

```
from google.colab import drive
drive.mount('/content/drive')

→ Drive already mounted at /content/drive; to attempt to forcibly remount, call drive.mount("/content/drive", force_remount=True).

!nvidia-smi

→ Sat May 18 15:17:25 2024
+-----+
| NVIDIA-SMI 535.104.05      Driver Version: 535.104.05    CUDA Version: 12.2 |
+-----+
| GPU  Name        Persistence-M | Bus-Id     Disp.A  | Volatile Uncorr. ECC | |
| Fan  Temp  Perf  Pwr:Usage/Cap | Memory-Usage | GPU-Util  Compute M. |
|                   |              |             | MIG M.               |
+-----+
|   0  Tesla T4           Off  | 00000000:00:04.0 Off |          0 | |
| N/A   37C   P8          9W /  70W |      0MiB / 15360MiB |     0%      Default |
|                   |              |             | N/A                  |
+-----+
+-----+
| Processes:                               |
| GPU  GI  CI      PID  Type  Process name        GPU Memory |
| ID  ID             ID              Usage          |
+-----+
| No running processes found               |
+-----+


#CLIP Architecture
!git clone https://github.com/openai/CLIP.git

→ fatal: destination path 'CLIP' already exists and is not an empty directory.

# TAMING-TRANSFORMER ARCHITECTURE
!git clone https://github.com/CompVis/taming-transformers

→ fatal: destination path 'taming-transformers' already exists and is not an empty directory.

!pip install --no-deps ftfy regex tqdm

→ Collecting ftfy
  Downloading ftfy-6.2.0-py3-none-any.whl (54 kB)
    54.4/54.4 kB 1.7 MB/s eta 0:00:00
Requirement already satisfied: regex in /usr/local/lib/python3.10/dist-packages (2023.12.25)
Requirement already satisfied: tqdm in /usr/local/lib/python3.10/dist-packages (4.66.4)
Installing collected packages: ftfy
Successfully installed ftfy-6.2.0


!pip install omegaconf==2.0.0 pytorch-lightning==1.0.8
Requirement already satisfied: tqdm>=4.41.0 in /usr/local/lib/python3.10/dist-packages (from pytorch-lightning==1.0.8) (4.66.4)
→ Requirement already satisfied: fsspec>=0.8.0 in /usr/local/lib/python3.10/dist-packages (from pytorch-lightning==1.0.8) (2023.6.2)
Requirement already satisfied: tensorboard>=2.2.0 in /usr/local/lib/python3.10/dist-packages (from pytorch-lightning==1.0.8) (2.2.0)
Requirement already satisfied: absl-py>=0.4 in /usr/local/lib/python3.10/dist-packages (from tensorboard>=2.2.0->pytorch-lightning) (2.2.0)
Requirement already satisfied: grpcio>=1.48.2 in /usr/local/lib/python3.10/dist-packages (from tensorboard>=2.2.0->pytorch-lightning) (1.48.2)
Requirement already satisfied: google-auth<3,>=1.6.3 in /usr/local/lib/python3.10/dist-packages (from tensorboard>=2.2.0->pytorch-lightning) (1.6.3)
Requirement already satisfied: google-auth-oauthlib<2,>=0.5 in /usr/local/lib/python3.10/dist-packages (from tensorboard>=2.2.0->pytorch-lightning) (0.5)
Requirement already satisfied: markdown>=2.6.8 in /usr/local/lib/python3.10/dist-packages (from tensorboard>=2.2.0->pytorch-lightning) (2.6.8)
Requirement already satisfied: protobuf!=4.24.0,>=3.19.6 in /usr/local/lib/python3.10/dist-packages (from tensorboard>=2.2.0->pytorch-lightning) (3.19.6)
Requirement already satisfied: requests<3,>=2.21.0 in /usr/local/lib/python3.10/dist-packages (from tensorboard>=2.2.0->pytorch-lightning) (2.21.0)
Requirement already satisfied: setuptools>=41.0.0 in /usr/local/lib/python3.10/dist-packages (from tensorboard>=2.2.0->pytorch-lightning) (41.0.0)
Requirement already satisfied: six>1.9 in /usr/local/lib/python3.10/dist-packages (from tensorboard>=2.2.0->pytorch-lightning==1.0.8) (1.9)
Requirement already satisfied: tensorboard-data-server<0.8.0,>=0.7.0 in /usr/local/lib/python3.10/dist-packages (from tensorboard>=2.2.0->pytorch-lightning) (0.7.0)
Requirement already satisfied: werkzeug>=1.0.1 in /usr/local/lib/python3.10/dist-packages (from tensorboard>=2.2.0->pytorch-lightning) (1.0.1)
Requirement already satisfied: filelock in /usr/local/lib/python3.10/dist-packages (from torch>=1.3->pytorch-lightning==1.0.8) (filelock)
Requirement already satisfied: sympy in /usr/local/lib/python3.10/dist-packages (from torch>=1.3->pytorch-lightning==1.0.8) (1.14.1)
Requirement already satisfied: networkx in /usr/local/lib/python3.10/dist-packages (from torch>=1.3->pytorch-lightning==1.0.8) (2.4.1)
Requirement already satisfied: jinja2 in /usr/local/lib/python3.10/dist-packages (from torch>=1.3->pytorch-lightning==1.0.8) (3.1.2)
Collecting nvidia-cuda-nvrtc-cu12==12.1.105 (from torch>=1.3->pytorch-lightning==1.0.8)
  Using cached nvidia_cuda_nvrtc_cu12-12.1.105-py3-none-manylinux1_x86_64.whl (23.7 MB)
Collecting nvidia-cuda-runtime-cu12==12.1.105 (from torch>=1.3->pytorch-lightning==1.0.8)
  Using cached nvidia_cuda_runtime_cu12-12.1.105-py3-none-manylinux1_x86_64.whl (823 kB)
Collecting nvidia-cuda-cupti-cu12==12.1.105 (from torch>=1.3->pytorch-lightning==1.0.8)
```

```
Using cached nvidia_cusolver_cu12_11.7.0.107_py3-none-manylinux1_x86_64.whl (147.4 MB)
Collecting nvidia-cusparse-cu12==12.1.0.106 (from torch>=1.3->pytorch-lightning==1.0.8)
  Using cached nvidia_cusparse_cu12-12.1.0.106-py3-none-manylinux1_x86_64.whl (196.0 MB)
Collecting nvidia-nccl-cu12==2.19.3 (from torch>=1.3->pytorch-lightning==1.0.8)
  Using cached nvidia_nccl_cu12-2.19.3-py3-none-manylinux1_x86_64.whl (166.0 MB)
Collecting nvidia-nvtx-cu12==12.1.105 (from torch>=1.3->pytorch-lightning==1.0.8)
  Using cached nvidia_nvtx_cu12-12.1.105-py3-none-manylinux1_x86_64.whl (99 kB)
Requirement already satisfied: triton==2.2.0 in /usr/local/lib/python3.10/dist-packages (from torch>=1.3->pytorch-lightning==1.0.8)
Collecting nvidia-nvjitlink-cu12 (from nvidia-cusolver-cu12==11.4.5.107->torch>=1.3->pytorch-lightning==1.0.8)
  Using cached nvidia_nvjitlink_cu12-12.4.127-py3-none-manylinux2014_x86_64.whl (21.1 MB)
Requirement already satisfied: cachetools<6.0,>=2.0.0 in /usr/local/lib/python3.10/dist-packages (from google-auth<3,>=1.6.3->te)
Requirement already satisfied: pyasn1-modules>=0.2.1 in /usr/local/lib/python3.10/dist-packages (from google-auth<3,>=1.6.3->ten)
Requirement already satisfied: rsa<5,>=3.1.4 in /usr/local/lib/python3.10/dist-packages (from google-auth<3,>=1.6.3->tensorboard)
Requirement already satisfied: requests-oauthlib>=0.7.0 in /usr/local/lib/python3.10/dist-packages (from google-auth<3,>=1.6.3->te)
Requirement already satisfied: charset-normalizer<4,>=2 in /usr/local/lib/python3.10/dist-packages (from requests<3,>=2.21.0->te)
Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.10/dist-packages (from requests<3,>=2.21.0->tensorboard>=2)
Requirement already satisfied: urllib3<3,>=1.21.1 in /usr/local/lib/python3.10/dist-packages (from requests<3,>=2.21.0->tensorbo)
Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.10/dist-packages (from requests<3,>=2.21.0->tensorbo)
Requirement already satisfied: MarkupSafe>=2.1.1 in /usr/local/lib/python3.10/dist-packages (from werkzeug>=1.0.1->tensorboard>=2)
Requirement already satisfied: mpmath>=0.19 in /usr/local/lib/python3.10/dist-packages (from sympy>torch>=1.3->pytorch-lightning)
Requirement already satisfied: pyasn1<0.7.0,>=0.4.6 in /usr/local/lib/python3.10/dist-packages (from pyasn1-modules>=0.2.1->google)
Requirement already satisfied: oauthlib>=3.0.0 in /usr/local/lib/python3.10/dist-packages (from requests-oauthlib>=0.7.0->google)
Installing collected packages: omegaconf, nvidia-nvtx-cu12, nvidia-nvjitlink-cu12, nvidia-nccl-cu12, nvidia-curand-cu12, nvidia-cuda-nvrtc-cu12-12.1.105 nvidia-cuda-r
Successfully installed nvidia-cublas-cu12-12.1.3.1 nvidia-cuda-cupti-cu12-12.1.105 nvidia-cuda-nvrtc-cu12-12.1.105 nvidia-cuda-r
```

```
!pip uninstall torchtext --yes
```

→ Found existing installation: torchtext 0.17.1
 Uninstalling torchtext-0.17.1:
 Successfully uninstalled torchtext-0.17.1

```
!pip install einops
```

→ Collecting einops
 Downloading einops-0.8.0-py3-none-any.whl (43 kB)
 ━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 43.2/43.2 kB 1.8 MB/s eta 0:00:00
 Installing collected packages: einops
 Successfully installed einops-0.8.0

```
import PIL
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
```

```
import torch , os, imageio, pdb, math
import torchvision
import torchvision.transforms as T
import torchvision.transforms.functional as TF
```

```
import yaml
from omegaconf import OmegaConf
```

```
from CLIP import clip
```

```
import warnings
warnings.filterwarnings("ignore")
```

```
## Helper functions
```

```
def show_from_tensor(tensor):
    img = tensor.clone()
    img = img.mul(255).byte()
    img = img.cpu().numpy().transpose(1,2,0)
    plt.axis('off')
    plt.imshow(img)
    plt.show()

def norm_data(data):
    return (data.clip(-1,1) + 1) / 2 ## range btn 0 & 1 in result
```

```
## Parameters
learning_rate= .5
batch_size= 1
wd = .1
noise_factor = .22
```

```
total_iter = 400
im_shape = [450,450, 3] #height, width, channel
size1, size2, channels = im_shape
```

```

##CLIP Model
clipmodel, _ = clip.load('ViT-B/32', jit=False)
clipmodel.eval()
print(clip.available_models())

print("Clip model visual resolution:" ,clipmodel.visual.input_resolution)

device = torch.device("cuda:0")
torch.cuda.empty_cache()

→ 100%|██████████| 338M/338M [00:04<00:00, 74.3MiB/s]
['RN50', 'RN101', 'RN50x4', 'RN50x16', 'RN50x64', 'ViT-B/32', 'ViT-B/16', 'ViT-L/14', 'ViT-L/14@336px']
Clip model visual resolution: 224

%cd taming-transformers

!mkdir -p models/vqgan_imagenet_f16_16384/checkpoints
!mkdir -p models/vqgan_imagenet_f16_16384/configs

if len(os.listdir('models/vqgan_imagenet_f16_16384/checkpoints/')) == 0:
    !wget 'https://heibox.uni-heidelberg.de/f/867b05fc8c4841768640/?dl=1' -O 'models/vqgan_imagenet_f16_16384/checkpoints/last.ckpt'
    !wget 'https://heibox.uni-heidelberg.de/f/274fb24ed38341bfa753/?dl=1' -O 'models/vqgan_imagenet_f16_16384/configs/model.yaml'

→ /content/taming-transformers
--2024-05-18 15:28:38-- https://heibox.uni-heidelberg.de/f/867b05fc8c4841768640/?dl=1&
Resolving heibox.uni-heidelberg.de (heibox.uni-heidelberg.de)... 129.206.7.113
Connecting to heibox.uni-heidelberg.de (heibox.uni-heidelberg.de)|129.206.7.113|:443... connected.
HTTP request sent, awaiting response... 302 Found
Location: https://heibox.uni-heidelberg.de/seafhttp/files/6c332f72-8a7d-41d2-8513-44654b525748/last.ckpt [following]
--2024-05-18 15:28:40-- https://heibox.uni-heidelberg.de/seafhttp/files/6c332f72-8a7d-41d2-8513-44654b525748/last.ckpt
Reusing existing connection to heibox.uni-heidelberg.de:443.
HTTP request sent, awaiting response... 200 OK
Length: 980092370 (935M) [application/octet-stream]
Saving to: 'models/vqgan_imagenet_f16_16384/checkpoints/last.ckpt'

models/vqgan_imagen 100%[=====] 934.69M 12.8MB/s in 74s

2024-05-18 15:29:54 (12.6 MB/s) - 'models/vqgan_imagenet_f16_16384/checkpoints/last.ckpt' saved [980092370/980092370]

--2024-05-18 15:29:54-- https://heibox.uni-heidelberg.de/f/274fb24ed38341bfa753/?dl=1&
Resolving heibox.uni-heidelberg.de (heibox.uni-heidelberg.de)... 129.206.7.113
Connecting to heibox.uni-heidelberg.de (heibox.uni-heidelberg.de)|129.206.7.113|:443... connected.
HTTP request sent, awaiting response... 302 Found
Location: https://heibox.uni-heidelberg.de/seafhttp/files/b0b29822-0461-4345-bec5-3eec17b9e8be/model.yaml [following]
--2024-05-18 15:29:55-- https://heibox.uni-heidelberg.de/seafhttp/files/b0b29822-0461-4345-bec5-3eec17b9e8be/model.yaml
Reusing existing connection to heibox.uni-heidelberg.de:443.
HTTP request sent, awaiting response... 200 OK
Length: 692 [application/octet-stream]
Saving to: 'models/vqgan_imagenet_f16_16384/configs/model.yaml'

models/vqgan_imagen 100%[=====] 692 --.KB/s in 0s

2024-05-18 15:29:55 (407 MB/s) - 'models/vqgan_imagenet_f16_16384/configs/model.yaml' saved [692/692]

from taming.models.vqgan import VQModel

def load_config(config_path, display=False):
    config_data = OmegaConf.load(config_path)
    if display:
        print(yaml.dump(OmegaConf.to_container(config_data)))
    return config_data

def load_vqgan(config, chk_path=None):
    model = VQModel(**config.model.params)
    if chk_path is not None:
        state_dict = torch.load(chk_path, map_location="cpu")["state_dict"]
        missing, unexpected = model.load_state_dict(state_dict, strict=False)
    return model.eval()

def generator(x):
    x = taming_model.post_quant_conv(x)
    x = taming_model.decoder(x)
    return x

taming_config = load_config("./models/vqgan_imagenet_f16_16384/configs/model.yaml", display=True)
taming_model = load_vqgan(taming_config, chk_path="./models/vqgan_imagenet_f16_16384/checkpoints/last.ckpt").to(device)

# here we got error then clicked on /content/taming-transformers/taming/data/utils.py in <module> in error window and remove 10-13 lin

```

```

→ model:
  base_learning_rate: 4.5e-06
  params:

```

```

ddconfig:
  attn_resolutions:
    - 16
  ch: 128
  ch_mult:
    - 1
    - 1
    - 2
    - 2
    - 4
  double_z: false
  dropout: 0.0
  in_channels: 3
  num_res_blocks: 2
  out_ch: 3
  resolution: 256
  z_channels: 256
  embed_dim: 256
lossconfig:
  params:
    codebook_weight: 1.0
    disc_conditional: false
    disc_in_channels: 3
    disc_num_layers: 2
    disc_start: 0
    disc_weight: 0.75
  target: taming.modules.losses.vqperceptual.VQLPIPSWithDiscriminator
  monitor: val/rec_loss
  n_embed: 16384
target: taming.models.vqgan.VQModel

```

Working with z of shape (1, 256, 16, 16) = 65536 dimensions.
 Downloading: "<https://download.pytorch.org/models/vgg16-397923af.pth>" to /root/.cache/torch/hub/checkpoints/vgg16-397923af.pth
 100% [██████████] 528M/528M [00:05<00:00, 92.5MB/s]
 Downloading vgg_lipips model from <https://heibox.uni-heidelberg.de/f/607503859c864bc1b30b/?dl=1> to taming/modules/autoencoder/lpip...
 8.19kB [00:00, 328kB/s]
 loaded pretrained LPIPS loss from taming/modules/autoencoder/lpip/vgg.pth
 VQLPIPSWithDiscriminator running with hinge loss.

Declare the values that we are going to optimize

```

class Parameters(torch.nn.Module):
  def __init__(self):
    super(Parameters, self).__init__()
    self.data = .5*torch.randn(batch_size, 256, size1//16, size2//16).cuda() # 1x256x14x15 (225/16, 400/16)
    self.data = torch.nn.Parameter(torch.sin(self.data))

  def forward(self):
    return self.data

def init_params():
  params=Parameters().cuda()
  optimizer = torch.optim.AdamW([{'params':[params.data]}, {'lr': learning_rate}], weight_decay=wd)
  return params, optimizer

```

```
### Encoding prompts and a few more things
normalize = torchvision.transforms.Normalize((0.48145466, 0.4578275, 0.40821073), (0.26862954, 0.26130258, 0.27577711))

def encodeText(text):
    t=clip.tokenize(text).cuda()
    t=clipmodel.encode_text(t).detach().clone()
    return t

def createEncodings(include, exclude, extras):
    include_enc=[]
    for text in include:
        include_enc.append(encodeText(text))
    exclude_enc=encodeText(exclude) if exclude != '' else 0
    extras_enc=encodeText(extras) if extras !='' else 0

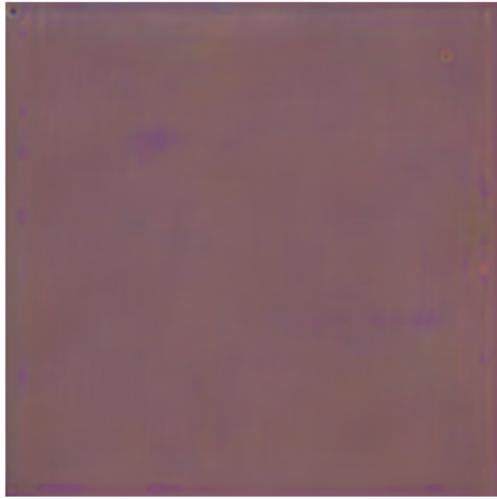
    return include_enc, exclude_enc, extras_enc

augTransform = torch.nn.Sequential(
    torchvision.transforms.RandomHorizontalFlip(),
    torchvision.transforms.RandomAffine(30, (.2, .2), fill=0)
).cuda()

Params, optimizer = init_params()

with torch.no_grad():
    print(Params().shape)
    img= norm_data(generator(Params()).cpu()) # 1 x 3 x 224 x 400 [225 x 400]
    print("img dimensions: ",img.shape)
    show_from_tensor(img[0])

→ torch.Size([1, 256, 28, 28])
img dimensions: torch.Size([1, 3, 448, 448])
```



```
#check midjourney.co/generator
```

```

### create crops

def create_crops(img, num_crops=32):
    p=size1//2
    img = torch.nn.functional.pad(img, (p,p,p,p), mode='constant', value=0) # 1 x 3 x 448 x 624 (adding 112*2 on all sides to 224x400)

    img = augTransform(img) #RandomHorizontalFlip and RandomAffine

    crop_set = []
    for ch in range(num_crops):
        gap1= int(torch.normal(1.2, .3, ()).clip(.43, 1.9) * size1)
        offsetx = torch.randint(0, int(size1*2-gap1),())
        offsety = torch.randint(0, int(size1*2-gap1),())

        crop=img[:, :, offsetx:offsetx+gap1, offsety:offsety+gap1]

        crop = torch.nn.functional.interpolate(crop,(224,224), mode='bilinear', align_corners=True)
        crop_set.append(crop)

    img_crops=torch.cat(crop_set,0) ## 30 x 3 x 224 x 224

    randnormal = torch.randn_like(img_crops, requires_grad=False)
    num_rands=0
    randstotal=torch.rand((img_crops.shape[0],1,1,1)).cuda() #32

    for ns in range(num_rands):
        randstotal*=torch.rand((img_crops.shape[0],1,1,1)).cuda()

    img_crops = img_crops + noise_factor*randstotal*randnormal

    return img_crops


### Show current state of generation

def showme(Params, show_crop):
    with torch.no_grad():
        generated = generator(Params())

    if (show_crop):
        print("Augmented cropped example")
        aug_gen = generated.float() # 1 x 3 x 224 x 400
        aug_gen = create_crops(aug_gen, num_crops=1)
        aug_gen_norm = norm_data(aug_gen[0])
        show_from_tensor(aug_gen_norm)

    print("Generation")
    latest_gen=norm_data(generated.cpu()) # 1 x 3 x 224 x 400
    show_from_tensor(latest_gen[0])

    return (latest_gen[0])

```

```

# Optimization process

def optimize_result(Params, prompt):
    alpha=1 ## the importance of the include encodings
    beta=.5 ## the importance of the exclude encodings

    ## image encoding
    out = generator(Params())
    out = norm_data(out)
    out = create_crops(out)
    out = normalize(out) # 30 x 3 x 224 x 224
    image_enc=clipmodel.encode_image(out) ## 30 x 512

    ## text encoding w1 and w2
    final_enc = w1*prompt + w1*extras_enc # prompt and extras_enc : 1 x 512
    final_text_include_enc = final_enc / final_enc.norm(dim=-1, keepdim=True) # 1 x 512
    final_text_exclude_enc = exclude_enc

    ## calculate the loss
    main_loss = torch.cosine_similarity(final_text_include_enc, image_enc, -1) # 30
    penalize_loss = torch.cosine_similarity(final_text_exclude_enc, image_enc, -1) # 30

    final_loss = -alpha*main_loss + beta*penalize_loss

    return final_loss

def optimize(Params, optimizer, prompt):
    loss = optimize_result(Params, prompt).mean()
    optimizer.zero_grad()
    loss.backward()
    optimizer.step()
    return loss

### training loop

def training_loop(Params, optimizer, show_crop=False):
    res_img=[]
    res_z=[]

    for prompt in include_enc:
        iteration=0
        Params, optimizer = init_params() # 1 x 256 x 14 x 25 (225/16, 400/16)

        for it in range(total_iter):
            loss = optimize(Params, optimizer, prompt)

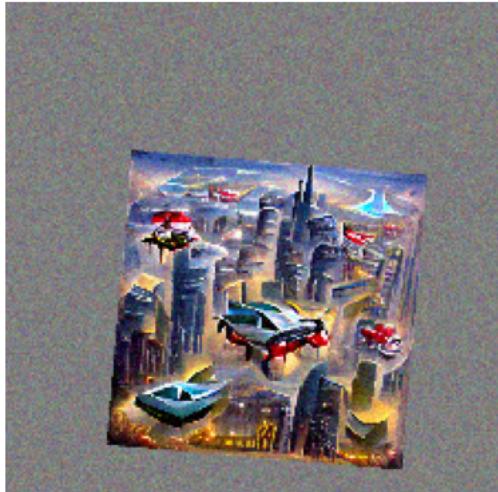
            if iteration>=80 and iteration%show_step == 0:
                new_img = showme(Params, show_crop)
                res_img.append(new_img)
                res_z.append(Params()) # 1 x 256 x 14 x 25
                print("loss:", loss.item(), "\niteration:",iteration)

            iteration+=1
            torch.cuda.empty_cache()
    return res_img, res_z

torch.cuda.empty_cache()
include=['cityscape with flying cars']
exclude='watermark'
extras = ""
w1=1
w2=1
noise_factor=.22
total_iter=110
show_step=10 # set this to see the result every 10 interations beyond iteration 80
include_enc, exclude_enc, extras_enc = createEncodings(include, exclude, extras)
res_img, res_z=training_loop(Params, optimizer, show_crop=True)

```

Augmented cropped example



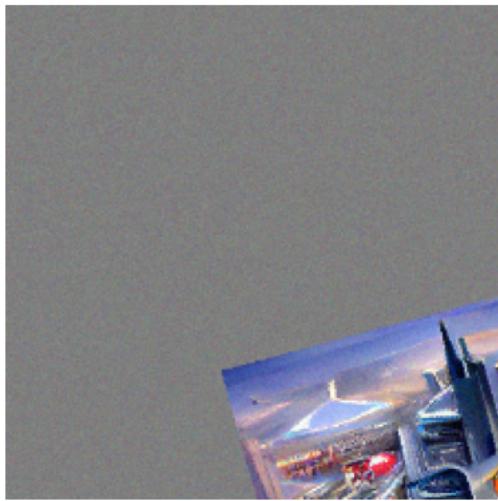
Generation



loss: -0.253173828125

iteration: 80

Augmented cropped example



Generation





loss: -0.280029296875
iteration: 90
Augmented cropped example



Generation



loss: -0.272216796875
iteration: 100

```
torch.cuda.empty_cache()  
include=['A mystical phoenix rising from its ashes in a burst of vibrant flames']  
exclude='watermark'  
extras = ""  
w1=1  
w2=1  
noise_factor=.22  
total_iter=110  
show_step=10 # set this to see the result every 10 interations beyond iteration 80  
include_enc, exclude_enc, extras_enc = createEncodings(include, exclude, extras)  
res_img, res_z=training_loop(Params, optimizer, show_crop=True)
```

Augmented cropped example



Generation



loss: -0.2392578125

iteration: 80

Augmented cropped example



Generation





loss: -0.2337646484375

iteration: 90

Augmented cropped example



Generation

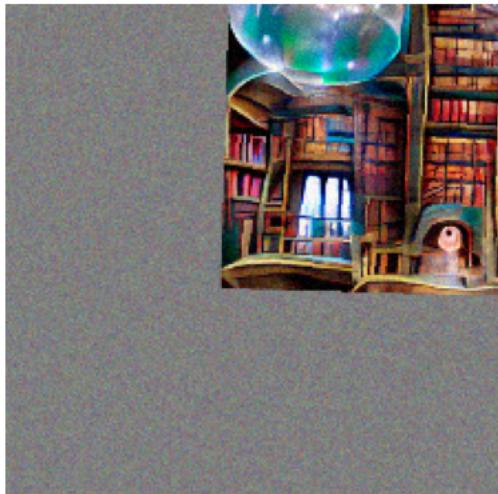


loss: -0.2371826171875

iteration: 100

```
torch.cuda.empty_cache()
include=['An enchanted library with towering bookshelves and a magical, floating orb of light']
exclude='watermark'
extras = ""
w1=1
w2=1
noise_factor=.22
total_iter=110
show_step=10 # set this to see the result every 10 iterations beyond iteration 80
include_enc, exclude_enc, extras_enc = createEncodings(include, exclude, extras)
res_img, res_z=training_loop(Params, optimizer, show_crop=True)
```

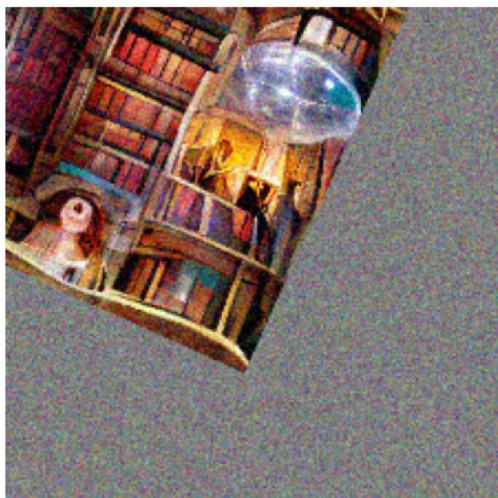
➡ Augmented cropped example



Generation



loss: -0.234375
iteration: 80
Augmented cropped example

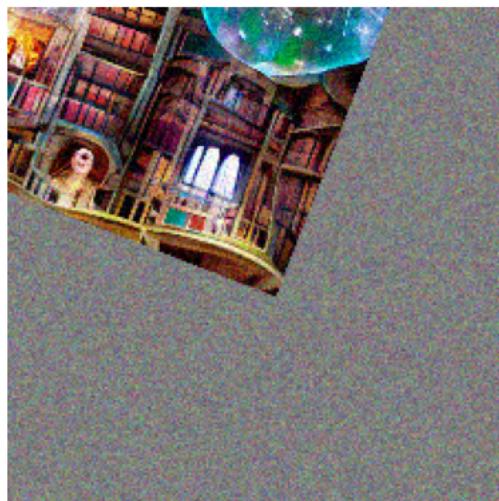


Generation





loss: -0.2724609375
iteration: 90
Augmented cropped example



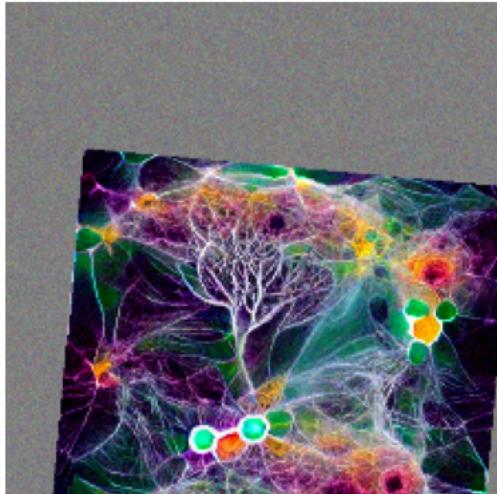
Generation



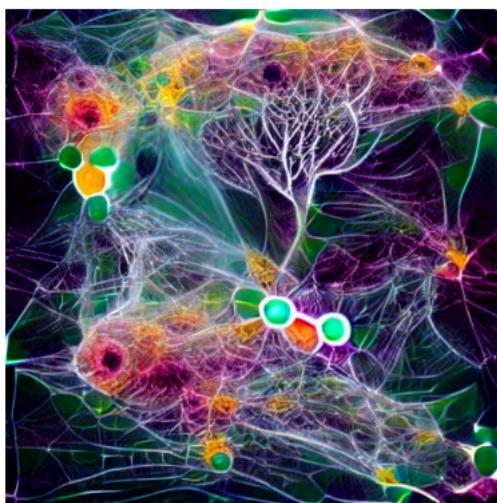
loss: -0.265869140625
iteration: 100

```
torch.cuda.empty_cache()
include=['A vibrant visualization of a neural network with interconnected nodes and data flow',
        'A futuristic laboratory with advanced AI robots conducting scientific experiments',
        'cityscape with flying cars',
        'space and galaxy',
        'A whimsical fairytale castle perched atop a floating island in the clouds',
        'An enchanted library with towering bookshelves and a magical, floating orb of light',
        'A mystical phoenix rising from its ashes in a burst of vibrant flames', ]
exclude='watermark'
extras = ""
w1=1
w2=1
noise_factor=.22
total_iter=110
show_step=10 # set this to see the result every 10 interations beyond iteration 80
include_enc, exclude_enc, extras_enc = createEncodings(include, exclude, extras)
res_img, res_z=training_loop(Params, optimizer, show_crop=True)
```

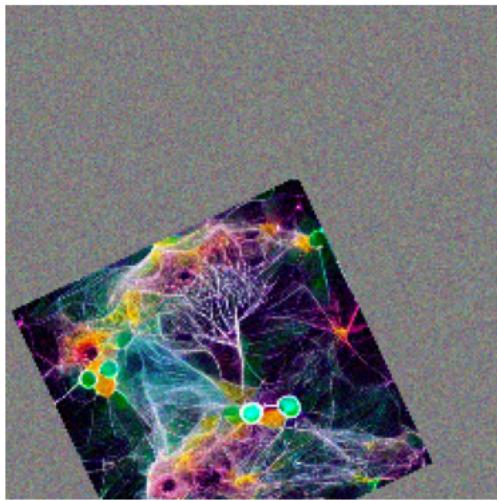
⤓ Augmented cropped example



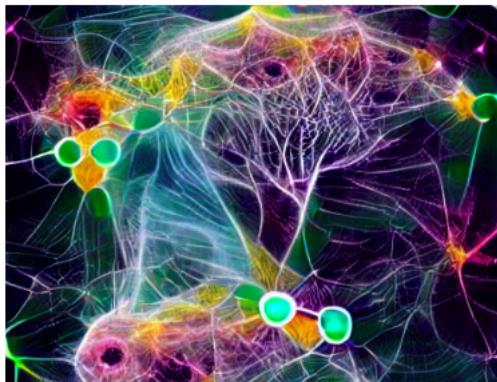
Generation



loss: -0.271484375
iteration: 80
Augmented cropped example

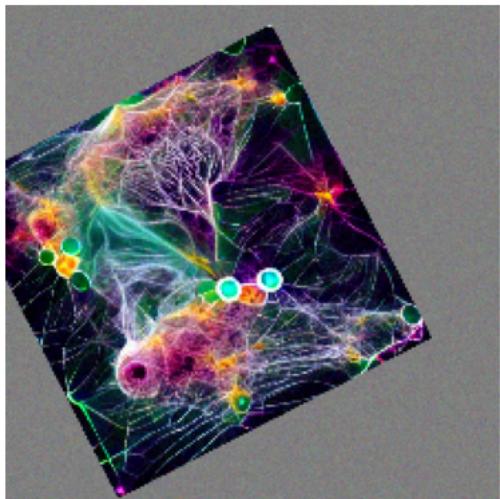


Generation

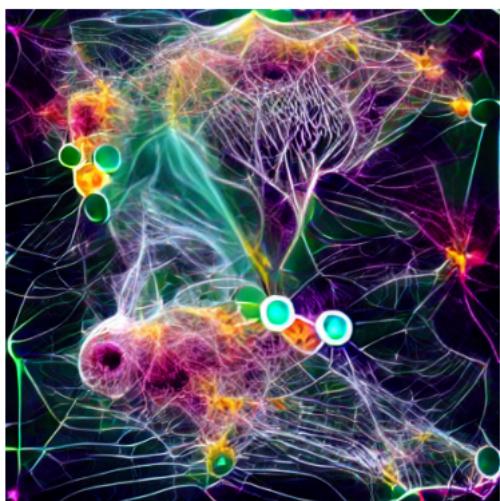




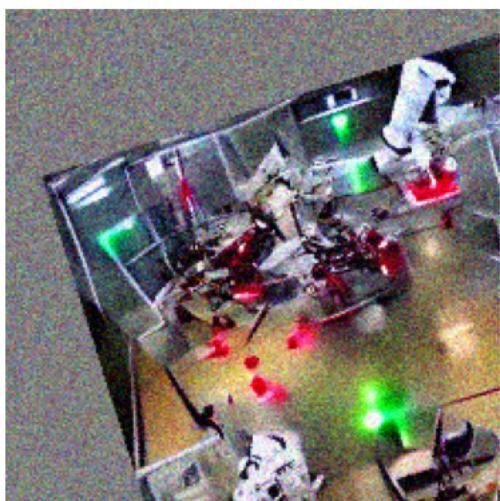
loss: -0.265869140625
iteration: 90
Augmented cropped example



Generation

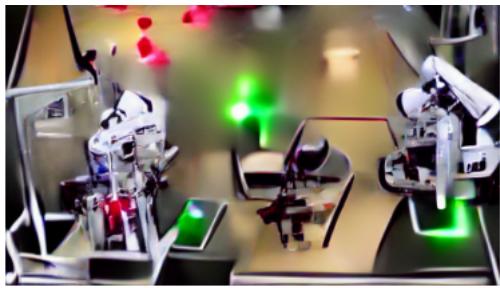


loss: -0.260009765625
iteration: 100
Augmented cropped example



Generation

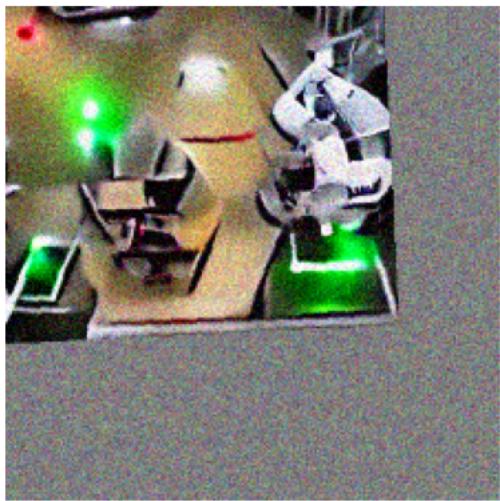




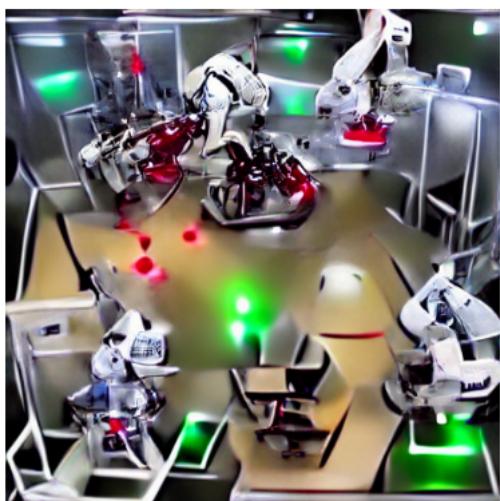
loss: -0.266845703125

iteration: 80

Augmented cropped example



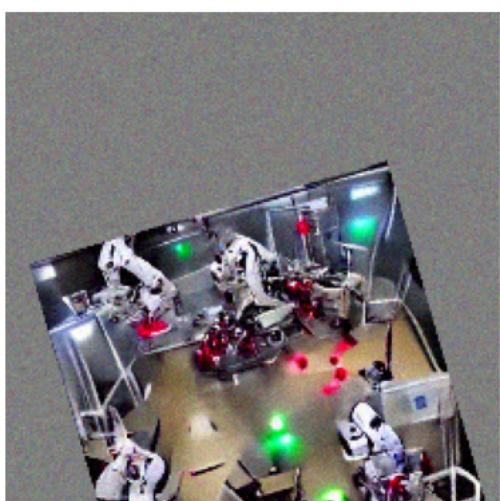
Generation



loss: -0.25439453125

iteration: 90

Augmented cropped example



Generation





loss: -0.2418212890625

iteration: 100

Augmented cropped example



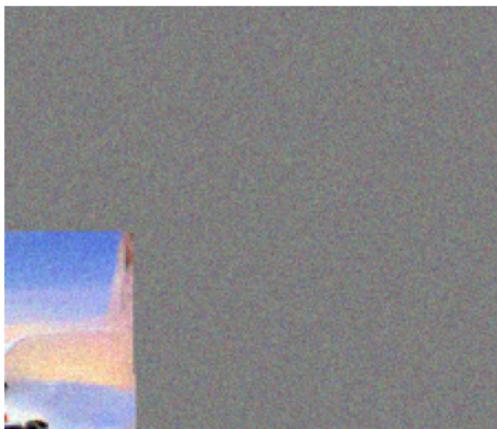
Generation



loss: -0.264892578125

iteration: 80

Augmented cropped example





Generation



loss: -0.2783203125

iteration: 90

Augmented cropped example



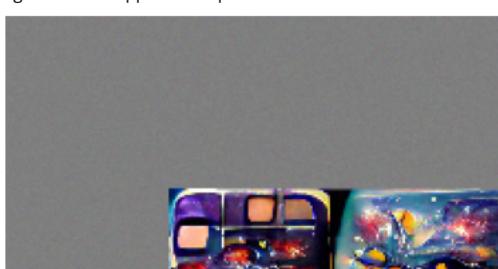
Generation



loss: -0.270751953125

iteration: 100

Augmented cropped example





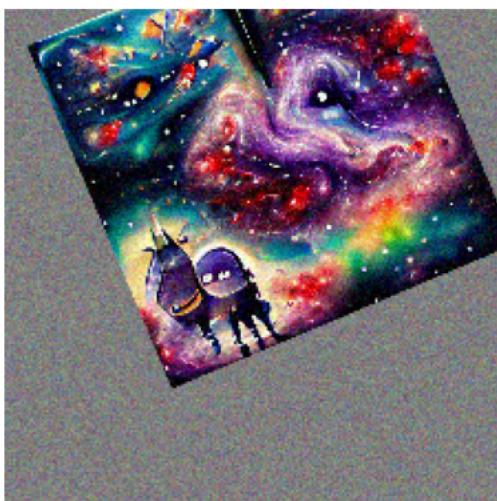
Generation



loss: -0.25244140625

iteration: 80

Augmented cropped example



Generation



loss: -0.24853515625

iteration: 90

Augmented cropped example





Generation



loss: -0.246826171875

iteration: 100

Augmented cropped example



Generation

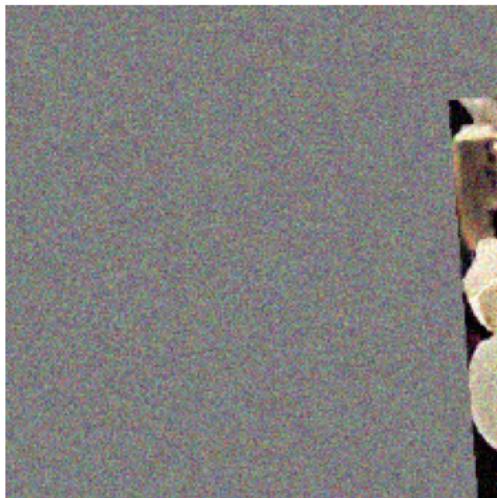


loss: -0.203701171075

5/18/24, 10:10 PM

17 MAY 24- GEN AI - Colab

loss: -
iteration: 80
Augmented cropped example



Generation



loss: -0.302734375
iteration: 90
Augmented cropped example

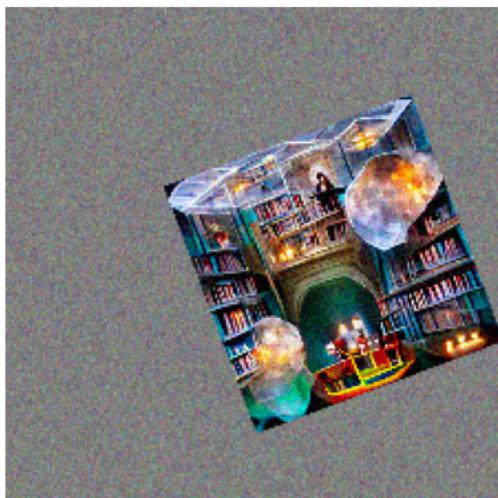


Generation





loss: -0.314697265625
iteration: 100
Augmented cropped example



Generation



loss: -0.26171875
iteration: 80
Augmented cropped example



Generation





loss: -0.253662109375

iteration: 90

Augmented cropped example



Generation



loss: -0.256591796875

iteration: 100

Augmented cropped example



Generation

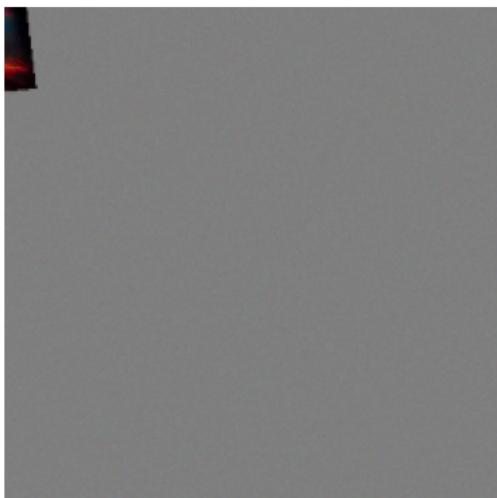




loss: -0.2626953125

iteration: 80

Augmented cropped example



Generation



loss: -0.2568359375

iteration: 90

Augmented cropped example





Generation

