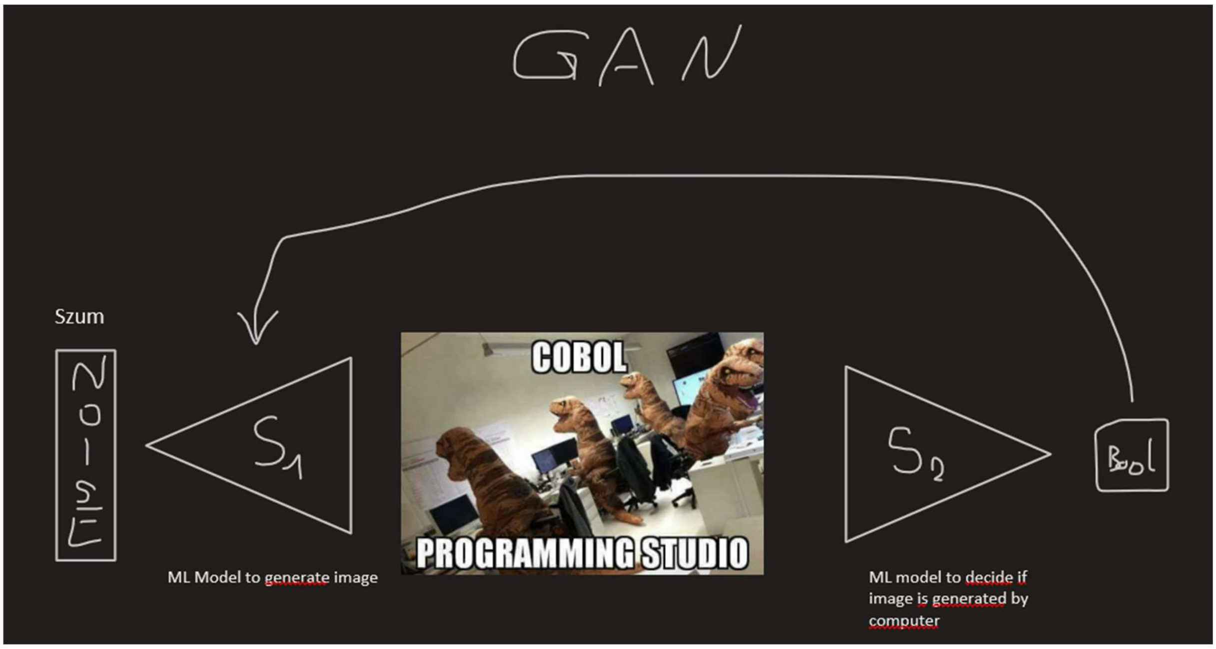
Stable Diffusion – Use-Cases

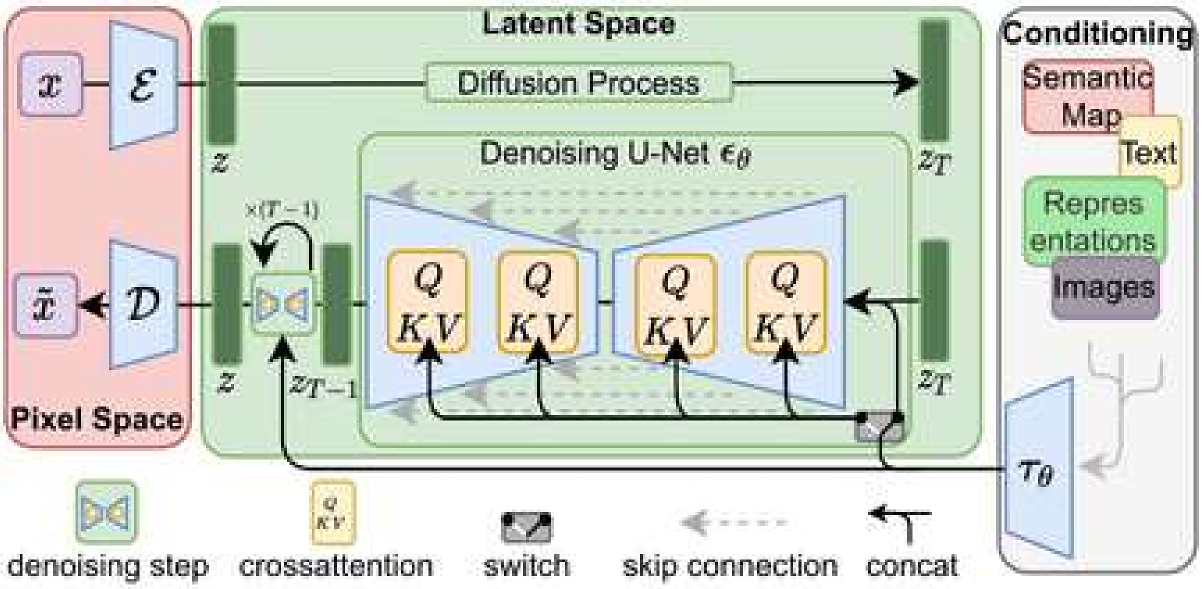
1. Models used before – Stable Diffusion:



Disadvantages:

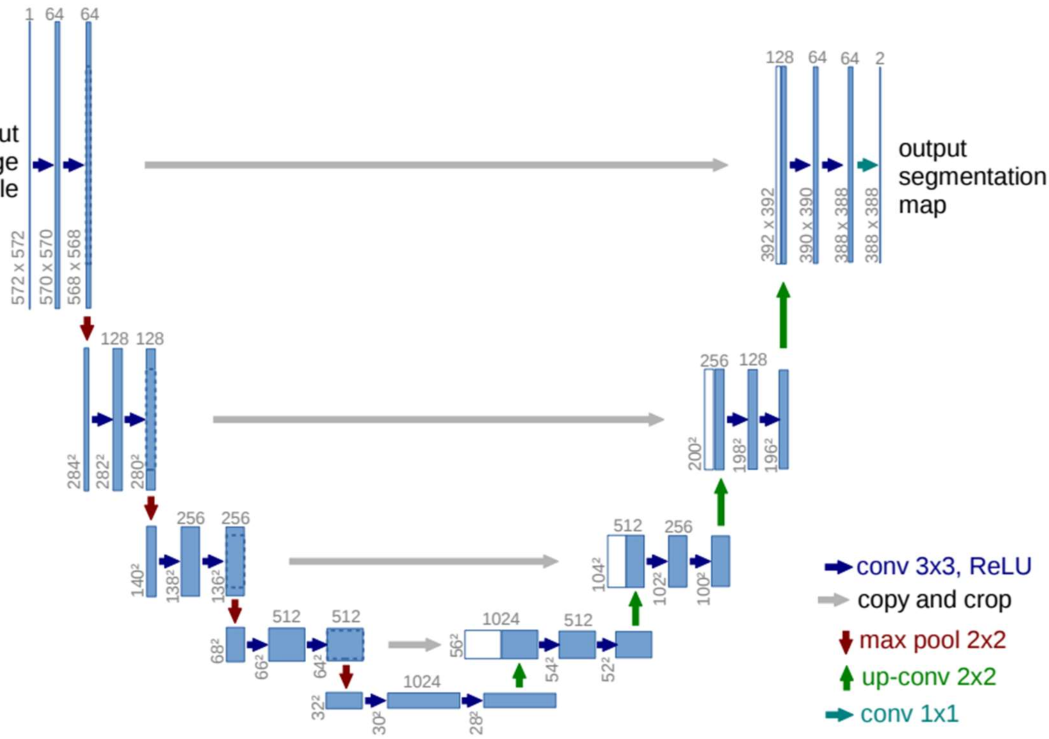
- Generating by S1 the same picture which S2 really “liked” over and over again.
- Lack of possibility of going inside the process of generating the image and thus - manipulating this process.

2. How Stable Diffusion Works:



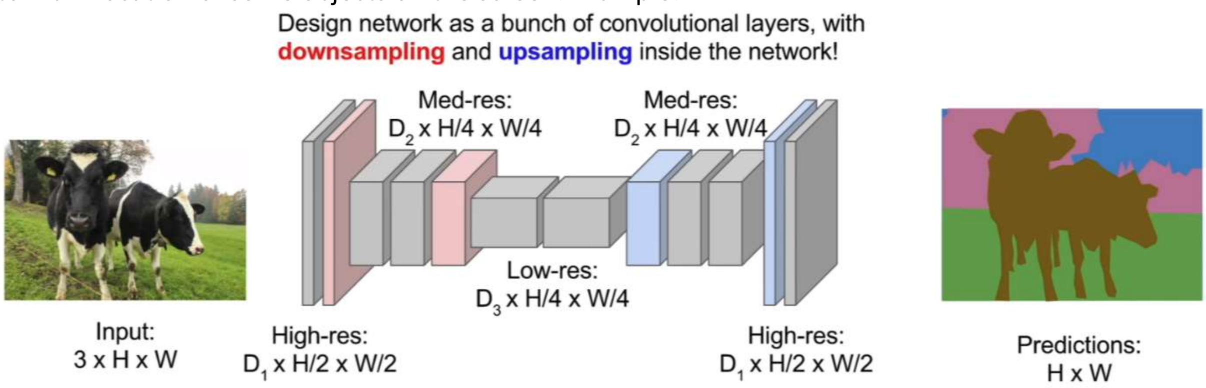
Advantages:

- Iterable process of image gereration.



U-Net architecture from "*U-Net: Convolutional Networks for Biomedical Image Segmentation*" paper. Source: https://arxiv.org/abs/1505.04597

In summary U-Net is more sophisticated form of CNN (Convolutional Neural Network) and its goal is to mark location of some objects on the screen. Example:



Source: https://calvinfeng.gitbook.io/machine-learning-notebook/supervised-learning/convolutional-neural-network/segmentation

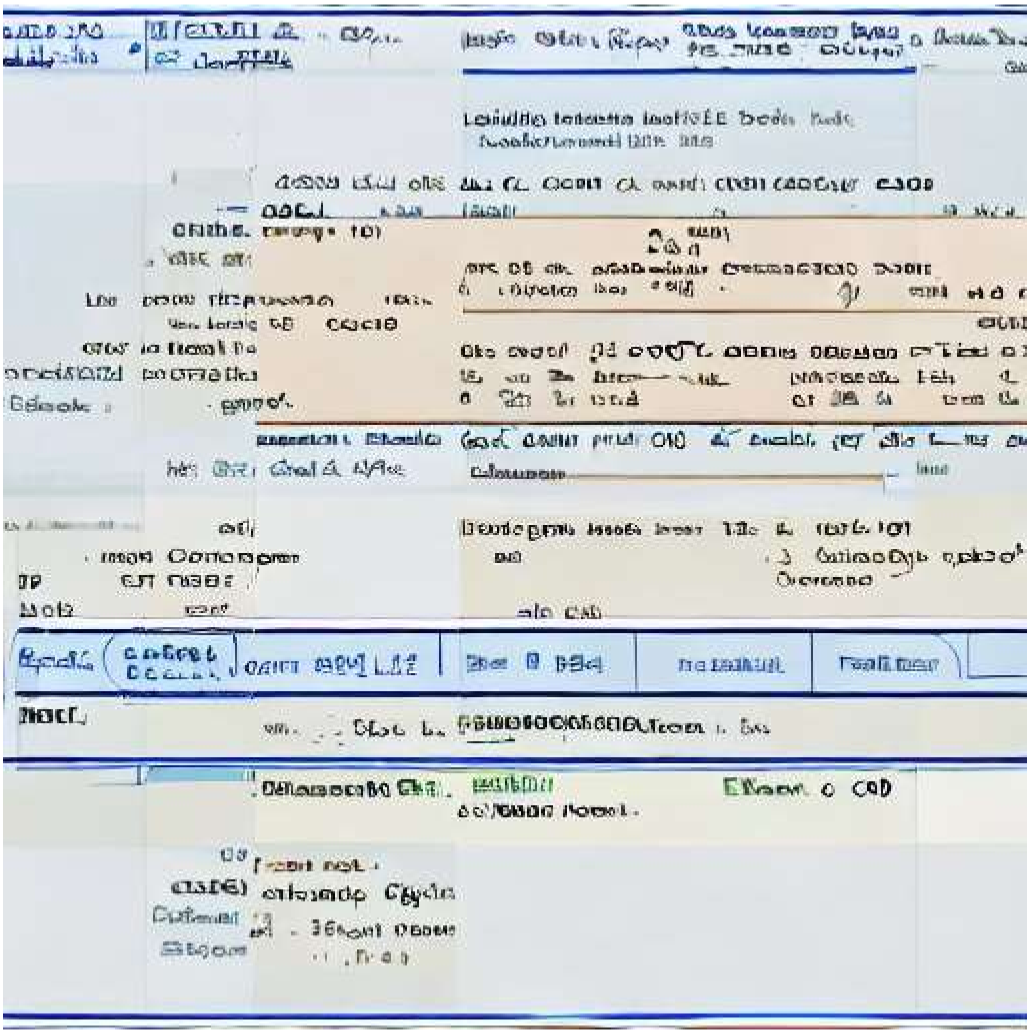
In Stable Diffusion U-Net contains also a "vector" with dimensions (772,772) of parameters which are taken from the conversion of prompt text.

3. Stable Diffusion Use-Cases:

3.1. Text2Img (*Text to Image*):

Input: “COBOL code”

Output:



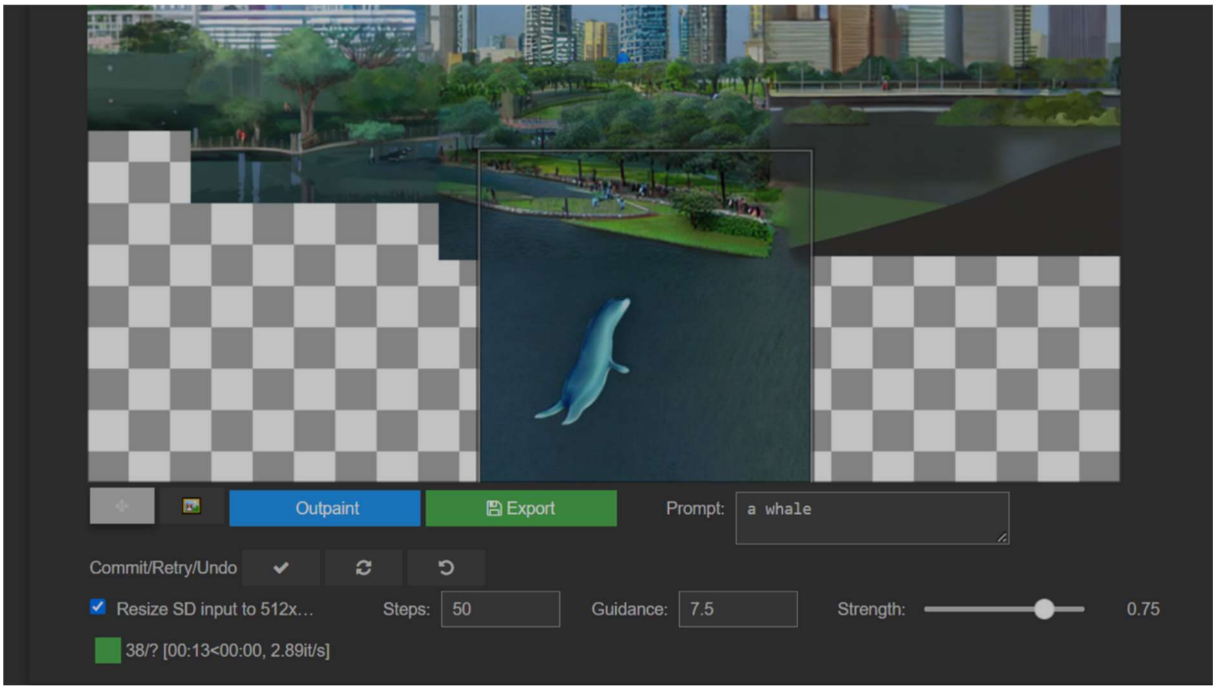
3.2. Img2Img (*Image to Image* or *Text + Image to Image*)

3.2.1. Inpainting:

image	mask_image	prompt		Output
		*Face of a yellow cat, high resolution, sitting on a park bench*	=>	

Inpainting approach tries to create a mask on the original photo which will restrict the Stable Diffusion generation process.

3.2.2. Outpainting:

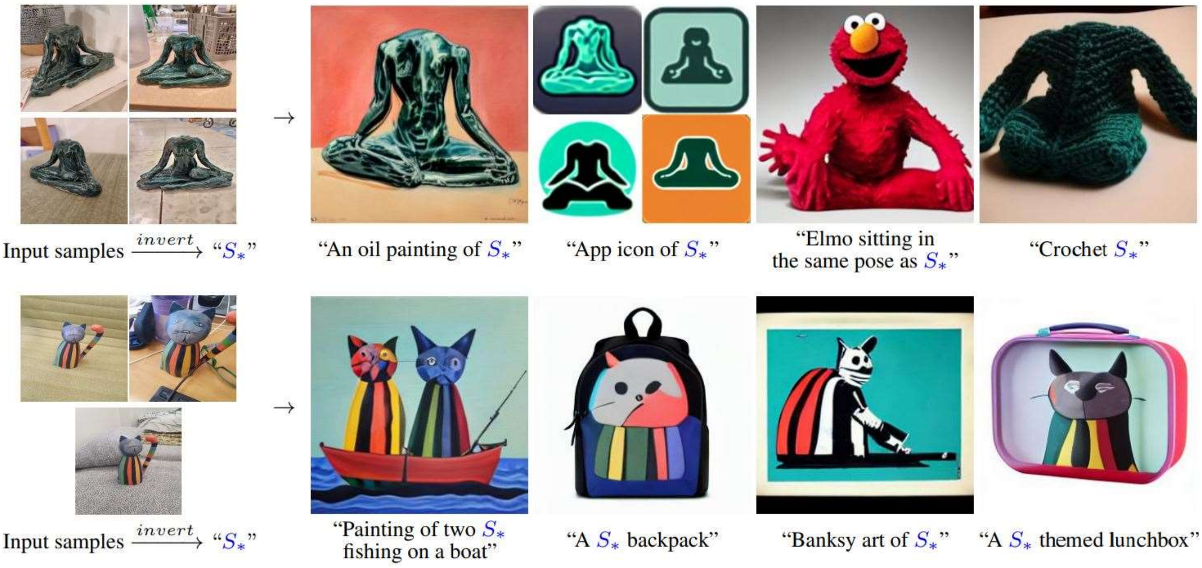


Outpainting approach tries to create more of data around the current image.

3.2.3. Depainting:

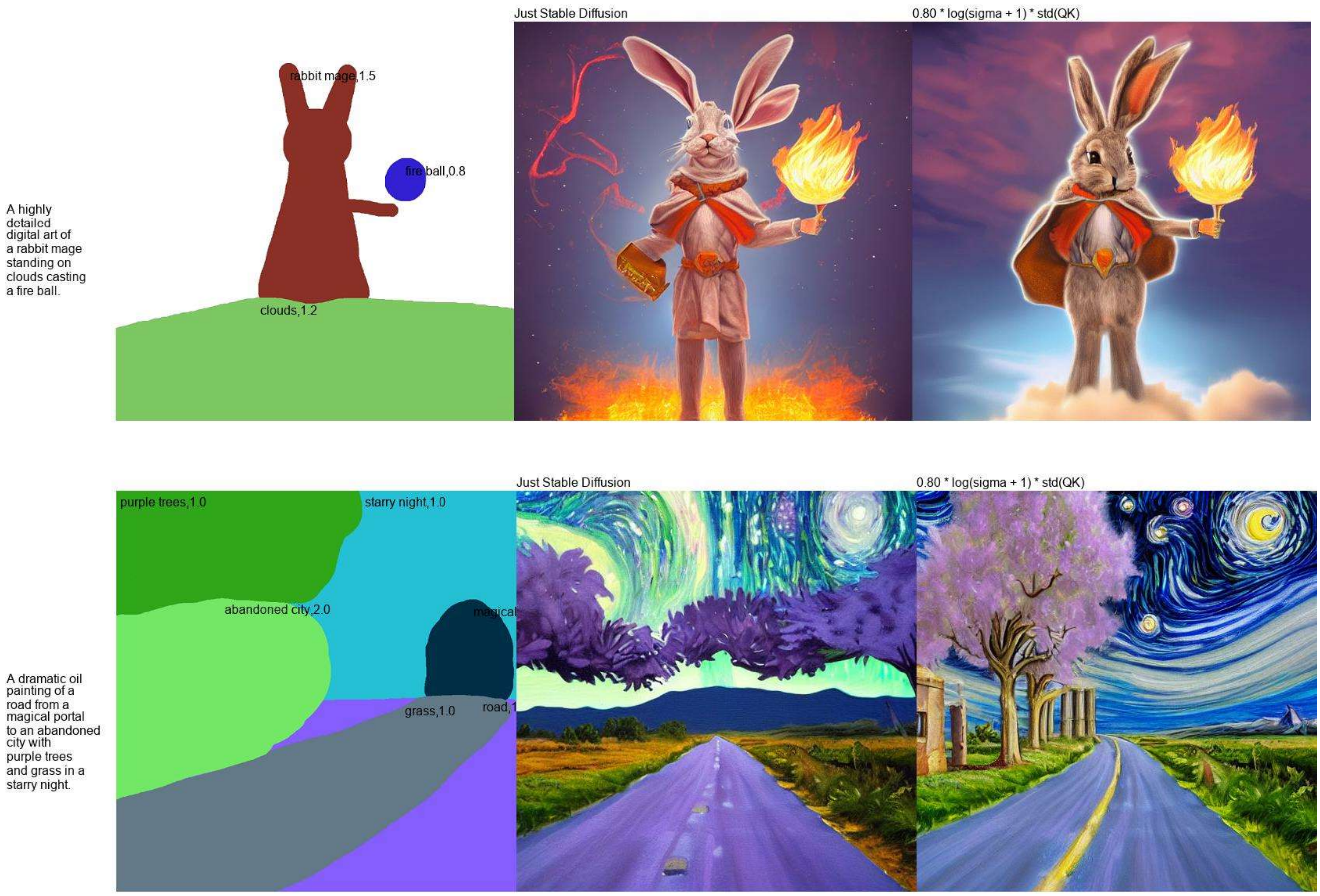
Depainting approach tries to recreate image in some different style than original one. It uses the prompt text as a hint for recreation of the desired image.

3.2.4. Textual Inversion:



Similarly do Depainting Textual Inversion approach tries to recreate an image in some different style than original one, but also tries to understand specified object as some S*. Then it uses the prompt text as a hint for generation of new image in different context.

3.2.5. “Paint With Words”: https://github.com/cloneofsimo/paint-with-words-sd



“Paint With Words” approach tries to create an image from text prompt, but also give a hint to a model where parts of prompt text should be painted.

3.3. Prompt2Prompt (by Google - *Prompt to prompt*)

https://github.com/google/prompt-to-prompt

This approach tries to understand latent space and add proper text manipulation to an image without obligation to create a mask for the original photo.

3.4. Filters:

3.4.1. Video Prompt-Based Filter (*Img2Img Videos*):

https://youtu.be/xtFFKDgyJ7A?t=36

3.5. https://www.youtube.com/watch?v=PddIlnAdv68

3.6. Video Generation (Not from Stable Diffusion):

https://imagen.research.google/video/

3.7. Other information:

3.7.1. Good introduction to machine learning and neural networks:

3.7.1.1. https://calvinfeng.gitbook.io/machine-learning-notebook/

3.7.2. Frequently used project:

3.7.2.1. https://github.com/AUTOMATIC1111/stable-diffusion-webui

3.7.2.2. https://ebsynth.com/

3.7.2.3. https://colab.research.google.com/github/Sxela/DiscoDiffusion-Warp/blob/main/Disco_Diffusion_v5_2_Warp.ipynb#scrollTo=DefMidasFns

3.7.2.4.

3.7.3. Frequently used auxiliary machine learning models:

3.7.3.1. https://github.com/TencentARC/GFPGAN - Making faces more realistic (mostly used for upscaling photos of faces)

3.7.4. Stable-Diffusion Artists / Enthusiasts:

3.7.4.1. https://www.youtube.com/channel/UClSBolYONOzQjOzE4cMHfpw

3.7.4.2. https://www.youtube.com/@NerdyRodent

3.7.4.3. https://www.youtube.com/c/OlivioSarikas