Mohamed Aymen BOUYAHIA
Asma Khalil

TP Bonus
IA323

27/01/2024

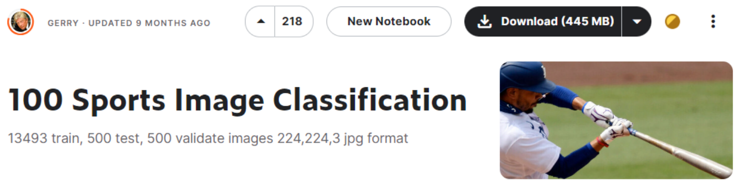
1 Introduction:

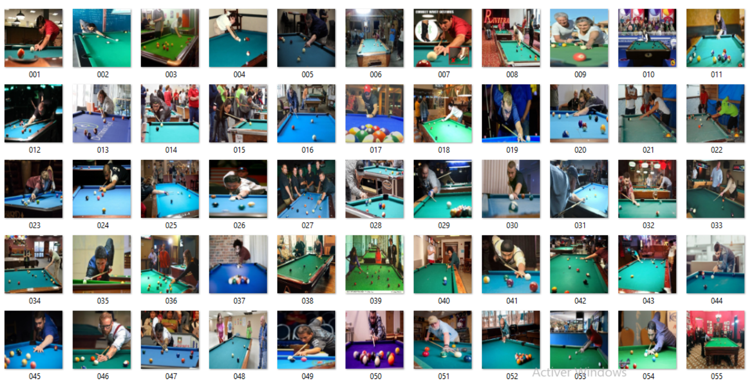
In this report, we will be talking about stable diffusion and how we fine tuned it using DreamBooth to generate images related to sports. The goal is to DreamBooth with LoRA a heavy pipline like Stable Diffusion XL to see the level of results we can get.

2 Dataset Used:

We used a dataset from Kaggle that contains sports images. We focused on images related to billiard. (Here is the link for the dataset: https://www.kaggle.com/datasets/gpiosenka/sports-classification)

The goal is to fine tune stable diffusion model on this dataset to generate, famous people playing billiard.





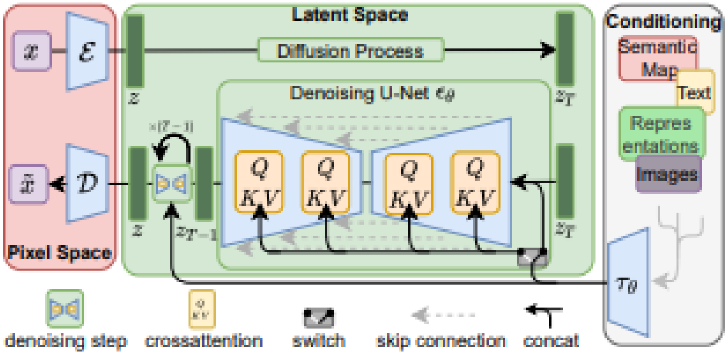
3 Stable Diffusion:

Stable diffusion is a generative model that was trained using denoising making it capable to extract images on which it was trained with only a noise as input.

The diffusion modeling process involves gradually refining an initial noisy image until it matches the desired target. At each stage of this process, the model assesses the compatibility of the generated image with the input text prompt and adjusts the noise accordingly.

This iterative approach allows Stable Diffusion to effectively capture the semantics and aesthetic nuances of the text description while maintaining consistency and realism in the generated image. The model's ability to handle high-dimensional representations of both text and images enables it to produce intricate details and complex patterns that align with the user's intended vision.

By leveraging diffusion modeling and latent spaces, Stable Diffusion has established itself as a powerful tool for text-to-image generation. Its ability to produce photorealistic and semantically consistent images has opened up new avenues for creative expression and visual storytelling

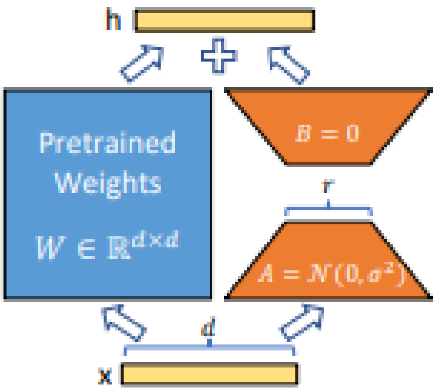


4 DreamBooth and LoRA:

4.1 LoRA (Low-Rank Adaptation of Large Language Models):

LoRA, which stands for Low-Rank Adaptation, is a fine-tuning method for large models that aims to reduce computational costs and improve performance. It achieves these goals by only updating a small subset of the model's parameters, rather than updating all of them as is typically done in full-parameter fine-tuning.

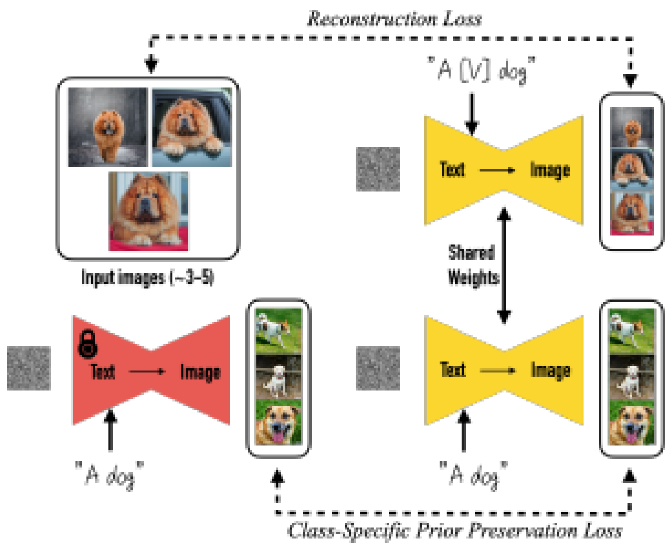
The key idea behind LoRA is to identify the most important parameters that contribute to the task at hand and focus on updating those parameters. This is done by using a technique called low-rank approximation, which essentially reduces the dimensionality of the parameter space. (Generally we try to freeze the parameters we don't need to update and we update the others using a certain matrix decomposition to decrease the dimensionality and decrease the computational cost. We use generally the SVD decomposition.)



4.2 DreamBooth:

Dreambooth is a technique for fine-tuning diffusion models like Stable Diffusion to generate images that are more in line with a specific subject or style. This process involves providing the model with a few example images of the subject or style and a designated keyword or phrase that acts as a unique identifier.

This fine-tuning process enhances the model's ability to identify and incorporate the desired features into the generated images, resulting in more accurate and aesthetically pleasing results.



5 Code used:

We used a certain colab found on the internet that fine tuned diffusion models using LoRA to help us implement the work.
We started, at first, by uploading the images on the colab and visualizing them.

```
import glob

# change path to display images from your local dir
img_paths = "/content/dog/dog/dog/billiard/*.jpg"
imgs = [Image.open(path) for path in glob.glob(img_paths)]

num_imgs_to_preview = 5
image_grid(imgs[:num_imgs_to_preview], 1, num_imgs_to_preview)
```

Then, we downloaded the diffusion library from Hugging face, from which we will be taking the Stable Diffusion XL model.

```
[3] !pip install git+https://github.com/huggingface/diffusers.git -q

      Installing build dependencies ... done
      Getting requirements to build wheel ... done
      Preparing metadata (pyproject.toml) ... done
      Building wheel for diffusers (pyproject.toml) ... done
```

Then, in order to be able to use LoRA and not to program it from scratch, we downloaded the Peft library to be able to do the fine tuning.

```
[2] # Install dependencies.
    !pip install bitsandbytes transformers accelerate peft -q

      ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 105.0/105.0 MB 9.1 MB/s eta 0:00:00
      ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 270.9/270.9 kB 29.3 MB/s eta 0:00:00
      ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 168.3/168.3 kB 22.5 MB/s eta 0:00:00
```

Then, we defined the parameters we wanted to use to dreambooth the Stable Diffusion XL model.

```bash
#!/usr/bin/env bash
!accelerate launch train_dreambooth_lora_sdxl.py \
  --pretrained_model_name_or_path="stabilityai/stable-diffusion-xl-base-1.0" \
  --pretrained_vae_model_name_or_path="madebyollin/sdxl-vae-fp16-fix" \
  --dataset_name="billiard" \
  --output_dir="messi_billard_LoRA" \
  --caption_column="prompt"\
  --mixed_precision="fp16" \
  --instance_prompt="a photo of messi playing billiard" \
  --resolution=1024 \
  --train_batch_size=1 \
  --gradient_accumulation_steps=3 \
  --gradient_checkpointing \
  --learning_rate=1e-4 \
  --snr_gamma=5.0 \
  --lr_scheduler="constant" \
  --lr_warmup_steps=0 \
  --mixed_precision="fp16" \
  --use_8bit_adam \
  --max_train_steps=500 \
  --checkpointing_steps=717 \
  --seed="0"
```

Finally, we trained the model using the parameters we defined and using LoRA which was taken from Peft library.

```python
import torch
from diffusers import DiffusionPipeline, AutoencoderKL

vae = AutoencoderKL.from_pretrained("madebyollin/sdxl-vae-fp16-fix", torch_dtype=torch.float16)
pipe = DiffusionPipeline.from_pretrained(
    "stabilityai/stable-diffusion-xl-base-1.0",
    vae=vae,
    torch_dtype=torch.float16,
    variant="fp16",
    use_safetensors=True
)
_ = pipe.to("cuda")
```

Here are some images we got when generating from the fine tuned model:



6 Conclusion:

Fine-tuning Stable Diffusion models unlocks a wealth of possibilities for customizing and enhancing their generative capabilities. By adapting the model to specific domains or styles, users can achieve remarkable results in image generation, text-to-image synthesis, and creative applications. Fine-tuning allows for finer control over the generated content, enabling the creation of more realistic, stylized, and even personalized imagery. This flexibility makes Stable Diffusion models even more powerful and versatile tools for artists, designers, and anyone interested in exploring the potential of AI-driven image generation.