

Image generation from freehand sketches using Diffusion Models

by

Farhan Tanvir
20101387

Md. Fahim Haque
20101014

Kazi Rifatul Islam
20101438

A thesis submitted to the Department of Computer Science and Engineering
in partial fulfillment of the requirements for the degree of
B.Sc. in Computer Science

Department of Computer Science and Engineering
Brac University
May 2024

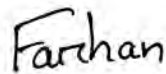
© 2024. Brac University
All rights reserved.

Declaration

It is hereby declared that

1. The thesis submitted is my/our own original work while completing degree at Brac University.
2. The thesis does not contain material previously published or written by a third party, except where this is appropriately cited through full and accurate referencing.
3. The thesis does not contain material which has been accepted, or submitted, for any other degree or diploma at a university or other institution.
4. We have acknowledged all main sources of help.

Student's Full Name & Signature:



Farhan Tanvir
20101387



Md. Fahim Haque
20101014



Kazi Rifatul Islam
20101438

Approval

The thesis titled “Image generation from freehand sketches using Diffusion Models” submitted by

1. Farhan Tanvir (20101387)
2. Md. Fahim Haque (20101014)
3. Kazi Rifatul Islam (20101438)

Of Spring, 2024 has been accepted as satisfactory in partial fulfillment of the requirement for the degree of B.Sc. in Computer Science on May 19, 2024.

Examining Committee:

Supervisor:
(Member)



Dr. Md. Ashraful Alam
Associate Professor
Department of Computer Science and Engineering
Brac University

Program Coordinator:
(Member)

Dr. Md. Golam Rabiul Alam
Professor
Department of Computer Science and Engineering
Brac University

Head of Department:
(Chair)

Dr. Sadia Hamid Kazi
Chairperson and Associate Professor
Department of Computer Science and Engineering
Brac University

Acknowledgement

Firstly, all praise to the Great Allah for whom our thesis have been completed without any major interruption.

Secondly, to our supervisor Dr. Ashrafal Alam sir for his kind support and advice in our work. He helped us whenever we needed help.

And finally to our parents for their throughout support. With their kind support and prayer we are now on the verge of our graduation.

Table of Contents

Declaration	i
Approval	ii
Acknowledgment	iii
Table of Contents	iv
List of Figures	vi
List of Tables	vii
Nomenclature	viii
Abstract	ix
1 Introduction	1
1.1 Problem statement	3
1.2 Research Objective	4
1.3 Research Orientation	4
2 Literature Review	5
2.1 GAN based methods	5
2.2 Diffusion model based methods	11
3 Research Methodology	13
3.1 Workflow	13
3.2 Input Dataset	14
3.2.1 ImageNet Dataset	14
3.2.2 Sketchy Dataset	16
3.3 Dataset Preprocessing	16
3.4 Proposed Approach	17
3.4.1 KAN-Sketch Guider	17
3.4.2 Sketch to Image Generation Process	18
3.5 Description of the model	20
3.5.1 Kan Sketch Guider	20
3.5.2 KAN	21
3.5.3 Clip Interrogator	22
3.5.4 Diffusion Model	22
3.5.5 Latent Diffusion Model	23

3.5.6	Attention Mechanism	24
3.5.7	U-Net	24
3.6	Loss Functions	26
3.6.1	MSE Loss	26
3.6.2	SSIM	26
3.7	Evaluation Metrics	26
3.7.1	FID	26
3.7.2	LPIPS	26
3.7.3	Inception Score	27
4	Experiments and Results	28
4.1	Experimental Setup	28
4.2	Training Details	28
4.2.1	Training Analysis	29
4.3	Results	30
4.4	Comparisons	32
4.4.1	Qualitative Evaluation	32
4.4.2	Quantitative Evaluation	33
4.5	Analysis	35
4.5.1	Ablation Study	35
4.5.2	Comparison with other methods	36
4.6	Applications	36
4.7	Limitations and Future Work	37
5	Conclusion	38
	Bibliography	42

List of Figures

1.1	Some images generated with SOTA diffusion models	2
3.1	Flowchart of workflow	13
3.2	Construction process of the training dataset	14
3.3	Sketch Samples of our training dataset	15
3.4	Real Samples of our training dataset	15
3.5	Sketchy dataset sample	16
3.6	Our proposed approach	17
3.7	Training process of our model	18
3.8	The architecture of our model	20
3.9	A simple KAN	21
3.10	CLIP Interrogator working process (Souce: [39])	22
3.11	Diffusion Process	23
3.12	Latent Diffusion Model Framework	24
3.13	Simplified architecture of U-Net	25
4.1	Training comparison	29
4.2	Overall Training Loss	29
4.3	Generated images of our model	30
4.4	More generation samples	31
4.5	Comparison of generated images	32
4.6	Image quality comparison with varying steps	35
4.7	Impact of different β values on image quality	35
4.8	Comparison with SDEdit[36]	36
4.9	Limitations of our model	37

List of Tables

4.1	Comparison between LEP_{MLP} [46] and our model	33
4.2	Human Evaluation Comparison	34

Nomenclature

The next list describes several symbols & abbreviation that will be later used within the body of the document

AE Auto-Encoder

DCNN Deep Convolutional Neural Networks

GAN Generative Adversarial Network

KAN Kolmogorov Arnold Networks

MLP Multi-layer Perceptron

MRU Masked Residual Unit

Abstract

Significant advancements have been made in the field of image-to-image translation and image synthesis in recent years. Generation of images from sketches is a popular topic in this field. It has many use cases in day-to-day life especially for artists. One useful kind of generative model that has recently come into use for this purpose are diffusion models. In this thesis, we investigate this topic further by developing an efficient approach to generate sufficiently similar images from simple sketch inputs using diffusion models. We utilize a custom Kolmogorov Arnold Network (KAN) based model to provide guidance to a pre-trained diffusion model, so that it generates an image following the input sketch. We also compare our approach with other existing methods and also evaluate their performance. Additionally, we experiment our model with various types of sketch styles containing varying levels of details to demonstrate its robustness. The results show that our method is able to produce images from freehand sketches efficiently.

Keywords: KAN; Diffusion Models; Sketch-to-Image; Generative AI

Chapter 1

Introduction

Drawing a sketch is one of the easiest and fastest ways to picture something. It gives a vague representation of the scene/image. Sketches are used for various purposes. For example: drawing the outline of a face, making the blueprint of a building, drawing the outline of a scenery, designing the layout and structure of characters & maps in games, drawing cartoons, digital painting etc. Drawing a sketch is very simple and requires very less effort. However, the tricky part comes when converting this abstract sketch into an actual image or 3D model. Even for professionals, it is quite a difficult and time-consuming task. One skill that humans have that we would want computers to replicate is the ability to create a fully realistic visual representation of an object or scenery from a rough sketch. Due to recent advancements in the field of artificial intelligence, it is now possible to effortlessly generate images from a sketch, reducing the hassle of a human artist. However, creating a realistic and accurate image from an arbitrary drawn sketch is still a task which machines can't do perfectly. This is because there are various angles to a sketch and it is quite difficult to accurately understand the artist's intent from a simple sketch with very little detail. A single sketch can be visualized by different people in various different ways with varying levels of detail. The simple the sketch the harder it is to depict what it truly meant. This is a very difficult task for a computer as there exists millions of possibilities of how the output image might be. For this reason, there is a necessity of a textual prompt to accurately understand the artist's depiction of the sketch.

Lots of different approaches have been introduced in the last few years to solve this problem. In the beginning, this problem was approached with image retrieval based methods ([4], [2]), which in reality, do not have any creation ability. These models work by searching a large database of images for the given input sketch by extracting and matching common features within them. Then came GANs[22] which had true generative capability however had other limitations. These methods were quickly replaced after the rise in popularity of Diffusion Models.

Image to Image translation was first introduced by Isola et al.[10]. Their suggested model was called pix2pix, which can be used in various tasks like labels to facade, sketch to image, image recoloring etc. Since then, there have been various new developments in this field which opened newer applications and possibilities using sketch-to-image translations. For example, the paper [31] suggests a method of sketch-based hairstyle design by extending the original pix2pix method. The paper

[34] proposes an architecture image translation method to synthesize realistic building images from freehand architecture sketches.

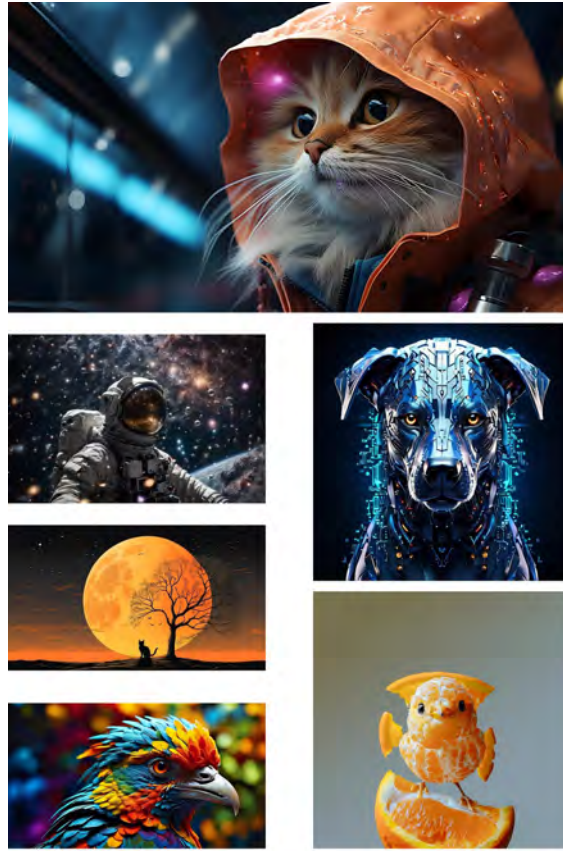


Figure 1.1: Some images generated with SOTA diffusion models

However, the dominance of GANs quickly faded after the introduction of image generation techniques using diffusion models. DDPM[23] introduced an approach for image generation by randomly adding noise to the original distribution following a markov chain and then learning to remove that noise completely to produce another similar distribution. Moreover, the paper [26] showed that diffusion models outperform GANs in every possible way in the task of image generation. From then, diffusion models became the new SOTA for image synthesis. The process of adding iterative noise is the forward diffusion process and the process of denoising is the reverse diffusion process. Diffusion models are a probabilistic generative model employed for the purpose of image production. The process operates by repeatedly applying a series of alterations to an initial image in order to generate a different image. Every stage of transformation adds a specific level of distortion to the image, which increases as the process continues. The noise gradually spreads out throughout the screen, leading to intricate and varied visual patterns.

A diffusion model typically consists of a sequence of diffusion stages, which are the forward and reverse process. In the forward process, Gaussian noise is added iteratively into the original image throughout a number of timesteps t . At the end of this process, the original distribution is completely destroyed and a completely random noise is produced. In the reverse process, the model learns to remove the

noise slowly to produce a new image similar to the original distribution. In this way, diffusion models synthesize images. Diffusion models produce higher quality images compared to GANs [26]. Because these models can learn complicated data distributions and are resilient to changes in the input shapes, designers and artists can effortlessly transform their ideas into visually appealing artwork. Following similar ideas, we propose a method for image synthesis from human hand-drawn sketches using diffusion models. In particular, we propose a custom model for edge detection during the generation stage of latent diffusion models, which can be used to generate images from sketches by controlling the generation process.

Following similar ideas, in this thesis, we propose a method for image generation from human hand-drawn sketches using diffusion models. In particular, we propose a custom model for edge detection in the latent space of diffusion models, which can be used to generate images from sketches.

Our contributions in this study include:

- We utilize a custom KAN based model for edge detection in the latent space.
- Utilize SSIM as a loss during inference process to generate more realistic images matching the sketch input.
- Find an optimal set of hyperparameters for our model through parameter tuning.
- Reduce the number of inference steps needed to generate images compared to our baseline model.
- Integrate a sketch simplification network to make the model generate images efficiently.
- Integrate CLIP-Interrogator in the pipeline to make user given text prompts optional.
- Improve efficiency of our baseline model and reduce the time required for generation.
- Suggest a better method for generating images from freehand sketches.

1.1 Problem statement

Generation of realistic and accurate images from sketches is an important task and an area of active research in the field of computer vision. It has many applications in law enforcement, films and animation, digital art, video games etc. This can also help amateur artists to easily generate realistic images from sketches. Previous approaches in solving this task were met with various challenges like lacking precise details, unable to preserve the unique style of the sketches, failure to generate realistic images with rich texture and details, failure to capture in-depth details etc. We aim to tackle these issues by providing a fast and efficient diffusion model based method to generate images automatically from input sketches. Moreover, our goal is to achieve state-of-the-art evaluation metrics on our proposed method.

1.2 Research Objective

The purpose of this research paper is to examine diffusion model based approaches for image generation. Various methods are evaluated and tweaked to provide more accurate image generation using augmented datasets. The research objectives of this thesis are:

1. To understand the most recent approaches in sketch-to-image tasks.
2. To synthesize realistic looking photos from novice sketches using a diffusion model.
3. To evaluate existing diffusion models for image generation.
4. To create a deep learning based model for image generation from sketches.
5. To explore various diffusion model based approaches for sketch to image translation
6. Enhancing performance, accuracy and generation quality further by adjusting existing models.

1.3 Research Orientation

In this thesis paper, we have organized our work into different chapters for easier understanding. The 1st chapter consists of the introduction, the 2nd chapter includes literature review, the 3rd chapter has the research methodology, the 4th chapter has the results & analysis and finally the 5th chapter contains the conclusion. Throughout chapter 2, we have examined numerous research papers related to the topic and have provided a brief summary of those research's approaches, models used, results etc. We provide our methodology and model description in chapter 3 and follow up chapter 4 with our results.

Chapter 2

Literature Review

2.1 GAN based methods

Image to Image translation models can be used for sketch to image generation tasks. Pix2Pix[10], a conditional GAN, was the first model for image to image translation, which can be used in various tasks like labels to facade, sketch to image, image recoloring etc. Since then, there have been various new developments in this field which opened newer applications and possibilities using sketch-to-image translation. CycleGAN[11] proposed a method for unsupervised image to image translation. AODA [42] suggested a method for open domain unsupervised sketch to image translation similar to the idea of cycleGAN. SketchyGAN [13] proposed an end-to-end trainable approach for sketch to image synthesis using GANs. Chen et al. in paper [13] mentions that the process of quickly visualizing a scene or object is by using sketches and then using these sketches to produce realistic images is challenging as there is limited dataset. To solve this problem, they proposed SketchyGAN, an end-to-end trainable approach for sketch to image synthesis using GAN. Previously, sketch to image synthesis required image retrieval techniques and complex post-processing. Now, with the emergence of deep convoluted neural networks and generative adversarial networks, a deep learning model can be employed for image synthesis. The authors expanded an existing sketch dataset with a bigger dataset of paired edge maps and images in an effort to circumvent the difficulties of getting paired photos. They taught the model to go from making images from edges to making images from sketches. Additionally, they included several loss terms to improve image quality as well as a MRU, which uses an internal mask to determine information flow. The study shows comparisons with pix2pix[10] model which shows that in terms of faithfulness of input sketch, augmented pix2pix [10] (65.9%) compared to SketchyGAN (47.4%). However, in terms of realism, SketchyGAN shows 53.7% better realism than pix2pix.

In [20], the authors have proposed a new approach, to generate realistic images containing many objects from hand drawn sketches and have also built a new dataset called SketchyCOCO, which is based on COCO-Stuff [12]. Their proposed architecture, EDGEGAN, is the first deep neural network architecture for generating images from freehand sketches containing multiple objects. They mainly divided the task of image generation from sketches into two separately-trained sequential stages: foreground generation and background generation. The foreground generation module

focuses on generating a foreground image from the freehand sketch given by the user, strictly maintaining the user requirements detail by detail. Instead of directly generating the foreground image from the input sketch, it is generated from an attribute vector learned by the model during its training stage. In the background generation stage, the authors train pix2pix[10] to convert the image synthesized in the foreground generation stage to produce the final image. To match the background with the foreground, pix2pix is leveraged and the foreground image is used as a constraint. The model is compared against other models like pix2pix [10], SketchyGAN[13] etc. The results prove that this model produces much more realistic and diverse images than the others and is far superior compared to them by comparing various metrics.

In the paper [19], the authors introduce a conditional GAN with self-attention to generate realistic human-face images from lines/edge maps that vaguely describe a face. The output images are well-aligned with the input image. The paper talks about the issues for which previous GAN based models failed to render highly detailed realistic face images from lines/edge maps/sketches. In solution to the issues, the authors propose a conditional self-attention module (CSAM). Using CSAM, information can be parsed by the higher layers from the input image while capturing deep-level dependencies. Moreover, the authors build a multi-scale discriminator while capturing details of the image from various depths. This discriminator pushes the generator to yield realistic and detailed images with accurate facial features. The conditional self-attention module is added before the last convolution layer. The authors train their model on the CelebA-HD dataset by randomly selecting 24k images and extracting the edges using a deep edge detector to generate edge maps. Various evaluation metrics like Inception scores [7], Fréchet inception distance [9] etc. are used to measure the model’s performance. The experiments produce promising results compared to previous approaches. This proposed model allowed the generation of high-detailed images from edgmaps even when some facial structures are missing in the input images.

In paper [27], it states that the conversion of black-and-white facial sketches into lifelike colored images is a crucial issue in image processing and machine vision, and it is covered in this work. There are three different categories of facial sketches: seen, forensic, and composite. Due to variations in structural and morphological traits, it can be difficult to match sketches with realistic images. To overcome this, the research suggests a framework for converting face sketches into high-resolution, high-quality, and colorful photographs. The model transforms an intermediate latent vector into the final image using a GAN. In order to increase the similarity between the synthesized photo and the input face sketch, it captures high-level qualities from the input face sketch as a feature vector, maps those traits into the latent space of the GAN, and then optimizes the latent vector. Unlike paper [13], the proposed model does not require training on paired sketch-photo data, and the experimental results demonstrate its effectiveness in terms of both qualitative and quantitative measures. The study showed results comparing with other models like DualGAN, CycleGAN[11], psp in terms of realism which showed the proposed model (75.89%) beating DualGAN (8.03%), CycleGAN (4.35%) and psp (11.72%).

The authors of paper [17] did a study that addresses the creation of a conditional GAN-based interactive picture generation system. The ability to convert abstract inputs into actual images has been achieved by existing models, but the current user interfaces make it challenging for users to generate drawings gradually since they demand the complete edge or label map as input like in paper [13]. The suggested system is a recommender system that creates a complete picture from only a few of the user’s partial strokes or sketches. Through the use of a gating-based conditioning method, it uses a single conditional GAN model to represent numerous image classes. The technique uses a two-stage methodology to give the artist input on the overall item shape, allowing for speedy refining of higher-level geometries. Performance is better when the completion and image production processes are separated than when partial outlines are converted straight into images. The second step makes use of a multi-class generator that is dependent on a user-supplied class label and uses a gating mechanism to concentrate on key components unique to a given class. With this method, a single generator and discriminator may be trained across many object classes, producing a deployable model that can be used in a variety of scenarios. The study shows that in single class generation, using two stage produces more accurate images compared to single stage across multiple datasets. In multi class generation, the system produces a 97.38% average accuracy.

The paper [21] discusses the task of Lab2Pix, which involves generating realistic images from sketch labels. Realistic images have been successfully produced by supervised approaches, but unsupervised scenarios do not work well with current architectures. The CycleGAN[11] model has been a ground-breaking effort in leveraging a cycle model to translate labels to actual images. It can synthesize intricate items in scenarios with multiple objects and use an excessive amount of resources to show the finer features of a single object. The unsupervised version of this problem is difficult to solve because of the disparity between the labels that are input and the images that are generated. The authors presented a unique framework that they named Lab2Pix with the goal of synthesising real-life images in a consistent manner as a solution to these challenges. The generator was developed to produce images with increasingly greater resolutions during a single forward step. The final couple of layers of the generator are where the framework places label guided spatial co-attention (LSCA) blocks. These blocks integrate features from multiple levels and refine outputs by making use of low-level attributes that are guided by the labels. A segmentation component is used to verify the produced outputs against the input labels. The authors recommend using the network for three different functions in order to increase the likelihood that it would produce lifelike photos. First, genuine photos are scaled up and down to create fuzzy samples in order to define a sharp enhancement loss. In order to differentiate between synthetic and real-world data, discriminators must be trained, which forces the generator to produce sharper images. A second picture consistency loss is included so that multi-scale images can be aligned at the feature level and the training process can be stabilised. Third, in the adversarial loss function, the synthesis of the foreground is improved. Experimental results show that the proposed method performs significantly better compared to CycleGAN. It shows that adjusted Lab2Pix achieves comparable Fréchet Inception Distance scores or FID[9] scores (98.0 vs 83.9), reduced training time (0.4 times less) and less computational resources (11 GB vs 8 GB).

In paper [41], the authors introduce DeepPortraitDrawing as a deep generative approach for generating lifelike human images from low-quality, hand-drawn input. The method addresses the challenges of sketch-based synthesis by leveraging part-level shape spaces, refining sketch and parsing maps, and employing global synthesis and face refinement networks. The experimental results validate the effectiveness and practicality of the proposed method, offering visually pleasing results with realistic local details. The researchers conducted a study to prove the effectiveness of their approach for synthesizing lifelike human portraits from hand-drawn sketches. Their method was compared against various other well-known methods. All models were trained on the same dataset. Their method produced the best FID scores which is 50.36, then GauGAN (51.92), pix2pixHD (70.87) and pix2pix (71.12).

The paper [28] introduced a self-supervised learning approach for exemplar-based sketch-to-image synthesis, disregarding the necessity of sketch-image paired data. They present a unique way of generating images from sketches using Auto-Encoders and GANs. Firstly, the authors propose generating line-sketches for RGB only datasets, which allows synthesis of multiple sketches for a single image. Then using a self-supervised Auto-Encoder, the style features and contents are decoupled from the sketches and RGB images from the dataset. The AE is formed of two separate encoders: a style encoder and a content encoder. A decoder takes the extracted features from both the encoders and produces the final image. The auto encoder generates images that maintain the details of the sketch and are similar in style with the RGB images. The authors also discuss various optimization techniques for performance and synthesis quality. A GAN is used to refine the output produced by the AE, making the model more efficient. The authors evaluate their model on the CelebA-HQ and WIKIART dataset, which yields a top-notch performance on 1024^2 resolution. The self-supervision mechanisms presented by the paper gives a significant performance boost on the tasks of sketch to image translation.

In paper [16], the authors present a novel approach to synthesize photorealistic face photos from sketches using GANs. The generator utilizes a deep residual U-Net architecture while the discriminator adopts a Patch-GAN with residual blocks. Skip connections and residual blocks help the generator produce high-resolution with rich details. The generator's encoder uses a 4×4 kernel and 2×2 stride filters. The last layer of the decoder uses a tanh activation function while the basic structure is formed by combining deconv+bn+res+relu. 5 layers of down-sampling are used by the discriminator among which are 4 residual blocks. The authors think that just using convolution layers are inefficient to properly discriminate the image, so they add residual blocks. The authors also introduce three effective loss functions that enforce pixel, edge, and high-level feature restrictions on the synthesized face images. The performance of the proposed approach is further enhanced by these loss functions. The paper also proposes a data augmentation strategy for photo-sketch pairings to alleviate data scarcity. This reduces overfitting and improves generation efficiency. Using a NVIDIA TITAN X gpu, the model is trained for a thousand iterations. The authors evaluate their method by means of qualitative and quantitative experiments. Furthermore, the paper evaluates the synthesized face photos in the context of face recognition tasks. Comparing the results against other state-of-the-

art methods, the authors state that the photos generated by their method achieves consistently better performance and improves the accuracy of the sketch-to-image tasks.

In the paper [24], the authors have suggested a sketch-to-image synthesis method that is end-to-end trainable and can produce objects from various classes. Their proposed approach accepts sketches of objects as input, and produces realistic images of the same objects in similar postures. The process of image generation is of two stages. In the first stage, their goal is to synthesize an image from the class label in order to establish a uniform baseline for each class. Then in the second stage, they combined the input sketch and the uniform baseline to create the realistic image. The second stage is implemented with the same backbone structure as SketchyGAN [13]. The authors describe a GAN-based sketch-to-image synthesis method that can create images straight from sketches without the need for image retrieval during testing. Their proposed method can learn more accurate representations of the publicly available unpaired data by using a CGAN, which allows it to take full advantage of the data. The authors utilize Inception Scores[7] to evaluate their method on their dataset.

In the paper [42], the authors introduced a first-of-its-kind Adversarial Open Domain Adaptation (AODA) framework that is taught to combine the missing hand-drawn input sketches and allows for unsupervised open-domain adaptation. The authors proposed a translation method for image to sketch and vice-versa, with the intent of converting open-domain pictures into sketches using GANs. The suggested framework and training methodology can produce real looking results, even for inputs of hypothetical classes. The authors compared their suggested method with existing previously developed similar approaches and evaluated their proposed method better since results from both qualitative and quantitative studies demonstrate that this method performs better on both seen and unseen data. To solve the problem of missing details in input sketches, the authors trained a model using their suggested framework. Besides, the authors implemented an open-domain training technique in order to lessen the generator’s bias toward synthesized drawings and take advantage of the generalization of adversarial domain adaptation and this strategy allowed them to generate open-domain classes more faithfully. Moreover, the authors describe their introduced network as an excellent freehand drawing extractor for random photographs. Extensive research, user testing, and analysis of multiple datasets show that the described model in this paper can correctly synthesize genuine images for various types of open-domain freehand sketches.

In the paper[32], the paper introduces a Sketch Transformer network for generation of realistic human faces from sketches. The network incorporates a self-attention mechanism to generate realistic face sketches from photos. The modules that make it up are an MFPEncoder, a residual self-attention block and lastly a MSPADE-Decoder. The MFPEncoder extracts feature embeddings and positional encodings at different scales, while the residual self-attention layer captures long-range spatial dependencies. The MSPADE-Decoder reconstructs the target image using the output of the self-attention module, multi-scale feature embeddings, and positional encodings. Quantitative and qualitative evaluations show that the suggested ap-

proach achieves better output quality than state-of-the-art methods. Evaluation metrics such as LPIPS, FID, and FSIM are used to measure the generated images' quality, with lower LPIPS and FID scores and higher FSIM scores indicating better quality. The Sketch-Transformer model effectively preserves global structures and produces sketch-like textures. The experiments are conducted on public databases, including the CUFS and CUFSF datasets, using standardized image sizes and comparison settings.

In paper [25], the authors state that investigators often utilize forensic artists' facial sketches to recreate images from verbal statements. The subject's genuine features are depicted in these sketches. These sketches aren't good for biometric identification. Edges dominate sketches' pixel information. However, these edge features can contain structural information that helps create high-quality visual representations. Due to the lack of a style guideline, turning sketches into realistic photographs might be difficult. Translation from picture to image, sketch to image, or edge to image has been established. Conditional generative models and image content and style separation are used in these methods. Combining deep convolutional neural networks with generative models like generative adversarial networks (GANs), VAEs, and auto-regressive models has improved data distribution modelling. In that respect, the authors proposed a model where sketches are synthesized into high-resolution graphics. A generator produces photos with diverse target properties while preserving the subject's identification. The model also has a verifier to assure identity consistency and a hybrid discriminator to identify actual and synthesized photographs based on desired qualities. The model uses a quality-guided network to reduce perceptual discrepancies between synthesized and actual images in different network sections to improve image quality. An identity preserving network keeps the subject's biometric identity. The results show a marked improvement over existing models like BP-GAN (86.1 FID), C-GAN (43.2 FID), CA-GAN (36.1), SCA-GAN (34.2), the author's model (34.1).

Generative Adversarial Networks [22] have true generative capability and are superior to older methods like image retrieval methods and VAEs. GANs, by learning the underlying distribution of data, can truly generate images. For this reason, GANs are very popular for image-to-image-translation based tasks. GAN consists of two networks, which work in contrast to each other. They play a minimax like game which results in the production of accurate output. However, the training process of GANs are prone to higher instability due to the complicated nature of training. They suffer from mode collapse and vanishing gradient most of the time. GAN based approaches learn to map an image from a domain to another domain. They work well, however, their big downside is that they cannot generate outside their training domain. Moreover, they lack support for text prompts, which are essential for clearly understanding the artist's intent.

2.2 Diffusion model based methods

Another newer approach for generating realistic images from sketches is by using diffusion models[23]. Various papers proposed ways to synthesize images from sketches using diffusion models like [40], [43]–[46].

The paper [40] uses diffusion models for sketch-controlled image generation. The proposed framework consists of a forward diffusion process and a reverse diffusion process. In the forward diffusion process, the input image is converted into latent noise by slowly adding Gaussian noise to destroy the pixel value distribution. In the process of reverse generation, the latent noise is removed in steps to produce the desired image. The classifier plays an important role in the image generation process. It makes sure that generated images match the category that we want to produce. The authors’ proposed model is called DiffSketching. This is one of the first works done on sketch to photo generation with a diffusion model. It uses diffusion models to generate realistic looking images as well as overcoming the limitations of previous GAN based works.

In paper [43] a unique method to image generation was introduced called DiSS. This approach generates images by taking input from sketches and strokes using diffusion models. This approach differs from other methods because this gives granular control over the position, colour and realism of the generated images in 3D. It also gives control in how closely the generated image would match the given input sketch. In other words, it lets you choose if you want the generated image to be more similar to the sketches or more realistic. This has numerous advantages from using sketches to generate images of different styles to generating a specific part of an input sketch. Also, the results given shows that the author’s claims are valid.

The paper [46] a new approach to manipulate a pre-trained model that transforms text into visual representations by utilizing both sketches and textual instructions. They utilize a Latent Guidance Predictor (LGP) to effectively control the visual representation of objects in the generated images, despite the presence of noisy input. This enables the generation of varied visuals that adhere to the framework of a drawing, even across various artistic styles. The approach is highly economical, as it only requires a minimal dataset for training, and it demonstrates excellent performance across many domains and drawing styles. Additionally, they showcase its application in repairing absent sections of photographs and altering the horizon line. An important benefit is its ability to handle various sketch styles and provide a diverse range of graphics with meticulous control over the displayed elements.

This paper [44] introduces an innovative method for image-to-image translation using diffusion models. In contrast to current approaches, the authors represent img2img translation as a stochastic process, eliminating the reliance on conditional generation. Unlike other methods, where the diffusion model is conditioned with a sketch input to generate the image, the authors approach this problem by using brownian bridge process to map an image into another image directly. This approach leverages bidirectional diffusion to acquire knowledge of the mapping between image domains directly, thereby augmenting the efficiency of translation. By operating in

the latent space of a pre-trained VQGAN, both generalization and efficiency are enhanced. The experimental findings demonstrate that BBDM exhibits competitive performance in both visual and quantitative domains, effectively tackling challenges encountered by alternative methods such as training instability and mode collapse. Notable contributions consist of the elimination of conditional inputs in the prediction phase and the provision of a translation process that is more consistent and diverse.

The paper [45] presents a technique that produces high-quality facial images from basic sketches. Contrary to earlier methods, SGLDM preserves intricate facial characteristics and ensures precise rendering. The system uses a Multi-Auto-Encoder to transform various segments of a sketch into a more streamlined representation, while preserving the geometric properties. The model is trained using paired sketch-face data and then enhanced with a Stochastic Region Abstraction technique to handle different sketch styles. SGLDM surpasses other methods, such as Pix2pixHD and Psp, in generating intricate facial images with diverse expressions and characteristics. The success of the system is supported by qualitative evaluations and user studies, which consistently show its ability to construct realistic facial representations from simple sketches.

Chapter 3

Research Methodology

3.1 Workflow

In this section, we discuss the workflow and methodology of our study. The entire procedure can be summarized by the high level flowchart below:

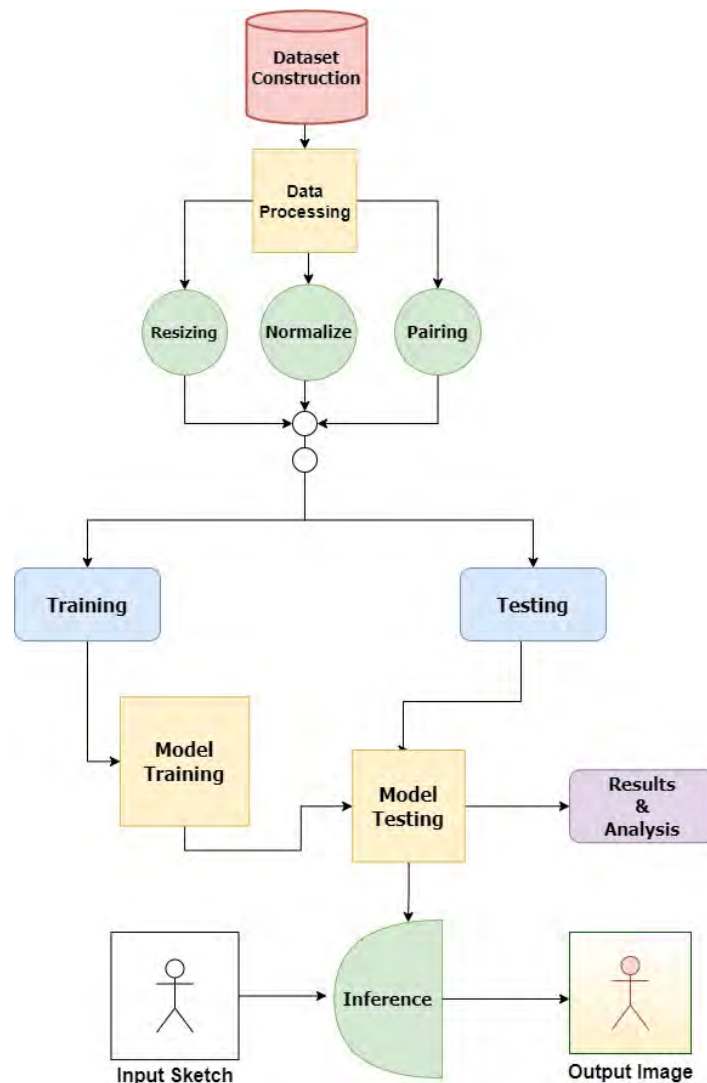


Figure 3.1: Flowchart of workflow

3.2 Input Dataset

The foundation of any successful machine learning model is a high-quality, well-balanced dataset. By guaranteeing an equal distribution of data from a variety of classes, it allows a much better generalization of the data. By feeding a diverse set of inputs, the model learns to handle real-world scenarios better. Moreover, a balanced dataset comes with its own added benefits like being less prone to underfitting/overfitting, having lesser bias towards specific classes, faster convergence etc. Training models on big and balanced datasets helps it to learn to generate better quality images and improve overall performance. We will use two different datasets for training and testing our model:

3.2.1 ImageNet Dataset

Training our model doesn't require any readily available sketch-image pair dataset. As the purpose of our model is to learn edge structures, we can generate edgemaps from any images and train it. For the purposes of our research, we have selected the widely established ImageNet dataset[3] as our training dataset. The dataset is diverse and consists of many different categories with a large number of samples. There are a total of 14 million images in the ImageNet dataset. However, our model

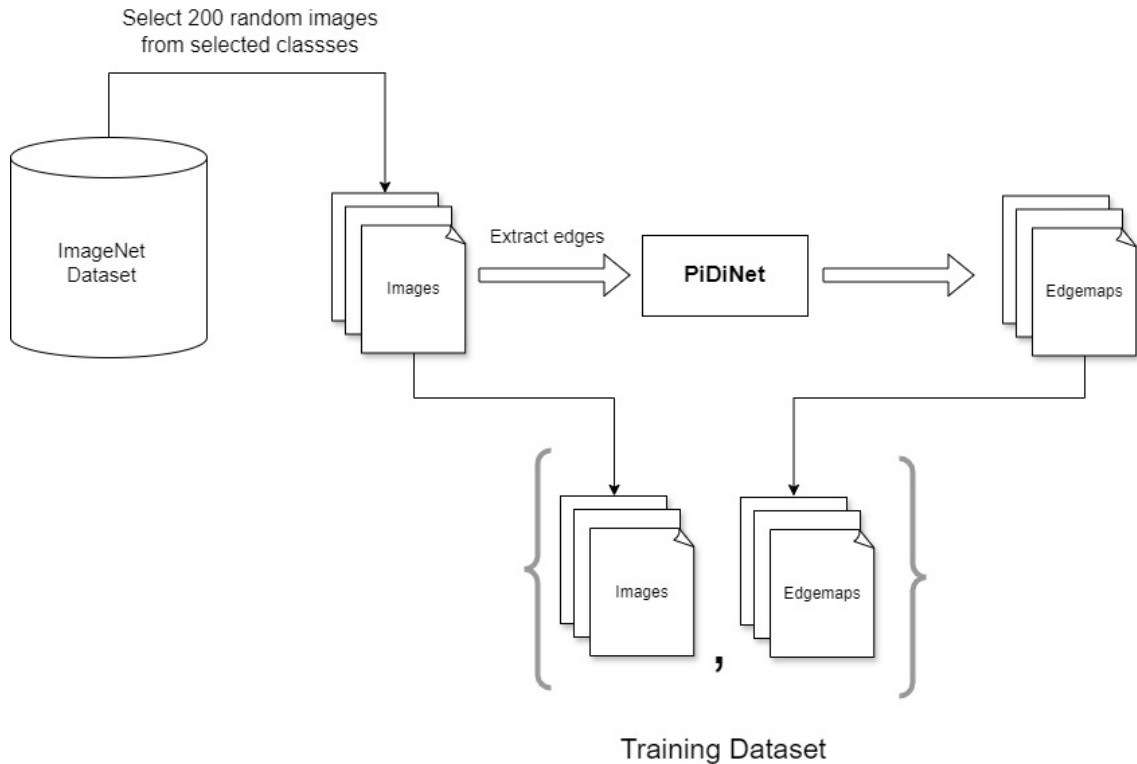


Figure 3.2: Construction process of the training dataset

requires only a few thousand images to be trained. So, we handpick 55 classes from the ImageNet dataset and download random 200 images per class. We collect a total of 11,000 images from the ImageNet dataset. Then, to extract edge-maps from the samples we used PiDiNet [30]. PiDiNet is a CNN that extracts useful edges from

images by using pixel difference. It is useful because it can automatically discard background elements and generate edgemap of the subject only. In this way, we have generated our dataset with 11,000 images and sketches/edgemaps. Samples of the dataset are provided below:

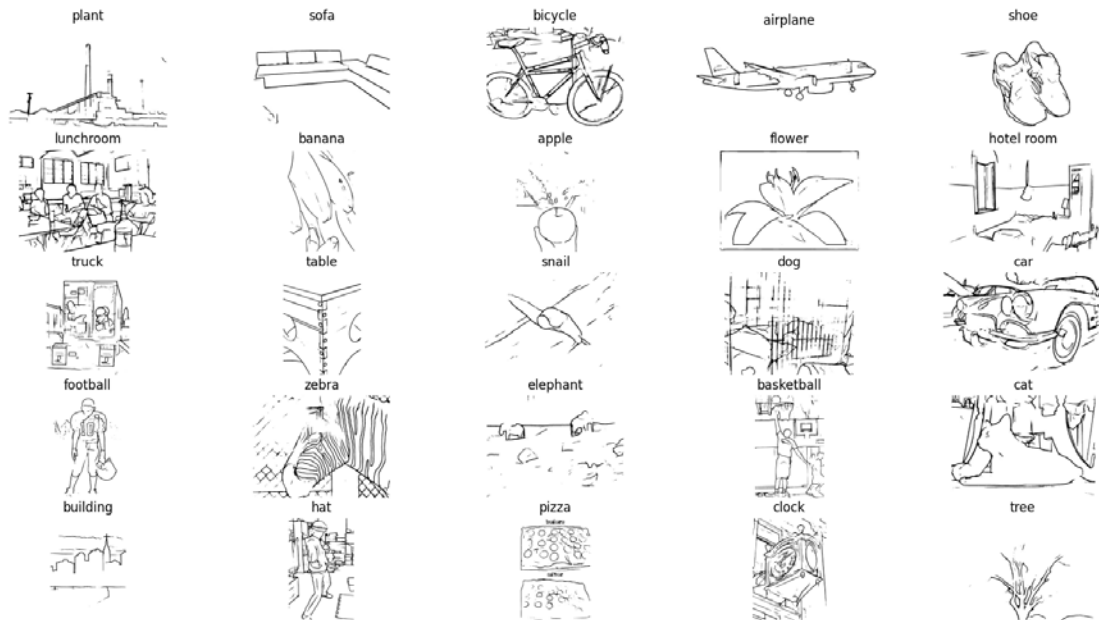


Figure 3.3: Sketch Samples of our training dataset



Figure 3.4: Real Samples of our training dataset

3.2.2 Sketchy Dataset

The Sketchy dataset [8] is an extensive dataset of images and human drawn sketches. We will use it to test our model. The dataset consists of 12,500 images from 125 classes and corresponding 75,000 sketches of each image. Each image in the dataset contains equivalent 5-10 sketches drawn by various artists. Multiple variations of the sketches allows a wide range of sketching styles and degrees of abstraction. We will use this dataset to generate images from our model for evaluation. We will randomly pick a sketching style from randomly selected images of the dataset and evaluate our model with it.

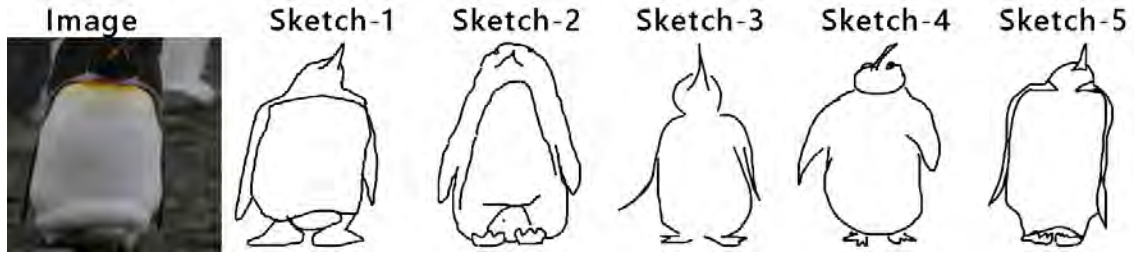


Figure 3.5: Sketchy dataset sample

3.3 Dataset Preprocessing

Due to the variation of sizes and high levels of noise in the dataset's images, preprocessing is required in order to convert them to the standard format. So, our preprocessing includes some of the following:

Resizing: We resize the original input sketch into 512x512 size, so that the images are consistent with the model's input size. The images in our dataset are of varying sizes, so they need to be resized.

Normalization: To stabilize the training process, we will bring the pixel value distribution of the images between -1 to 1. This will help in generalizing better.

3.4 Proposed Approach

The purpose of this research is to analyze the working procedures and limitations of existing sketch-to-image generation models and suggest a method for doing this task with improved performance and accuracy. To achieve this goal, we will use a Diffusion Model. However, our method doesn't require training a conditional latent diffusion model from scratch to generate images from sketches. Rather, we will train a KAN (Kolmogorov Arnold Network) based latent sketch guidance network and use that to provide guidance to a pre-trained latent diffusion model based on input sketches during inference time. The motive behind the sketch guidance model is to learn edge representations based on noisy encoded images and predict edges during the inference stage. The input sketches will be simplified using sketch simplification network [14]. This will allow us to clean up rough sketches or unclean pencil sketches and generate images from it. We will also utilize CLIP-Interrogator to automatically generate text prompts from the input sketch, so that the necessity of giving input prompts from the user can be left optional. After simplifying the input sketch, we will utilize our sketch guidance network during the inference process of the pre-trained diffusion model.

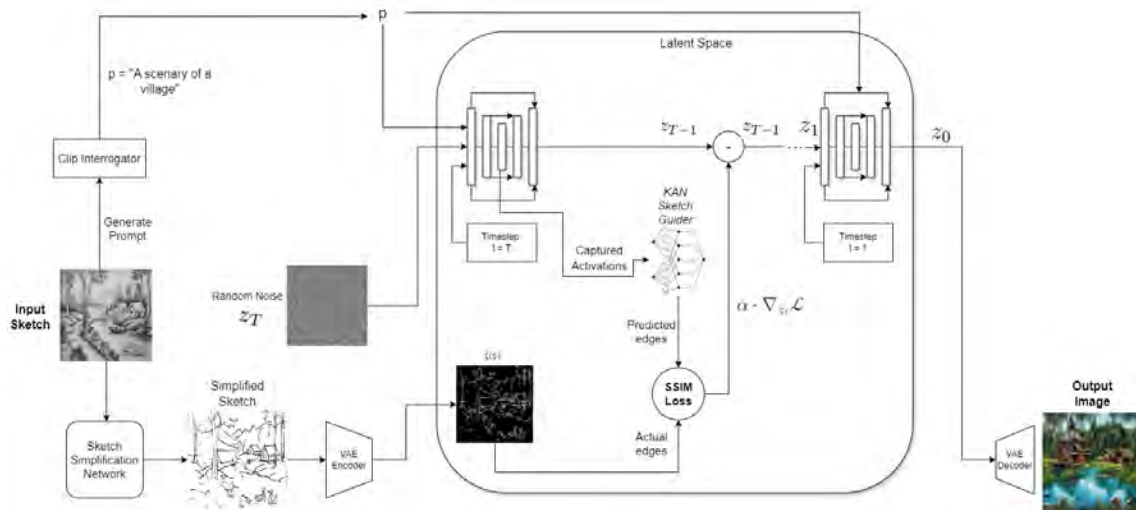


Figure 3.6: Our proposed approach

Our sketch guidance model, KAN Sketch Guider, will estimate the edges generated by the diffusion model by capturing the low level features from the core of the diffusion model. Based on the estimations of the sketch guidance model, we can encourage the generation process of the diffusion model to follow the input sketch and generate a realistic image. This will ensure that the diffusion model is able to generate images corresponding to the freehand input sketch. Below we describe our sketch guidance model and image generation process in further details.

3.4.1 KAN-Sketch Guider

In order to predict edges during the image generation process or inference time, we will utilize a custom model called KAN Sketch Guider (KSG). Our model works similarly to the LEP proposed by Voynov et al. [46]. However, instead of using only

an MLP (Multi-Layer Perceptron), our latent sketch guider is based on a KAN (Kolmogorov Arnold Networks) [47]. Moreover, we use CLIP Interrogator to generate captions of the training dataset and train our model with it, in contrast to [46] who used the class labels for training. The purpose of KSG is to learn edge structures of encoded latent images. It does so by extracting internal features from the core (U-net) of the pretrained diffusion model. Our KSG is trained using pixelwise Mean-Squared Error loss. This ensures that the model learns to map individual pixels to their equivalent edge-maps in the latent space. Due to this pixelwise training, it isn't only limited to the training dataset's images or sketch styles, but rather learns open domain adaptation. So, our model can generate images which are beyond the training dataset.

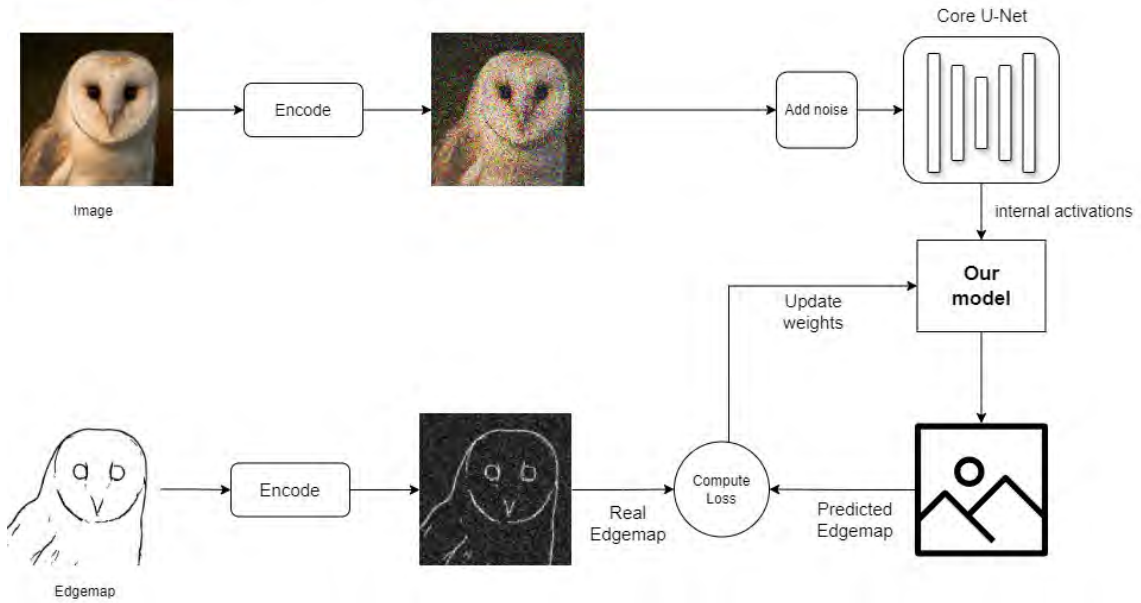


Figure 3.7: Training process of our model

The KSG mainly takes as input a joined list of the internal features of the core U-Net of the diffusion model. The internal features are fetched after the core U-Net is fed with an input x along with a text prompt p and a timestep t . The fetched features are later resized and concatenated to be passed as input to the KSG. Let, the conjoined list of fetched internal features be F . After F is forwarded to the sketch guider, it predicts a latent edge-map of the generated photo from the input x at the given timestep t . Then the loss between the predicted latent spatial map and the real encoded edge map from the training dataset will be computed using MSE loss. In this way, the sketch guider model will be trained. After it fully learns to estimate, it can be used to predict the edgemaps of the noisy latent images and guide the generation process to follow the outline of the input sketch.

3.4.2 Sketch to Image Generation Process

Our goal is to efficiently generate images from a given freehand input sketch s and a text description/prompt p . We have integrated sketch simplification network in this step to cleanup any rough lines or edges in the input sketch. The sketch simplification network returns a simplified version of the sketch with an overall clean outline, which

is helpful for generating high quality images. In addition to that, we have used CLIP-Interrogator to automatically generate a text prompt from the input sketch. This will now make it optional for the user to enter a prompt during the generation process. Our KSG model is used to lead the generation process. The core diffusion model is built with an U-Net to produce a less noisy latent z_{t-1} from a latent z_t at each timestep iteratively from $t=T$ until $t=1$. This is the reverse process. This takes place in the latent space to make it more efficient. A latent space is just a low dimensional encoded space. We will use our sketch guider to use the input sketch as a conditional control to produce an image following the sketch’s structure. At every timestep t , our KSG model will predict an edgemap $P(e)$ in the latent space after the denoised latent z_{t-1} is produced by the backbone U-Net. The simplified input sketch s will also be encoded in the latent space into $\mathcal{E}(s)$ and a noisy version will be produced for each noise level. We will then use SSIM to compare the predicted edgemap $P(e)$ and the encoded input sketch $\epsilon(s)$. In contrary to [46], who used MSE loss to compare the two, we use SSIM as this will help measure how close the two sketches are not only in terms of structural similarity but also in perceived quality. The gradients are then computed based on SSIM. Then the normalized gradient will be subtracted from the produced noisy latent z_{t-1} so that it matches the input sketch. In this way, guidance will be provided to the denoising process in order to produce an output image similar to the input sketch. A parameter β is used to control the strength of guidance from the gradient step. It acts as the parameter to balance between the edge similarity of the final output image and the input sketch. The denoised latent z_{t-1} is updated using the gradient controlled by β . In this way guidance is provided. The sketch-based guidance is provided for only some i steps of the denoising process and the rest $T - i$ steps are left untouched so that the diffusion process isn’t hampered. After the final denoising step is completed, i.e. after $t=1$, the final latent z_0 is decoded by the core model’s decoder to produce the final image.

3.5 Description of the model

Here we provide a brief description of the models used in our suggested approach and also some components of diffusion models.

3.5.1 Kan Sketch Guider

This is the architecture of our custom model, Kan Sketch Guider, as described in section 3.4.1.

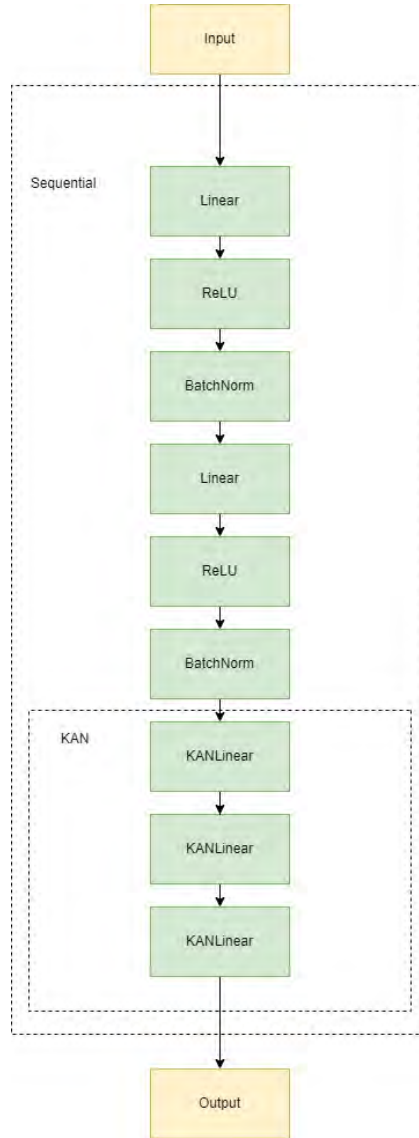


Figure 3.8: The architecture of our model

The input and output size of our KSG model is $(9324,4)$ respectively. The architecture of our KSG model is simple. The first layer is a Fully Connected Layer with input dimension of 9324, hidden layer of size 512 and output size of 256. The next layer is a multi-layered KAN with input size 256 with hidden layers of size $[128,64]$ and output dimension equal to 4. The output size is 4 since this is the number of output channels of the encoder of Stable Diffusion, which is the pre-trained diffusion model we chose.

3.5.2 KAN

Kolmogorov-Arnold Network (KAN) [47] is a new type of neural network architecture which works as a replacement for traditional MLPs. The theory behind KAN is the Kolmogorov-Arnold representation theorem, while the theory behind MLP is the universal approximation theorem. The Kolmogorov-Arnold representation theorem makes it possible to define any multivariate continuous function as a composition of multiple univariate functions. Kolmogorov-Arnold networks use this theory to arrange the network in a way that facilitates learning complex multivariable functions. The formula can be expressed by:

$$f(\mathbf{x}) = \sum_{q=1}^{2n+1} \Phi_q \left(\sum_{p=1}^n \phi_{q,p}(x_p) \right)$$

KANs learn these functions i.e. have learnable functions instead of learnable weights. There are two primary layers of the network. Each node in the first layer of the network structure computes a continuous function that is subsequently used to alter the input. In the network structure's second layer, each node applies a weighted sum of the first layer's outputs to a continuous function. The final output equals to the sum of the second layer outputs. One of the differences between KANs and MLPs is that the activation functions of KANs are on the edges while the activation functions of MLPs are on nodes. The authors of [47] have shown that KANs outperform MLPs in various ways. KANS learn faster than MLPs and have higher accuracy. For these reasons, we have decided to build our model based on KANs.

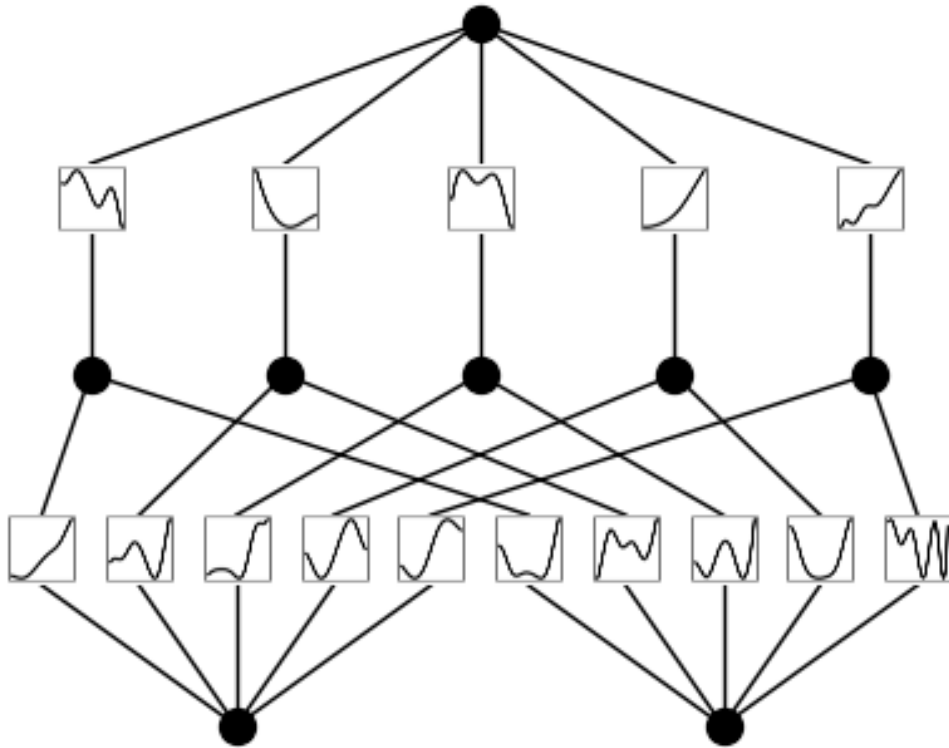


Figure 3.9: A simple KAN

3.5.3 Clip Interrogator

CLIP Interrogator is a tool that automatically generates prompts for images. This is done using BLIP[35] and CLIP[29]. The process by which CLIP Interrogator works is by feeding an image to BLIP, it will get a description as output and the same image is also fed to CLIP which gets the text embedding. By comparing the image embedding with label embedding from many lists, four most similar embeddings are selected. Finally, by combining these selected texts, it produces the final prompt, which provides an optimal visual description of the image.

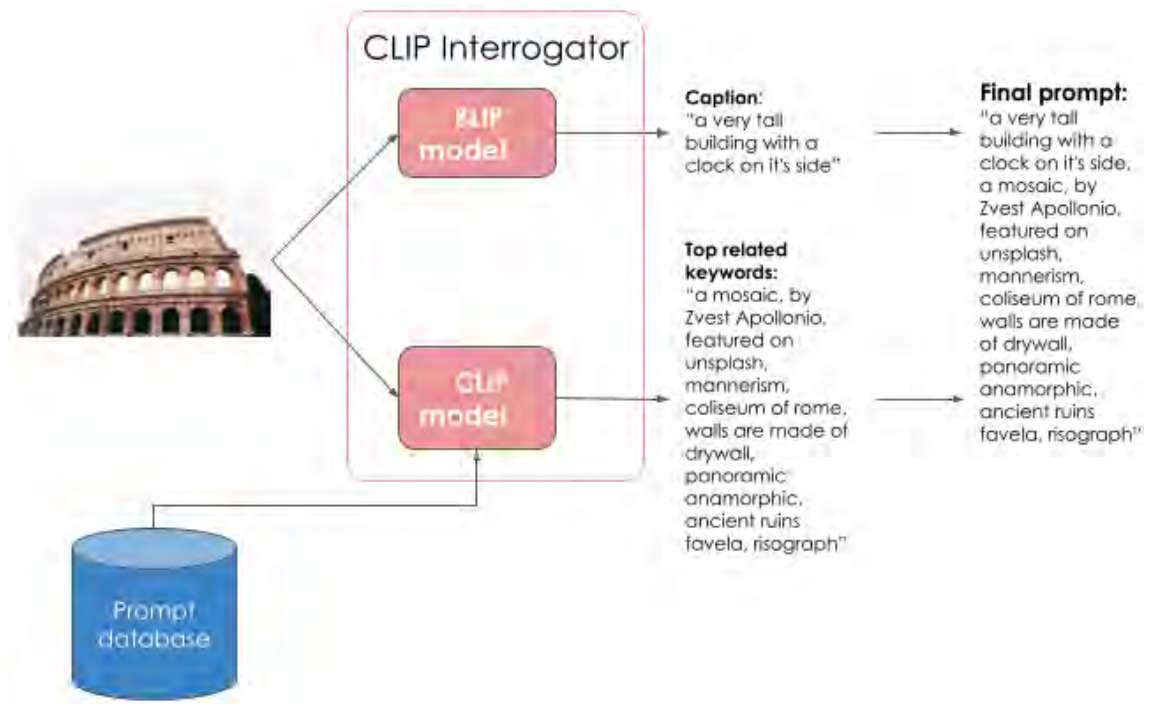


Figure 3.10: CLIP Interrogator working process (Souce: [39])

3.5.4 Diffusion Model

Diffusion models became a popular choice for image synthesis after DDPM[23] proposed a new method to generate images using them. Diffusion models consistently beat GAN based models in generating images. The working principles behind Diffusion models are described below:

DDPM [23] generates images by adding noise to them and later denoising them iteratively. The forward process and the reverse process are the two main processes involved in this. The model follows a Markov chain to transform the input photo into a pure random noise during the forward step. Progressively, Gaussian noise is iteratively introduced to the initial data sample, resulting in the production of a completely new random noise at the end of this process. Adding noise completely destroys the overall distribution of the initial image. As the step size increases, the attributes of the original data sample decrease and the resulting image gradually resembles a Gaussian distribution. Subsequently, during the process of reverse diffusion, the model gradually eliminates this noise through repeated steps, resulting

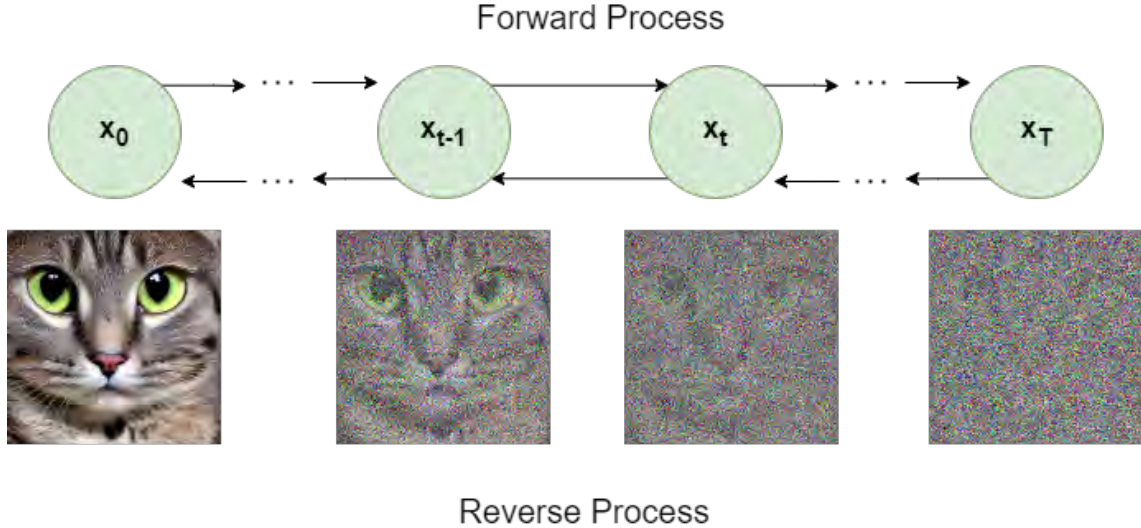


Figure 3.11: Diffusion Process

in the generation of another distribution that closely resembles the original one. This technique utilizes Variational Autoencoders and a U-net. The reverse process operates as a sequence of denoising autoencoders, often implemented using a U-Net architecture with weight sharing. The purpose of training these autoencoders is to predict the denoised versions of their noisy inputs.

3.5.5 Latent Diffusion Model

To reduce the computation requirements, [37] suggested to move the diffusion process in the latent space. A pre-trained encoder and decoder are used to facilitate the transformation of the image from the pixel-space to the latent space and vice versa. By shifting the diffusion process in the latent space, we can reduce the computational cost required for training by a significant amount. This is what the Latent Diffusion Model (LDM) achieves. LDMs produces good looking and high quality images comparable to traditional diffusion models.

LDM uses a method called perceptual compression. Here, using a patch based adversarial objective and perceptual loss, an auto encoder is trained. LDM lowers the dimensionality of the data by mapping it to the low dimensional latent space, which removes information that are imperceptible. This strategy aims to enhance the efficiency of model training and inference by reducing computational complexity and minimising storage requirements.

The Latent Diffusion Model framework illustrated above, trained an auto encoder, which includes an encoder and a decoder. The diffusion process is carried out on the latent representation space following the encoder ϵ 's compression of the image x to latent representation z . The diffusion process of Latent Diffusion Model and regular Diffusion Model are comparable. Latent Diffusion Model takes a data sample z from noise z_T and it uses a decoder D to restore the sample to the original pixel space and thus producing the final image which is \hat{x} .

A pre-trained perceptual compression model, including the encoder and the decoder, allows LDM to obtain noise samples z_t in the latent space and perform the forward

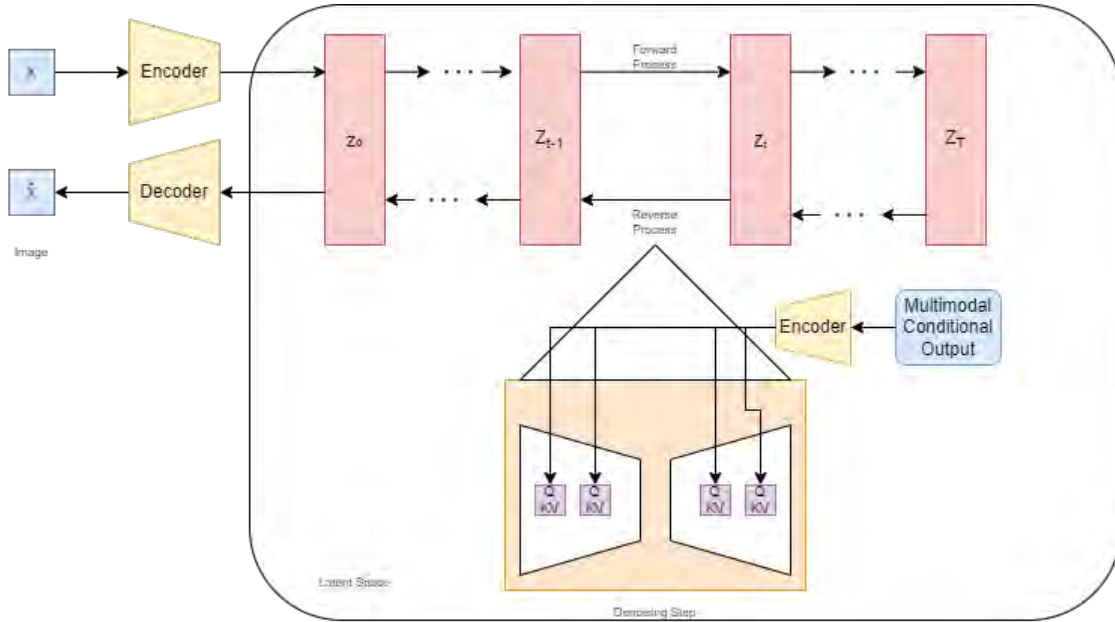


Figure 3.12: Latent Diffusion Model Framework

diffusion process there. An LDM is trained with the following loss:

$$L_{\text{LDM}} := \mathbb{E}_{\mathcal{E}(x), \epsilon \sim \mathcal{N}(0,1), t} [\|\epsilon - \epsilon_{\theta}(z_t, t)\|_2^2]$$

3.5.6 Attention Mechanism

The attention mechanism is crucial for computer vision tasks such as object detection and image recognition. It enables the models like Transformer, BERT, and GPT to concentrate on important regions or features of the input. The two types of attention mechanism are self attention mechanism and cross attention mechanism. Cross-attention analyses the connections between different components of multimodal inputs. On the other hand, self-attention focuses on the relationships between the same input.

More precisely, cross-attention allows for the incorporation of different forms of data that are not related. One mode is employed for inquiries, while another mode is utilized for both keys and values in cross-attention. Latent Diffusion Models utilize this procedure to produce conditional images by including cross-attention into the U-Net backbone network. Latent Diffusion Models offer many forms of conditioning throughout the process of image synthesis. This is achieved by employing a domain-specific encoder to translate conditional information, such as text or drawings, to an intermediate representation.

3.5.7 U-Net

U-Net [5] is a deep learning network designed for semantic segmentation tasks. It utilizes an encoder-decoder architecture, which allows it to efficiently finish segmentation using less amount of training data. The encoder is composed of convolutional and pooling layers, which serve to decrease the dimensions of the input image and extract information at a higher degree of abstraction. The decoder improves the

accuracy of segmentation and restores the image size by preserving contextual information. This is achieved by employing transposed convolutions and including skip connections from the encoder.

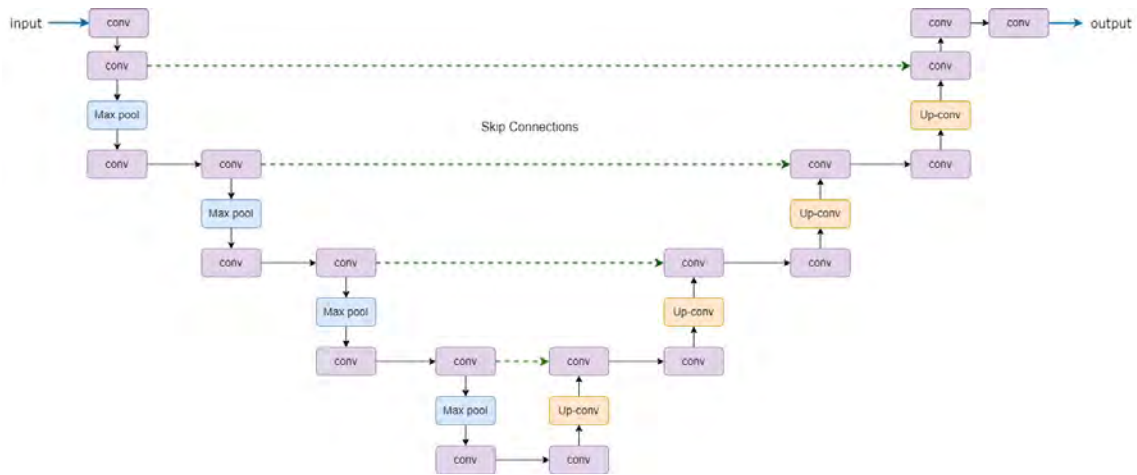


Figure 3.13: Simplified architecture of U-Net

In the Latent Diffusion Model, U-Net is enhanced by incorporating a temporal embedding module and a spatial transformer module (cross-attention). Time embedding enables the model to utilize temporal connections by converting time information into a continuous vector space. The Latent Diffusion Model incorporates temporal information to repeatedly forecast noise, enabling U-Net to gradually enhance noise predictions. The cross-attention module in the Latent Diffusion Model enhances the interaction between text and picture data by linking specific textual information with corresponding regions of the noise matrix.

3.6 Loss Functions

We describe below some of the loss functions used for training our model.

3.6.1 MSE Loss

Mean Squared Error loss is a very popular and well known loss. Diffusion models are mostly trained with mse loss. It is also very effective in guiding the model during the inference process. In case of training diffusion models, this loss is the most popular. It measures the average of the squared error of the ground truth and the predicted noise of the diffusion model, for every pixel of the latents, in each time interval. We utilize this loss for training our model. It is defined by:

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

3.6.2 SSIM

Structural Similarity Index is used to evaluate structural similarity between the generated images and real images. Although it was introduced as an evaluation metric years ago, it can also be used as a loss function. The values of SSIM are always within -1 and 1. The higher the value, the higher the structural similarity between the images and vice versa. It not only measures the outline of the two images but also other factors like structure and luminance. This will be utilized in our inference process to generate images with high visual perception. The SSIM value is calculated by:

$$SSIM(x, y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)}$$

3.7 Evaluation Metrics

Evaluation metrics are necessary to judge the generated output quality of generative models. Metrics are also used to compare between different models and frameworks. We describe below some of the metrics we used for evaluation:

3.7.1 FID

Fréchet Inception Distance (FID) [9] is used to evaluate the efficacy of generated output images. If the values of FID are low, then the distributions of the produced and real data are very comparable. An Inception-v3 network is used to extract features for this measure. So, lower FID scores are preferable.

3.7.2 LPIPS

LPIPS is designed to be visually compatible with human vision by concentrating on perceptual image similarity and comparing images using a learned perceptual

measure. Higher levels of visual similarity are indicated by lower LPIPS scores. The combination of these criteria allows for a thorough quantitative evaluation of diffusion models, highlighting distributional alignment and perceptual quality.

3.7.3 Inception Score

One way to measure the quality and diversity of images generated by generative models like Diffusion Models is with the Inception Score (IS). It is a measure of the originality and diversity of the generated images. To calculate the score, the photographs are first classified using a pretrained Inception v3 model. The entropy of the class distribution is taken into account for both the total distribution of the produced images and for each individual image. Inception Score is a way to compare two distributions. This metric does a good job at capturing visual diversity and quality. A higher Inception Score indicates better fidelity of generation, including both exceptional image quality and a wide range of variations.

Chapter 4

Experiments and Results

4.1 Experimental Setup

We performed various experiments with different model architectures and datasets to create our custom model. Moreover, we trained our model with various batch size, edge maps styles and hyperparameters to finally select a model architecture and training hyperparameters. We also performed various inference/generation tests with our model and other state-of-the-art models, including various GAN based models, for analysis and comparison. We ran all the models on the default hyperparameters provided by the respective authors to prevent any bias in our analysis. We chose Stable Diffusion v1.5 as the pre-trained diffusion model for our approach. It is trained on 5 billion images from the LAION-5B dataset [38]. For the edgemap generation, we generated edgemaps initially with Im2Pencil [18] on various datasets but found much better results with the edgemaps generated with PiDiNet[30] on a subset of ImageNet dataset. All our experiments were performed on device with hardware configurations:

- Intel Core i9 13900K CPU
- Nvidia RTX 4090 GPU
- 64 GB RAM
- Windows 10 OS

After running various tests, we picked the right epoch, batch size, hyperparameters, along with loss functions and evaluation metrics for training and testing our model. The details of those along with the results and analysis are provided below.

4.2 Training Details

We trained our Kan Sketch Guider model using the images and generated edgemaps from samples we collected from the ImageNet dataset. The images and edge maps were normalized from -1 to 1. Pixel intensity less than 0.5 will be converted to 0 and greater than 0.5 will be converted to 1. This makes it easier for the model to converge. Our training dataset is formed with triplets of {image,sketch,caption}. The sketches / edgemaps were generated with PiDiNet and image descriptions with

Clip-Interrogator. The use of proper prompts instead of only the class label as caption will help the model learn better. Next, our model is trained similarly as described in section 3.4.1. We have used pretrained Stable Diffusion v1.5 as the core diffusion model. The internal activations are captured from the core diffusion model’s 9 feature blocks. The images and edgemaps are encoded into the latent space by the diffusion model’s encoder. The text encoder of Stable Diffusion is used to encode the text captions into tokens. The architecture of our custom sketch guider network is quite simple. The first layer is made with Fully Connected Layers with input size equal to the total size of the extracted activations. However, they are resized to 9324 as the size of all the activations are not the same. We use 9324 as the input dimension of our model. So, our model’s first layer’s input size is 9324 and output size is 256, with a hidden dimension of size 512. Then we have utilized a KAN based multilayered structure. The KAN is formed of 3 total layers, with the input size of first layer being 256 and hidden layers of sizes 128,64. The output size of the last layer is 4, which is equal to the number of output channels of diffusion model’s encoder. For training we have used the MSE loss to measure the pixelwise loss between the predicted edgemaps and real edgemaps in the latent space. We have used the Adam optimizer to optimize the learnable parameters. The learning rate is set to 1.5×10^{-4} . We have used a constant schedule with warmups of 300 steps as the learning rate scheduler. The diffusion timesteps were set to 250 steps for training. We train our model for a total of 6 epochs or 66,000 steps. Each epoch or 11,000 steps took around 30 minutes on a single RTX 4090 GPU.

4.2.1 Training Analysis

We trained the MLP LEP[46] with the default settings as suggested by the author and compared our model with it. Our model outperforms the LEP_{MLP} . Our KAN based sketcher guider learns much faster and better than the MLP based Latent Edge Predictor. The model converges very fast compared to LEP and the training loss of our model is much lower.

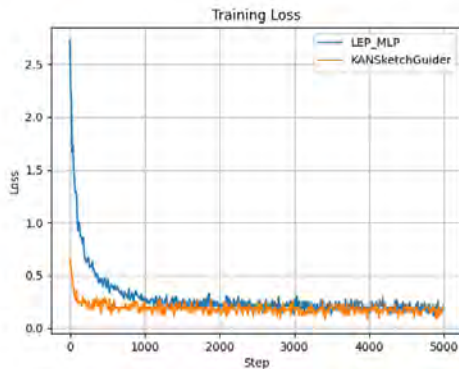


Figure 4.1: Training comparison

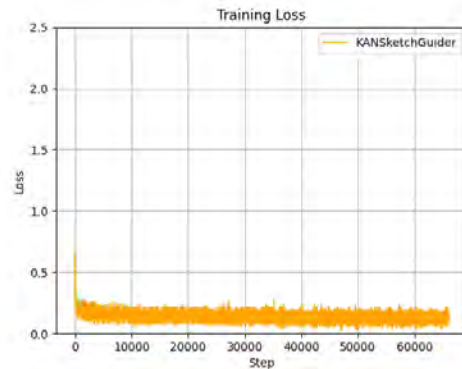


Figure 4.2: Overall Training Loss

As seen from the figure 4.1, our model starts stabilizing very early, only after 1000 steps whereas the LEP_{MLP} [46] takes around 5000 steps to start stabilizing. Figure 4.2 shows the overall training loss of our model for all the 6 epochs.

4.3 Results

In this part, we will describe the inference process details along with their results, comparisons and analysis with other baseline models.

The image generation process is described in the section 3.4.2. For the hyperparameters of our model, we chose $\beta = 0.65$ and classifier free guidance[33] level = 6.5. We set our inference steps=55. The sketch-based guidance from our model is provided for 50% of the total steps. The input sketches are simplified with sketch simplification. The images were generated with random sketch samples from the Sketchy dataset[8]. The output image size is set to 512×512 . Below are some of the images generated with our approach:

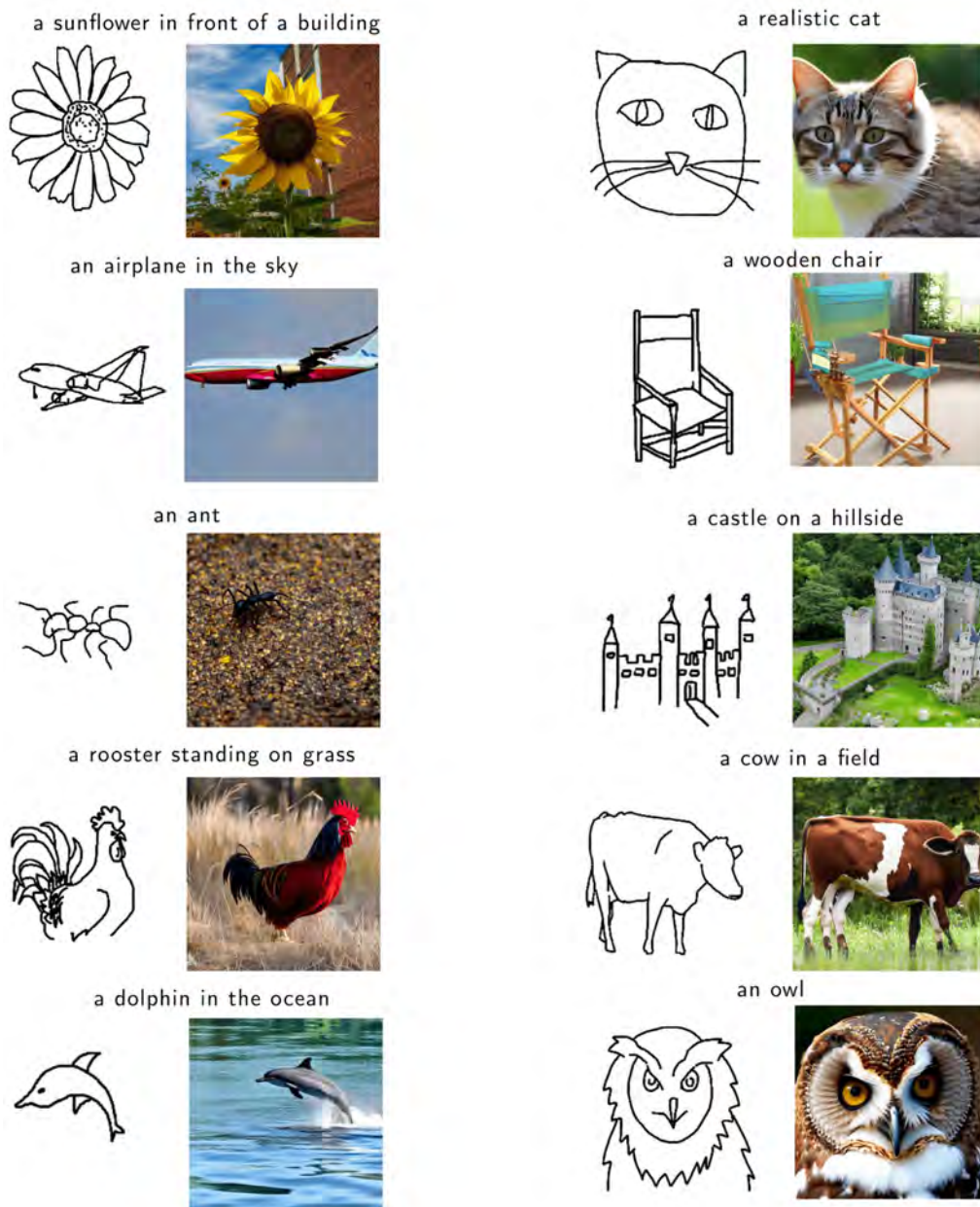


Figure 4.3: Generated images of our model

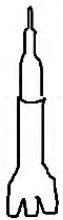
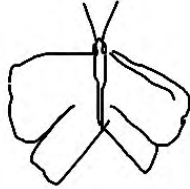
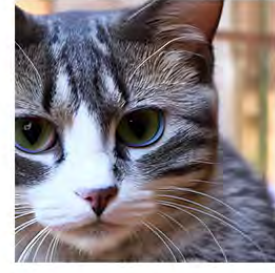


Figure 4.4: More generation samples

4.4 Comparisons

We will compare our model and approach with other similar models in this section. We choose the latent edge predictor (LEP) from [46] as our baseline for comparison. LEP_{MLP} was trained for the same number of epochs as ours, which is 6 epochs. The default parameters suggested by the authors were used to generate images from the LEP_{MLP} , which is $\beta = 1.6$ and MSE loss for inference. For our model, we used the parameters described in the previous section, which is $\beta = 0.65$ and SSIM[1] as inference loss. The seed value, number of inference steps and guidance level were all kept the same for both the models in order to prevent any bias between the comparisons. We synthesize images from both the models using the same prompts we generated using CLIP-Interrogator, for fair comparison.

4.4.1 Qualitative Evaluation

We conduct qualitative analysis between our model’s outputs and the output of our baseline model LEP_{MLP} . Below we provide the images generated from both the models.

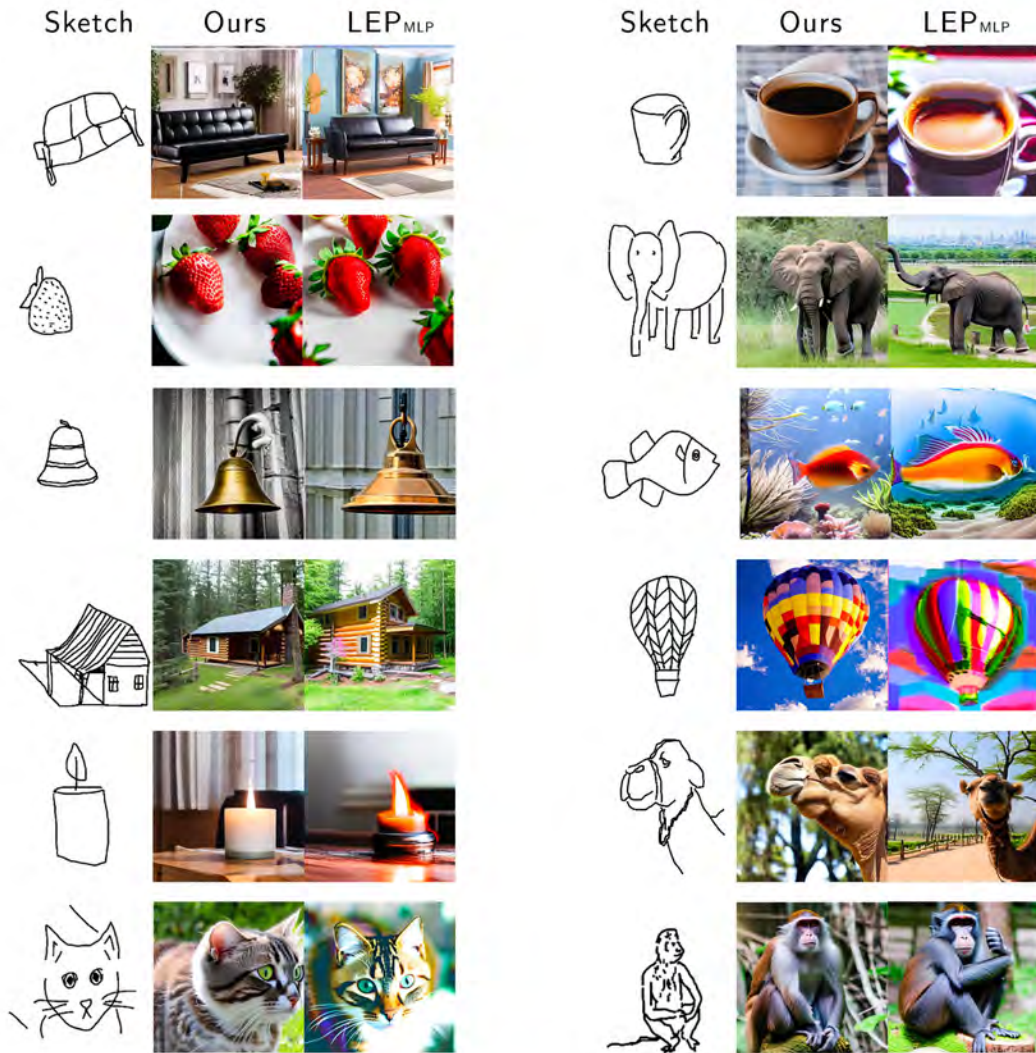


Figure 4.5: Comparison of generated images

We can see from the figure 4.5 that our model’s generation capability is much higher and better. The images generated with our approach align much better with the input sketch. This is due to the fact that we used SSIM loss during the generation process instead of the MSE Loss. In some cases where the LEP_{MLP} produces outputs capturing the sketch’s outline, it comes at the cost of realism of the image. However, our model doesn’t suffer from this issue because of the SSIM loss we used. SSIM not only measures structural similarity between images but also other factors which impact perceived quality. Due to this, our model can generate more realistic images with much better alignment to the freehand input sketch.

4.4.2 Quantitative Evaluation

Here, we compare our model which is the KAN Sketch Guider (KSG) with the MLP based Latent Edge Predictor (LEP) and found that KSG outperforms LEP_{MLP} comprehensively in a wide range of evaluation metrics including Frechet Inception Distance (FID) [9], Inception Score (IS) [6] and Learned Perceptual Image Patch Similarity (LPIPS) [15]. We used these metrics to evaluate the quality of our generated images. For that purpose, we have generated 2,200 samples from each of the two models. It took around 2.5 hours to generate the requisite amount of samples per model with our machine. The LEP_{MLP} model has generated images with stock parameters according to the paper and both models have generated images with 55 steps with each iteration taking around 6 seconds to complete. We also generate another 2200 images using LEP_{MLP} using our own hyperparameters (e.g. $\beta = 0.65$) to show the comparison results. The results are given below:

Model	FID ↓	IS ↑	LPIPS ↓
LEP_{MLP}	59.46	27.293	0.819
$LEP_{MLP}(\beta = 0.65)$	39.51	38.349	0.807
Ours	35.56	42.317	0.791

Table 4.1: Comparison between LEP_{MLP} [46] and our model

FID score measures the similarity of the original image with the generated image. This shows how closely the generated image resembles the original image. A lower FID score means better quality images. FID scores also indicate a greater sample variety with the generated images. As we can see from the table above, our model has achieved a FID score of 35.56 which is much lower than LEP whose FID score is 59.46. This indicates our model produces images much closer to the original sample than LEP. Moving over to the Inception Score (IS), we can see that our model achieves a higher Inception Score (42.32) than LEP (27.293). In this case, a higher score is better than a lower one. Inception Score measures the quality of an individual generated image. It provides a measure of each of the individual images from a collection of generated images based on factors like diversity and quality. As the Inception Score is better in our model compared to LEP, the quality of generated images are much greater and diverse. The LPIPS metric on the other hand measures perceptual similarity in contrast to the quality evaluation of the previous metrics. This metric has been shown to match with human perceptions as well. Here, a lower score indicates a better image. As we can see from the table,

our model has also achieved a lower Lpips score than LEP as well. It scored 0.791 compared to LEPs 0.819. So, taking all of the scores in account, we can conclude that our model produces better looking images that match with human perception as well as sketches with a greater variety.

Human Evaluation

We conducted human evaluation of our model’s output. The images generated by generative models cannot be properly evaluated without human feedback. For this reason, we conducted an anonymous survey with the outputs of our model and LEP_{MLP}. We provided some of the sketches and outputs of the two models and told users to select which of the images they preferred. The provided outputs were randomly selected. The users were not informed about which model generated which output. The survey was conducted in between a total of 19 participants. The users were asked to select an output image of their preference according to the sketch and also rate the two outputs on a scale of 1 to 5, based on two factors: sketch similarity and realism. The results are provided below:

Model	User Preference↑	User Rating↑	
		Sketch Similarity	Realism
LEP _{MLP}	29.48%	3.112	3.187
Ours	70.52%	3.656	4.025

Table 4.2: Human Evaluation Comparison

As we can see from the table above, our model was consistently chosen by survey participants over the LEP_{MLP} model. It shows that 70.52% of survey participants chose our generated images compared to 29.48% of LEP_{MLP}. It also shows that they found our images more realistic with a rating of 4.025 compared to LEP_{MLP} (3.187). They have also found our images to be more similar (3.656) to the input sketches compared to LEP_{MLP} (3.112). So, in conclusion, the survey showed our images to be more realistic and similar to the sketches and thus the survey participants chose our images more consistently.

4.5 Analysis

We perform various tests with our and other models for analysis and ablation study.

4.5.1 Ablation Study

We test the impact of different parameters in our model and compare it with our baseline model. We test the impact of β and the number of inference steps in the generation process. We generate images with our model and LEP_{MLP} with varying inference steps while keeping everything else the same. The resulting images are given below:



Figure 4.6: Image quality comparison with varying steps

The results show that our model can generate good looking images with as low as 15 inference steps only. However, for realistic high quality images it takes 25 steps on average. However, the authors of [46] suggested 250 inference steps for their model. That is a 90% difference in inference steps. So, our model has a faster inference time.



Figure 4.7: Impact of different β values on image quality

The parameter β is a control over the strength of guidance from our model. Higher β values will cause the denoising process to follow the sketch more strongly. The figure 4.7 shows the results of the same image produced with different β values.

4.5.2 Comparison with other methods

Our model generates images similar to [46], by using a model for edge prediction in the latent space of a pretrained diffusion model. There are other approaches for image to image translation using generative models. Here, we discuss how our compares with those. Pix2Pix[10] generates images by training a dedicated generator against a discriminator. It was the first model to introduce image to image translation. However, it cannot generate outside its training dataset. So, it is not very useful. SDEdit[36] suggested an approach for image to image translation with Latent Diffusion models[37]. However, it cannot perform sketch to image translation. When it is given an input sketch, it returns another output sketch. However, our approach can generate realistic images from sketch inputs.

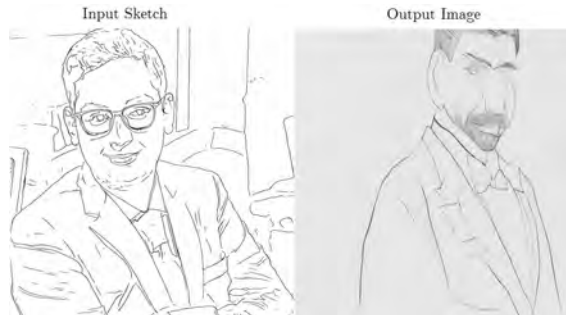


Figure 4.8: Comparison with SDEdit[36]

4.6 Applications

1. Artists can generate sample images to see how a given sketch might look like.
2. Entry level comic book authors can generate images to streamline the process of creating panels.
3. Concept arts can be quickly created using the generated images.
4. Marketers can quickly produce images for ad presentation.
5. Can be useful for generating illustrations quickly for books or other educational materials.

4.7 Limitations and Future Work

There are some limitations in our approach. Firstly, our model cannot generate images with matching faces. We found through our experiments that, when a human face sketch is provided to the model, it fails to produce images of similar faces. Moreover, if there are multiple subjects in a single sketch, our model sometimes gets confused and mixes up the subjects in the final image. In addition to that, we saw that sometimes the model struggles to produce images of different artistic styles.

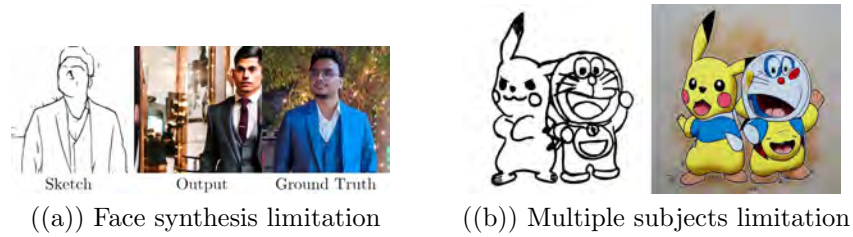


Figure 4.9: Limitations of our model

We left it as a future work to fully inspect these problems and come up with a better solution.

Chapter 5

Conclusion

Before the arrival of diffusion models and GANs, image generation was done using image retrieval methods like Photosketcher [4], Sketch2photo [2]. This approach worked well for some time but it required much precise data and had no true generation capability. On the other hand, diffusion models have made a tremendous impact on the field of image generation. Diffusion models have also provided enormous benefits in other applications such as video games, texture enhancement, image up-scaling etc. Our paper focuses on improving the existing methods on producing a realistic image from a sketch.

In this paper, for generating realistic images from sketches, we suggested a diffusion model based approach that improves on the foundation of previous approaches. We utilize a pre-trained diffusion model, in particular Stable Diffusion 1.5, and produce images from freehand sketches with great quality. Our main contribution lies in generating images from source sketches by using a custom KAN based model for edge estimation in the latent space and providing a few adjustments to the model for improving performance. We propose a custom kan based model to detect edges in the latent space. This model will be used to guide the image generation process of a pretrained diffusion model from human drawn sketch inputs. We replace MSE loss with SSIM loss and integrate other modules like sketch simplification and clip interrogator, in the generation process, which improves generation quality of our model. Our model has been able to produce images from input sketches drawn by amateur artists efficiently and improved the performance of existing models with similar approaches. We conducted various studies along with human evaluations, which shows the efficacy of our method compared to previous methods. The research undertaken in this paper will help in the field of criminology, digital art, video game development etc. In general, diffusion based models have revolutionized the field of image generation and have become a vital tool in various fields, but additional research and development are necessary to solve the limitations and make it more widely available.

Bibliography

- [1] Z. Wang, A. Bovik, H. Sheikh, and E. Simoncelli, “Image quality assessment: From error visibility to structural similarity,” *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600–612, 2004. DOI: 10.1109/TIP.2003.819861.
- [2] T. Chen, M.-M. Cheng, P. Tan, A. Shamir, and S.-M. Hu, “Sketch2photo: Internet image montage,” *ACM transactions on graphics (TOG)*, vol. 28, no. 5, pp. 1–10, 2009.
- [3] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, “Imagenet: A large-scale hierarchical image database,” in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, 2009, pp. 248–255. DOI: 10.1109/CVPR.2009.5206848.
- [4] M. Eitz, R. Richter, K. Hildebrand, T. Boubekur, and M. Alexa, “Photosketcher: Interactive sketch-based image synthesis,” *IEEE Computer Graphics and Applications*, vol. 31, no. 6, pp. 56–66, 2011.
- [5] O. Ronneberger, P. Fischer, and T. Brox, *U-net: Convolutional networks for biomedical image segmentation*, 2015. arXiv: 1505.04597 [cs.CV].
- [6] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen, *Improved techniques for training gans*, 2016. arXiv: 1606.03498 [cs.LG].
- [7] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen, “Improved techniques for training gans,” *Advances in neural information processing systems*, vol. 29, 2016.
- [8] P. Sangkloy, N. Burnell, C. Ham, and J. Hays, “The sketchy database: Learning to retrieve badly drawn bunnies,” *ACM Trans. Graph.*, vol. 35, no. 4, Jul. 2016, ISSN: 0730-0301. DOI: 10.1145/2897824.2925954. [Online]. Available: <https://doi.org/10.1145/2897824.2925954>.
- [9] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter, “Gans trained by a two time-scale update rule converge to a local nash equilibrium,” *Advances in neural information processing systems*, vol. 30, 2017.
- [10] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, “Image-to-image translation with conditional adversarial networks,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 1125–1134.
- [11] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, “Unpaired image-to-image translation using cycle-consistent adversarial networks,” in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2223–2232.
- [12] H. Caesar, J. Uijlings, and V. Ferrari, “Coco-stuff: Thing and stuff classes in context,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 1209–1218.

- [13] W. Chen and J. Hays, “Sketchygan: Towards diverse and realistic sketch to image synthesis,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 9416–9425.
- [14] E. Simo-Serra, S. Iizuka, and H. Ishikawa, “Mastering Sketching: Adversarial Augmentation for Structured Prediction,” *ACM Transactions on Graphics (TOG)*, vol. 37, no. 1, 2018.
- [15] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, *The unreasonable effectiveness of deep features as a perceptual metric*, 2018. arXiv: 1801.03924 [cs.CV].
- [16] W. Chao, L. Chang, X. Wang, J. Cheng, X. Deng, and F. Duan, “High-fidelity face sketch-to-photo synthesis using generative adversarial network,” in *2019 IEEE International Conference on Image Processing (ICIP)*, IEEE, 2019, pp. 4699–4703.
- [17] A. Ghosh, R. Zhang, P. K. Dokania, *et al.*, “Interactive sketch & fill: Multiclass sketch-to-image translation,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 1171–1180.
- [18] Y. Li, C. Fang, A. Hertzmann, E. Shechtman, and M.-H. Yang, *Im2pencil: Controllable pencil illustration from photographs*, 2019. arXiv: 1903.08682 [cs.CV].
- [19] Y. Li, X. Chen, F. Wu, and Z.-J. Zha, “Linestofacephoto: Face photo generation from lines with conditional self-attention generative adversarial networks,” in *Proceedings of the 27th ACM International Conference on Multimedia*, 2019, pp. 2323–2331.
- [20] C. Gao, Q. Liu, Q. Xu, L. Wang, J. Liu, and C. Zou, “Sketchycoco: Image generation from freehand scene sketches,” in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 5174–5183.
- [21] L. Gao, J. Zhu, J. Song, F. Zheng, and H. T. Shen, “Lab2pix: Label-adaptive generative adversarial network for unsupervised image synthesis,” in *Proceedings of the 28th ACM International Conference on Multimedia*, 2020, pp. 3734–3742.
- [22] I. Goodfellow, J. Pouget-Abadie, M. Mirza, *et al.*, “Generative adversarial networks,” *Communications of the ACM*, vol. 63, no. 11, pp. 139–144, 2020.
- [23] J. Ho, A. Jain, and P. Abbeel, *Denoising diffusion probabilistic models*, 2020. arXiv: 2006.11239 [cs.LG].
- [24] Z. Li, C. Deng, E. Yang, and D. Tao, “Staged sketch-to-image synthesis via semi-supervised generative adversarial networks,” *IEEE Transactions on Multimedia*, vol. 23, pp. 2694–2705, 2020.
- [25] U. Osahor, H. Kazemi, A. Dabouei, and N. Nasrabadi, “Quality guided sketch-to-photo image synthesis,” in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, 2020, pp. 820–821.
- [26] P. Dhariwal and A. Nichol, *Diffusion models beat gans on image synthesis*, 2021. arXiv: 2105.05233 [cs.LG].

- [27] N. M. Farid, M. S. Fard, and A. Nickabadi, “Face sketch to photo translation using generative adversarial networks,” *arXiv preprint arXiv:2110.12290*, 2021.
- [28] B. Liu, Y. Zhu, K. Song, and A. Elgammal, “Self-supervised sketch-to-image synthesis,” in *Proceedings of the AAAI conference on artificial intelligence*, vol. 35, 2021, pp. 2073–2081.
- [29] A. Radford, J. W. Kim, C. Hallacy, *et al.*, *Learning transferable visual models from natural language supervision*, 2021. arXiv: 2103.00020 [cs.CV].
- [30] Z. Su, W. Liu, Z. Yu, *et al.*, *Pixel difference networks for efficient edge detection*, 2021. arXiv: 2108.07009 [cs.CV].
- [31] C. Xiao, D. Yu, X. Han, Y. Zheng, and H. Fu, “Sketchhairsalon: Deep sketch-based hair image synthesis,” *arXiv preprint arXiv:2109.07874*, 2021.
- [32] M. Zhu, C. Liang, N. Wang, X. Wang, Z. Li, and X. Gao, “A sketch-transformer network for face photo-sketch synthesis.,” in *IJCAI*, 2021, pp. 1352–1358.
- [33] J. Ho and T. Salimans, *Classifier-free diffusion guidance*, 2022. arXiv: 2207.12598 [cs.LG].
- [34] S. Jiang, Y. Yan, Y. Lin, X. Yang, and K. Huang, “Sketch to building: Architecture image translation based on gan,” in *Journal of Physics: Conference Series*, IOP Publishing, vol. 2278, 2022, p. 012036.
- [35] J. Li, D. Li, C. Xiong, and S. Hoi, *Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation*, 2022. arXiv: 2201.12086 [cs.CV].
- [36] C. Meng, Y. He, Y. Song, *et al.*, *Sdedit: Guided image synthesis and editing with stochastic differential equations*, 2022. arXiv: 2108.01073 [cs.CV].
- [37] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, *High-resolution image synthesis with latent diffusion models*, 2022. arXiv: 2112.10752 [cs.CV].
- [38] C. Schuhmann, R. Beaumont, R. Vencu, *et al.*, *Laion-5b: An open large-scale dataset for training next generation image-text models*, 2022. arXiv: 2210.08402 [cs.CV].
- [39] T. T. Tran, *Diversify photo database with clip interrogator*, Nov. 2022. [Online]. Available: <https://trungtranthanh.medium.com/diversify-photo-database-with-clip-interrogator-5dd1833be9f5>.
- [40] Q. Wang, D. Kong, F. Lin, and Y. Qi, “Diffsketching: Sketch control image synthesis with diffusion models,” in *33rd British Machine Vision Conference 2022, BMVC 2022, London, UK, November 21-24, 2022*, BMVA Press, 2022, p. 67. [Online]. Available: <https://bmvc2022.mpi-inf.mpg.de/67/>.
- [41] X. Wu, C. Wang, H. Fu, A. Shamir, S.-H. Zhang, and S.-M. Hu, “Deepportraitdrawing: Generating human body images from freehand sketches,” *arXiv preprint arXiv:2205.02070*, 2022.
- [42] X. Xiang, D. Liu, X. Yang, Y. Zhu, X. Shen, and J. P. Allebach, “Adversarial open domain adaptation for sketch-to-photo synthesis,” in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2022, pp. 1434–1444.

- [43] S.-I. Cheng, Y.-J. Chen, W.-C. Chiu, H.-Y. Tseng, and H.-Y. Lee, “Adaptively-realistic image generation from stroke and sketch with diffusion model,” in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2023, pp. 4054–4062.
- [44] B. Li, K. Xue, B. Liu, and Y.-K. Lai, *Bbdm: Image-to-image translation with brownian bridge diffusion models*, 2023. arXiv: 2205.07680 [cs.CV].
- [45] Y. Peng, C. Zhao, H. Xie, T. Fukusato, and K. Miyata, *Difffacesketch: High-fidelity face image synthesis with sketch-guided latent diffusion model*, 2023. arXiv: 2302.06908 [cs.CV].
- [46] A. Voynov, K. Aberman, and D. Cohen-Or, “Sketch-guided text-to-image diffusion models,” in *ACM SIGGRAPH 2023 Conference Proceedings*, ser. SIGGRAPH ’23, , Los Angeles, CA, USA, Association for Computing Machinery, 2023, ISBN: 9798400701597. DOI: 10.1145/3588432.3591560. [Online]. Available: <https://doi.org/10.1145/3588432.3591560>.
- [47] Z. Liu, Y. Wang, S. Vaidya, *et al.*, “Kan: Kolmogorov-arnold networks,” *arXiv preprint arXiv:2404.19756*, 2024.