Conditional LDM with Medical Images

Tony Yang

October 2024

1 Summary

The focus of the work was to follow the method outlined in the referenced paper, which combines edge detection and semantic mapping as conditional inputs for training the LDM.

Overall the data include 360 subjects from the M&M's Challenge dataset, with a split of 350 subjects for training and 10 for validation. It is worth noting that the LDM was trained with the combination of both long-axis and short-axis cardiac images. The model architecture consisted an autoencoder trained over 120 epochs (approximately 29k steps) and a UNet trained for 100 epochs (approximately 24k steps). The naive hyperparameter configuration did not deliver out-performing results, leaving potentials for exploration.

Performance evaluation on the 10 left-out test subjects delivers the following performances:

- FID: 140.54
- SSIM: 0.40
- NMSE: 1.22

2 Data

2.1 Dataset Overview

Data from the M&M's Challenge dataset, includes 360 subjects. Each subject's data includes distinct types of images: original images and semantic maps for end-diastolic, end-systolic, and CINE sequences, in both long-axis and short-axis views. The CINE sequences were excluded, due to the absence of corresponding semantic maps.

2.2 Preprocessing Pipeline

The preprocessing workflow consists of several steps:

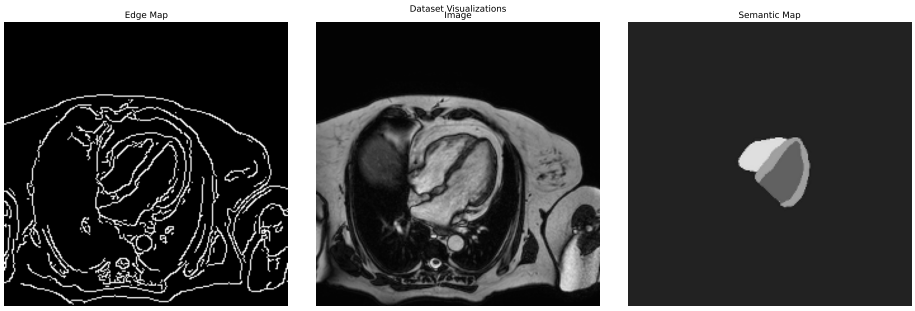

Figure 1: Visual representation data (from left to right): detected edges, original image and semantic map.

1. **Edge Detection**: Binary boundary maps were generated using the Canny Edge Algorithm, with additional filtering and tricks to reduce noise.
2. **Semantic Map Processing**: The semantic maps are quantized into four distinct values, followed by transformation to one-hot encoded representation, resulting in four channels.
3. **Normalization**:
   - All data (images, edges, and semantic maps) were resized to 256×256 resolution, aligning with ImageNet specifications
   - Original images were normalized to the range of $[-1, 1]$

Figure 1 shows an example of the processed data.

2.3 Data Storage

The processed data for each subject were stored in separate HDF5 format, for fast access during the training.

3 Model Overview

The model architecture comprises two components: VQVAE as the autoencoder, and UNet as the backbone for the LDM. The architectural and training parameters are detailed below.

3.1 Model Architecture

- **Latent Space Configuration**:
  - Latent representation shape: $32 \times 32 \times 3$
  - Codebook size: 16,834

- **Network Structure**:
  - Down/Up/Mid block depth: 2 (uniform across all blocks)
  - Down-block channels: [256, 384, 512, 768]
  - Mid-block channels: [768, 512]
  - Self-Attention heads: 8
- **Diffusion Parameters**:
  - Number of diffusion steps: 1,000
  - Noise scheduler: Linear

3.2 Training Configuration

- **Optimization Parameters**:
  - VQVAE learning rate: 8.0e-5
  - UNet learning rate: 8.0e-6
  - Batch size: 36
  - Dropout rate: 0.1

3.3 Conditional Input

The conditional input, consisting of one channel of edges and four channels of semantic map, summarizing to shape [256,256,5]. Its dimension is reduced to [32,32,1] before concatenating with the image's latent features.

4 Results

4.1 VQVAE Autoencoder Performance

The progression of VQVAE reconstruction quality is visualized across different training epochs in Figure 2, Figure 3 and Figure 4. It delivers good reconstruction performance after training for 120 epochs.

4.2 LDM Performance

While the VQVAE delivered good reconstruction capabilities, the LDM showed limitations in generation quality. The comparisons between ground truth images and the generated outputs are presented in Figure 5 and Figure 6.

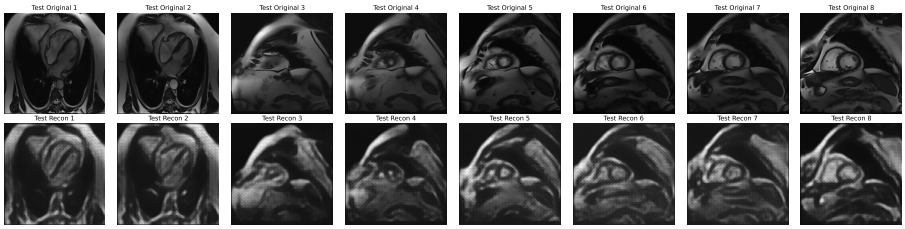

Figure 2: Early-stage reconstruction results (Epoch 3)

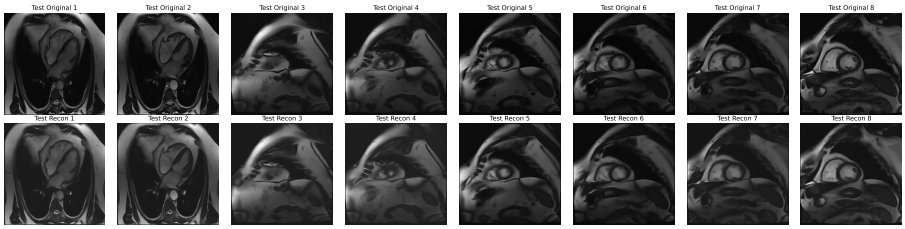

Figure 3: Mid-training reconstruction results (Epoch 74)

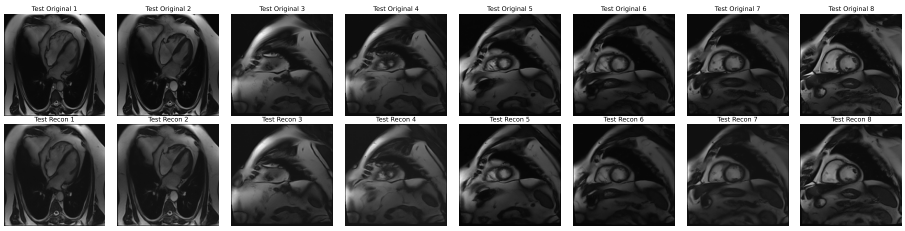

Figure 4: Late-stage reconstruction results (Epoch 20)

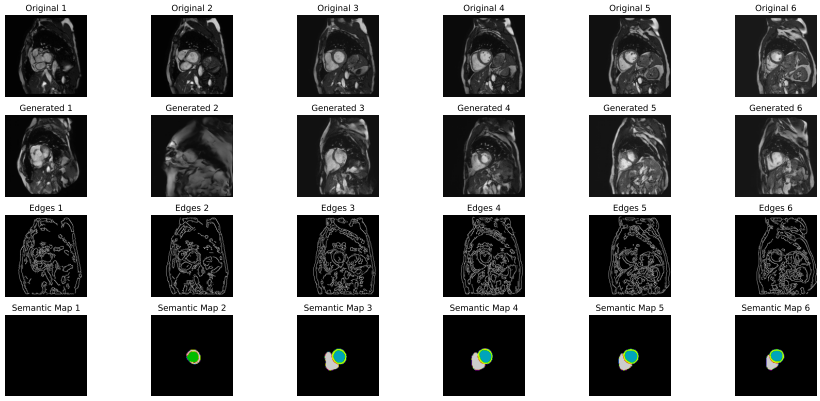

Figure 5: Comparison of generated images (1)

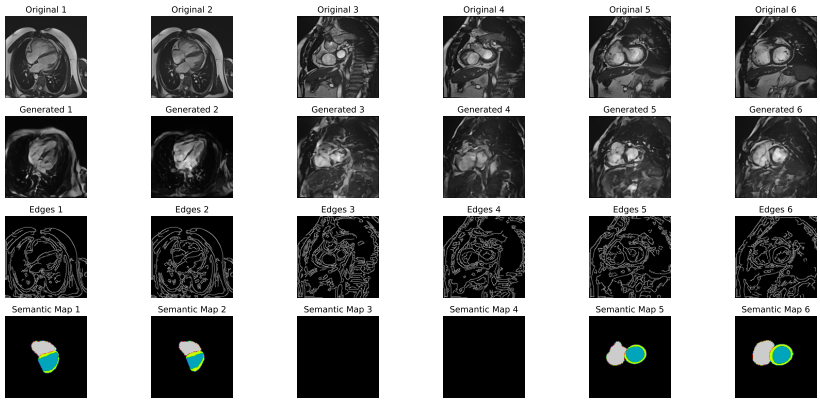

Figure 6: Comparison of generated images (2)

5 Reflections

The intuition behind suboptimal quality of generated images may be attributed to two factors. First, the limited exploration of hyperparameters with insufficient training steps. Second, the mixture of both long-axis and short-axis cardiac images introduced additional complexity. While the challenge guidelines permitting a unified model for both image types, the significant differences in spatial characteristics between them are huge. The long-axis views occupy most of the spatial space, while short-axis views occupying much less.

A consideration emerged while conducting the work, regarding the use of edge maps as conditional inputs. While detailed edge maps effectively capture the characteristics of the original images, this approach may raise concerns on practical usage. In real-world applications, user-provided edge maps are likely to be less detailed than those detected from real medical images. This may be an issue when considering deployment in practical clinical scenarios.

An idea was thought of from considering advances in fine-tuned Llama for clinical applications. The combination of generated doctor-like natural language descriptions specifying the target organ systems and the semantic maps as indicator of the spatial features, may worth investigating.