Test (2 samples / class)

(e) Train (5 samples / class)

(f) Test (5 samples / class)

Figure B.5: Linear DAAM with Optimized prompt IoU

Appendix C

Dataset

This appendix includes the 200 images of the dataset used in the experiments, which were generated synthetically for this purpose. The dataset consists of four classes: "car" (Figs. C.1), "person" (Figs. C.2), "traffic light" (Figs. C.3), and "rider" (Figs. C.4).

The images were generated using the Stable Diffusion 2-base architecture trained by StabilityAI [1]. Specifically, revision d28fc8045793886e512c5389771d3b3d560f9575 of the model was utilized.

To produce images with different outcomes while using the same text prompt per class, a random seed was varied. A total of 50 numbers were randomly generated and used as seeds to generate the images for each of the four classes. The same set of seeds was applied to different classes intentionally to maintain similar spatial complexity across classes and minimize external factors that could influence the results. Images generated with the same seed (and therefore initialized with the same random noise vector) tend to exhibit similar spatial compositions. In the dataset images, one can observe how images in the same position within the image matrix of their class share similarities with images from other classes, occasionally featuring common elements such as a specific building or even a watermark.

The dataset was generated using an Apple M1 Max laptop with 64GB of memory and a GPU with 32 cores. The generation process required approximately 30 seconds per image.

[1] https://huggingface.co/stabilityai/stable-diffusion-2-base, accessed June 2023

Figure C.1: Images 1-50. Class “car”. Images generated with the prompt “A car in an urban environment“ varying the random seed.

Figure C.2: Dataset images 51-100. Class “Person”. Images generated with the prompt “A person in an urban environment“ varying the random seed.

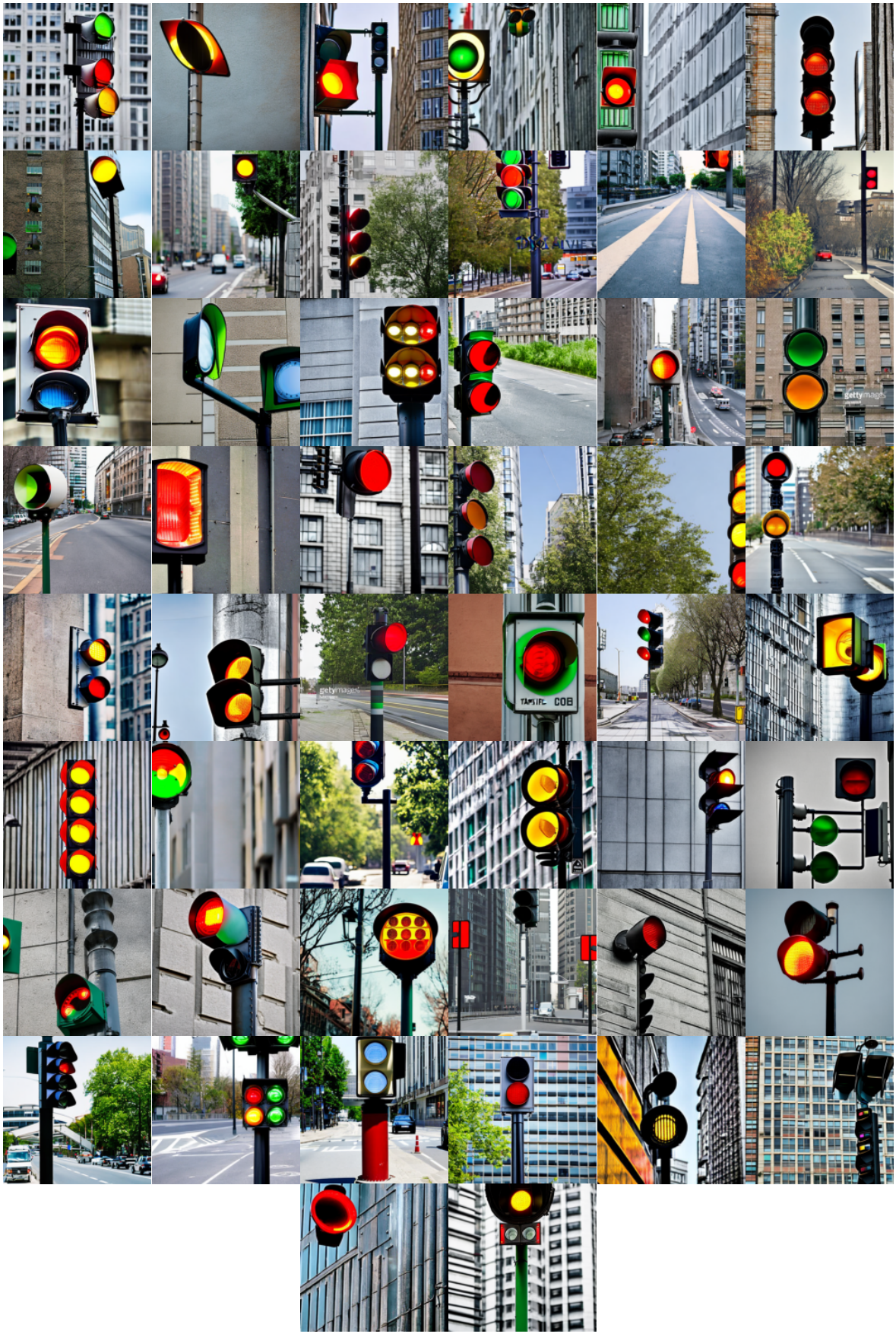


Figure C.3: Dataset images 101-150. Class “Traffic Light”. Images generated with the prompt “A traffic light in an urban environment“ varying the random seed.

Figure C.4: Images 151-200. Class “Rider”. Images generated with the prompt “A rider in an urban environment“ varying the random seed.