Using Multiple Latent Layers in Stable Diffusion for Image Semantic Matching

Yu Ting Cheng[1], supervised by Prof. Kai Han[1]

[1]Department of Statistics and Actuarial Science, Faculty of Science, The University of Hong Kong

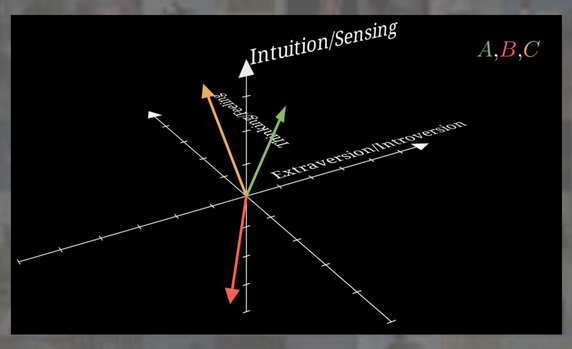
Introduction

This research aims to establish robust point-wise semantic matches between images, which is to establish matches between parts of images with similar features. In our method, we aim to utilise multiple latent layers within the Stable Diffusion pipeline as feature maps for a more robust semantic matching algorithm, especially when the margin of error is small.

What are Features

In short, features are vectors that carry semantic meaning, with each entry of the vector defining some kind of definition. We do not truly know the definitions of the features, but it can capture the meaning of text & images quite well.

For instance, if we rate a person by their MBTI values, we may see the following

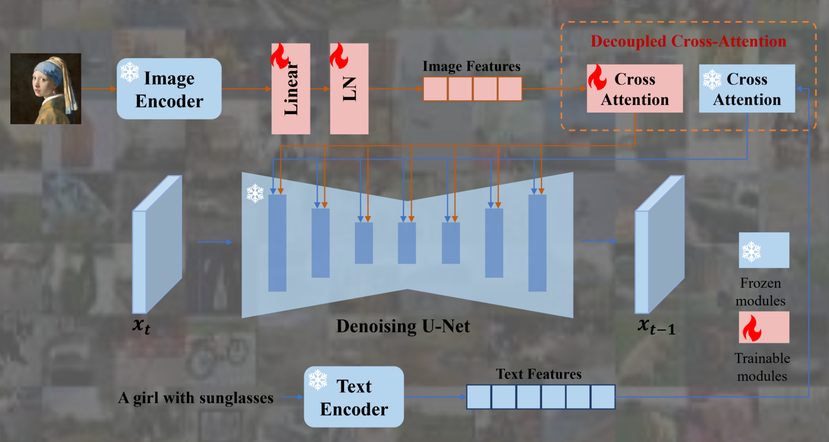


Different people may have different characteristics, if we rate every trait of a person and compare their similarities, we can see which people may be similar to each other.
In this scenario, person A may be more similar to person C than person B.

Similarly, by comparing the similarity between the features, we can see whether the information represented by the two features are similar.

What is Stable Diffusion

Stable Diffusion is a pretrained image generating model, which includes a forward and backward process during training. In the forward process, random (Gaussian) noise is added to the image until the image itself is just random noise. In the reverse process, the model will learn and try to predict the noise added to the image at a certain time. Once the noise added was predicted, the noise will be removed to reveal the original image. By providing the model with some form of input, such as text, or images, the model will generate an image based on the input from random noise.

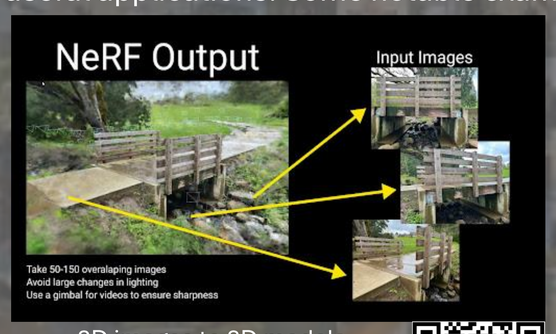


General pipeline of Stable Diffusion
https://docs.openvino.ai/2024/notebooks/stable-diffusion-ip-adapter-with-output.html

Learn More

Applications of Semantic Matching

While semantic matching itself seems quite abstract, it acts as a basis for many useful applications. Some notable examples include the following.



2D images to 3D models
Example: NeRF from NVIDIA
Photos taken from different angles will be used to generate 3D model of objects or even environments. The model must first understand the relationship between the points in different viewpoints to plot accurate 3D voxels

Website

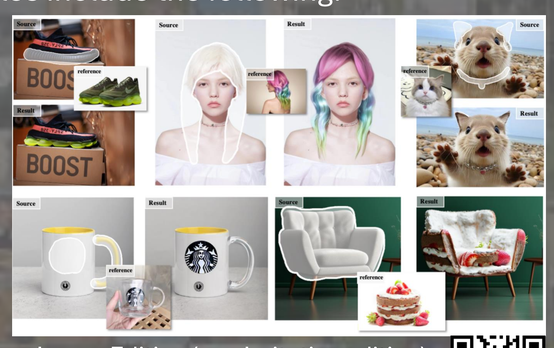


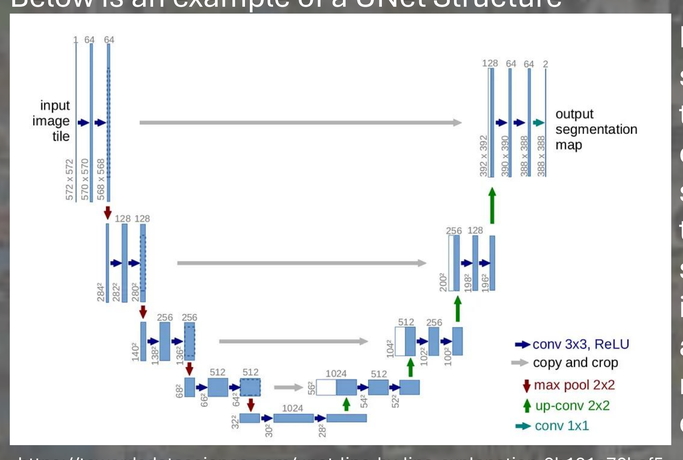
Image Editing (e.g. Imitative editing)
Example: Mimic Brush
By only marking regions on the source image, the image editing model can understand the related regions in the reference image and imitate the results on the source image
This requires the model to relate semantically similar parts between two images

Demo

Website

Point-wise Semantic Matching Using Stable Diffusion

For predicting the noise added to the image, Stable Diffusion uses the UNet model. Below is an example of a UNet Structure



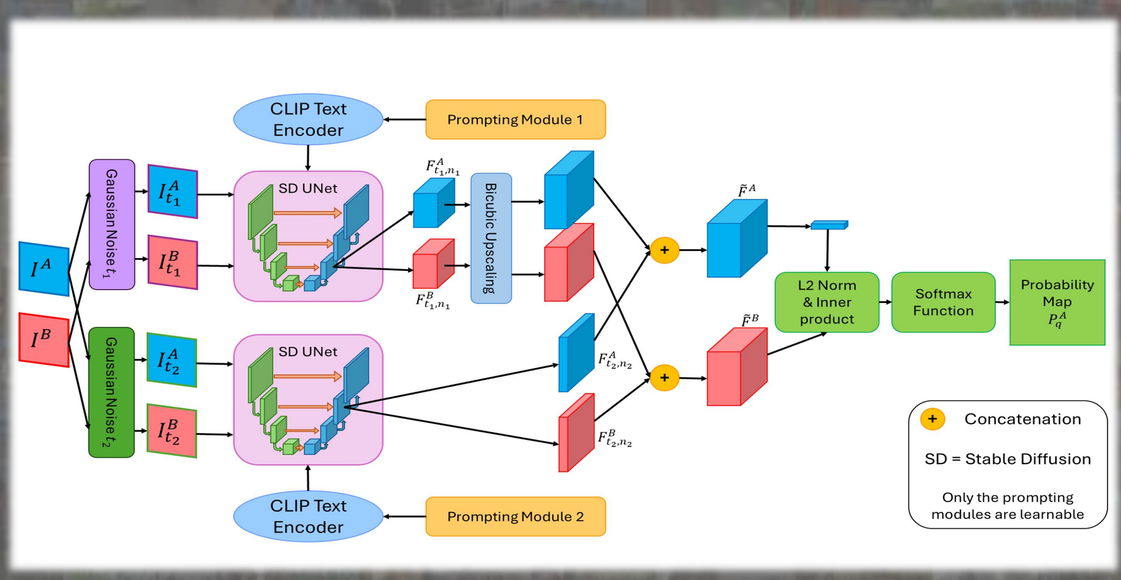
https://towardsdatascience.com/unet-line-by-line-explanation-9b191c76baf5

From the UNet model, there are some intermediate layers between the input and the output, those are called latent layers. It was previously suggested that the latent layers in the decoder side (right side) may store semantic information of the image, and therefore can be used as a feature map. The Stable Diffusion model can be utilised as a feature extractor.

Using Multiple Latent Layers for Point-wise Semantic Matching

It was suggested that different latent layer and timestep combinations in the Stable Diffusion UNet captures different semantic information, where abstract semantic information can be found in earlier layers of the decoder with a larger timestep, local details and texture with higher resolution can be found in later layers with a smaller timestep. Therefore, we have suggested the following method, where the second and the third latent layer of the UNet, each with different timestep and prompt embedding to allow for more variation in semantic information.

A visualisation of our pipeline, where two separate latent layers was extracted from the Stable Diffusion (SD) UNet structure for semantic matching.



From the table below, we can see that our method performs better than previous similar methods when the margin of error is small.

Method	Aero	Bike	Bird	Boat	Bottle	Bus	Car	Cat	Chair	Cow	Dog	Horse	Motor	Person	Plant	Sheep	Train	TV	All
DIFT	45.78	37.01	59.23	17.80	29.03	31.40	26.86	70.86	24.44	52.44	37.90	41.87	33.09	31.46	31.11	29.30	43.78	32.77	37.26
DIFT[6] bicubicly upsampled	46.67	40.04	63.09	18.44	29.86	32.42	27.62	72.31	26.99	54.80	39.75	42.26	34.30	33.51	34.89	29.53	46.17	35.85	39.09
DIFT[6] with dual layer	48.96	42.00	61.53	21.68	31.87	35.32	28.68	69.92	30.45	55.55	40.31	41.83	36.25	32.50	38.51	29.31	50.76	47.84	41.14
SD4Match-Single[5]	61.04	47.34	62.65	41.72	42.69	61.55	57.47	71.96	48.91	64.36	51.58	55.44	46.88	56.64	44.13	36.64	65.58	56.41	53.64
SD4Match-Class[5]	65.23	49.19	67.56	47.25	43.13	73.87	58.76	75.14	49.22	61.10	52.09	55.48	54.65	66.72	40.96	40.19	72.80	68.14	57.40
SD4Match-CPM[5]	64.27	48.41	63.94	44.03	45.37	71.79	58.68	74.57	49.95	64.85	53.36	57.26	52.47	57.38	42.38	44.68	70.69	60.50	56.48
Single (Ours)	65.82	54.59	68.64	48.70	42.10	76.43	69.96	76.99	61.55	72.34	60.25	60.14	58.54	65.33	51.19	**48.57**	75.24	73.38	62.18
Class (Ours)	70.60	53.67	**76.34**	**53.00**	46.05	81.55	67.59	77.14	**65.53**	70.21	58.33	59.89	58.22	**75.16**	**55.62**	43.98	79.54	**77.27**	64.58
CPM (Ours)	**70.65**	**57.37**	74.16	51.14	**47.77**	**81.61**	**73.36**	**80.75**	62.71	**74.99**	**61.52**	**62.93**	**61.66**	64.66	51.71	47.30	**81.28**	74.42	**65.01**

Class-wise Evaluation of SPair-71k Dataset (which makes up the background) with α =0.05 the best results in each class is **bolded**.

Conclusion

By further utilizing the potential within different latent layers of the Stable Diffusion model, we can further improve the accuracy and reduce the margin of error for image point-wise semantic matching

香 港 大 學
THE UNIVERSITY OF HONG KONG

LAIDLAW FOUNDATION