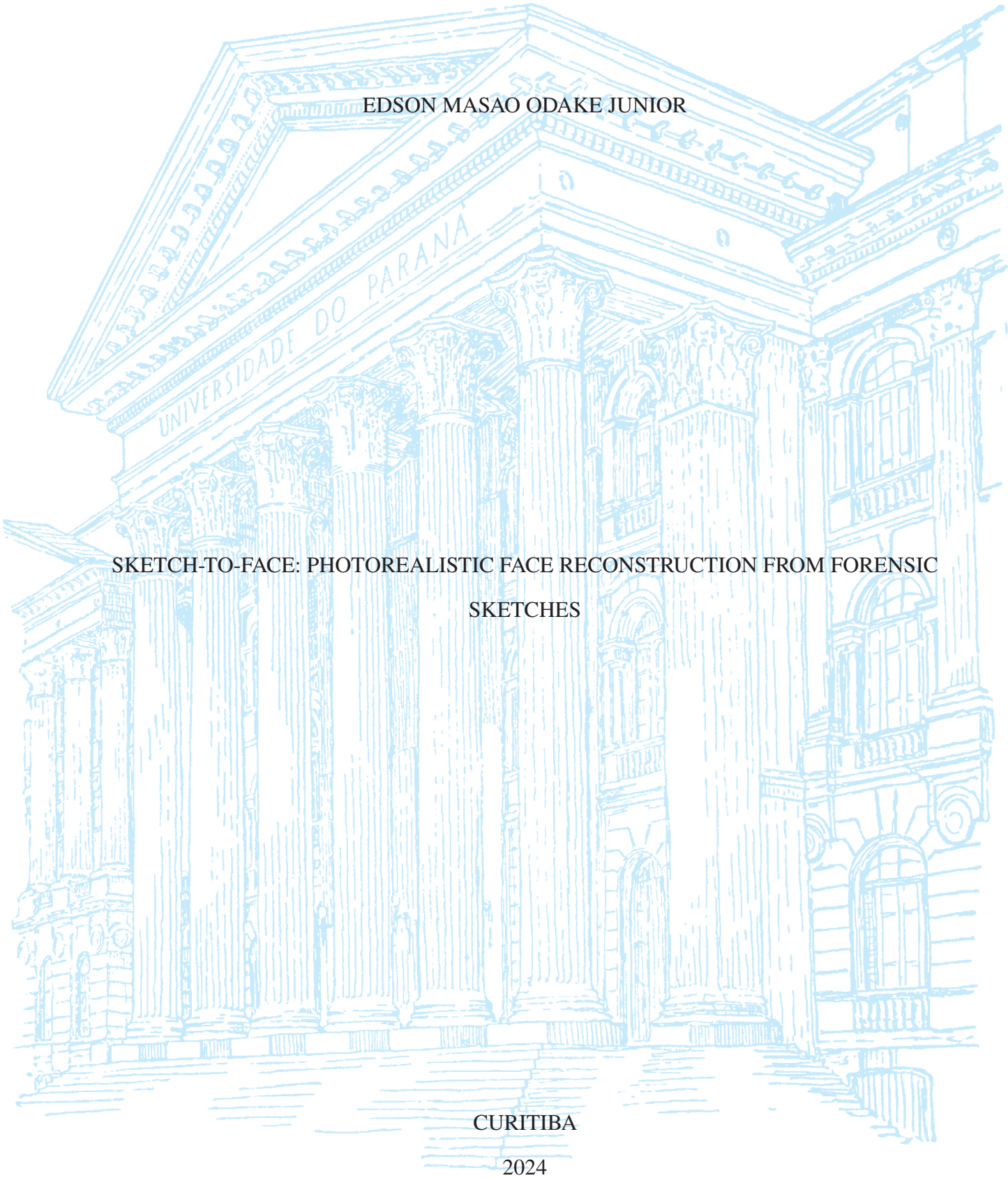
UNIVERSIDADE FEDERAL DO PARANÁ

EDSON MASAO ODAKE JUNIOR



SKETCH-TO-FACE: PHOTOREALISTIC FACE RECONSTRUCTION FROM FORENSIC SKETCHES

CURITIBA

2024

EDSON MASAO ODAKE JUNIOR

SKETCH-TO-FACE: PHOTOREALISTIC FACE RECONSTRUCTION FROM FORENSIC SKETCHES

Dissertação apresentada como requisito parcial à obtenção do grau de Mestre em Engenharia Elétrica no Programa de Pós-Graduação em Engenharia Elétrica, setor de Ciências Exatas, da Universidade Federal do Paraná.

Área de concentração: *Sistemas Eletrônicos*.

Orientador: Prof. Dr. Eduardo Parente Ribeiro.

CURITIBA

2024

DADOS INTERNACIONAIS DE CATALOGAÇÃO NA PUBLICAÇÃO (CIP)
UNIVERSIDADE FEDERAL DO PARANÁ
SISTEMA DE BIBLIOTECAS – BIBLIOTECA DE CIÊNCIA E TECNOLOGIA

Odake Junior, Edson Masao
Sketch-to-face: photorealistic face reconstruction from forensic sketches / Edson Masao Odake Junior. – Curitiba, 2024.
1 recurso on-line : PDF.

Dissertação (Mestrado) - Universidade Federal do Paraná, Setor de Ciências Exatas, Programa de Pós-Graduação em Engenharia Elétrica.

Orientador: Eduardo Parente Ribeiro

1. Criminalística. 2. Processamento Digital de Imagens. I. Universidade Federal do Paraná. II. Programa de Pós-Graduação em Engenharia Elétrica. III. Ribeiro, Eduardo Parente. IV . Título.

Bibliotecário: Leticia Priscila Azevedo de Sousa CRB-9/2029

MINISTÉRIO DA EDUCAÇÃO
SETOR DE TECNOLOGIA
UNIVERSIDADE FEDERAL DO PARANÁ
PRÓ-REITORIA DE PESQUISA E PÓS-GRADUAÇÃO
PROGRAMA DE PÓS-GRADUAÇÃO ENGENHARIA ELÉTRICA - 40001016043P4

TERMO DE APROVAÇÃO

Os membros da Banca Examinadora designada pelo Colegiado do Programa de Pós-Graduação ENGENHARIA ELÉTRICA da Universidade Federal do Paraná foram convocados para realizar a arguição da Dissertação de Mestrado de **EDSON MASAO ODAKE JUNIOR** intitulada: **Sketch-to-Face: Photorealistic Face Reconstruction from Forensic Sketches**, sob orientação do Prof. Dr. EDUARDO PARENTE RIBEIRO, que após terem inquirido o aluno e realizada a avaliação do trabalho, são de parecer pela sua APROVAÇÃO no rito de defesa.

A outorga do título de mestre está sujeita à homologação pelo colegiado, ao atendimento de todas as indicações e correções solicitadas pela banca e ao pleno atendimento das demandas regimentais do Programa de Pós-Graduação.

Curitiba, 06 de Agosto de 2024.

Assinatura Eletrônica
06/08/2024 17:49:56.0
EDUARDO PARENTE RIBEIRO
Presidente da Banca Examinadora

Assinatura Eletrônica
06/08/2024 18:03:29.0
GIDEON VILLAR LEANDRO
Avaliador Interno (UNIVERSIDADE FEDERAL DO PARANÁ)

Assinatura Eletrônica
07/08/2024 09:43:48.0
JULIO CÉSAR NIEVOLA
Avaliador Externo (PONTIFÍCIA UNIVERSIDADE CATÓLICA DO PARANÁ)

RESUMO

Um assalto normalmente ocorre rapidamente, deixando pouca ou nenhuma evidência na cena do crime. Em muitos casos, a única pista para a identidade do agressor é a memória visual da vítima. A reconstrução dessas pistas visuais é importante quando as evidências físicas são escassas. O esboço forense é um método tradicional de transformar a descrição da vítima em uma representação visual do suspeito. No entanto, esses esboços frequentemente carecem do detalhe e realismo necessários para uma identificação pública eficaz. Uma representação mais realista e precisa do suspeito pode aumentar o engajamento do público e facilitar a integração com a tecnologia de reconhecimento facial. Este trabalho apresenta uma nova arquitetura de Autoencoder Variacional Condicional (*CVAE*) para transformar esboços forenses em imagens faciais fotorrealistas. O modelo proposto utiliza um *pipeline* estocástico de pré-processamento para extrair diversos mapas de contorno a partir de imagens, evitando a necessidade de bancos de dados de esboço-fotografias pareados manualmente. Ao incorporar entradas condicionais, o *CVAE* permite um processo de geração interativo baseado em atributos específicos e interpretáveis. O desempenho do modelo é avaliado utilizando múltiplos bancos de dados, incluindo esboços desenhados digitalmente e não digitalmente. Quando avaliado usando *Facenet*, o *CVAE* gera imagens 56,8% mais semelhantes à foto original do suspeito em comparação com abordagens mais simples. Além disso, o modelo foi testado em um cenário onde a imagem gerada foi usada para identificar um suspeito hipotético a partir de um pequeno conjunto de imagens, demonstrando sua aplicabilidade prática na aplicação da lei. Nossos resultados indicam que a estrutura proposta oferece uma solução viável para aplicações forenses e outras tarefas de conversão de imagem para imagem.

Palavras-chave: Autoencoder Variacional Condicional. Síntese Fotorealista de Rostos. Ciência Forense. Tradução de Imagem para Imagem. Processamento Digital de Imagens.

ABSTRACT

A robbery often occurs quickly, leaving little to no evidence at the crime scene. In many cases, the only clue to the assailant's identity is the victim's visual memory. Reconstructing these visual clues is important when physical evidence is scarce. Forensic sketching is a traditional method of turning a victim's description into a visual representation of the suspect. However, these sketches often lack the detail and realism required for effective public identification. A more realistic and accurate depiction of the suspect could enhance public engagement and facilitate integration with facial recognition technology. This work presents a novel Conditional Variational Autoencoder (CVAE) architecture for transforming forensic sketches into photorealistic facial images. The proposed model utilizes a stochastic preprocessing pipeline to extract diverse edge maps from images within a dataset, bypassing the need for manually paired sketch-photo databases and enhancing scalability. By incorporating conditional inputs, the CVAE enables an interactive generation process based on specific interpretable attributes. The model's performance is evaluated using multiple databases, including digitally and non-digitally drawn sketches. When evaluated with a Facenet-based metric, the CVAE generates images that are 56.8% more similar to the suspect's original picture compared to simpler approaches. Furthermore, the model was tested in a scenario where the generated image was used to identify a hypothetical suspect from a small image set, demonstrating its practical applicability in law enforcement. Our results indicate that the proposed framework offers a viable solution for forensic applications and other image-to-image translation tasks.

Keywords: Conditional Variational Autoencoder, Photorealistic Face Synthesis, Forensic Science, Image-to-Image Translation, Digital Image Processing

List of Figures

List of Tables

LIST OF ACRONYMS

AE	Autoencoder
VAE	Variational Autoencoder
CVAE	Conditional Variational Autoencoder
MSE	Mean Squared Error
FID	Frechet Inception Distance
SSIM	Structural Similarity Index Measure
LLM	Large Language Model
GAN	Generative Adversarial Network
NMS	Non-Maximum Suppression
ELBO	Evidence Lower Bound
AI	Artificial Intelligence
KL	Kullback-Leibler
ResNet	Residual Network
ReLU	Rectified Linear Unit
SAPGAN	Sketch and Paint Generative Adversarial Network
PSNR	Peak Signal-to-Noise Ratio

Contents

1 INTRODUCTION

The ability to develop an idea based on an initial insight is a recurrent skill in any creative field. Traditionally, this has been considered a uniquely human capability, as machines have not been able to replicate this process. However, this notion has changed in recent years with the advent of AI tools that assist in various creative processes. Engineers, for instance, often use Large Language Models (LLMs) to provide technical solutions for diverse problems more quickly, thereby improving productivity and quality. Beyond technical applications, these AI tools are frequently used in more subjective fields, such as marketing strategies, content writing, and entertainment. While AI tools serve to mold and refine ideas, the user must still input the initial concept, present essential information and analysis, and verify the output to ensure it aligns with reality.

Artists often create analog or digital images to showcase ideas or for entertainment purposes. AI tools can help generate images from text descriptions or reference images. These tools can be utilized to create logos for company brands, social media content, construction architecture designs, and historical reconstructions. More technical applications include data augmentation for training other AI models, such as generating medical pathology images (Kebaili et al., 2023) or training autonomous vehicles (Cunneen et al., 2019).

In the specific area of face generation, a model could be developed to reconstruct a target facial photo based on a sketch. Since sketches naturally dismiss color information, additional inputs could be provided to the model to incorporate this data and generate a realistic facial image (Zhao et al., 2019). Such a model could be especially useful in forensic sketching and suspect identification. Forensic artists create facial sketches based on witness descriptions. These sketches, along with additional information, could be fed into the model to produce a more realistic image. Realistic images are more likely to capture public attention and generate valuable tips for identifying suspects. These images could also be integrated with digital technologies for recognition and identification of a suspect against a database (Devakumar and Sarath, 2023).

The generation of images based on sketches has a wide array of applications aside from forensic investigations. In initial stages of product development, conceptual sketches can be transformed into realistic and manufacturable 2D images and 3D prototypes. This process facilitates rapid and clearer design iteration, allowing designers to explore a broader range of concepts faster (Edwards et al., 2024).

Generative models can accelerate the architectural design process by transforming simple sketches into detailed floorplans and 3D models. Enhancing the ideation phase and reducing the time required to develop creative architectural solutions (Li et al., 2024). Integrating sketch-to-image generators in architectural education offers innovative methods for teaching architectural history. AI tools like Leonardo AI can improve student's engagement and comprehension. This

approach provides new perspectives and interactive tools for learning, helping students to better visualize and understand architectural concepts and styles from different historical periods (Fareed et al., 2024). With frameworks like Sketch-And-Paint GAN (SAPGAN), artists can transform simple sketches into colored Chinese landscape paintings. This method enables artists to explore more creative possibilities, reduce production time, and open up new paradigms for interactive entertainment and digital media (Xue, 2021).

Despite advances in generative algorithms, the methods currently used to create faithful facial representations from sketches usually depend on limited databases that pair sketches with original photos (Nikkath Bushra and Uma Maheswari, 2021; Sun et al., 2022), a practice that reduces scalability and restricts style diversity. Furthermore, existing models trained on unpaired images are primarily evaluated on artificial or digital sketches (Li et al., 2020; Chen et al., 2020). This evaluation fails to account for non-digital, hand-drawn sketches, which limits the accuracy of the generated images across more traditional forms of sketching. Another gap in the literature is the absence of specific metrics to assess the fidelity of the generated facial features to their original counterparts.

Unpaired facial generation techniques propose extracting edge maps to convert both realistic sketches and photorealistic images to the same domain, thus eliminating the need for pairing (Liu et al., 2020). However, these models often disregard evaluations using real sketches. Some studies evaluate digitally drawn sketches with simple details but fail to assess non-digital sketches (Li et al., 2020). Most previous works also don't provide quantitative evaluations, which are necessary to analyze the quality and similarity between the generated image and the target image.

The characteristics of sketches can vary depending on the artist or the software used (in the case of digital sketches). Therefore, extensive evaluation is important to ensure the model's robustness and ability to interpret diverse, unseen sketches.

1.1 OBJECTIVES

This work presents a generative model that transforms facial sketches into photorealistic images. Unlike traditional approaches, our model employs a stochastic edge map extraction method, enhancing generalizability by accommodating a wider range of styles. We introduce a novel architecture that combines a Conditional Variational Autoencoder (CVAE) with skip connections (Ronneberger et al., 2015) and attention mechanisms (He et al., 2016) to improve image quality. Utilizing a CVAE, the generated images capture essential conditional information such as skin and hair color, compensating for details not provided by the sketch.

Our model is trained on the CelebA (Liu et al., 2015) database and validated against a variety of previously unseen hand-drawn sketches, ensuring robust performance across different artistic styles. We evaluate the generated images using several perceptual metrics to verify the

preservation of facial features. This approach broadens the practical applicability in fields such as forensic science and art, where handling different sketch styles is required.

We aim to develop a model capable of generalizing the necessary features to reconstruct a specific face without relying on paired images. The model should have a disentangled latent space, allowing additional information, such as hair color or facial expression, to modify only the specific aspects of the image. By generalizing with unpaired data, the model disregards the need to create an extensive sketch database. We also assess the model's accuracy in recognition and identification tasks and analyze the performance of different metrics on the generated images. The practical application of this model is to use these generated images to identify suspects in an image database or to present them to the public for identification, thereby improving forensic investigations.

In summary, the goals of this work are:

- Develop a generative model that transforms facial sketches into photorealistic images.
- Train the model without the need for paired sketch-photo data.
- Improve generalizability across different sketch styles with a stochastic edge map extraction method.
- Provide simple parameters to adjust the generated image.
- Evaluate the model's accuracy in recognition tasks and its use in forensic investigations.

The dissertation is structured as follows. In section 2, we review the related work in this field. section 3 describes the technologies and methods used to develop this application. section 4 provides an overview of the training process and the evaluation of different models on a diverse set of unseen data. Finally, section 5 concludes the work with insights and sets the path for future improvements.

2 RELATED WORK

The synthesis of photos guided by sketches is essentially an image-to-image translation problem, which seeks to optimize the mapping between two domains: the domain of sketches and the domain of photorealistic images. This task involves complex challenges, such as capturing the intended details of sketches while dismissing bad strokes to accurately reproduce a realistic appearance of faces. The most commonly used models for this task are Variational Autoencoders (VAEs) and Generative Adversarial Networks (GANs) (Pang et al., 2022).

VAEs are designed to learn a probabilistic representation of the input data, allowing them to generate new samples by sampling from the learned latent space (Odaibo, 2019). GANs, on the other hand, consist of two neural networks: a generator that creates images from sketches and a discriminator that evaluates the authenticity of the generated images. The adversarial training process helps GANs produce high-quality, realistic images. Both VAEs and GANs have been widely adopted due to their ability to learn complex data distributions and generate high-fidelity images.

However, each of these models comes with its own set of drawbacks. VAEs often struggle with producing sharp images, as they tend to generate blurrier outputs due to their loss function. GANs, while capable of producing sharper and more detailed images, can be difficult to train due to issues like mode collapse and instability during the adversarial training process. In forensic applications, the blurrier output of VAEs is not necessarily a problem. The primary goal is to retain essential features for suspect identification, rather than producing the sharpest or most appealing image.

Also, the field has seen the integration of conditional inputs to guide and improve the generation process. By incorporating additional information such as facial attributes or color details, these models can produce more accurate and contextually relevant images. This has led to the development of Conditional VAEs (CVAEs) and Conditional GANs (CGANs) (Deshpande et al., 2017), which offer enhanced control over the generated outputs by conditioning the generation process on supplementary inputs.

Nastaran Moradzadeh Farid (Moradzadeh Farid et al., 2023) developed a sketch-to-image GAN that achieved impressive results in generating photorealistic images from sketches. However, their evaluation was limited to a single-style database, which raises concerns about the model's generalizability across different sketch styles and datasets. This limitation suggests that the model might not perform as well when applied to more diverse and complex sketches encountered in more practical scenarios.

Yongyi Lu (Lu et al., 2018) proposed a Contextual GAN that utilized sketches as weak contextual constraints to guide the image generation process. This method demonstrated robustness when applied to less detailed sketches, producing very impressive images. However, it

did not ensure the preservation of the subject's identity, which is necessary for applications such as forensic sketch reconstruction. The lack of identity preservation limits the practical utility of this approach in scenarios where an accurate depiction of individuals is necessary.

Phillip Isola (Isola et al., 2017) introduced a Conditional GAN with a U-Net-based generator designed to translate various styles of images. While their approach showed versatility in handling different image translation tasks, their focus was on more generic image-to-image translation rather than the specific area of sketch-to-photo synthesis. Consequently, while their model is powerful, its application to sketch-guided face synthesis may not fully exploit its potential without further adaptation and specialized training.

Several GAN-based models have specifically targeted the conversion of sketches to facial photos (Wang et al., 2018; Hu and Guo, 2020; Li et al., 2022). However, their evaluations were confined to the same databases used for training, which limited the assessment of their performance on unseen data. This restriction raises concerns about the generalizability and robustness of these models in real-world applications, where the diversity of sketch styles and quality can vary greatly.

Jun Yu (Yu et al., 2021) attempted to address this limitation by extending the evaluation to a variety of sketch styles outside the training database. While this approach produced results that were blurry but still interesting, it demonstrated the difficulty of maintaining image clarity and detail when the model encounters unfamiliar sketch styles. The requirement for paired images for training remained a big limitation, constraining the scalability and practical deployment of these models.

Mingming Hu (Hu and Guo, 2020) employed the xDoG filter (Winnemöller et al., 2012) to generate sketches from facial photos for training inputs. This method achieved good results but did not evaluate the model on real sketches, thereby missing an important aspect of practical application. The use of synthetic sketches can lead to models that perform well under controlled conditions but fail to generalize to hand-drawn or real-world sketches.

Other researchers (Osahor and Nasrabadi, 2022; Xia et al., 2021) have used the powerful pre-trained StyleGAN model, creating a latent space mapping from sketches and text descriptions to generate appealing results. Despite the visual appeal of these generated images, their evaluations were again limited to the training database, which raises questions about the model's adaptability to diverse and unstructured real-world data.

The varying styles between training databases and sketches produced by different artists present a serious issue for generating high-quality images from sketches. This variability requires models to be robust and flexible enough to handle diverse input styles while maintaining the fidelity of the generated output.

Yuhang Li (Li et al., 2020) achieved impressive results with less detailed digital sketches by synthesizing sketches from original photos using various methods. This approach helped in capturing the essential features of the face, but it focused on less detailed, digital sketches, which might not fully represent the complexity of forensic hand-drawn sketches.

Shu-Yu Chen (Chen et al., 2020) developed an architecture that focused on generating separate facial components (eyes, mouth, nose) by applying windows to encode and decode these features. This method achieved impressive results on free-hand digital sketches breaking down the complex task of face generation into smaller, more manageable parts. However, the evaluation of this approach was restricted to digitally drawn sketches. This limitation means that the model's performance on more variable and noisier hand-drawn sketches remains untested.

The reliance on digitally drawn sketches for evaluation can lead to models that perform well under controlled conditions but may struggle with the unpredictability and diversity of forensic sketches. Therefore, it is important to extend evaluations to include a wider variety of sketch styles.

Xing Di (Di and Patel, 2019) employed a complex three-stage training architecture that combined Conditional Variational Autoencoders (CVAE) and GANs in a three-stage training process to generate facial photos from sketches. This hybrid approach used the strengths of both CVAEs and GANs to produce notable results, capturing detailed features and generating realistic images from sketches. The CVAE component helped in learning a probabilistic representation of the data, while the GAN component enhanced the sharpness and realism of the generated images.

However, GAN-based models are difficult to train. One of the primary issues is the convergence problem, where the generator and discriminator networks must be carefully balanced to ensure stable training. If one network outpaces the other, it can lead to poor performance and instability. Additionally, GANs are prone to mode collapse, a phenomenon where the generator produces a limited variety of outputs, failing to capture the full diversity of the input data. Also, their architecture seems to add too much complexity to the training process. Our model offers a simpler and more stable training process, with a preprocessing pipeline that improves generalizability across different sketch styles and eliminates the need for manually pairing sketch-photo images during training. While GANs are often known for generating visually sharp images, our model prioritizes preserving the essential features of the original sketch, ensuring that the generated image can be used for suspect identification.

3 METHODOLOGY

In this section, we present an in-depth explanation of the steps and decisions made to develop this generative model. In section 3.1 we address the preprocessing techniques used to guarantee diverse yet well-behaved input to fit the model. section 3.2 - 3.9 contemplates the justification of each component included in the proposed architecture. In section 3.10 we discuss the cost function used to train the model and its advantages and disadvantages. section 3.11 provides the training hyper parameters used to train each model. In section 3.12 we present the metrics used to quantify the quality of the generated images and the similarity with the original face. In section 3.13 we describe the database used for both training and evaluating the model.

3.1 PREPROCESSING PIPELINE

This pipeline was developed to ensure that every style of image is mapped into the same domain, maintaining the essential features required to generate a photorealistic version of the input. By mapping the input into the same domain, it becomes possible to utilize different styles to generate the final image. This process involves standardizing input types, such as hand-drawn sketches, digital sketches, and photographs, into an edge map representation. The core advantage of this approach lies in its ability to handle diverse input styles without necessitating specific adjustments for each type. In essence, this preprocessing pipeline is designed to standardize the diverse input images into a unified domain, thereby enabling the generative model to interpret and process them.

A different aspect of this pipeline is the stochastic variation in preprocessing parameters. By randomly adjusting the threshold values for edge detection and the kernel size for Gaussian filtering, the pipeline introduces variability that mimics different sketching styles and conditions. This stochasticity ensures that the model learns to generalize across a wide range of edge map characteristics, making it more robust to variations in input sketches during practical application.

The preprocessing pipeline, illustrated in Figure 3.1, starts with training images sized at 178×218. For new images, input sketches are cropped to this proportion and reshaped. Edge detection, important for unpaired training, maps images to the "edge space", ensuring consistency between training photos and evaluation sketches.

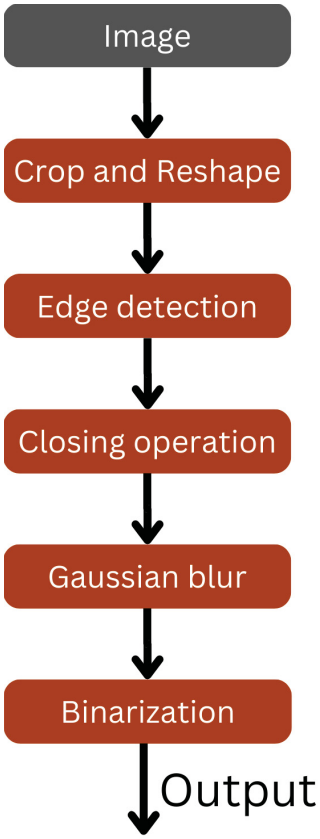


Figure 3.1: Processing pipeline.

The edge is extracted from the original images using the OpenCV Canny detector (Canny, 1986). This algorithm was chosen for its simplicity, ease of parameter tuning, and computational efficiency while providing accurate results. The algorithm steps are as follows:

- **Filter**: A Gaussian filter with a 5x5 kernel is applied to the image to remove undesired high-intensity gradients in parts not associated with edges. This step smooths the image, reducing noise and minor details that could interfere with edge detection.
- **Gradient intensity**: Sobel filters are applied in both x and y directions to compute the gradient intensity and direction at each pixel. The final gradient magnitude G and direction θ are calculated as follows.

$$\begin{aligned} G &= \sqrt{G_x^2 + G_y^2} \\ \theta &= arctan\left(\frac{G_x}{G_y}\right) \end{aligned} \tag{3.1}$$

- **Non-maximum suppression (NMS)**: This step aims to thin out the edges by preserving all local maxima in the gradient image while discarding all other gradient values. High-intensity gradients indicate edges. After obtaining the gradient and direction, NMS suppresses pixels with smaller gradients than their neighbors along the θ direction.

- **Hysteresis**: After NMS, the resulting image consists of thin lines representing potential edges. A double threshold hysteresis is applied to differentiate strong edges from weak ones. This step uses two thresholds: pixels with gradient intensity above the high threshold are accepted as edges, those below the low threshold are rejected, and those between the thresholds are accepted only if they are connected to strong edges. This process ensures continuity and reduces fragmentation in the edge map.

The closing operation (Sreedhar, 2012) is important for enhancing edge quality by closing gaps and reinforcing continuity through sequential dilation and erosion. Figure 3.2 illustrates examples of edges extracted from a digital sketch. Sketches with thick traces often display different features when edges are extracted, often resulting in a double-line structure. This occurs because the Sobel filter detects high-intensity slopes at both ends of a line, while the middle section does not activate the filter. In contrast, edges extracted from photos are often discontinuous, contain several gaps, and do not exhibit this double-line characteristic.

Dilation (Sreedhar, 2012) is a morphological operation that extends the image while maintaining its shape, often used to close gaps or connect separate parts of an image. However, it also increases edge thickness. Conversely, erosion (Sreedhar, 2012) shrinks the image while preserving its shape.

The closing operation involves first applying dilation to extend the edges and close gaps, followed by erosion to refine the edges and reduce any excessive thickness. This sequence removes double-line edges and fills in gaps, resulting in a cleaner and more continuous edge map.

- **Dilation**:
  - Expands the boundaries of foreground pixels.
  - Uses a predefined kernel to set each pixel to the maximum value in its neighborhood.
  - Closes small gaps and thickens edges.
  - **Equation**:
$$A \oplus B = \{z \in E \mid (B_z \cap A) \neq \emptyset\}$$
- **Erosion**:
  - Shrinks the boundaries of foreground pixels.
  - Uses a predefined kernel to set each pixel to the minimum value in its neighborhood.
  - Removes small objects and thins edges.
  - **Equation**:
$$A \ominus B = \{z \in E \mid B_z \subseteq A\}$$
- **Closing Operation**:
  - Combines dilation and erosion.

- Closes small holes and gaps while maintaining object shape.
- Maintains the original size of the image.
- **Sequence**:
  1. Apply dilation.
  2. Apply erosion.
- **Equation**:

$$A \bullet B = (A \oplus B) \ominus B$$

where:

- A: The input image.
- B: The structuring element (or kernel) used in the operation.
- $\oplus$: Dilation operation.
- $\ominus$: Erosion operation.
- B_z: The structuring element B translated by z (similar to a convolution operation).
- E: The Euclidean space (the set of all pixel coordinates).
- $\emptyset$: The empty set.
- $\cap$: The intersection operation, representing the common elements between two sets.
- $\subseteq$: The subset operation, indicating that one set is contained within another.

A stochastic Gaussian blur is then applied to smooth edges, eliminate unwanted details, and introduce variability in edge width. The Gaussian blur is implemented using a Gaussian filter with stochastic kernel size, which smooths the image by averaging the pixels based on a Gaussian distribution.

The Gaussian blur is described by the following equation:

$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \tag{3.2}$$

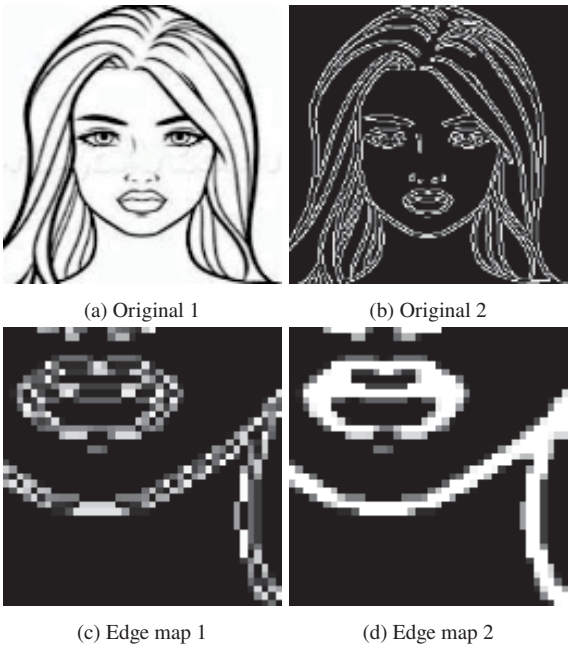
where σ is the standard deviation of the Gaussian distribution.

Since analog sketches, digital sketches, and the training database differ significantly for instance, digital sketches often have strong edge definitions, whereas analog sketches may include shadows and diverse stroke types, the edge map can vary greatly in pixel intensity. To address this, a binarization procedure is employed to standardize the intensity. This process converts the edge map into a binary representation of zeros and ones, defining whether a pixel is an edge or not, and disregarding intensity variations. This standardization ensures consistency in edge detection across different sketch styles. Of course, it also causes some information loss.

The binarization is performed using a thresholding method, where each pixel intensity $I(x, y)$ is compared to a predefined threshold T:

$$B(x, y) = \begin{cases} 1 & \text{if } I(x, y) \geq T \\ 0 & \text{if } I(x, y) < T \end{cases} \tag{3.3}$$

where $B(x, y)$ is the binarized edge map, and T is the defined with value 0.5.



(a) Original 1 (b) Original 2

(c) Edge map 1 (d) Edge map 2

Figure 3.2: Double-edge

The parameters in this preprocessing pipeline are selected stochastically and vary between images, providing diverse edge map extraction. Figure 3.3 demonstrates the full pipeline applied to the same image with different parameter settings. The images are resized to 64x64 to fit the model input shape, showcasing the diversity of edge thickness and sensitivity of edge detection.

(a) Original sketch (b) Variation 1 (c) Variation 2

(d) Variation 3 (e) Variation 4 (f) Variation 5

(g) Variation 6 (h) Variation 7 (i) Variation 8

Figure 3.3: Edge map stochastic variation

The model also receives conditional variables as input, though not all have a significant impact on the final generated image. To facilitate interpretability and improve user interaction, only a few selected conditions are utilized. These specific conditional inputs are detailed in Section 3.5.

3.2 AUTOENCODER

A straightforward approach to map a set of data to the latent space and then back to the original domain is by using autoencoders. An autoencoder is a specific neural network architecture designed to compress data into a lower-dimensional form and subsequently reconstruct the original data from this compressed representation (Tschannen et al., 2018). The decoder could also be used to map the latent representation into other domains (sketch to photorealistic images is one example).

The process begins with a high-dimensional input, which is passed through successive layers that progressively reduce the number of neurons, compressing the information. This phase is known as the encoding process. During encoding, the network captures the essential features of the input data, while discarding redundant information. The result of this compression phase is a bottleneck layer, which contains the latent representation of the input data in a much lower dimension.

Once the data is encoded into the latent space, the next phase is the decoding process. Here, the bottleneck layer is upsampled by gradually increasing the number of neurons in each subsequent layer. This upsampling continues until the data is restored to its original dimensions. The goal of the decoding process is to reconstruct the input data as accurately as possible from the compressed latent representation.

The architecture of an autoencoder is often symmetrical, with the encoder and decoder usually being mirror images of each other. The encoding and decoding enable the autoencoder to learn a meaningful compressed representation of the input data, which can then be used for various applications, such as dimensionality reduction, denoising, and feature extraction.

A simplified version of the autoencoder and its variations are illustrated in Figure 3.4, showing both the compression (encoding) and reconstruction (decoding) phases.

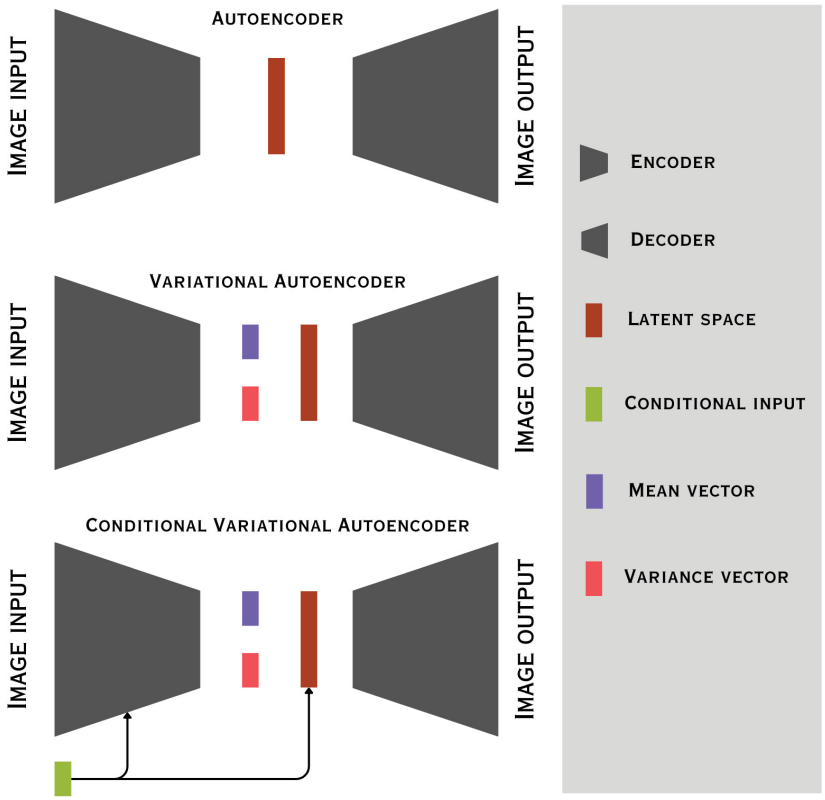


Figure 3.4: Autoencoder based architechtures.

To ensure the reconstructed image closely resembles the original input, a reconstruction loss is applied to the network. This loss function compares the output image with the original one and backpropagates the error to update the network parameters, thereby improving the reconstruction quality. A simple loss function used in this context is the Mean Squared Error (MSE), which performs a pixel-by-pixel comparison between the input and the output.

Autoencoders are effective for data compression, especially for image applications, where convolutional layers are often used to extract relevant features. This architecture is not limited to simple compression, it can also be applied to tasks like image denoising. In denoising

applications, the input image is corrupted with noise, and the network is trained to reconstruct the original, noise-free image (Vincent et al., 2008). Similarly, the same concept can be extended to tasks such as watermark removal, where the network learns to remove unwanted marks from the images while preserving the essential content.

While autoencoders are excellent tools for compressing and decompressing images, they are not well-suited for generating new samples. This limitation arises because, although the model can map known data to the latent space, the resulting latent space representation is often asymmetrical, and does not adequately separate different types of data, leading to gaps within the space. Since the sole objective of an autoencoder is reconstruction, there is no regularization applied to the latent space, resulting in a disorganized and messy latent space. Consequently, predicting the output image from a specific point in the latent space becomes challenging due to the lack of continuity and clear correlation, as illustrated in Figure 3.5.

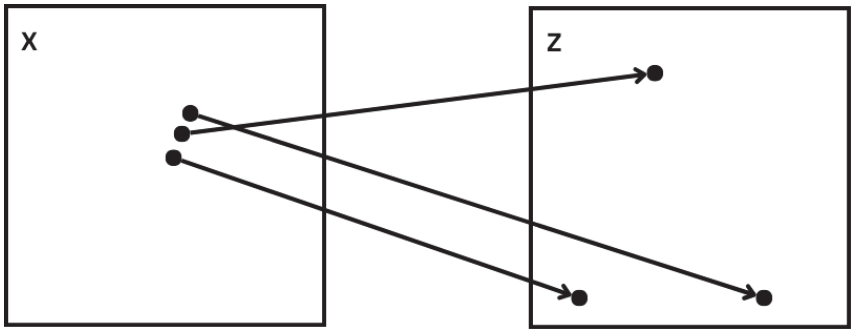


Figure 3.5: Non-continuous latent space illustration.

Mathematically, an autoencoder consists of two main components: the encoder E and the decoder D. The encoder maps the input x to a latent space representation z, and the decoder reconstructs the input from this latent space representation:

$$z = E(x) \tag{3.4}$$

$$\hat{x} = D(z) = D(E(x)) \tag{3.5}$$

The loss function measures the difference between the input x and the reconstructed output $\hat{x}$, the difference could be measured by several metrics. Here is an example using Squared Error:

$$L(x, \hat{x}) = \|x - \hat{x}\|^2 \tag{3.6}$$

3.3 VARIATIONAL AUTOENCODER

Variational autoencoders (VAEs) offer a solution to better understand and control the latent space of the model, facilitating the modification of images and the generation of new samples. The

primary objective of VAEs is to create a smooth, continuous latent space from which the decoder can generate coherent images. Unlike traditional autoencoders, VAEs incorporate regularization mechanisms that structure the latent space in a meaningful way, ensuring that small changes in the latent space correspond to smooth variations in the generated images. This allows for better interpolation between data points and the generation of novel samples that are not present in the training data.

The encoding and decoding components of VAEs are similar to those of traditional autoencoders. However, the main difference lies in the bottleneck portion of the model. Instead of compressing the data into a single layer and then decompressing it, VAEs model the latent space as a probability distribution. At the end of the encoding process, two vectors are produced: the mean vector and the variance vector. These vectors represent the parameters of the probability distribution of the latent space. Both the mean and variance vectors have dimensions equal to the size of the latent space. This probabilistic representation allows for the generation of new samples by sampling from the distribution defined by these vectors.

A significant difference between the autoencoder and the VAE is that the autoencoder is a deterministic model. Given an input array x, the output y is always the same. In contrast, the VAE introduces a sampling step after the encoding process, adding stochastic behavior to the model. This ensures that instead of mapping the image to a single point in the latent space, the VAE maps it to an area or volume. Depending on the sample, the data is projected to different points within a region near the mean. It also provides a more continuous space, with similar features mapped more closely. This concept is illustrated in Figure 3.6, where the VAE mapping demonstrates the spread of data points in the latent space.

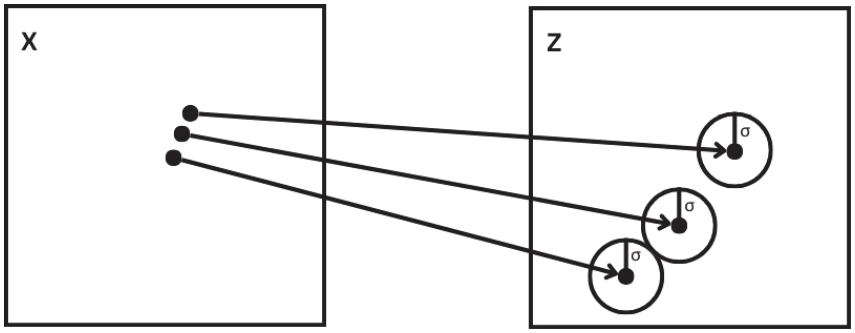


Figure 3.6: Continuous latent space illustration.

The initial idea would be to use the mean and variance vectors to sample the latent space directly and then use the decoder to reconstruct the initial image, but since we are using the backpropagation algorithm to train the network, it wouldn't be possible to do gradient descent on the sample, so we apply the reparameterization trick to solve this problem. It's included an ε standard distribution, with mean zero and variance one instead of sampling directly from the mean, and covariance vectors. The latent space is created by the following equation $z = \mu + \sigma^2\varepsilon$. This process adds a stochastic approach to the autoencoder. The process is illustrated in figure 3.7.

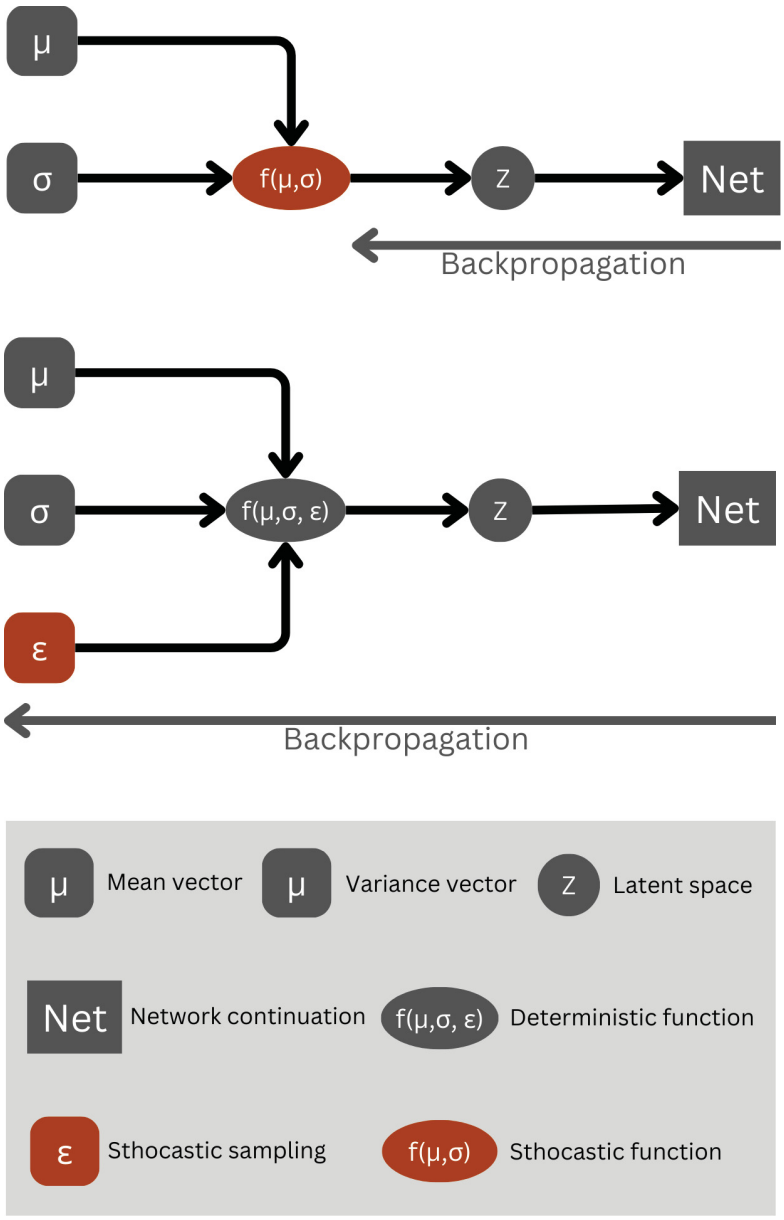


Figure 3.7: Reparameterization trick.

The objective now is to map the data to specific continuous regions of the latent space, ensuring that the features are well-defined within their respective regions. This approach facilitates navigation through the latent space to create new samples. The decoding part receives this new stochastic latent space z and proceeds with the upsampling process, similar to the standard autoencoder.

The VAE employs two key components in its loss function: the reconstruction loss and the KL divergence.

The reconstruction loss is analogous to that of a standard autoencoder. It compares the generated image with the original one, aiming to make them as similar as possible. Since the VAE uses a stochastic approach, mapping the data to a volume rather than a single point, this loss

ensures that the volume is sufficiently distinct so the decoder can accurately differentiate between various samples.

The second component, the KL divergence, is a metric that measures the similarity between two probability distributions. This term acts as a regularizer, encouraging the encoder to shape the encoded data into a Gaussian distribution. By doing so, it prevents the model from creating an overly complex latent space that doesn't generalize well to new data. The KL divergence ensures that the latent space maintains a meaningful representation of the data, avoiding overfitting.

Combining these two loss functions results in the Evidence Lower Bound (ELBO). The reconstruction loss focuses on the accuracy of the output image compared to the input, while the KL divergence regularizes the latent space distribution, ensuring it aligns with a Gaussian distribution. Together, these components balance the trade-off between accurately reconstructing the input and maintaining a well-structured, generalizable latent space.

Let's derive the VAE loss function. Information can be defined as a measure inversely proportional to probability, meaning that events with a high probability of occurring carry low information. This relationship is expressed mathematically as follows:

$$I_p(x) = -\log(p(x)) \tag{3.7}$$

Here, $I_p(x)$ represents the information content of event x, and $p(x)$ is the probability of the event occurring.

Given a sample x from the random variable X, $P(x)$ represents the probability of a certain event occurring. The difference in information between $p(x)$ and another distribution $q(x)$ can be calculated as follows:

$$\Delta I = I_p - I_q = -\log(p(x)) - (-\log(q(x))) \tag{3.8}$$

This can be simplified to:

$$\Delta I = \log\left(\frac{p(x)}{q(x)}\right) \tag{3.9}$$

This equation shows the change in information when moving from distribution $q(x)$ to distribution $p(x)$.

The Kullback-Leibler (KL) divergence is defined as the expected value ($\mathbb{E}$) of the difference in information between two probability distributions. The KL divergence is formulated as follows:

$$D_{KL}(q(x) \parallel p(x)) = \mathbb{E}_q[\Delta I] \tag{3.10}$$

This can be expanded to:

$$D_{KL}(q(x) \parallel p(x)) = \int (\Delta I)\, q(x)\, dx \tag{3.11}$$

Substituting the expression for ΔI, we get:

$$D_{KL}(q(x) \parallel p(x)) = \int q(x)\, \log\left(\frac{q(x)}{p(x)}\right) dx \tag{3.12}$$

This can also be expressed as:

$$D_{KL}(q(x) \parallel p(x)) = \int \left[\log(q(x))\, q(x) - \log(p(x))\, q(x)\right]\, dx \tag{3.13}$$

The term $\int q(x)\, \log\left(\frac{q(x)}{p(x)}\right) dx$ shows that the difference in information ΔI is weighted by $q(x)$. If we switch the calculation from $D_{KL}(q(x) \parallel p(x))$ to $D_{KL}(p(x) \parallel q(x))$, the weighting would be done by $p(x)$, and ΔI would be inverted. This asymmetry is why the KL divergence is referred to as a divergence rather than a distance.

An important property to prove is that the KL divergence is always non-negative, i.e., it is equal to or greater than zero. To demonstrate this, consider the following inequality:

$$\log(t) \leq t - 1 \tag{3.14}$$

Using this inequality, we can derive the KL divergence as follows:

$$D_{KL}(q(x) \parallel p(x)) = \int q(x)\, \log\left(\frac{q(x)}{p(x)}\right) dx \tag{3.15}$$

Substituting $\log(t) \leq t - 1$ into the equation, we get:

$$\int q(x)\, \log\left(\frac{q(x)}{p(x)}\right) dx \leq \int q(x)\, \left(\frac{q(x)}{p(x)} - 1\right) dx \tag{3.16}$$

Since:

$$\begin{aligned} \int q(x)\, \left(\frac{q(x)}{p(x)} - 1\right) dx &= \int q(x)\, \frac{q(x)}{p(x)}\, dx - \int q(x)\, dx \\ &= \int q(x)\, dx - \int q(x)\, dx = 0 \end{aligned} \tag{3.17}$$

Therefore, we have:

$$D_{KL}(q(x) \parallel p(x)) = \int q(x)\, \log\left(\frac{q(x)}{p(x)}\right) dx \geq 0 \tag{3.18}$$

This shows that the KL divergence is always non-negative, as required.

To proceed with the derivation of the loss function, it's essential to understand Bayes Theorem, which is given by the following equation:

$$P(B \cap A) = P(A \cap B) \implies P(A|B)P(B) = P(B|A)P(A) \tag{3.19}$$

From this, we can derive:

$$p(A|B) = \frac{p(B|A)p(A)}{p(B)} \tag{3.20}$$

Given that $q_\theta(z|x_i)$ is the encoder probability function given a sample x_i and $p_\phi(x|z_i)$ is the decoder, while $p(z|x_i)$ is the real distribution, we start the formulation by calculating the KL divergence from $q_\theta(z|x_i)$ to $p(z|x_i)$:

$$D_{KL}(q_\theta(z|x_i)||p(z|x_i)) = -\int q_\theta(z|x_i) \log\left(\frac{p(z|x_i)}{q_\theta(z|x_i)}\right) dz \tag{3.21}$$

Since $p(z|x_i)$ is intractable, we apply Bayes' theorem and use $p_\phi(x_i|z)$ instead:

$$\begin{aligned} D_{KL}(q_\theta(z|x_i)||p(z|x_i)) = \\ -\int q_\theta(z|x_i) \log\left(\frac{p_\phi(x_i|z)p(z)}{q_\theta(z|x_i)p(x)}\right) dz \end{aligned} \tag{3.22}$$

So,

$$\begin{aligned} 0 \leq D_{KL}(q_\theta(z|x_i)||p(z|x_i)) = \\ -\int q_\theta(z|x_i) \log\left(\frac{p_\phi(x_i|z)p(z)}{q_\theta(z|x_i)}\right) dz \\ +\int q_\theta(z|x_i) \log(p(x)) dz \end{aligned} \tag{3.23}$$

Integrating the last component, we have:

$$\log(p(x)) \geq \int q_\theta(z|x_i) \log\left(\frac{p_\phi(x_i|z)p(z)}{q_\theta(z|x_i)}\right) dz \tag{3.24}$$

$$\begin{aligned} \log(p(x)) \geq \int q_\theta(z|x_i) \log\left(\frac{p(z)}{q_\theta(z|x_i)}\right) dz \\ +\int q_\theta(z|x_i) \log p_\phi(x_i|z) dz \end{aligned} \tag{3.25}$$

The ELBO, represented by $\log(p(x))$, is the function we want to maximize. The ELBO's first component is the regularization term, which measures the distance between the encoder's mapping to the latent space and the real distribution mapping. The second component is the reconstruction term, which indicates the probability of reconstructing the desired x_i.

$$\begin{aligned} \log(p(x)) \geq & -D_{KL}(q_\theta(z|x_i)||p(z)) \\ & +\mathbb{E}_{\sim q_{z|x_i}}[\log(p_\phi(x_i|z))] \end{aligned} \tag{3.26}$$

To derive a more applicable equation, we substitute the distributions in the regularization term with Gaussian distributions. The $p(z)$ is set with $\mu = 0$ and $\sigma = 1$. This substitution leads to the loss function defined as $\mathcal{L}$:

$$\begin{aligned} \mathcal{L} = & -\frac{1}{2}\sum_i \left(1 + \log(\sigma_i^2) - \sigma_i^2 - \mu_i^2\right) \\ & - \sum_l \mathbb{E}_{z\sim q_\theta(z|x_i)}\left[\log p(x_i|z^{(i,l)})\right] \end{aligned} \tag{3.27}$$

The VAE offers the ability to navigate through the latent space, generating data with smooth transitions between adjacent points. Different nodes in the latent space capture various input features, allowing for modifications to edit the generated image. However, the features in the latent space remain abstract, making it non-intuitive to discern what each node represents. This abstraction complicates the generation of images with specific attributes. For example, to generate the face of a man, one would need to randomly sample from the latent space until reaching a region corresponding to gender-specific attributes.

The next section introduces a framework to better understand and manipulate the latent space, making generating images with desired features more practical.

3.4 CONDITIONAL VARIATIONAL AUTOENCODER

The idea behind CVAE is to reduce the abstraction of the VAE's latent space (z) and provide more control over the generated outputs. In the context of face generation models, a standard VAE receives an image as input and attempts to reconstruct a similar image. During the generation of new samples, the decoder component of the VAE is used to produce these samples by randomly sampling from the latent space z.

CVAE enhances this process by incorporating conditional information into both the encoder and decoder inputs. Instead of using just the image, CVAE takes the image along with additional attributes, such as hair color, age, or any other facial detail, as input. In the encoder, the image and its corresponding attributes are concatenated and passed through the network to produce the mean and variance vectors of the latent space. During decoding, these attributes are concatenated with the latent vector z to guide the generation of new images.

This approach allows for more precise control over the generated images, enabling the model to generate faces with specific attributes rather than relying solely on the abstract latent space representation. For example, if the goal is to generate an image of a person with brown

hair and glasses, these attributes are fed into the model along with the image, ensuring that the generated face meets these criteria.

The CVAE architecture, illustrated in Figure 3.4, demonstrates a simplified version of how the conditional information is integrated into the model.

The training process for a CVAE follows the same principles as that of a VAE, involving the optimization of both the reconstruction loss and the KL divergence to ensure the model accurately learns the data distribution. However, the distinction lies in the decoder's ability to generate new data post-training.

After training, the CVAE decoder allows for two distinct ways to manipulate generated new samples: the abstract z component and the conditional z component.

CVAEs do not fully resolve the issue of image blurriness, a common problem in VAE-generated samples. However, the inclusion of conditions can improve the sharpness of generated images to some extent. The level of detail of the CVAE has demonstrated enough sharpness for forensic applications.

To reduce the blurriness issue, additional post-processing treatments can be applied to the output. These treatments might include techniques such as super-resolution algorithms, which enhance image details, or GAN-based refinement methods, which can further sharpen the generated images. By combining CVAEs with these post-processing techniques, it is possible to produce higher-quality, more detailed images that better meet the requirements of specific applications.

3.5 CONDITIONAL INPUTS

The model will use conditional inputs more related to color which complements the information not provided by the sketch. The inputs are provided in the CelebA annotation, which contains 40 different attributes from the image.

However, not all attributes have the same level of impact. For instance, attributes like black hair are very obvious and easily detectable, while others, such as a pointy nose, are less apparent, especially due to the dimension reduction to 64 pixels. Some attributes are inherently captured by the sketch itself, whereas color attributes are more harder to infer. In these cases, the model relies on predefined concepts and prior knowledge to make accurate predictions.

While testing on the CelebA dataset, we found that all attributes contributed to the model's performance. However, manually setting 40 attributes for generating new images that are not pre-annotated is time-consuming and impractical. Therefore, we reduced the number of attributes to simplify the analysis and the generation process.

This approach ensures that essential features are accurately represented without overwhelming the model or the user with excessive data input. The selected attributes provide a sufficient level of complementation to the sketch to generate the desired images while minimizing the complexity of the input data.

3.6 RESNET

The Residual Network architecture (Figure 3.8) was introduced to address performance stagnation in deeper neural networks. Theoretically, increasing network depth should enhance performance if overfitting is controlled. However, before ResNet, training very deep models faced significant challenges, mainly due to the vanishing gradient problem, where gradients become too small to effectively update earlier layers during backpropagation, leading to suboptimal performance (He et al., 2016). Also, deeper networks often encountered degradation, where adding more layers increased training error rates.

A major contribution of ResNet is the use of shortcut connections, which bypass one or more layers, enabling the model to learn identity mappings. These connections are formulated $H(x) = F(x) + x$, where $H(x)$ is the output after adding the residual connection, $F(x)$ is the output after passing through the convolutional layers, and x is the input that is added as the residual connection.

The residual connections ensure that important features are not lost during the forward pass, preserving essential information. This technique also significantly improves the training process by addressing the issue of degradation that deeper networks previously faced. Besides, residual networks reduce the risk of overfitting, since the model can more easily learn identity mappings, and unnecessary layers can be skipped, allowing the network to focus on the most relevant transformations (He et al., 2016).

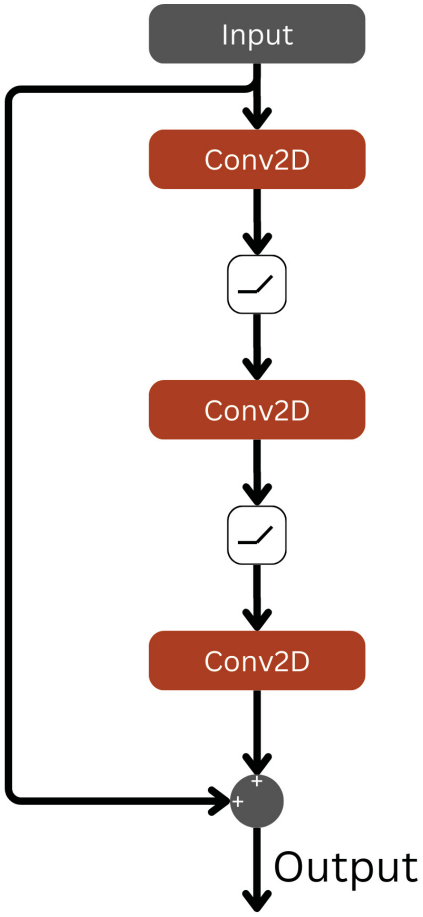


Figure 3.8: Resnet component.

For a given layer in a neural network, the transformation applied is $\mathcal{F}(x)$. Where $\mathcal{F}$ is a series of operations (such as convolutions, batch normalization, and activation functions) applied to the input x.

In a residual block, instead of just learning $\mathcal{F}(x)$, the network learns the residual function:

$$\mathcal{H}(x) = \mathcal{F}(x) + x \tag{3.28}$$

Here's a breakdown of the components:

- x: Input to the residual block.
- $\mathcal{F}(x)$: Residual function, representing the operations applied to x. This usually includes convolutional layers, batch normalization, and ReLU activation.
- $\mathcal{H}(x)$: Output of the residual block, which is the sum of the residual function and the original input x.

The output of a residual block can be formally written as:

$$\mathcal{H}(x) = \mathcal{F}(x, \{W_i\}) + x \tag{3.29}$$

where:

- $\mathcal{H}(x)$: Output of the residual block.
- $\mathcal{F}(x, \{W_i\})$: Residual function applied to the input x with weights $\{W_i\}$.
- x: Original input to the block.

A simple residual block with three convolutional layers with ReLU activation functions can be expressed as:

$$\mathcal{F}(x) = W_3 * \sigma(W_2 * \sigma(W_1 * x)) \tag{3.30}$$

where:

- W_i: Weights of the i-th convolutional layer.
- σ: ReLU activation function.

Then, the output of this block would be:

$$\mathcal{H}(x) = W_3 * \sigma(W_2 * \sigma(W_1 * x)) + x \tag{3.31}$$

3.7 SKIP CONNECTIONS

The U-Net architecture, originally proposed for medical image segmentation (Ronneberger et al., 2015), offers a method to ensure the model preserves structural information about the sketch. This architecture's greatest feature is its use of skip connections, as illustrated in Figure 3.10.

During the convolutional downsampling process, the image is progressively reduced in size until it reaches the latent space. At each step of this process, the feature map generated by each convolutional layer is concatenated with its corresponding feature map on the upsampling side. This design allows the latent space to capture semantic information about the sketch, while the skip connections retain the structural details necessary for a more accurate reconstruction of the original image. This retention of spatial information is necessary for tasks that require precise localization and structural integrity, such as sketch-based image generation.

Let's assume that x is the output of a convolutional layer in the encoder and y is the output of an upsampling layer in the decoder. The concatenation of these two feature maps x and y is performed before further processing in the decoder.

$$z = \text{concat}(x, y) \tag{3.32}$$

where:

- x: Feature maps from a layer in the encoder.
- y: Feature maps from a corresponding layer in the decoder.
- concat(x, y): Concatenation operation that combines the feature maps x and y along the channel dimension.

After concatenation, usually a convolution operation is applied to the combined feature maps:

$$o = \sigma(W * z + b) \tag{3.33}$$

where:

- $*$: Convolution operation.
- b: Bias term.
- σ: Activation function (ReLU).
- o: Output feature maps after applying the convolution.

3.8 ATTENTION MECHANISM

The attention mechanism serves as a regularization technique to enhance the training process of neural networks (Oktay et al., 2022). Figure 3.10 illustrates an autoencoder with skip connections, incorporating an attention cell placed before the concatenation stage.

The attention cell, depicted in Figure 3.9, operates by calculating the relevance of each part of the image. It does this by combining the information from the skip connection with the output of the previous layer in the upsampling process. The resulting relevance map points out which areas of the image are most important for accurate reconstruction.

This relevance map is then used to scale the input from the previous layer. Specifically, regions deemed relevant are amplified, while less relevant areas are attenuated.

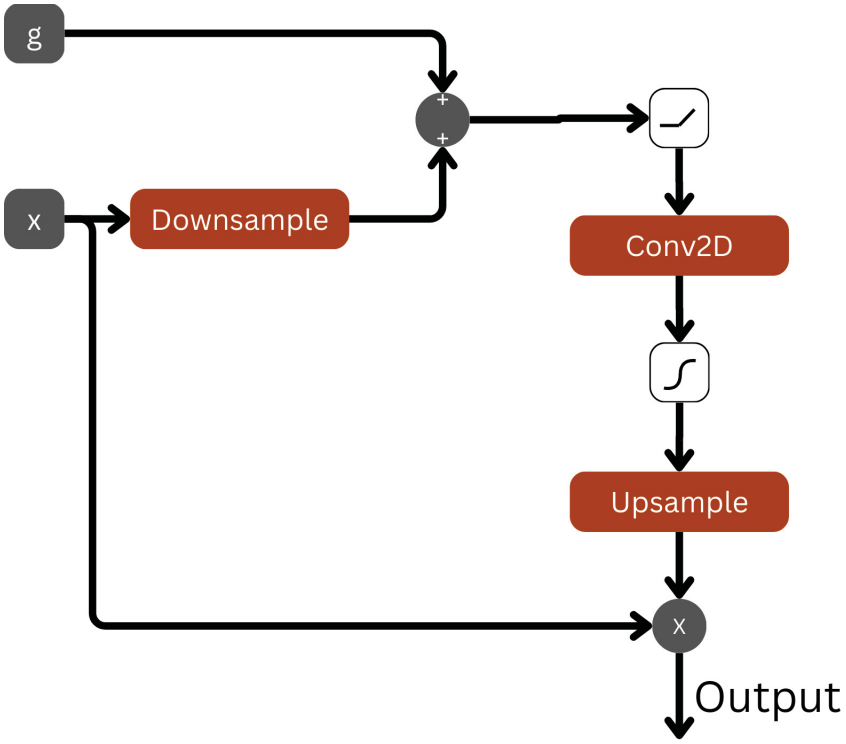


Figure 3.9: Attention cell.

Attention cells in U-Net are used to focus on relevant features in the input data, helping the model to better segment the target areas by suppressing irrelevant regions.

An attention cell involves the following steps:

1. **Input maps**: Two inputs, one from the encoder (skip connection) and one from the decoder (gating signal), are processed through convolutional layers. These inputs are denoted as x (encoder feature map) and g (decoder feature map).

2. **Linear Combination**:

$$\psi = g + \text{Downsample}(x) \tag{3.34}$$

3. **Non-linear Activation**:

$$\psi' = \text{ReLU}(\psi) \tag{3.35}$$

4. **Attention Coefficients**:

$$\alpha = \sigma(W_\psi \psi' + b_\psi) \tag{3.36}$$

5. **Upsampling**:

$$\alpha' = \text{Upsample}(\alpha)$$

6. **Multiplication with the Input**:

$$x' = \alpha' \cdot x \tag{3.37}$$

where:

- W_ψ: Weights of the convolutional layer.
- b_ψ: Bias term.
- ReLU: Rectified Linear Unit activation function.
- σ: Sigmoid activation function.
- x: Input feature map from the encoder.
- g: Gating signal from the decoder.
- Downsample(x): Downsampling operation applied to the encoder feature map.
- Upsample(α): Upsampling operation applied to the attention coefficients.
- x': Output feature map after applying the attention cell.

3.9 PROPOSED ARCHITECTURE

The proposed architecture incorporates several components to enhance the performance and accuracy of the model, as illustrated in Figure 3.10. The design transforms sketch inputs into photorealistic images while maintaining structural integrity and enabling fine-tuning through conditional inputs and adjustments in preprocessing parameters. The input image of this model is the edge map extracted through the preprocessing pipeline, and the output is supposed to be as similar as possible to the original photo of the person.

Conditional inputs are incorporated into both the encoder and decoder sections of the network. These inputs provide additional information, such as color and specific facial attributes, which complement the features extracted from the sketch, enabling the model to generate more accurate representations. Also, by adjusting the conditional inputs, it is possible to alter the resulting image's attributes, offering more flexibility and control over the generated outputs.

The attention mechanism is placed before the concatenation of skip connections. The attention cell calculates the relevance of different parts of the image, ensuring that the most important features and areas are emphasized during the reconstruction process (Oktay et al., 2022).

Skip connections mitigate the risk of losing important structural details as the image is processed through multiple layers, ensuring a more accurate and detailed reconstruction. (He et al., 2016). These connections bridge the encoder and decoder layers, ensuring that features

with higher resolution from the early stages of the network are available during the upsampling process.

The latent space representation is derived using the mean (μ) and variance (σ) vectors, as described by the reparameterization trick in VAE (Odaibo, 2019). This technique ensures that the latent space provides a continuous and smooth mapping from the sketch domain to the photorealistic domain.

As will be discussed in Section 3.12, this comprehensive setup not only improves image generation but also allows for the fine-tuning of outputs through conditional inputs and preprocessing parameters. The combination of CVAE, attention mechanisms, and skip connections creates a robust framework capable of producing, faithfully, photorealistic images from sketch inputs.

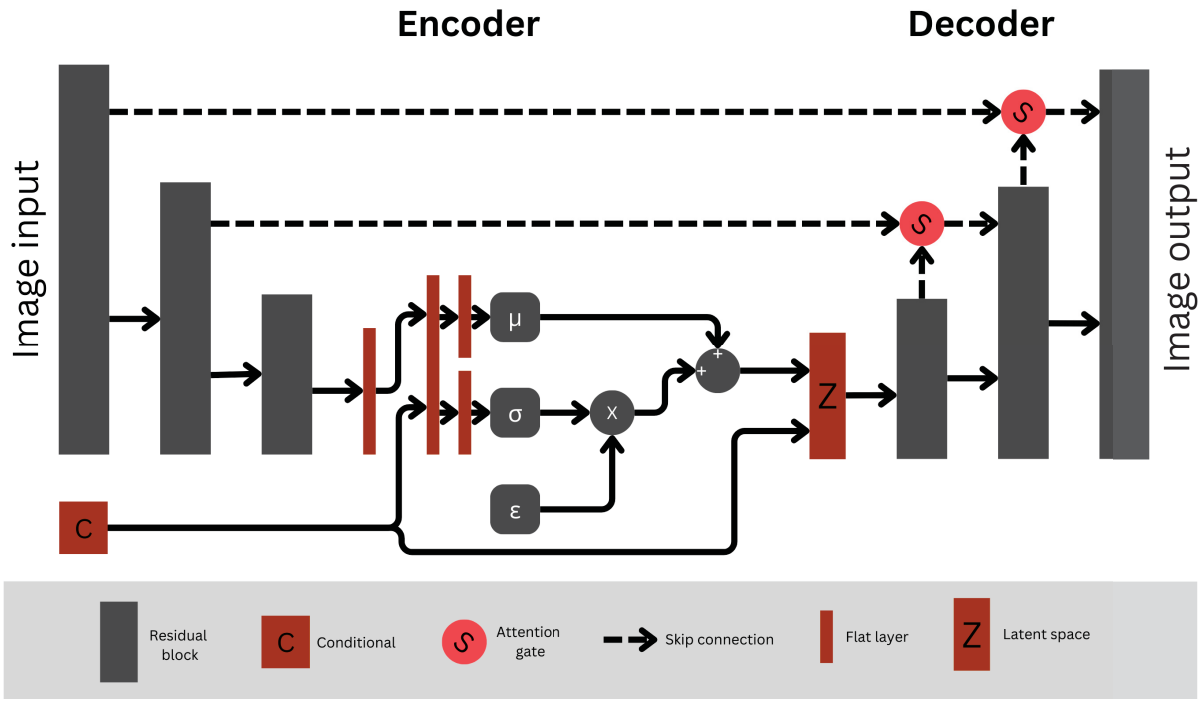


Figure 3.10: Final model architecture.

3.10 OBJECTIVE FUNCTION

In the previous sections, we derived the overall loss function for the VAE, which combines the reconstruction loss and the KL divergence term to ensure a smooth and meaningful latent space. However, the reconstruction loss component, which measures how accurately the generated image resembles the input image, was not explicitly defined. This section expands on the reconstruction loss, explaining its formulation and significance in the VAE framework. Choosing an appropriate reconstruction loss has been proved experimentally in the course of this work to strongly impact the final result.

A constant value (named β) is multiplied with the regularization term to modulate its weight in the loss function, balancing the latent space constraint (Higgins et al., 2022).

When β is significantly greater than one, the regularization term dominates the loss function. This can lead to blurrier images since the emphasis on reconstruction loss diminishes, causing the model to prioritize fitting the latent space distribution over accurately reconstructing the input. Conversely, if β is close to zero, the influence of the regularization term is reduced, resulting in a less predictable and less continuous latent space. This insufficient constraint can make it more complicated to understand and control the latent space. Therefore, choosing an appropriate β value helps to achieve a balance between preserving image quality and maintaining a well-structured latent space.

To enhance the perceptual quality of the reconstructed images, the Structural Similarity Index Measure (SSIM) loss is chosen as the reconstruction component. SSIM provides a more perceptual evaluation compared to MSE or Peak Signal-to-Noise Ratio (PSNR), which calculates the loss in a pixelwise manner. Unlike MSE or PSNR, SSIM evaluates the similarity between images by considering changes in structural information, luminance, and contrast (Wang et al., 2004). This approach results in a loss that is less susceptible to minor pixelwise differences, providing a more human-like measure of image similarity. The SSIM loss is defined as:

$$\text{SSIM}(x, y) = l(x, y) \cdot c(x, y) \cdot s(x, y) \tag{3.38}$$

The luminance component can be expanded to:

$$l(x, y) = \frac{2\mu_x\mu_y + C_1}{\mu_x^2 + \mu_y^2 + C_1} \tag{3.39}$$

The contrast component can be expanded to:

$$c(x, y) = \frac{2\sigma_x\sigma_y + C_2}{\sigma_x^2 + \sigma_y^2 + C_2} \tag{3.40}$$

And structural component can be expanded to:

$$s(x, y) = \frac{\sigma_{xy} + C_3}{\sigma_x\sigma_y + C_3} \tag{3.41}$$

So, the complete SSIM formula is defined by:

$$\text{SSIM}(x, y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)} \tag{3.42}$$

where μ_x and μ_y are the mean values of the images x and y, σ_x^2 and σ_y^2 are the variances, and σ_{xy} is the covariance of x and y. Constants C_1 and C_2 are small values included to stabilize the division with weak denominators. By incorporating SSIM loss, the model can better capture perceptual differences between the reconstructed and original images, leading to more visually pleasing results.

3.11 TRAINING

In this study, three distinct models are trained for comparison: a simple Autoencoder (AE) model, a proposed AE architecture enhanced with an attention mechanism and skip connections, and the complete CVAE as proposed. The objective of training these three models is to justify the increased architectural complexity through demonstrated improvements in image reconstruction quality.

The simple AE model serves as a baseline, representing a conventional approach to image compression and reconstruction without any advanced enhancements. This model will allow us to establish a performance benchmark and understand the fundamental limitations in terms of image quality and feature retention.

The proposed AE model incorporates attention mechanisms and skip connections, as detailed earlier. The inclusion of skip connections ensures that structural information is preserved throughout the network, which is particularly decisive for maintaining the integrity of fine details in the reconstructed images. The attention mechanism not only enhances the model by focusing on the most relevant parts of the image, thereby improving the overall reconstruction but also serves as another regularization technique. This model is expected to create more detailed and sharper images when compared to the simple AE.

The complete CVAE model incorporates attributes such as color, age, and other facial details. Training the CVAE model will showcase its ability to add specific attributes to the generated photorealistic images based on varying conditional inputs. The overall quality is supposed to be at least slightly better than the proposed AE. Yet, the most important idea for developing this model is to simplify the adjustments and interaction for the user when generating new images.

The models were structured to have similar depths and parameters, ensuring a fair comparison. Each model consisted of multiple convolutional layers with 3x3 kernel sizes, followed by batch normalization and ReLU activation functions to introduce non-linearity. The simpler AE architecture included 13 layers. While sharing a similar core structure, the proposed AE and the CVAE architectures included additional layers for residual connections, attention mechanisms, and conditional input processing, bringing the total to 26 layers. These additions primarily serve to improve regularization. All models were trained for 15 epochs with a batch size of 64, using the ADAM optimizer (Kingma and Ba, 2015) with a learning rate of 0.001. Batch normalization was applied after each convolutional layer to mitigate issues like exploding gradients.

An early stopping callback was not implemented because the models often ceased training while producing very blurry images. To maintain consistency and ensure a fair comparison, an equal number of epochs was chosen for each model. This approach allows for a direct comparison of the validation losses across the models throughout the training epochs.

3.12 EVALUATION METRIC OVERVIEW

Evaluating generative image models is a complex task, as it requires balancing both objective and perceptual assessments. pixelwise metrics, such as MSE, offer a straightforward way to quantify the accuracy of image reconstruction by measuring the average squared differences between corresponding pixels of the original and generated images. While these metrics are useful for providing a basic understanding of reconstruction fidelity, they fall short of capturing the human visual perception quality, which means that images with low MSE may still appear visually unappealing or inaccurate.

Employing perceptual metrics that more accurately reflect human visual interpretation proves beneficial to overcome the limitations of pixelwise metrics. Perceptual metrics focus on evaluating images by considering their overall structure and appearance, rather than focusing solely on local pixel-by-pixel differences. These metrics provide a more holistic assessment of image quality. By incorporating both types of metrics, we can achieve a more comprehensive evaluation of the generative model's performance.

One commonly used perceptual metric is the SSIM, which evaluates images based on their luminance, contrast, and structural information, providing a measure of similarity that correlates well with human perception (Wang et al., 2004). Unlike pixelwise metrics, SSIM assesses how well the overall structure of an image is preserved, making it a better tool for evaluating generative models.

Another important perceptual metric is the Frechet Inception Distance (FID). FID calculates the distance between the distributions of high-level features extracted from the original and generated images using a pre-trained model, normally Inception V3 (Dowson and Landau, 1982). By comparing these feature distributions, FID provides a quantitative measure of how similar the generated images are to real images. This metric is valuable because it captures more abstract, qualities of the images, such as texture and patterns, which are often more relevant to human observers.

The FID employs a model trained on a diverse set of images to provide a general assessment of image quality. However, for tasks involving face generation, it is necessary to ensure that the generated face closely resembles the original person, not just in overall quality but in identifiable features.

Facenet, a pre-trained model specifically trained for face recognition and identification, is also used in the evaluation. Facenet is designed to extract facial features helpful for identifying individuals (Schroff et al., 2015). Facenet can evaluate whether the generated face image retains enough similarity to the original face for accurate identification.

For this purpose, we compute the Euclidean distance between the feature vectors extracted by Facenet from the original and generated images. This metric provides a quantitative measure of similarity, ensuring that the generated face is not only visually appealing but also recognizable as the same individual. By incorporating Facenet, we enhance our evaluation

framework to better capture the nuances of facial similarity, which is essential for applications in forensic contexts.

Since a potential application of this model is to generate photorealistic images from sketches for suspect identification in a database, the Facenet-based metric is particularly critical. By tracking the Euclidean distance between the features extracted by Facenet, we can assess the quality of the generated images in practical forensic scenarios, making this metric the most important one to monitor.

Although the model is trained exclusively on the CelebA database, its evaluation is conducted using a diverse set of databases to ensure its generalization capability. The evaluation includes both full-reference samples, where the original image is available for comparison, and more subjective analysis with samples that contain only the sketch without a reference picture.

3.12.1 Mean Squared Error (MSE) for Images

Consider two images: the original image I and the reconstructed (or predicted) image $\hat{I}$, both of size $H \times W$ (height H and width W). The MSE between these two images is given by:

$$\text{MSE} = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} \left(I(i,j) - \hat{I}(i,j)\right)^2 \tag{3.43}$$

where:

- $I(i,j)$ is the pixel value at position (i,j) in the original image.
- $\hat{I}(i,j)$ is the pixel value at position (i,j) in the reconstructed image.
- H is the height of the images.
- W is the width of the images.

3.12.2 Fréchet Inception Distance

Given two sets of images, real images X_r and generated images X_g, let μ_r and Σ_r be the mean and covariance of the activations of the real images, and μ_g and Σ_g be the mean and covariance of the activations of the generated images. The FID is defined as (Dowson and Landau, 1982):

$$\text{FID} = \|\mu_r - \mu_g\|^2 + \text{Tr}(\Sigma_r + \Sigma_g - 2(\Sigma_r \Sigma_g)^{\frac{1}{2}}) \tag{3.44}$$

where:

- $\|\mu_r - \mu_g\|^2$ is the squared Euclidean output distance between the means of the real and generated images after passing the model.
- Tr denotes the trace of a matrix (sum of its diagonal elements).

- Σ_r and Σ_g are the covariance matrices of the activations for the real and generated images, respectively.

  These variables represent high-level features from the Inception V3 model.

3.12.3 Facenet Overview

Facenet is a pre-trained deep-learning model developed for face recognition, verification, and clustering. It uses a deep convolutional network to map images of faces into a compact Euclidean space where distances directly correspond to a measure of face similarity. Some important concepts about the Facenet model are (Schroff et al., 2015):

- **Embedding Space**:
  - Facenet maps face images to a high-dimensional space (128 dimensions) called the embedding space.
  - In this space, the Euclidean distance between two embeddings reflects the similarity of the corresponding faces.
- **Triplet Loss**:
  - Facenet is trained using a triplet loss function, which ensures that the distance between an anchor (a reference image) and a positive example (another image of the same person) is smaller than the distance between the anchor and a negative example (an image of a different person) by a margin.
  - The triplet loss is defined as:

$$L = \sum_{i}^{N} \left[\|f(A_i) - f(P_i)\|_2^2 - \|f(A_i) - f(N_i)\|_2^2 + \alpha \right]_+ \tag{3.45}$$

    where:
    * $f(A_i)$ is the embedding of the anchor image.
    * $f(P_i)$ is the embedding of the positive image.
    * $f(N_i)$ is the embedding of the negative image.
    * $\|\cdot\|_2^2$ represents the squared Euclidean distance.
    * α is a margin that ensures a gap between the positive and negative pairs.
    * $[\cdot]_+$ denotes the hinge function, which outputs the value inside the brackets if it is positive and zero otherwise.
- **Euclidean Distance for Facenet**:

  - Given two face embeddings f_1 and f_2, both of size d, the Euclidean distance between these embeddings is calculated as:

$$\text{Distance}(f_1, f_2) = \sqrt{\sum_{i=1}^{d} (f_{1,i} - f_{2,i})^2} \tag{3.46}$$

    where:

    * f_1 and f_2 are the face embeddings.
    * d is the dimensionality of the embeddings.
    * $f_{1,i}$ and $f_{2,i}$ are the i-th components of embeddings f_1 and f_2, respectively.

- **Applications**:
  - Face verification: Determine if two images are of the same person.
  - Face recognition: Identify a person from a database of known faces.
  - Clustering: Group similar faces together.

3.13 DATABASES

The models are trained using the CelebA database, which comprises over 202,599 images with dimensions of 178x218 pixels. Each image is annotated with a vector of 40 attributes. For this study, 14 attributes are selected and filtered to serve as conditional inputs for the CVAE. These attributes are chosen based on their relevance and ability to complement the sketch information. The CelebA database provides a rich and diverse set of facial images, enabling the model to learn and generalize across various facial features and conditions.

The images pass through the preprocessing pipeline and then are resized to 64x64 pixels to fit the model's input layer. Figure 3.11 illustrates the conditional inputs, the edge map generated after passing through the pipeline, and the original image. As observed, the edge map extracted from the original image is perfectly aligned with it. This alignment ensures that during training, the model learns to associate edge maps with their corresponding photorealistic images accurately. However, this perfect alignment does not hold true when evaluating sketches. Sketches often have traces that are slightly misplaced compared to the real image, which may have distinct behavior when evaluated with different metrics.

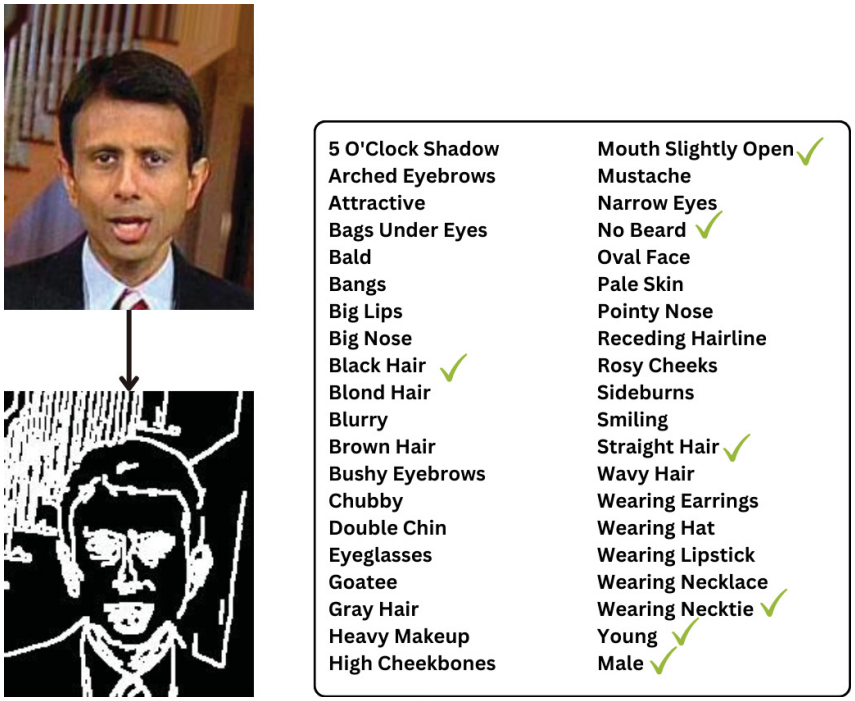


Figure 3.11: Training input example.

Since the CelebA database does not contain any sketches, the evaluation is extended using different sketch databases. One such database is the CUHK student database (Zhang et al., 2011), which contains pairs of sketches and original pictures of young Asian individuals.

This database, however, does not provide the 40 attributes available in CelebA. These attributes were manually annotated for each image (only needed in the CVAE model). This manual annotation ensures that the same conditional inputs used in the CelebA-trained model can be applied to the CUHK student database, facilitating a more consistent evaluation.

It is important to emphasize that in the CUHK student database, the sketch traces are not perfectly aligned with the facial contours of the original images. This misalignment introduces an additional layer of complexity to the evaluation. The sketches may also be an imperfect representation of the real person, stressing the model's ability to generalize and accurately reconstruct photorealistic images from less precise sketches.

By incorporating the CUHK student database into the evaluation, we can assess the model's robustness and flexibility in handling more realistic variations in sketch alignment and attribute annotation.

To further extend the evaluation, we collected some random digitally drawn images from the internet, introducing additional diversity to the dataset. Unlike the other databases, these images do not have matching pairs of original photos and sketches. As a result, the scope of this evaluation is limited to assessing a more subjective analysis of the generated images. This evaluation helps to understand how well the model performs with digitally drawn sketches.

By including these diverse sources, we can more comprehensively evaluate the model's ability to generate a good photorealistic representation of the images across various styles and drawing methods, ensuring its robustness in real-world applications.

4 RESULTS

In this section, we evaluate the images generated on a diverse set of databases to ensure a broad capability of generalizing sketch styles. We compare the performance of three models: a simple autoencoder, the proposed enhanced autoencoder, and the proposed CVAE.

In Section 4.1 we discuss the training procedure, explaining the differences in results on training and validation data among the models. Section 4.2 demonstrates the model's capability to reconstruct the original face and its features given a sketch, using metrics such as MSE, SSIM, FID, and FaceNet Euclidean Distance. Section 4.3 focuses on identifying the reconstructed face given a small set of images, utilizing FaceNet-based metric to evaluate the similarity between the generated and real images. In Section 4.4 we expand the evaluation to digital unpaired images, assessing the overall quality of the generated images and their interpretation on more variated styles of sketches (minimalist, hatching, and realistic). Section 4.5 explores the impact of conditional inputs on the quality of the generated images, demonstrating the flexibility of the CVAE model in generating more specific images based on the given conditions. Finally, Section 4.6 proposes a method to fine-tune the preprocessing when generating new images to use in more specific applications depending on the sketch artist.

4.1 TRAINING EVALUATION

The models were trained using 50,000 images from the CelebA dataset (Liu et al., 2015) over 15 epochs. Training with 10,000 images already produced good results when evaluated on the CelebA dataset. However, to ensure more variability, we decided to expand the training dataset. We also tried to use the full CelebA dataset with fewer epochs to maintain the same training time, but the results decayed in quality.

Figure 4.1 shows that the loss for the proposed models is lower than that of the simple AE. It is important to note that the loss function for the CVAE differs from that of the AE due to the inclusion of a regularization component, which often results in a slightly bigger overall loss.

The proposed AE loss demonstrated the best loss while training. Despite this, the proposed AE exhibited more signs of overfitting than the CVAE, leading to a better result on the CVAE in the evaluation step.

The regularization term in the CVAE's loss function encourages the encoder to map the input data into a continuous Gaussian latent space. This regularization facilitates better generalization by the decoder, as it helps to maintain the continuity and smoother transitions of the latent space. Additionally, the incorporation of conditional inputs provides more interpretable information to the model, thereby guiding the generation of specific attributes. The CVAE's ability to generalize better than the AE is possibly attributable to this regularization component and the conditional inputs.

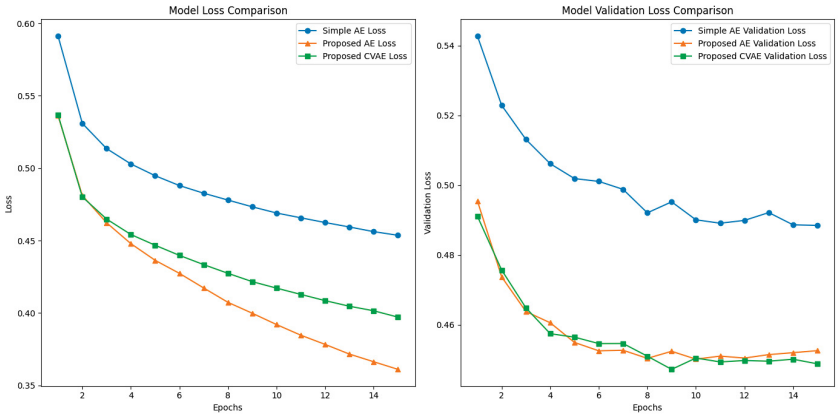


Figure 4.1: Training and validation history.

4.2 RECONSTRUCTION QUALITY

To assess whether the generated image can accurately reproduce the characteristics of the original face, we compare the outputs of the three models: a simple autoencoder, the proposed autoencoder with attention mechanisms and skip connections, and the complete CVAE. Figures 4.2 and 4.3 present the original images, their sketch representations, and the corresponding generated images from the CelebA database and the CUHK student database, respectively.

Subjectively speaking, the results from the CelebA database demonstrate a satisfactory degree of accuracy in reconstructing the original faces. The background of the generated images is often rendered in grayscale, mainly because most of the background details are filtered out during the preprocessing step. Consequently, the model predicts a neutral value for these areas to minimize reconstruction error in regions with unknown or irrelevant data.

The background could be removed from the image to ensure cleaner input data using segmentation techniques (Chiu et al., 2010). However, some images did not respond well to this approach, as some important areas were classified as background incorrectly. Since the background wasn't a major issue, it was maintained.

The simple autoencoder's generated images are noticeably blurrier and more generic compared to those produced by the other models. This blur and lack of detail point to the limitations of the simple AE in capturing and reconstructing fine features.

In contrast, the proposed autoencoder and the CVAE produced images with a lot more detail and clarity. While both proposed models yield similar results, the CVAE performs slightly better. This improvement is likely due to the conditional inputs, which provide additional contextual information, allowing the model to generate more accurate and detailed images. The regularization term in the CVAE also contributes to this enhancement by ensuring a well-structured latent space, facilitating more consistent and reliable image generation.

(a) Original 1 (b) Original 2 (c) Original 3

(d) Edge map 1 (e) Edge map 2 (f) Edge map 3

(g) Simple AE 1 (h) Simple AE 2 (i) Simple AE 3

(j) Proposed AE 1 (k) Proposed AE 2 (l) Proposed AE 3

(m) Prop. CVAE 1 (n) Prop. CVAE 2 (o) Prop. CVAE 3

Figure 4.2: CelebA image comparison

(a) Original 1 (b) Original 2 (c) Original 3

(d) Sketch 1 (e) Sketch 2 (f) Sketch 3

(g) Simple AE 1 (h) Simple AE 2 (i) Simple AE 3

(j) Proposed AE 1 (k) Proposed AE 2 (l) Proposed AE 3

(m) Prop. CVAE 1 (n) Prop. CVAE 2 (o) Prop. CVAE 3

Figure 4.3: CUHK image comparison

The images generated from the CUHK sketches also demonstrated similar representations of the original faces. To achieve these results, the stochastic behavior inherent in the preprocessing pipeline was removed, and the parameters were set to fixed values. Different parameters could be fine-tuned to optimize the model's input depending on the sketch style or the original image dimensions. However, in some cases the generated images displayed different interpretations from some traces of the sketches, leading to a less similar reconstruction when compared to the original image. These images are generated from a sketch. It's important to note that when the

sketch is drawn, it already contains some representation error that is summed with the model's error. If the sketch fails to represent some feature, the model will also capture this misinformation when generating the final image.

The parameters that proved most useful for fine-tuning were the thresholds for the weak and strong edges in the Canny edge detection algorithm and the size of the Gaussian filter applied after edge detection. Adjusting the edge thresholds was particularly beneficial when dealing with sketches that contained excessive details or shadows (perhaps an equalization could be used to solve this problem), which can be difficult for the encoder to interpret and map into the latent space accurately. By tweaking these thresholds, it was possible to filter out unnecessary details and focus on the most relevant features of the sketch.

Similarly, adjusting the size of the Gaussian kernel was helpful when the hair in the sketch contained too many details that the model might misinterpret as illumination reflections. By fine-tuning the Gaussian filter size, the model could better smooth out these details.

Section 4.6 provides a more detailed discussion on the importance of defining the correct parameters when inputting different sketches. This section will analyze the specific adjustments made and their impact on the quality of the generated images, highlighting the necessity of parameter optimization for achieving the best possible results from the model.

The simple AE generates noticeably blurrier and more generic images compared to the proposed models. This difference in quality is evident in both subjective visual assessments and perceptual objective metrics (less evident in pixelwise metrics). In Figure 4.4, we can observe the comparison of original and generated CelebA images using these metrics. Also, in table 4.1 we can see the metrics without any scale applied.

The MSE results were quite similar across all models, with the proposed CVAE achieving slightly better results. However, MSE is not ideal for evaluating perceptual features since it focuses on pixelwise differences, which do not necessarily reflect human visual perception.

For perceptual evaluation, the SSIM and the FID provide more meaningful insights. The proposed models demonstrated superior performance in these metrics, indicating better preservation of structural and perceptual details in the generated images. The SSIM metric is inverted (1 - SSIM), meaning that smaller SSIM values indicate better results. FID measures the distance between high-level feature distributions, aligning more closely with human perception.

However, these metrics are generic and not specifically tailored for facial recognition applications. To address this, we utilized the FaceNet model to evaluate the facial similarity between the original and generated images. FaceNet provides a more specialized metric. This metric showed that the proposed models, particularly the CVAE, yielded better results in maintaining facial similarities.

Overall, the proposed models, especially the CVAE, offer great improvements in image quality and facial similarity compared to the simple AE, with a 29.2% improvement using the Facenet metric and 18% using the FID Score. In this evaluation, we can see that as the metrics become more specific, the discrepancy between the models is also increased.

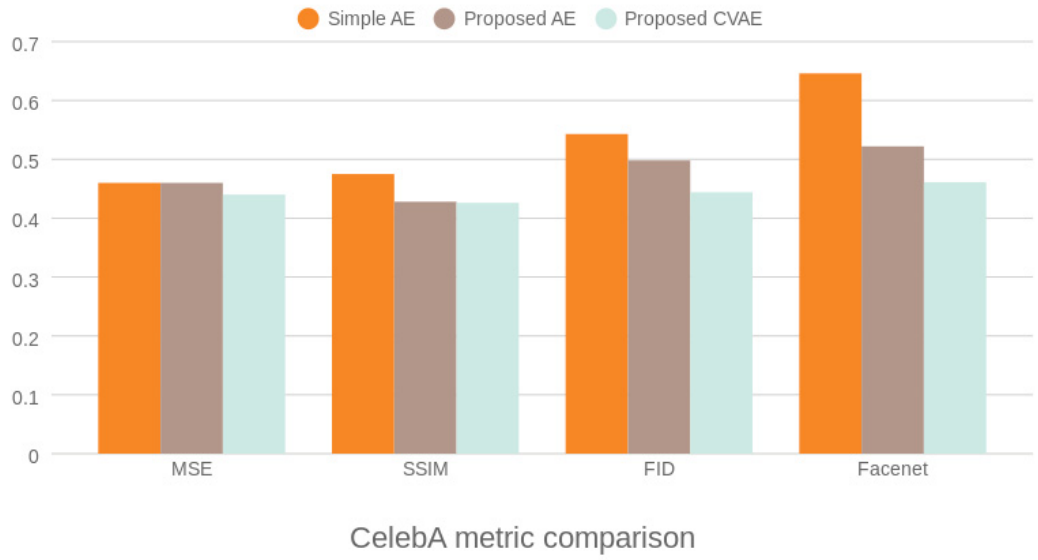


Figure 4.4: CelebA performance metric comparison for each model on 10 samples (Metrics were scaled to the same perspective).

	Dataset	MSE	SSIM	PSNR	FID	Facenet
Simpler AE	**CelebA**	0.05	0.52	14.24	180.58	0.65
	CUHK	0.07	0.32	11.69	176.18	0.82
Proprosed AE	**CelebA**	0.05	0.57	14.51	165.54	0.52
	CUHK	0.08	0.30	11.22	180.64	0.44
Proprosed CVAE	**CelebA**	0.04	0.57	14.69	147.91	0.46
	CUHK	0.06	0.33	12.01	180.46	0.35

Table 4.1: Performance metrics for CelebA and CUHK datasets

In Figure 4.5, we perform a similar comparison using the CUHK database. The results show that the more generic metrics (MSE, SSIM, and FID) presented very similar results across the different models. The MSE showed bigger variations, with the simple AE outperforming the proposed AE. However, given that MSE is a pixelwise metric, it's not the ideal metric to monitor.

The SSIM and FID also showed comparable performance between the models, though these metrics are more aligned with perceptual features. Despite these metrics providing useful insights, they may not fully reflect the specific nuances of facial features required for accurate recognition.

The most significant difference is observed with the FaceNet-based metric. The simple AE displayed a big discrepancy when evaluated with this metric, indicating that its generated images were less accurate in capturing the essential facial features necessary for recognition. The proposed AE showed an improvement of 53%, with fewer discrepancies than the simple AE. However, the CVAE demonstrated the best performance with a similarity 56.8% higher than the simple AE, with the smallest values on the FaceNet-based metric, indicating a higher degree of similarity between the generated and original faces.

Since FaceNet is specifically designed for face recognition, achieving good results with this metric is a strong indicator of the model's ability to generate realistic and recognizable facial images. This reinforces the superiority of the proposed models in accurately reconstructing facial features from sketches, making it a robust choice for applications requiring good facial recognition accuracy.

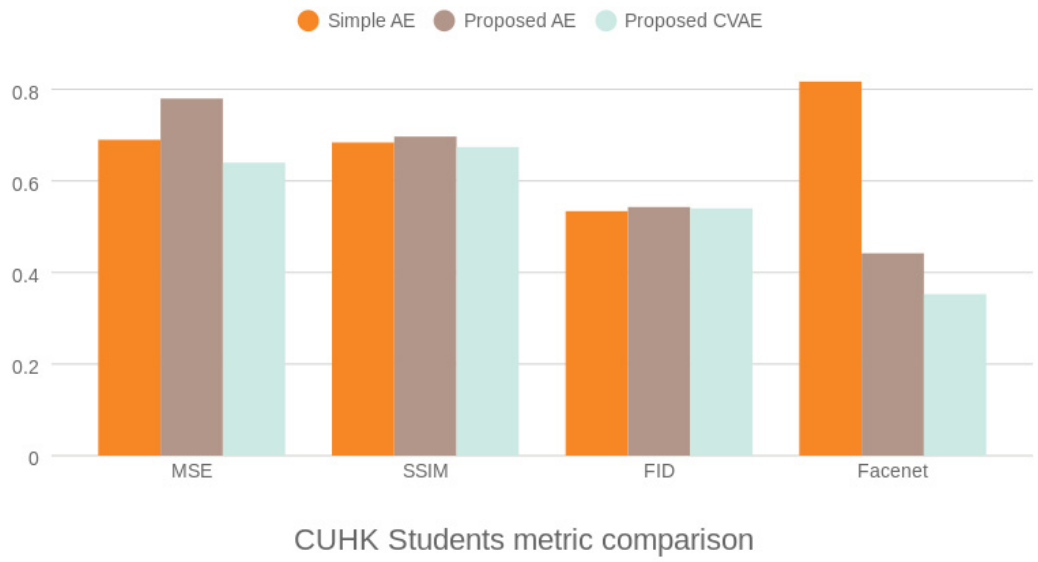


Figure 4.5: CUHK performance metric comparison for each model on 10 samples (Metrics were scaled to the same perspective).

4.3 IDENTIFICATION QUALITY

The FaceNet-based metric provides a good overview of the similarity between two faces. Now it's evaluated how accurately we can use this model alongside digital technology to detect a suspect-generated image against a database containing different people's faces. The idea is to simulate a practical scenario where a generated image from a sketch is compared to a database to identify potential matches accurately.

To achieve this evaluation, we take a small sample of random images and include the image of the suspect. Then we use the sketch to generate a face. We compare the generated face with each individual in the database by calculating the FaceNet Euclidean distance. Finally, we rank the similarity of the original suspect; if the suspect is the most similar face, the rank is one, if it's the second-best match, it is two, and so on.

The database size is varied from five images to fifty to understand the effects of database size on the results. The metrics were calculated based on a list of suspects. For each suspect, we add their image to the database and compare it with the generated image. Then, we remove this suspect from the database and repeat the process with another suspect. For the CelebA database, 30 suspect images were used, and for the CUHK database, 13 suspect images were used.

Figure 4.6 and 4.7 show both the accuracy score and the number of correctly identified faces versus the number of images in the database, respectively. For this task, the proposed AE provided the best results, correctly matching 53% of the images, while the CVAE achieved a 50% accuracy and the simpler AE provided the poorest result of 30%.

We can verify that most errors occur when the database size is less than 25 for the AEs and 35 for the CVAE. A possible explanation for this characteristic is that the suspect list used to generate the images was not fine-tuned (the preprocessing parameters were not adjusted). Consequently, some images may have lower quality, leading to mismatches in smaller databases. Poor-quality images are ranked low in initial states when the database is small. On the other hand, high-quality generated images maintain their best similarity ranking even as the database size increases, stabilizing the accuracy score.

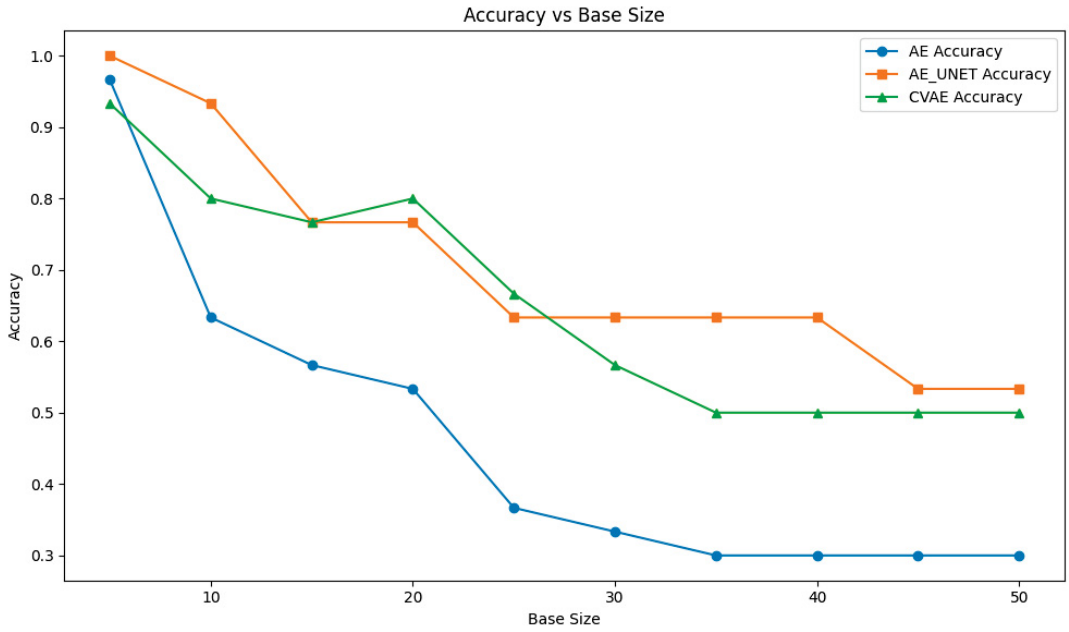


Figure 4.6: Accuracy of facial identification using CelebA.

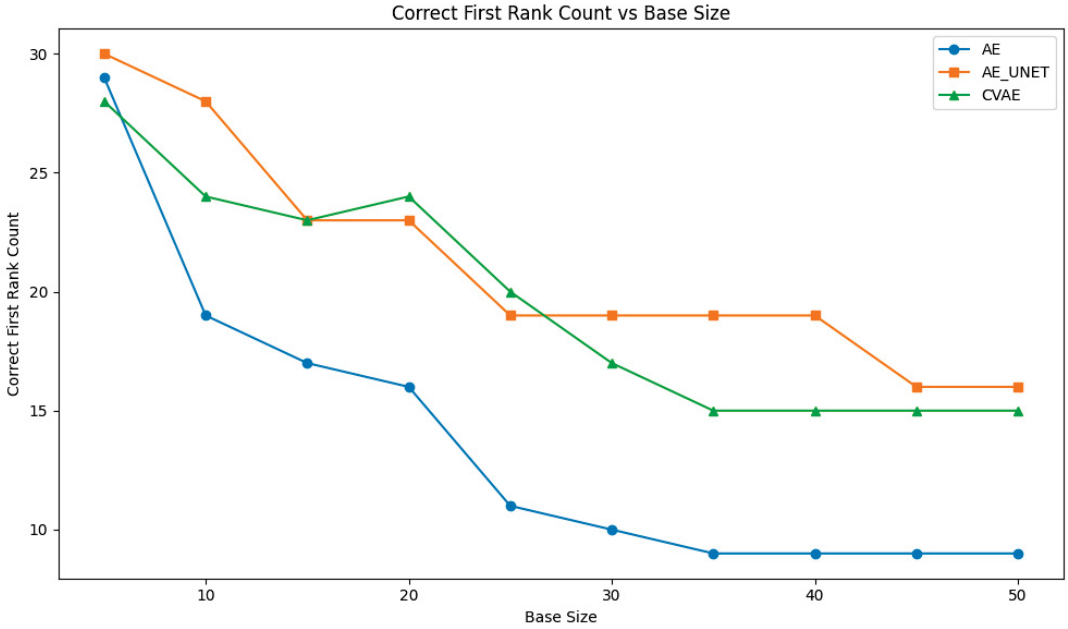


Figure 4.7: Quantity of correctly identified faces using CelebA.

We can explore in more detail the performance of images that were not correctly identified. Figure 4.8 shows the mean rank position of the images for each database size. The rank position of a correctly identified image is one, indicating it is the best match. From table 4.2, we observe that the mean rank of the simple autoencoder is 7.2, which is close to double the mean rank of the proposed models.

Even when the accuracy of the models remains stable, the mean rank position continues to rise with increasing database size. This phenomenon occurs because the incorrectly identified images receive progressively worse rankings as more images are added to the database. For the simple autoencoder, the mean rank position increases sharply, indicating that the incorrect matches are much further from the correct identification. The proposed models exhibit a slower increase in mean rank because most images are correctly identified with a rank of one. The fact that the mean rank is still increasing suggests that some images may be of lower quality and could benefit from further fine-tuning.

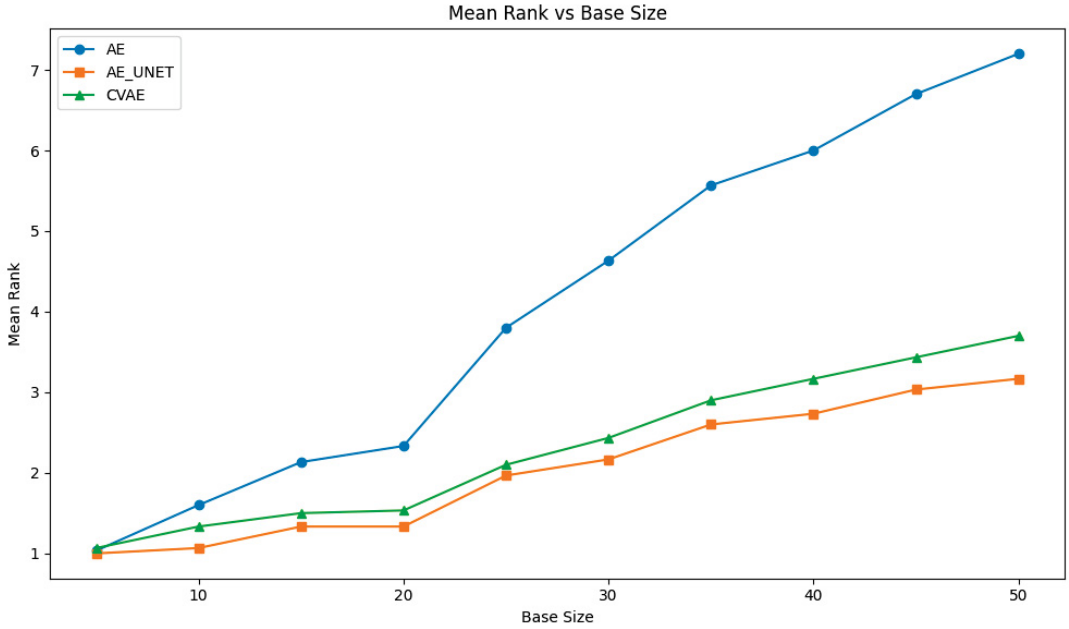


Figure 4.8: Mean rank of facial identification using CelebA.

In Figure 4.9, we can observe the maximum rank found for each model. Interestingly, the CVAE showed the worst result in this metric, indicating that at least one generated image had very poor quality. This again supports the idea that the CVAE's performance could be improved with more fine-tuned preprocessing parameters. The presence of an outlier with a high rank could also suggest that while the overall model performance is strong, it still struggles with certain images.

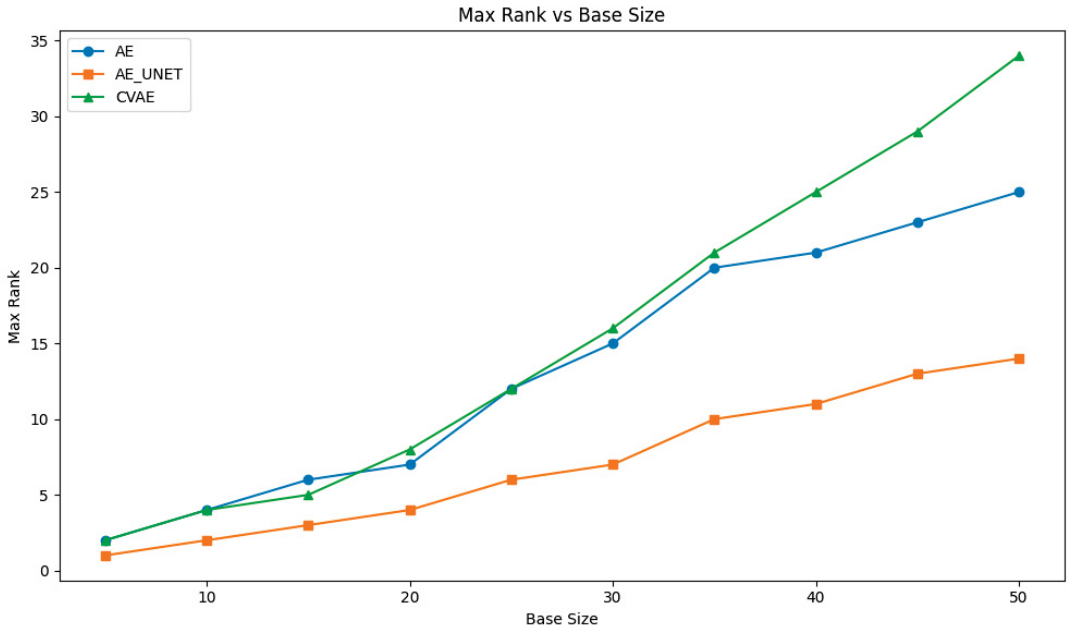


Figure 4.9: Max rank of facial identification using CelebA.

The same procedure was reproduced using the CUHK Student database. Figures 4.10 and 4.11 illustrate both the accuracy and the number of correctly identified images as a function of the database size. The simple autoencoder failed to correctly identify almost every face (only one face was correctly identified), which indicates the overfitting of the model to the CelebA database.

In contrast, the proposed models demonstrated great improvement when applied to the CUHK Student database. The proposed AE achieved an accuracy of 62% in the biggest database, while the CVAE model reached approximately 46%.

The improved performance of the proposed models on the CUHK Student database underscores their adaptability to unseen sketch styles. It also suggests that these models are less likely to overfit a specific dataset, making them more suitable for practical applications where data variability is a serious concern.

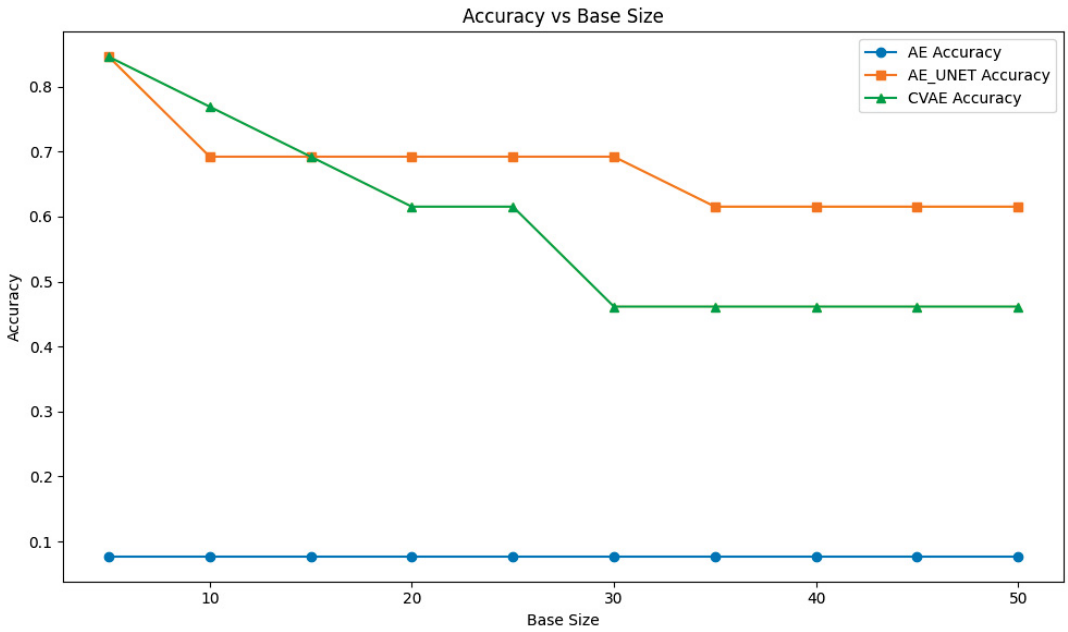


Figure 4.10: Accuracy of facial identification using CUHK.

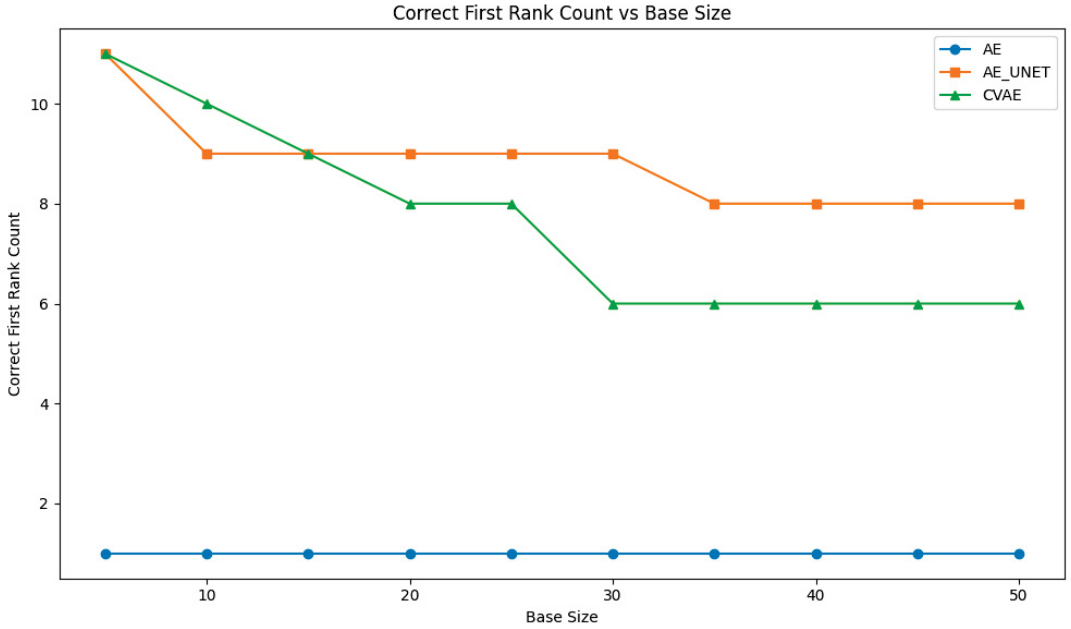


Figure 4.11: Quantity of correctly identified faces using CUHK.

The mean rank shown in Figure 4.12 is very similar to the one observed in the CelebA evaluation. However, the simpler autoencoder performed worse (16.77), while the proposed AE (4.54) and CVAE (5.54) were able to maintain consistent performance.

Figure 4.13 presents the maximum rank observed in the CUHK Student database evaluation. In this case, the CVAE exhibited a smaller maximum rank of 28, indicating better performance. On the other hand, the simpler autoencoder displayed a much worse result, with a higher maximum rank of 37.

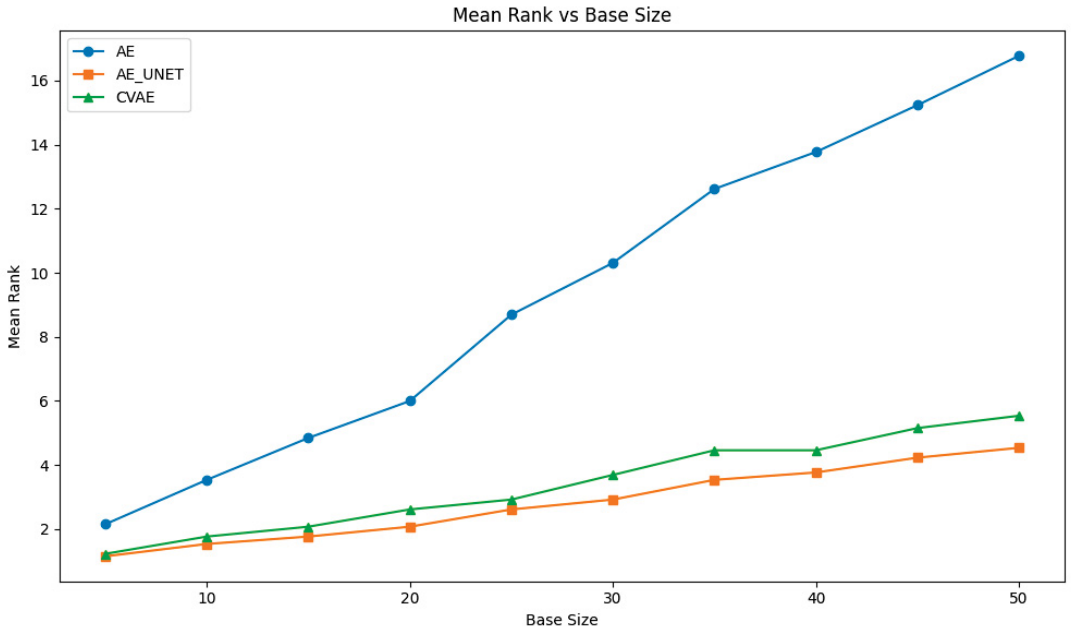


Figure 4.12: Mean rank of facial identification using CUHK.

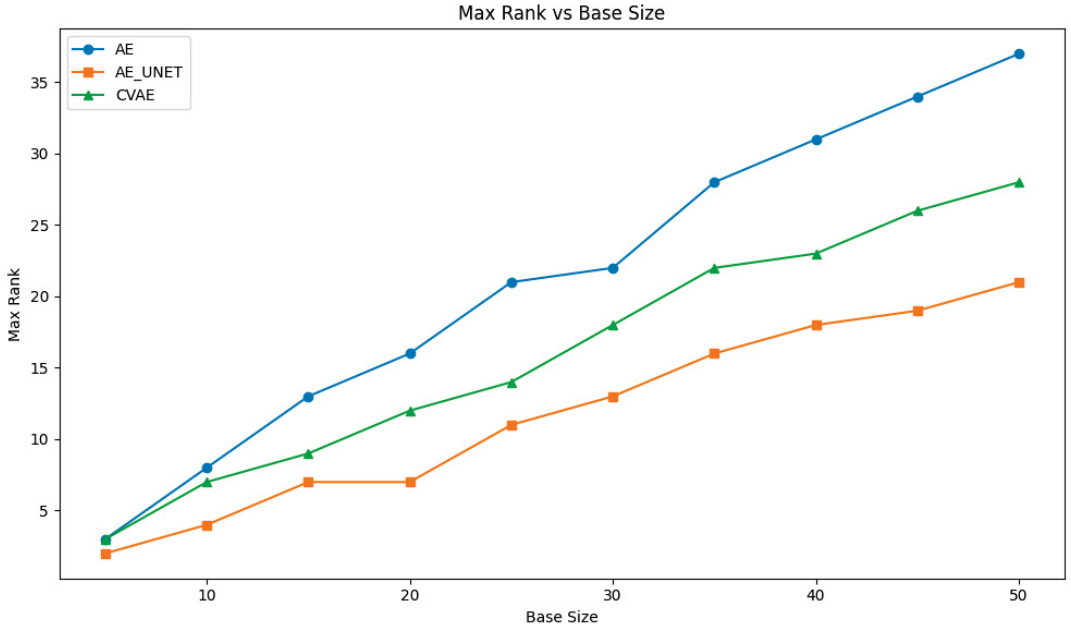


Figure 4.13: Max rank of facial identification using CUHK.

From a subjective analysis from the human point of view, the images generated from the CelebA database presented more similar features to their original counterparts than those from the CUHK Student database. However, when using the FaceNet Euclidean distance metric, the CUHK-generated images were easier to identify, or at least the generated images presented more stable qualities.

Even though the proposed AE presented better identification results compared to the CVAE, it is important to address the advantages of using the CVAE, especially in scenarios where the AE-generated image fails to capture specific features such as skin color, hair color, facial expressions, or even gender. The ability of the CVAE to incorporate conditional inputs allows it to generate images with more specific attributes.

In table 4.2 we can check the identification values.

4.4 DIGITAL SAMPLES

Previous evaluations focused solely on hand-drawn images created with pencil and paper. In this section, we extend our evaluation to include images generated using digital sketches with varying levels of detail. The process to generate the images is the same. Figures 4.14 and 4.15 showcase faces generated from a range of digitally drawn sketches, illustrating the model's adaptability to different artistic styles.

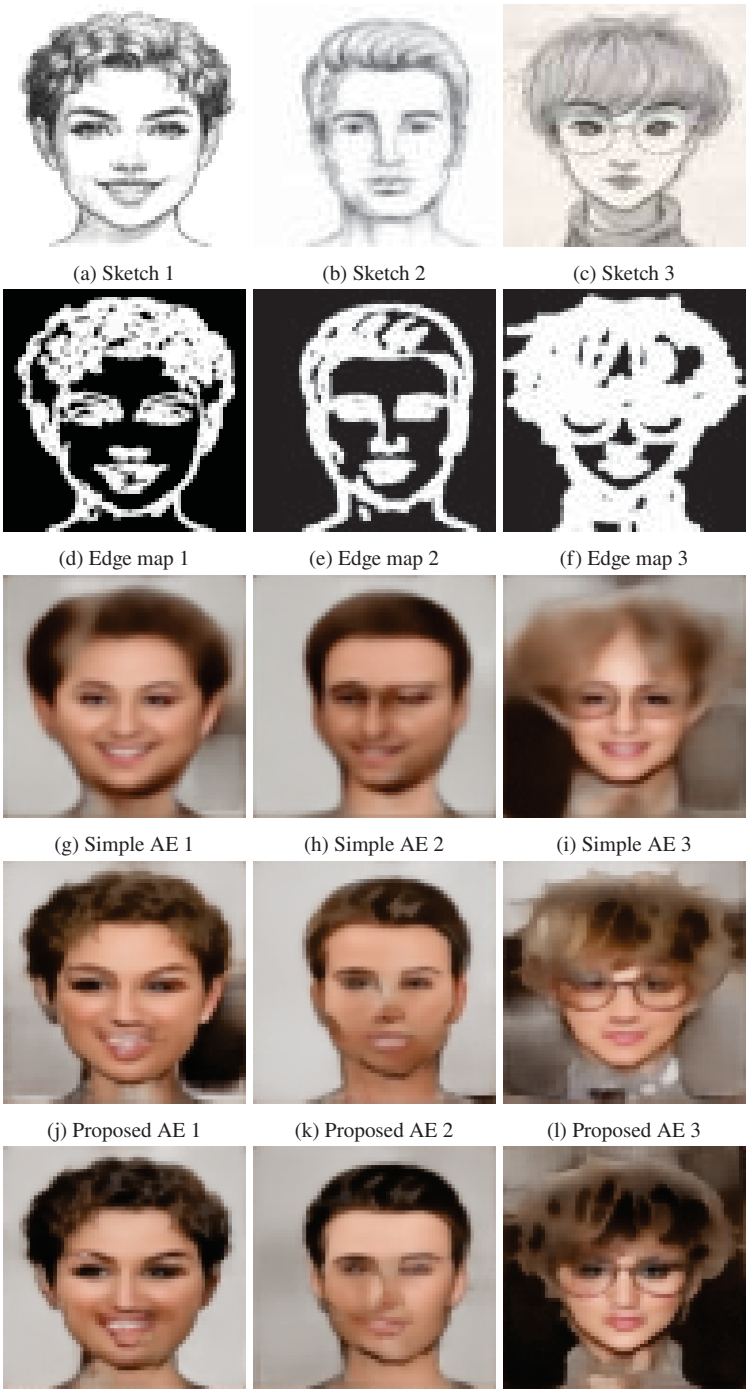
The styles of the digital sketches vary significantly. Some sketches are very realistic, capturing complex details, while others are more minimalistic, focusing on essential facial features with fewer strokes. Despite these variations, the proposed models were able to generate images that captured some notable features of the faces depicted in the sketches.

Dataset	Model	Metric	5	10	20	30	40	50
CelebA	Simp. AE	Accuracy	0.97	0.63	0.53	0.33	0.30	0.30
		Correct Count	29	19	16	10	9	9
		Mean Rank	1.03	1.60	2.33	4.63	6.00	7.20
		Max Rank	2	4	7	15	21	25
	Prop. AE	Accuracy	1.00	0.93	0.77	0.63	0.63	0.53
		Correct Count	30	28	23	19	19	16
		Mean Rank	1.00	1.07	1.33	2.17	2.73	3.17
		Max Rank	1	2	4	7	11	14
	CVAE	Accuracy	0.93	0.80	0.80	0.57	0.50	0.50
		Correct Count	28	24	24	17	15	15
		Mean Rank	1.07	1.33	1.53	2.43	3.17	3.70
		Max Rank	2	4	8	16	25	34
CUHK	Simp. AE	Accuracy	0.08	0.08	0.08	0.08	0.08	0.08
		Correct Count	1	1	1	1	1	1
		Mean Rank	2.15	3.54	6.00	10.31	13.77	16.77
		Max Rank	3	8	16	22	31	37
	Prop. AE	Accuracy	0.85	0.69	0.69	0.69	0.62	0.62
		Correct Count	11	9	9	9	8	8
		Mean Rank	1.15	1.54	2.08	2.92	3.77	4.54
		Max Rank	2	4	7	13	18	21
	CVAE	Accuracy	0.85	0.77	0.62	0.46	0.46	0.46
		Correct Count	11	10	8	6	6	6
		Mean Rank	1.23	1.77	2.62	3.69	4.46	5.54
		Max Rank	3	7	12	18	23	28

Table 4.2: Identification performance for CelebA and CUHK from a dataset ranging from 5 to 50 images

However, not all generated images were flawless. For instance, the second sketch in Figure 4.14 resulted in a face with distorted features, particularly the nose. This distortion can be attributed to the way the edge map detected the nose area. Because of the shadow, the model seems to misinterpret the nose pointing direction, creating a distorted nose when compared to the rest of the image. Similarly, the third sketch in Figure 4.15 generated a face where the neck appears to blend into the background. This issue arose because the sketch and its corresponding edge map did not clearly define the neck, leading to an ambiguous interpretation by the model.

Despite these imperfections, the overall quality of the generated images was satisfactory. The models demonstrated the ability to translate the essence of the digital sketches into photorealistic images.



(a) Sketch 1 (b) Sketch 2 (c) Sketch 3

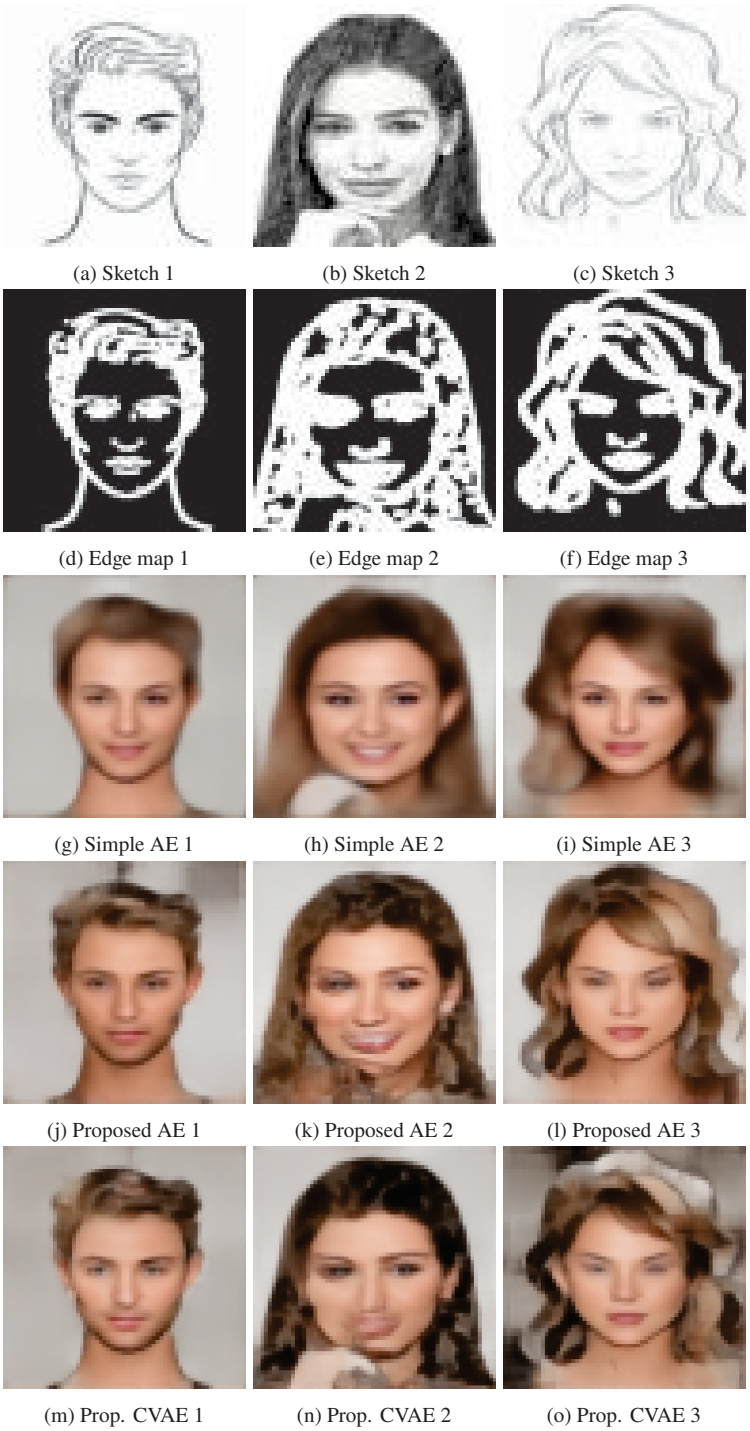
(d) Edge map 1 (e) Edge map 2 (f) Edge map 3

(g) Simple AE 1 (h) Simple AE 2 (i) Simple AE 3

(j) Proposed AE 1 (k) Proposed AE 2 (l) Proposed AE 3

(m) Prop. CVAE 1 (n) Prop. CVAE 2 (o) Prop. CVAE 3

Figure 4.14: Digital image evaluation 1



(a) Sketch 1 (b) Sketch 2 (c) Sketch 3

(d) Edge map 1 (e) Edge map 2 (f) Edge map 3

(g) Simple AE 1 (h) Simple AE 2 (i) Simple AE 3

(j) Proposed AE 1 (k) Proposed AE 2 (l) Proposed AE 3

(m) Prop. CVAE 1 (n) Prop. CVAE 2 (o) Prop. CVAE 3

Figure 4.15: Digital image evaluation 2

The edge map parameters in these images varied a lot due to the diverse styles, dimensions, and strokes used in the sketches. The variations in the sketches required different parameter settings to achieve optimal edge detection and subsequent image generation. In some instances, particularly on sketches with ambiguous or poorly defined features, the generated images did not meet the desired quality. This issue was evident in the eye area, where certain sketches led to poor interpretations and, consequently, lower-quality generations. For example, sketches with minimal detail often resulted in very blurry or low-quality images.

Some sketches required significant fine-tuning of the preprocessing parameters to achieve a satisfactory result. The adjustments included tweaking the edge detection thresholds and the Gaussian filter size, which handles different stroke styles and levels of detail in the sketches. Despite these adjustments, there were cases where the generated images did not accurately capture the intended features.

While the proposed models demonstrated a capability to handle a wide range of digital sketch styles, certain limitations were observed. Sketches with insufficient detail or poorly defined features often resulted in inferior generated images. The need for precise parameter adjustments in the preprocessing step indicates that further work is required to enhance the model's robustness.

4.5 CONDITIONAL ADJUSTMENTS

Conditional inputs enhance the model's ability to interpret and generate specific features such as skin tone, hair color, and facial expressions. By providing these attributes as part of the input, the model can produce better guided and realistic representations of the subject. This becomes useful when the sketch alone does not convey all the necessary details.

Figure 4.16 illustrates the generated images created by interpolating the conditional inputs. The conditionals are binary values (0 or 1). However, the interpolation ranges from -2 to 2, and the model could still return a good interpretation of these inputs. The only difference between each generated image is the associated attribute. For instance, the smiling attribute shown in the first sequence might be detectable from the sketch alone, while attributes such as a pale face or blond hair in the second and third sequences are inferred by the model based on the provided conditions. This demonstrates the model's capability to supplement missing details using conditional information, thus reducing potential biases and improving accuracy.

We can also verify the model's ability to disentangle these features by modifying the input conditions and observing the changes in the generated images. When a specific attribute is altered, only the corresponding feature in the image is modified, while other aspects remain unchanged. This function is useful for making final adjustments to the generated image, ensuring it aligns more closely with the user's description.

This ability to fine-tune the generated image based on conditional inputs is valuable in practical forensic investigations. For example, if a witness describes a suspect with specific facial features that are not clearly depicted in the sketch, these attributes can be added as conditional inputs to guide the model. As a result, the generated image will incorporate these details, leading to a more accurate and useful representation for identification purposes.

(a) Smiling

(b) Pale sking

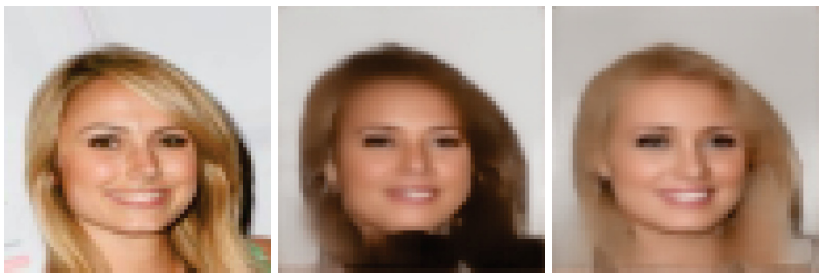
(c) Blond hair

Figure 4.16: Attribute interpolation across a range of values. The first image is the original, while the subsequent images show the effect of interpolating specific attributes from -2 to 2 in increments of 1

These conditional inputs are more helpful when the model struggles to identify specific attributes based on the sketches. For example, a sketch of a man with long hair might be incorrectly interpreted as a woman's sketch, as the model might associate long hair more commonly with women. In such cases, the conditional input ensures that the model generates the correct features.

Figure 4.17 illustrates this scenario. The proposed AE model misinterpreted the hair color, resulting in an inaccurate representation. In contrast, the CVAE model, aided by conditional input, accurately captured the correct hair color and other attributes.

The flexibility provided by conditional adjustments allows for the exploration of different scenarios. Investigators can generate multiple variations of a suspect's image by adjusting conditions such as hair color or facial expressions, thereby increasing the chances of matching the suspect in a database.



(a) Original image (b) Proposed AE (c) Proposed CVAE

Figure 4.17: Conditional evaluation

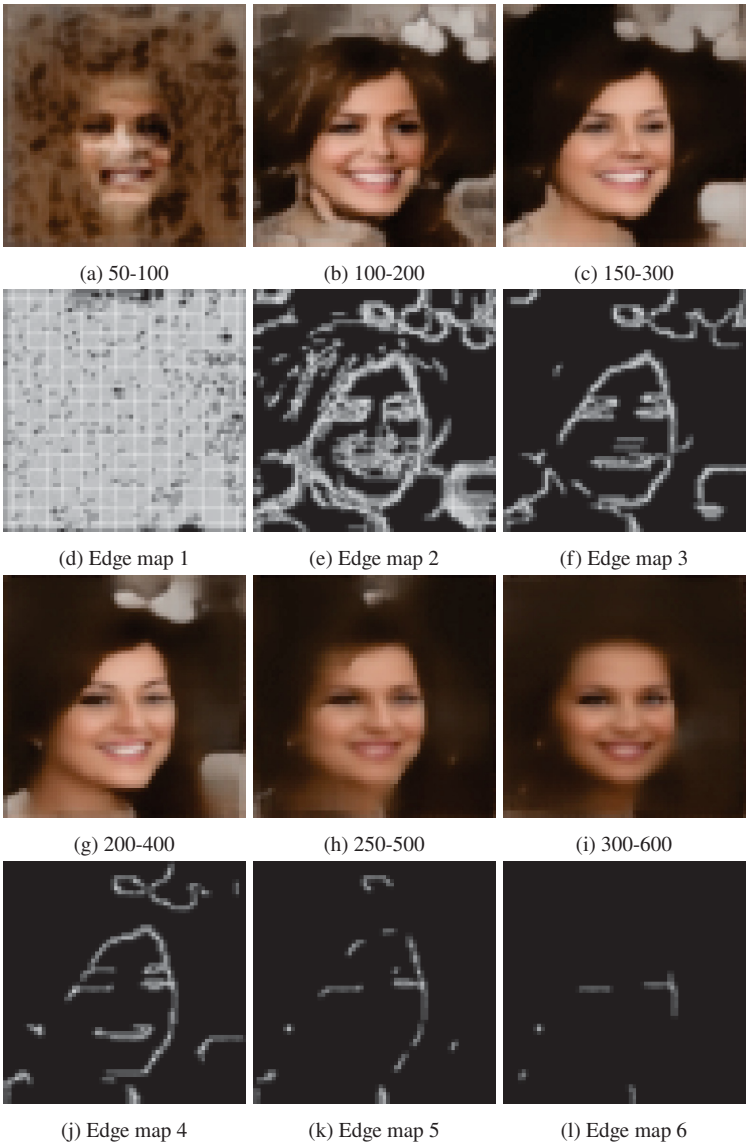
4.6 PREPROCESS FINE-TUNING

The preprocessing step during model training involves a stochastic approach to introduce variability in the training data. However, when generating a new image, it is interesting to define specific parameter values tailored to the details and style of the sketch. This fine-tuning process can strongly impact the quality of the generated image.

In Figure 4.18, we evaluate the effects of varying the Canny edge detection thresholds on a sketch. The edge map generated with a low threshold presents almost no information about the original image, capturing too many irrelevant details and noise. Conversely, a high threshold results in an edge map with very sparse information, only identifying some contours and eye locations. These extreme cases demonstrate that improper threshold settings can lead to poor quality in the generated images.

For a lower threshold, the generated image is very noisy and includes hallucinated features that do not help reconstruct the original face, making the final output very messy. On the other hand, a too-high threshold may miss essential details, resulting in an incomplete blurry reconstruction.

For instance, sketches with shadowing and finer details might require a higher threshold to avoid unnecessary noise, while sketches with bold, clear lines might benefit from a lower threshold to capture all the features.



(a) 50-100 (b) 100-200 (c) 150-300

(d) Edge map 1 (e) Edge map 2 (f) Edge map 3

(g) 200-400 (h) 250-500 (i) 300-600

(j) Edge map 4 (k) Edge map 5 (l) Edge map 6

Figure 4.18: Images generated on different canny thresholds (range from 50-100 (low-high) to 300-600)

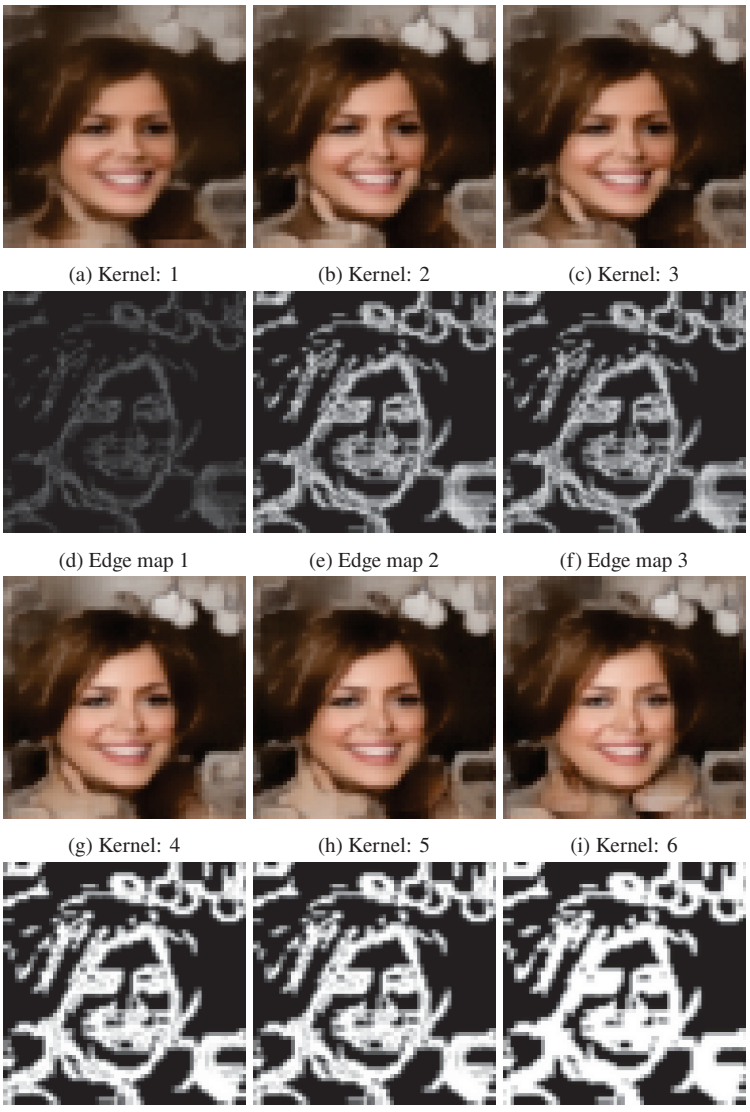
In Figure 4.19, we examine the effects of varying the Gaussian kernel size on the generated image. The Gaussian blur is applied during the preprocessing step to smooth the edges and reduce high-frequency noise. This step helps in creating a more stable and consistent edge map, which is important for the model to generate good-quality images.

We observe that the Gaussian kernel size is less sensitive compared to the Canny edge detection thresholds. Most kernel sizes result in a well-defined face, indicating that the model is relatively robust to changes in this parameter. However, the choice of kernel size can still impact the final image quality, especially in cases where the sketch has complex details.

For smaller kernel sizes, the smoothing effect is minimal, which may result in a slightly noisier edge map but retains more detailed features. Conversely, larger kernel sizes apply more noticeable blurring, which helps in reducing noise but may also obscure finer details.

In some instances, adjusting the Gaussian kernel size provides more notable improvements in the generated image quality. For example, sketches with a lot of high-

frequency noise or detailed textures might benefit from a larger kernel size to ensure smoother and cleaner edges. On the other hand, sketches with already clean and well-defined edges might only require a minimal blur to maintain clarity while still benefiting from some noise reduction.



(a) Kernel: 1 (b) Kernel: 2 (c) Kernel: 3

(d) Edge map 1 (e) Edge map 2 (f) Edge map 3

(g) Kernel: 4 (h) Kernel: 5 (i) Kernel: 6

(j) Edge map 4 (k) Edge map 5 (l) Edge map 6

Figure 4.19: Images generated on different Gaussian kernel sizes (range from 1x1 to 6x6)

5 CONCLUSIONS

In our work, we presented a novel CVAE architecture designed to generate photorealistic facial images from sketches and a few attributes. The model was trained only on artificially synthesized sketches using edge detection techniques among other preprocessing methods, and evaluated across different real sketches, demonstrating the capability to produce good-quality images across several styles. Despite being trained on unpaired images from the CelebA database and evaluated on unseen CUHK data, our models achieved good quality and coherence with the input, when compared to previous models trained and evaluated with paired images from CUHK database.

Unlike previous work, our model was rigorously evaluated on various styles of both digital and non-digital sketches. We also explored practical applications by using a limited database to identify the generated person within this database, thus validating the model's accuracy and determining the best metric for this task. Our evaluation included a combination of perceptual and pixelwise metrics. When evaluating with Facenet, the similarity of generated images using the CVAE is 56.8% better than the simpler AE, while the proposed AE is 53% better than the simpler AE, highlighting the strengths of the proposed models in generating facial images that are not only visually accurate but also recognizable using modern facial recognition models like Facenet.

The model was trained on 64x64 images, but it's possible to adjust the model for higher pixel resolutions, which was not done due to computational cost. For future improvements, we suggest enhancing the preprocessing pipeline by incorporating additional techniques such as HED, XDoG, or Photoshop to create more refined edge maps. Additionally, integrating an adversarial loss could further enhance image quality, provided it is applied carefully to preserve the essential features of the original image. These enhancements could help to further improve the model's performance and broaden its applicability.

These models are expected to enhance investigative efficiency by enabling quicker and more precise suspect identification, facilitated through seamless integration with digital recognition systems.

Bibliography

Canny, J. (1986). A Computational Approach to Edge Detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-8(6):679–698.

Chen, S.-Y., Su, W., Gao, L., Xia, S., and Fu, H. (2020). Deepfacedrawing: deep generation of face images from sketches. *ACM Trans. Graph.*, 39(4).

Chiu, C.-C., Ku, M.-Y., and Liang, L.-W. (2010). A Robust Object Segmentation System Using a Probability-Based Background Extraction Algorithm. *IEEE Transactions on Circuits and Systems for Video Technology*, 20(4):518–528.

Cunneen, M., Mullins, M., and Murphy, F. (2019). Autonomous Vehicles and Embedded Artificial Intelligence: The Challenges of Framing Machine Driving Decisions. *Applied Artificial Intelligence*, 33(8):706–731.

Deshpande, A., Lu, J., Yeh, M.-C., Chong, M. J., and Forsyth, D. (2017). Learning diverse image colorization. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2877–2885.

Devakumar, S. and Sarath, G. (2023). Forensic Sketch to Real Image Using DCGAN. *Procedia Computer Science*, 218:1612–1620.

Di, X. and Patel, V. M. (2019). Facial synthesis from visual attributes via sketch using multiscale generators. *IEEE Transactions on Biometrics, Behavior, and Identity Science*, 2:55–67.

Dowson, D. C. and Landau, B. V. (1982). The Fréchet distance between multivariate normal distributions. *Journal of Multivariate Analysis*, 12(3):450–455.

Edwards, K. M., Man, B., and Ahmed, F. (2024). Sketch2Prototype: rapid conceptual design exploration and prototyping with generative AI. *Proceedings of the Design Society*, 4:1989–1998.

Fareed, M. W., Bou Nassif, A., and Nofal, E. (2024). Exploring the Potentials of Artificial Intelligence Image Generators for Educating the History of Architecture. *Heritage*, 7(3):1727–1753.

He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep Residual Learning for Image Recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, Las Vegas, NV, USA. IEEE.

Higgins, I., Matthey, L., Pal, A., Burgess, C., Glorot, X., Botvinick, M., Mohamed, S., and Lerchner, A. (2022). Beta-VAE: Learning Basic Visual Concepts with a Constrained Variational Framework. In *International Conference on Learning Representations*.

Hu, M. and Guo, J. (2020). Facial attribute-controlled sketch-to-image translation with generative adversarial networks. *EURASIP Journal on Image and Video Processing*, 2020(1):2.

Isola, P., Zhu, J.-Y., Zhou, T., and Efros, A. A. (2017). Image-to-Image Translation with Conditional Adversarial Networks. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5967–5976. IEEE Computer Society.

Kebaili, A., Lapuyade-Lahorgue, J., and Ruan, S. (2023). Deep Learning Approaches for Data Augmentation in Medical Imaging: A Review. *Journal of Imaging*, 9(4):81.

Kingma, D. P. and Ba, J. (2015). Adam: A method for stochastic optimization. In *3rd International Conference for Learning Representations (ICLR)*.

Li, L., Tang, J., Shao, Z., Tan, X., and Ma, L. (2022). Sketch-to-photo face generation based on semantic consistency preserving and similar connected component refinement. *The Visual Computer*, 38(11):3577–3594.

Li, P., Li, B., and Li, Z. (2024). Sketch-to-architecture: Generative ai-aided architectural design. *ArXiv*, abs/2403.20186.

Li, Y., Chen, X., Yang, B., Chen, Z., Cheng, Z., and Zha, Z.-J. (2020). DeepFacePencil: Creating Face Images from Freehand Sketches. In *Proceedings of the 28th ACM International Conference on Multimedia*, pages 991–999.

Liu, R., Yu, Q., and Yu, S. X. (2020). Unsupervised sketch to photo synthesis. In *Computer Vision – ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part III*, page 36–52, Berlin, Heidelberg. Springer-Verlag.

Liu, Z., Luo, P., Wang, X., and Tang, X. (2015). Deep Learning Face Attributes in the Wild. In *Proceedings of International Conference on Computer Vision (ICCV)*.

Lu, Y., Wu, S., Tai, Y.-W., and Tang, C.-K. (2018). Image Generation from Sketch Constraint Using Contextual GAN. In Ferrari, V., Hebert, M., Sminchisescu, C., and Weiss, Y., editors, *Computer Vision – ECCV 2018*, pages 213–228, Cham. Springer International Publishing.

Moradzadeh Farid, N., Saeedi Fard, M., and Nikabadi, A. (2023). Face Sketch-to-Photo Translation Using Generative Adversarial Networks. *AUT Journal of Modeling and Simulation*, 55(1):39–52.

Nikkath Bushra, S. and Uma Maheswari, K. (2021). Crime Investigation using DCGAN by Forensic Sketch-to-Face Transformation (STF)- A Review. In *2021 5th International Conference on Computing Methodologies and Communication (ICCMC)*, pages 1343–1348.

Odaibo, S. G. (2019). Tutorial: Deriving the standard variational autoencoder (vae) loss function. *ArXiv*, abs/1907.08956.

Oktay, O., Schlemper, J., Folgoc, L. L., Lee, M., Heinrich, M., Misawa, K., Mori, K., McDonagh, S., Hammerla, N. Y., Kainz, B., Glocker, B., and Rueckert, D. (2022). Attention U-Net: Learning Where to Look for the Pancreas. In *Medical Imaging with Deep Learning*.

Osahor, U. and Nasrabadi, N. M. (2022). Text-Guided Sketch-to-Photo Image Synthesis. *IEEE Access*, 10:98278–98289.

Pang, Y., Lin, J., Qin, T., and Chen, Z. (2022). Image-to-Image Translation: Methods and Applications. *IEEE Transactions on Multimedia*, 24:3859–3881.

Ronneberger, O., Fischer, P., and Brox, T. (2015). U-net: Convolutional networks for biomedical image segmentation. In Navab, N., Hornegger, J., Wells, W. M., and Frangi, A. F., editors, *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, pages 234–241, Cham. Springer International Publishing.

Schroff, F., Kalenichenko, D., and Philbin, J. (2015). FaceNet: A unified embedding for face recognition and clustering. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 815–823, Boston, MA, USA. IEEE.

Sreedhar, K. (2012). Enhancement of Images Using Morphological Transformations. *International Journal of Computer Science and Information Technology*, 4(1):33–50.

Sun, J., Yu, H., Zhang, J. J., Dong, J., Yu, H., and Zhong, G. (2022). Face image-sketch synthesis via generative adversarial fusion. *Neural Networks*, 154:179–189.

Tschannen, M., Bachem, O., and Lucic, M. (2018). Recent advances in autoencoder-based representation learning. In *Third workshop on Bayesian Deep Learning (NeurIPS 2018)*.

Vincent, P., Larochelle, H., Bengio, Y., and Manzagol, P.-A. (2008). Extracting and composing robust features with denoising autoencoders. In *Proceedings of the 25th international conference on Machine learning - ICML '08*, pages 1096–1103, Helsinki, Finland. ACM Press.

Wang, L., Sindagi, V., and Patel, V. (2018). High-Quality Facial Photo-Sketch Synthesis Using Multi-Adversarial Networks. In *2018 13th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2018)*, pages 83–90.

Wang, Z., Bovik, A., Sheikh, H., and Simoncelli, E. (2004). Image quality assessment: From error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612.

Winnemöller, H., Kyprianidis, J. E., and Olsen, S. C. (2012). XDoG: An eXtended difference-of-Gaussians compendium including advanced image stylization. *Computers & Graphics*, 36(6):740–753.

Xia, W., Yang, Y., Xue, J.-H., and Wu, B. (2021). Tedigan: Text-guided diverse face image generation and manipulation. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2256–2265.

Xue, A. (2021). End-to-end chinese landscape painting creation using generative adversarial networks. In *2021 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 3862–3870.

Yu, J., Xu, X., Gao, F., Shi, S., Wang, M., Tao, D., and Huang, Q. (2021). Toward Realistic Face Photo–Sketch Synthesis via Composition-Aided GANs. *IEEE Transactions on Cybernetics*, 51(9):4350–4362.

Zhang, W., Wang, X., and Tang, X. (2011). Coupled information-theoretic encoding for face photo-sketch recognition. In *CVPR 2011*, pages 513–520. ISSN: 1063-6919.

Zhao, J., Xie, X., Wang, L., Cao, M., and Zhang, M. (2019). Generating Photographic Faces From the Sketch Guided by Attribute Using GAN. *IEEE Access*, 7:23844–23851.