RecVis project topic H: Unleashing Text-to-Image Diffusion Models for Visual Perception

SEVI Melvin
ENS Paris-Saclay
melvin.sevi@ens-paris-saclay.fr

Abstract

Diffusion models have gained a lot of attention recently, due to their remarkable capacity to produce high-quality images from diverse prompts such as images or textual inputs. The use of diffusion within a compact latent space, as introduced by Stable Diffusion (SD), has open the door for significant innovations for example in video generation particularly because of it's ability to apply diffusion or denoising faster in training and inference. This study builds upon the advancements made by the Visual Perception with Pre-trained Diffusion (VPD) model, presented at ICCV23 by Zhao, Rao and al. Our project seeks to elevate the model's performance by taking the most advantage of Stable Diffusion. However this model is expensive to train due to a high number of parameters. Consequently, we aim to find insights to improve the model without increasing the number of learnable parameters, we also modify the level of noise of the image used as input of the model and then propose future approach of research directly deduced from our work.

1. Introduction

The VPD model [9], tackles a range of visual perception tasks, including referring segmentation, semantic segmentation, and depth estimation. It showcased incredible advancements although being recently outperformed by newer models like Unitext [6]. In this work, we only focus on the referring image segmentation application of the model. In the training phase, the VPD takes as input an image (unoised) and uses the pre-trained SD U-Net to make the image go through it one last time in the aim of retaining all the knowledge from SD during this single step. To return the segmentation mask based on the caption, it simultaneously takes a caption and establishes cross-modal attention [5] between textual embeddings extracted from the caption and feature maps from the pre-trained U-Net. Various attention maps and filters derived from the U-Net model are fed to a "classifier" for executing the intended task. This procedure allow the VPD model to benefit from SD knowledge for the specific task. With the aim of boosting the VPD model's efficiency, this study is divided into four stages. Initially, we reproduce the original author's findings during the inference phase on the dataset RefCOCO. We then try to improve the model by freezing the SD U-Net in the goal of fully exploiting SD but we notice that the tokens of the caption are not able to know on their own which part of the image it should attend to which highlight the importance of an adapter to refine the caption token embeddings. By removing the adapter, we only lose degrees of freedom. After that we try to vary the noise of the initial image fed in the model and then we compare the attention and filter maps and performance to finally give interesting ideas for future research. (github available at: https://github.com/melvinsevi/RecVis/tree/main)

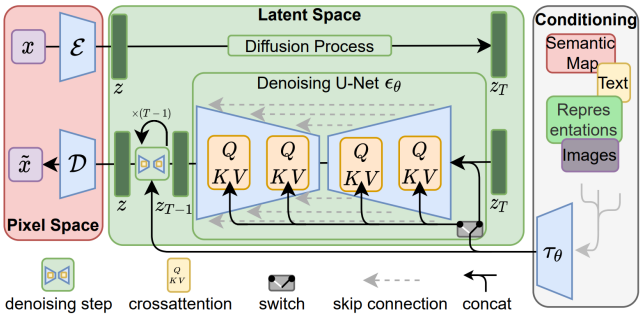


Figure 1. An encoder converts the input image into the latent space, noise is added to it over T steps. The U-Net predict the noise that was added and remove it over the T steps. This is followed by a decoder translating the encoded representation back into the original space. Meanwhile, cross-attention mechanisms link textual embeddings and the U-Net feature maps, guiding conditional image generation and temporal embeddings are fed into the U-Net to precise the step of denoising.

2. Background

Latent diffusion models Diffusion models represent a pioneering class of generative models capable of reconstructing the data distribution via learning the reverse process of a diffusion process. Consider a diffusion process characterized by z_t, representing the random variable at time step t (distribution corresponding to a noisy image at a certain step); each state transition conforms to a Gaussian distribution depicted below: The law of $z_t \in \mathbb{R}^{n \times n}$ knowing z_{t-1} is defined as:

$$z_t \sim \mathcal{N}(\sqrt{\alpha_t} z_{t-1}, (1-\alpha_t)\mathbf{I}), \tag{1}$$

where $\{\alpha_t, t \in \mathbb{N}\}$ denotes the collection of fixed coefficients governing the noise schedule. With reparametrization, we have:

$$z_t \sim \sqrt{\alpha_t} z_{t-1} + (1-\alpha_t)\epsilon, \quad \epsilon \sim \mathcal{N}(0, \mathbf{I}) \tag{2}$$

Using variational inference allows us to determine the reverse process through discovering the noise introduced at each time step, leading to the loss function:

$$L = \mathbb{E}_{t \sim \mathcal{U}[[1,T]], \mathbf{z}_0, \boldsymbol{\epsilon}} \left[\|\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta (\mathbf{z}_t, t)\|^2 \right] \tag{3}$$

.

In addition, there is also the image reconstruction loss which is optimized for the auto-encoder.

The particularity in SD is that the denoising process is done in a much smaller latent space to increase the noise and denoising speed.

3. VPD framework Inference and Training

Tailoring Stable Diffusion for Referred Image Segmentation This section focuses on the method used by VPD for referring image segmentation, which involves generating a segmentation mask based on an input image and an accompanying text caption. The VPD converts the image into a latent representation of size 4x64x64 with a VQ-GAN encoder [1]. The representation then goes through the SD U-Net, while the VPD framework also performs cross-attention between textual features and the current feature maps derived from the U-Net. Textual features are initially encoded via CLIP [3] before being refined by a MLP that the authors call a "text adapter" to suit the specific task. These features maps of the U-Net, along with associated attention maps from two precise resolutions **(32x32 and 16x16)** generated throughout the process, serve as inputs for a "classifier" which aims at producing the final segmentation mask. The classifier uses the same method as LAVT [7] to merge the different resolutions of feature maps together. It's just several convolution heads with concatenations and other convolutions to return the final segmentation mask. A visual representation of the complete VPD architecture can be found in the provided figure (2).

3.1. Reproduction of the Results in Inference and Experiments

Inference Setup and Results Just as in the paper, we perform the evaluation of the model on the widely used benchmark dataset RefCOCO. We will only focus on this dataset for computational reasons. It contains around 20K images and 50K annotated objects, with 142,209 expressions. We use the split from UC Berkeley with a training set of 42 404 images and 120 623 expressions. The validation has 3811 images and 10 833 expressions. We use the overall intersection-over-union (oIoU) as the metric for performance. The IoU metric calculates the ratio of common region between ground truth and predicted masks (intersection) to their collective coverage (union). The closest it is to 1, the better the segmentation is. The oIoU of a set is just the mean across all the images of the set.

We show exactly the same result as the paper in inference wich is an oIoU of 73.46. The checkpoints used for inference are the one provided in the codebase for the VPD model and the one used for SD is the v1-5-pruned-emaonly from Hugging-Face. We also test the performance of the model without normalization during inference and show that it decreases the performance. We tried to remove the adapter during inference but it obviously led to less good results. In the paper, they also affirm that training without the adapter led to lower performance. The table [1] display the impact on inference performance when excluding images normalization and the text adapter.

Table 1. **Comparison of Inference Results Across Normalization and/or Adapter Removal**

Threshold	P@0.5	P@0.6	P@0.7	P@0.8	P@0.9	oIoU	mIoU
Paper	**85.52**	**83.02**	**78.45**	**68.53**	**36.31**	**73.46**	**75.67**
Inference	85.52	83.02	78.45	68.53	36.31	73.46	75.67
No norm	84.74	81.77	76.58	66.25	33.34	72.11	74.55
No adaptater	83.63	80.77	75.38	65.74	31.23	71.43	73.82

3.2. Model Training

Training Setup and Results To assess the computational and qualitative efficiency of retraining the entire SD, which gather near 900M parameters and is initially trained on a single high-end NVIDIA A100 GPU (3$/hour on G-Cloud), we aimed to unlock its full potential while making it feasible to train on commonly available GPU on G-Cloud. Our strategy involved significantly reducing the number of trainable parameters. Concretely, we froze the SD U-Net and disabled the text adapter, thereby shrinking the parameter count from 899M to 39M. Despite this decrease, training continued to be computationally intensive even on top Nvidia GPUs like the T4, P100, or L4 also due to insufficient VRAM memory (unfeasible to obtain multiple GPU machines promptly).

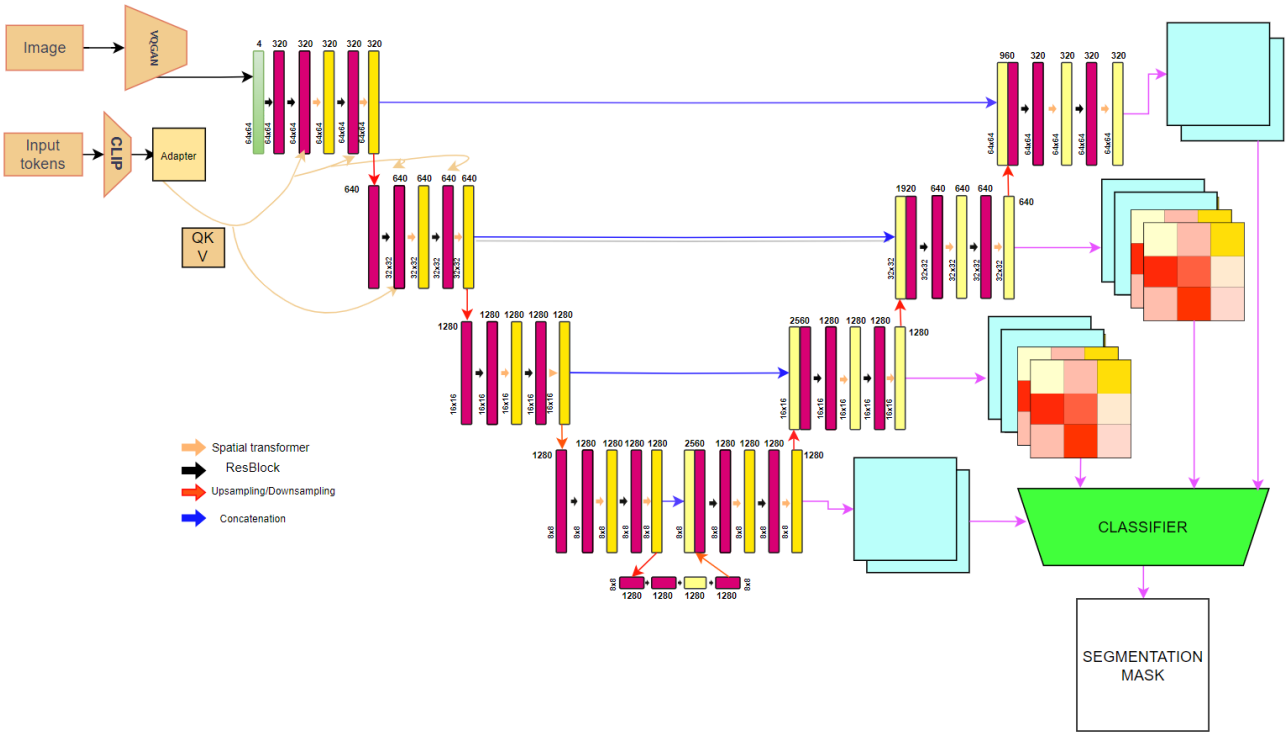


Figure 2. Figure that detail the VPD model architecture (made by me!)

To enhance training optimization, we chose a more manageable subset consisting of 3,000 referring expressions from the training set of RefCOCO with 1000 expressions in the validation set. The split is done by taking always the first images in the original split. By employing this split with a diminished batch size of 4 (32 in the paper), we completed training over two epochs using an identical initial learning rate of 0.005 and weight decay value of 0.01 mentioned in the original code. We also used Adam optimizer and cross entropy loss for training. Additionally, we attempted a complete dataset training run with a batch size of 4 (totaling 14 hours); however, the outcomes were poor (table [2]), likely attributed to the small batch size and quantity of epochs.

Experimental evidence suggests that enhancing the model through attention map is beneficial. So our real goal here is to provide the best attention maps to the "classifier". The problem is that implementing direct cross-attention between CLIP text features and the U-Net feature maps has shown less good performance. The removal of the adapter seems to only sacrifices flexibility without substantial improvement. Using only CLIP is clearly not enough to directly get good segmentation in the attention maps. **Indeed these attention maps in the latent space gets better during the generation process of an image by SD [8] but directly trying to get the attention maps from the cross attention between the caption tokens and the feature maps from the U-Net does not give the same results** (figure 2). We propose new ideas to address this problem at the end of the paper.

Table 2. **Performance on the RefCOCO dataset of the trained VPD with freezed SD with different thresholds and comparison with adapter/no adapter/full dataset with adapter**

Threshold	$P@0.5$	$P@0.6$	$P@0.7$	$P@0.8$	$P@0.9$	oIoU	mIoU
F-VPD + Adapter	0.74	0.21	0.00	0.00	0.00	5.41	4.63
F-VPD + No adapter	0.60	0.21	0.00	0.00	0.00	5.23	4.61
F-VPD + Full set + Adapter	0.00	0.00	0.00	0.00	0.00	0.41	0.43

Model Training and Use of Different Noise Scale The (VPD) model demonstrates remarkable performance, achieving 73.56 oIoU on the validation dataset of RefCoCO. However, when freezing SD parameters and removing the Adapter during training, the results are noticeably inferior. In pursuit of improved robustness, another approach explored varying levels of noise applied to the input images both through training and inference. To evaluate the impact of diverse noise scales, we compared attention maps generated across various timestamps during inference of the original VPD. Unfortunately, upon completing the training phase of the freezed SD VPD **(F-VPD)**, the model's performance on the validation set remains disappointing (table [3]), rendering attention map visualizations unnecessary. Interestingly, during inference (table [5]), at-

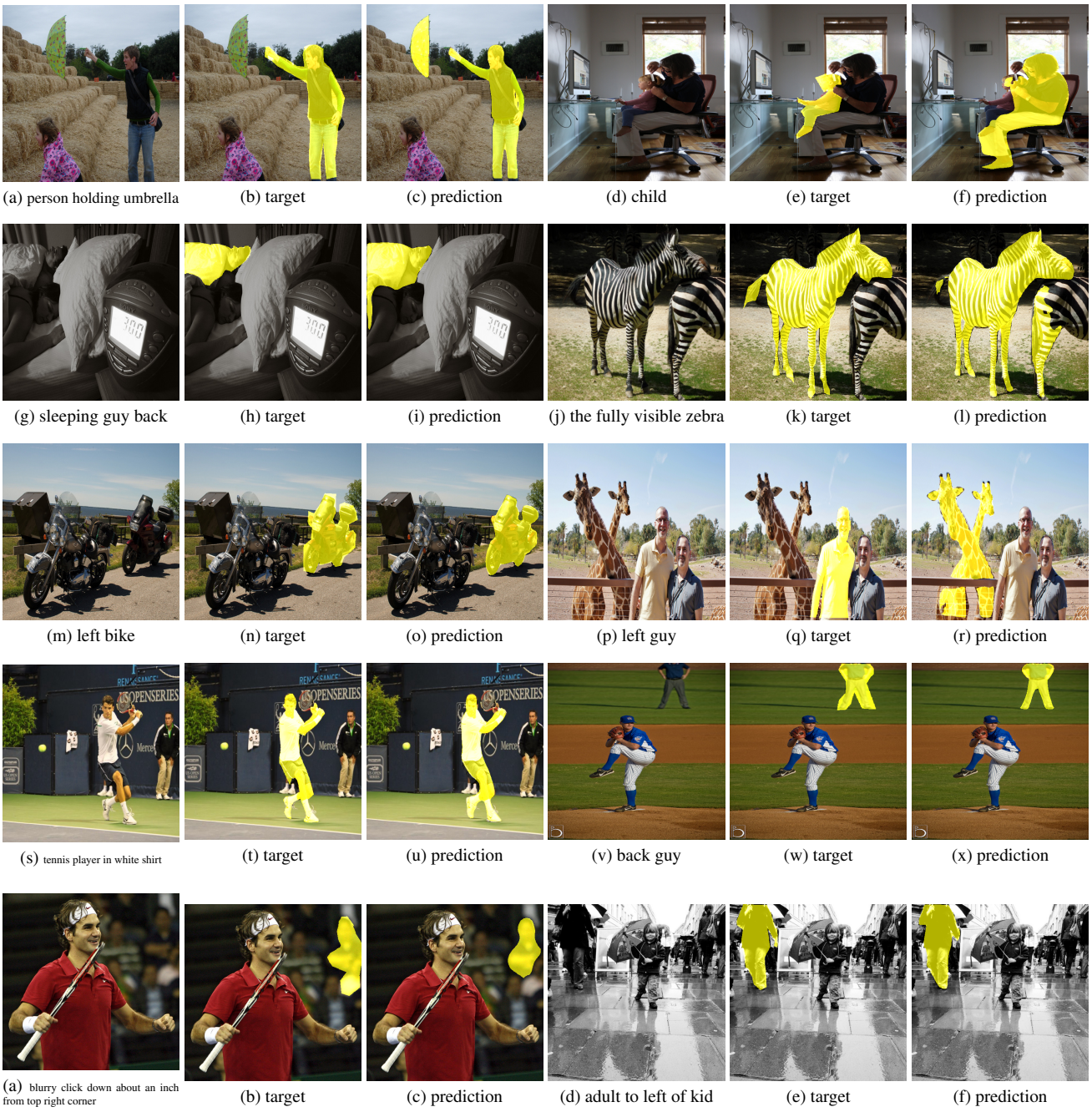
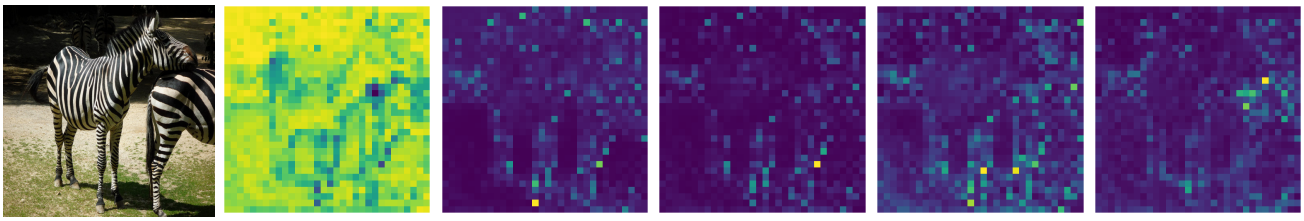


Figure 4. 512x512 segmentation mask from 10 images of the RefCoCo dataset. We superposed the predicted segmentation mask for the model and the expected mask

tention maps remain good (figure [7]) even under significant noise conditions up to a noise scale factor of 500. As a result, we hypothesize that either the latent representations of clean and noisy images retain considerable shared information, or the U-Net produces similar feature maps regardless of the imposed noise level. Ultimately, the objective involves obtaining optimal segmentations via enhanced attention map so we give insight in the last part on ideas to use different noise scale to obtain better attention maps.

4. Further research approach:

1. Utilizing High-Resolution Feature Maps: Our current implementation uses 32×32 and 16×16 resolution attention maps fed into the classifier. However, increasing the



(a) the fully visible zebra (b) [START] (c) "the" (d) "fully" (e) "visible" (f) "zebra"

Figure 5. Attention map of the VPD model without VPD pretrained weights but only SD pretrained weights with resolution 32x32 of an image of RefCoCo

Figure 6. Segmentation mask of 1 image of the RefCOCO dataset, each image is the respectively the segmentation mask for: the VPD model inference but with input noiscale image at 5, 100, 500 and the two last images are the segmentation mask for our trained VPD model with freezed SD for the noisecale 500 and 100 (we didn't just plot the mask but the confidence in each pixel wich is more interesting in this case since the results are still poor)

Table 3. **Performance on the Refcoco dataset at different thresholds and number of timestep for the VPD model trained with freezed SD but with the text adapter**

Timestep	P@0.5	P@0.6	P@0.7	P@0.08	P@0.9	oIoU	mIoU
0	0.60	0.21	0.00	0.00	0.00	5.41	4.63
5	0.57	0.19	0.00	0.00	0.00	5.39	4.57
100	0.49	0.8	0.00	0.00	0.00	5.31	4.54
200	0.48	0.08	0.00	0.00	0.00	5.39	4.51
500	0.5	0.18	0.00	0.00	0.00	5.23	4.53

Table 4. **Performance on the Refcoco dataset of VPD at inference but with different number of timesteps for the noise of the input image**

Timestep	P@0.5	P@0.6	P@0.7	P@0.8	P@0.9	oIoU	mIoU
T = 0	0.71	83.02	78.45	68.53	36.31	73.46	75.67
T = 5	85.45	82.95	78.40	68.48	36.25	73.13	75.34
T = 100	84.74	81.71	76.58	66.25	33.34	70.11	71.22
T = 200	84.68	81.65	76.53	66.20	33.29	71.06	73.50
T = 500	84.62	81.59	76.48	66.15	33.24	70.24	71.56

resolution to 64×64 may yield improved results by providing more detailed spatial information about objects within the image.

2. Fine-Tuning Using Low Rank Adaptation Matrices [2]: To avoid re-training the entire model, fine-tuning specific components can result in improved performance while reducing the required computation. By adjusting the different Q, K, and V weight matrices responsible for cross-attention between textual features and U-Net feature maps, lower rank matrices can be added to maintain information from pre-trained models without training numerous new parameters.
3. Obtaining Higher Resolution Attention Maps via Upsampling: Inspired by the (DAAM) paper [4], upsampling attention maps could deliver finer attention details and produce higher resolution output suitable for feeding into the subsequent classification module.
4. Generative Conditioned Denoising through Referring Captions: Adopting a generative approach to obtain better segmentation attention maps may be beneficial. Specifically, adding noise to the image and guiding its denoising based on the provided referring captions could potentially refine attention mechanisms and offer enhanced accuracy.

5. Conclusion

In this study, we aimed to improve the VPD model by Zha and Rao, focusing on enhancing attention maps for better performance. We initially attempted to simplify the model and fully exploit self-distillation, discovering the necessity of using an adapter for effective text embedding refinement. We also explored the impact of different noise scales as inputs, finding the VPD's robustness to higher noise scales and sensitivity to attention map quality. Unfortunately, due

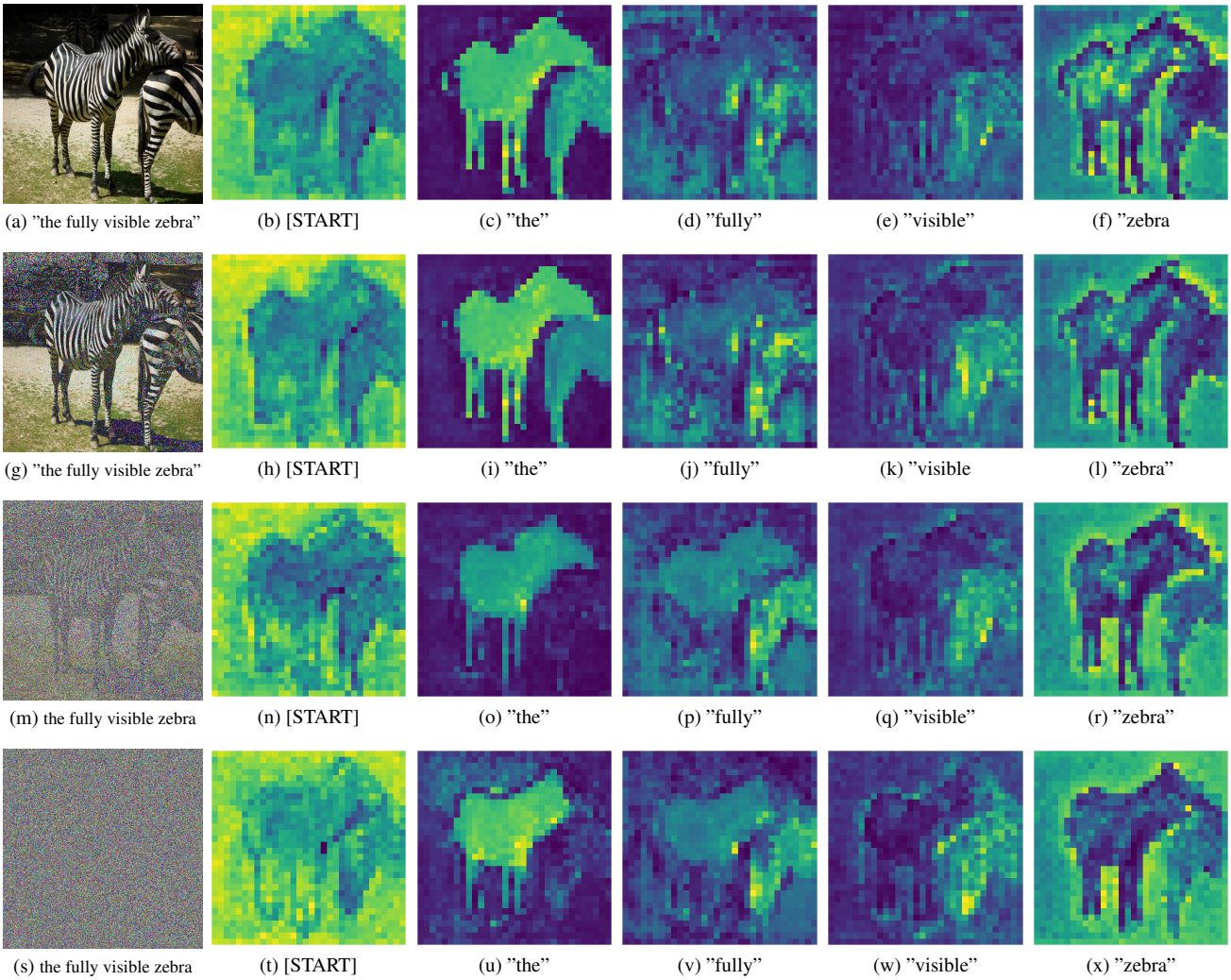


Figure 7. 32x32 attention maps at different noise scale (respectively t= 0,5,100,500) of the pretrained VPD model of 1 image of the validation set of the RefCOCO dataset

to computational limitations, we couldn't achieve conclusive results. Nonetheless, our findings provide valuable insights, suggesting promising directions for future research to refine the VPD model.

References

[1] Patrick Esser, Robin Rombach, and Björn Ommer. Taming transformers for high-resolution image synthesis. 2023. 2

[2] Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *Journal Name*, 2023. 5

[3] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. 2023. 2

[4] Raphael Tang, Linqing Liu, Akshat Pandey, Zhiying Jiang, Gefei Yang, Karun Kumar, Pontus Stenetorp, Jimmy Lin, and Ferhan Ture. Daam: Interpreting stable diffusion using cross attention. *Journal Name*, 2023. 5

[5] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. 2023. 1

[6] Bin Yan, Yi Jiang, Jiannan Wu, Dong Wang, Ping Luo, Zehuan Yuan, and Huchuan Lu. Universal instance perception as object discovery and retrieval. 2023. 1

[7] Zhao Yang, Jiaqi Wang, Yansong Tang, Kai Chen, Hengshuang Zhao, and Philip H.S. Torr. Lavt: Language-aware vision transformer for referring image segmentation. 2023. 2

[8] Tianyi Zhang, Zheng Wang, Jing Huang, Mohiuddin Muhammad Tasnim, and Wei Shi. A survey of diffusion-based image generation models: Issues and their solutions. 2024. 3

[9] Wenliang Zhao, Yongming Rao, Zuyan Liu, Benlin Liu, Jie

Zhou, and Jiwen Lu. Unleashing text-to-image diffusion models for visual perception. 2023. 1

,