Navigating the Latent

Exploring the Potentials of Islamic Calligraphy
with Generative Adversarial Networks

by

Arshia Sobhan Sarbandi

A thesis presented to OCAD University
in partial fulfillment of the requirements for
the degree of Master of Design in Digital Futures

Toronto, Ontario, Canada, 2021

Abstract

Islamic calligraphy is a substantial cultural element in the countries that share Arabic script. Despite its prevalence and importance, Islamic calligraphy has not significantly benefited from modern technologies, and there are considerable gaps between the affordances of digital tools and the subtle requirements of this domain. This project explores the use of Generative Adversarial Networks (GANs) as an option that can fill the gap between digital tools and fundamental aspects of Islamic calligraphy, and as a new tool that can expand the creative space of artists and designers who use Islamic calligraphy in their works. This study also promotes an informed approach toward using GANs with a focus on domain-specific requirements of Islamic calligraphy. Some of the potentials of using GANs in Islamic calligraphy are depicted through analyzing the results of a GAN trained on a custom-made and regularized dataset of *nasta'līq* script. The results are also used to make calligraphy pieces with a new mode of expression.

Acknowledgments

I acknowledge the ancestral and traditional territories of the Mississaugas of the Credit, the Haudenosaunee, the Anishinaabe and the Huron-Wendat, who are the original owners and custodians of the land on which this research was carried out.

I would like to express my deep gratitude to my supervisor, Dr. Adam Tindale, for his patient guidance and enthusiastic encouragement. His willingness to give his time so generously has been very much appreciated. My grateful thanks are also extended to Nicholas Puckett, my secondary supervisor, for his valuable advice, assistance and feedback throughout this project.

My sincere thanks to all faculty, staff and students of OCAD U Digital Futures Program, who create the strong interdisciplinary basis of this wonderful program, through which I was able to pursue this research.

My sincere and grateful thanks to Dr. Ashok Mathur for his generous help and support.

I would like to thank my dear friend, Mohamad Bozorgi, for all the wonderful discussions we had over the past few years that helped me establish the foundations of this project. I would also like to express my appreciation to Dr. Farhang Sahba and Dr. Ali Ashtari, who generously offered their help with the technical aspects of the project. My appreciation is also extended to my kind friend, Manijeh Maboudi, who helped me in creating my datasets.

Thanks to all my friends in the cohort, especially Manisha Laroia, Jun Li, and Sananda Dutta, the presence of whom made the passing of difficult times easier.

Finally, I want to thank my family and friends for their ongoing love and support.

This research was supported by OCAD U Graduate Student Project Grant.

Table of Contents

1 Introduction 1
1.1 The Path to This Study 4
1.2 Scope 6
1.3 My Position 7
1.4 Methods and Techniques 8
1.5 Challenges 9
2 Background and Context 11
2.1 Islamic Calligraphy 11
2.1.1 The Tool 12
2.1.2 How Islamic Calligraphy Is Practiced 13
2.1.3 Different Faces of Islamic Calligraphy in Modern Times 14
2.1.4 Knowledge of Islamic Calligraphy 15
2.2 Generative Adversarial Networks 16
2.2.1 Applications of GANs 16
2.2.2 Artistic Use of GANs 17
2.2.3 Working With GANs 18
3 Making a Calligraphy Dataset 19
3.1 Conceptual Challenge of Creating a Calligraphy Dataset 20
3.2 Technical Challenges of Creating a Calligraphy Dataset 23
3.3 Concept and Process 27
3.4 Nas3-10k Dataset 29
3.5 Nas4-60k Dataset 33
3.6 Summary 35
4 Training Process 37
4.1 Choosing the Network 38
4.2 Early Attempts in RunwayML 39
4.3 Adjusting Hyper-Parameters 43
4.4 Final Training Sessions 45
5 Results and Reflections 47
5.1 The Generation of Exact Letterforms 47

5.2 The Generation of Five and Six-Letter Combinations 48
5.3 Random Changes of Generated Results 48
5.4 Smooth Transitions Between Forms 49
5.5 What Is “New”? 51
5.6 Plasticity of Forms 52
5.7 Abstract Forms 56
5.8 Incorporating the Results in the Final Exhibited Work 58

6 Summary and Conclusion 61
6.1 Future Work 62

References 65

Table of Figures

Figure 1.1 Interactive installation “Invisible Kelk”, 2019 5

Figure 1.2 Results from my hand practice to create calligraphy using unconventional tools. 6

Figure 2.1 Right: A demonstration of different Islamic calligraphy scripts. Left: The measurement system using dots demonstrated in naskh script. Works by Habibollah Fazaeli (1922-1997)........ 12

Figure 2.2 The reed pen used in Islamic calligraphy. Photo by Aieman Khimji. Source: Wikimedia Commons......... 13

Figure 3.1 A cropped view of one of the 12 batches of data samples being prepared for Nas4-60 dataset, explained in 3.5......... 19

Figure 3.2 Examples of calligraphy pieces in different scripts and different compositions. Works by: Abu al-Ma’ali (d. 1209), Kamal al-Din (d. 1567), Mir Emad (1554-1615), Ala al-Din Tabrizi (1524-1593). Source: Wikimedia Commons. 22

Figure 3.3 A demonstration of the issues arising from size regularization of calligraphy pieces. Calligrapher: Mirza Gholamreza Esfahani (1830-1887)........ 24

Figure 3.4 Saxophone Dreams by Mike Tyka. Image source: https://miketyka.com 25

Figure 3.5 Samples of overlapping forms in calligraphy pieces in nasta’līq script (left) and thuluth script (right). Works by Fathollah Khan Jalali (d. 1918), Ali Badavi (d. 1940), Aziz al-Rafaei (d. 1934). 26

Figure 3.6 A comparison of the word “nasta’līq” written in different calligraphy fonts......... 28

Figure 3.7 In the reduced alphabet, letters are grouped based on their visual similarities and the simplest letter is kept. Also, letters with partial similarities of the initial and medial forms are grouped (highlighted with the same colour). 30

Figure 3.8 An example of creating alternative letterforms and possible elongations in the combination of letters mim, he and re. 30

Figure 3.9 Examples of manual spacing adjustments in Nas3-10k... 32

Figure 3.10 Random samples from Nas3-10k dataset 33

Figure 3.11 Random samples from Nas4-60k dataset 35

Figure 4.1 A schematic representation of different phases of network training and related activities. 37

Figure 4.2 StyleGAN2 network architecture... 39

Figure 4.3 Random samples from the first dataset used for training in RunwayML 40

Figure 4.4 Generated samples from the first training in RunwayML (left) and extracted calligraphic features (right). 41

Figure 4.5 Generated samples in RunwayML using Nas3-10k dataset and extracted calligraphic features. 42

Figure 4.6 Smooth transitions of letterforms in interpolation between points using the trained network in RunwayML. 42

Figure 4.7 An example of mode collapse and divergence in a training session 44

Figure 4.8 A comparison of randomly generated samples from networks trained on Nas4-60k in 512x512 px and 1024x1024 px with different training configurations.. 46

Figure 5.1 Random results from a network trained on Nas4-60k dataset for 1500 kimg, showing diversity in forms and extraction of fine calligraphic features. 47

Figure 5.2 Curated examples of generated five and six-letter combinations. 48

Figure 5.3 Examples of random changes in samples generated from fixed seeds. 49

Figure 5.4 A sequence of 48 samples generated through interpolation between two points of the latent space.. 50

Figure 5.5 Fine transition of letterforms in interpolation between close points in the latent space. 51

Figure 5.6 Interpolation between elongated and non-elongated forms. 51

Figure 5.7 Examples of plasticity in nasta'līq script. A: The elongated form of letter sin. B: The final forms of letter be. C: Initial form of letter he. Works by Mir Emad (1554-1615), Mirza Gholamreza Esfahani (1830-1887). 54

Figure 5.8 Left: Examples of generated samples. Right: The overlay of multiple samples generated from the latent space in the close proximity of the samples. 55

Figure 5.9 Curated results from earlier stages of training showing general visual features of the dataset without identifiable exact replication of letterforms. 57

Figure 5.10 Examples of abstraction in Islamic calligraphy. 58

Figure 5.11 Siyah-mashq pieces in nasta'līq script by Mirza Gholamreza Esfahani (1830-1887). 59

Figure 5.12 Still images of the final calligraphy pieces created using the components form the latent space interpolations of the trained network. Full pieces are accessible via: https://www.arshsobhan.com/latent 60

1 Introduction

An important technological turning point that had a significant influence on Islamic calligraphy and its relationship with technology was the advent of the printing press and movable type. The Printing press was practically put into use by the Ottomans in 1726 [1] and then followed into Iran in 1816 [2], around 300 years after its first use in Europe. This late adoption of a rather well-developed technology that was created for a different script, was the beginning of a deep discordance between Islamic calligraphy and modern technology. It led to the long-term issue of delaying the modification of the ever-evolving technology to meet the specific functional and aesthetic requirements of Islamic calligraphy. The introduction of each new tool created new issues for the use of Arabic script and often resulted in undesirable changes in the script to match those technologies rather than modifications to the technology to support the specific needs of the script [2]. Early fonts were created based on *naskh* script as this script was more readily adaptable to the new technology based on its structure. Other scripts, however, were much more complex to make horizontally partitionable glyphs that could then be put together to form words.

With the continuation of using separate glyphs to form combinations in digital typography, basic issues persisted in the use of Islamic calligraphy in computers. Although many improvements have happened over time, some of the most fundamental features of Islamic calligraphy are still not practically accessible in modern typography systems. Features such as multiple baselines, dynamic elongations, ascending baseline, proper positioning of the dots, proper spacing between letters and words, and plastic forms are either totally not available or not accessible via user-friendly processes in such systems.

These shortcomings have since been the subject of research [3][4]. An example of recent efforts is DecoType[1] by Thomas Milo which addressed some of the issues but the solution did not gain widespread public use. Another example is the research by Sahar Afshar that aims toward making dynamic elongations accessible through variable fonts [5]. A look into other digital tools shows how

[1] https://www.decotype.com

some of the common tools do not support the basic needs of Islamic calligraphy. An example is how digital pens do not support rotation of the nib due to their cylindrically symmetric structure.

The gap between Islamic calligraphy and modern technology has further widened due to a reluctance in the traditional calligraphy community to embrace the use of modern technology [6]. I have personally experienced the prevalence of this reluctance among calligraphy masters and some students during the years I spent learning calligraphy in The Society of Iranian Calligraphers (*anjuman-e khoshnevisan-e iran*) and later through many discussions with professional calligraphers. They fear that technology is in stark contrast with the spiritual nature of Islamic calligraphy. Others are afraid of automation and think that the skills that take years to acquire could be replaced with a push of a button. The reluctance of the classical school might itself be due to the fact that a smooth transition did not happen for Islamic calligraphy at the emergence of the printing press.

Be it the shortcomings of modern technology in supporting specific needs of Islamic calligraphy or the reluctance of classical school to adopt modern technology, it is hard to say if digital tools play a substantial role in augmenting artists and designers to expand their creative space. This gap has widened so much that today our expectations of modern technology to augment the creative space of Islamic calligraphy are not even clear anymore. What we, users of calligraphy in digital media, are more concerned with is the shortcomings of the technology. We are constantly trying to access, in digital media, the already available features of calligraphy in analog media.

What we know today as the traditional practice of Islamic calligraphy has benefited from technology in different ways during its development. The reed pen itself is natural that is cut using a sharp metal blade. Cutting the reed accurately so that it meets the expectations of the master calligrapher is thus dependent on the technology of making the blade. It is the same with the technology required for making the ink of the desired colour and the right viscosity, and the same for the making of a paper that is glossy enough to provide the desired traction for the reed without absorbing too much ink. Although we might think of such technologies as primitive, it is important to acknowledge that at some point in time in the history of Islamic calligraphy, these have been the most advanced technology available.

Presently, we are in the midst of another significant technological turning point with the emergence of Artificial Intelligence and the rapid development of different branches of it, especially deep neural networks. There have been significant advancements in this field and the use of these networks is becoming more prevalent in different domains every day. A family of deep neural networks are Generative Adversarial Networks (GANs) that have gained special attention in visual domains due to their generative features and have seen impressive improvements in recent years with growing applications in different fields. GANs are also used in artistic domains and have attracted many artist's attentions as it provides them with new modes of creative expression.

However, the main body of research in GANs is being developed primarily with a western-centric approach. Accordingly, GANs do not essentially support specific requirements of many different artforms like Islamic calligraphy that is not mainstream to where this technology is born and is being developed. These networks are moving fast towards being integrated into creative tools and are becoming publicly available in the form of commercial end-user software tools. This creates the urge to be more pro-active with this emerging technology before it gets deeply integrated into the tools that artists and designers use in its current form.

Despite the strong body of research around GANs and impressive recent advancements, the benefits of GANs for Islamic calligraphy are not necessarily spontaneous, nor obvious. Many GANs are being developed for various domains and most of these networks are designed to be used with data samples that have a radically different structure compared to Islamic calligraphy data. Finding the benefits and potentials requires domain-specific research on GANs to discover meaningful connections to Islamic calligraphy and to shape this tool for subtle requirements of Islamic calligraphy and the unique feature structure of data in this domain.

This study is to demonstrate the potentials of GANs in the domain of Islamic calligraphy and to discover possibilities of what GANs can contribute to this domain. Further, the study asks "how utilizing this technology can expand the creative and practical space of Islamic calligraphy." Through the study, I find meaningful connections between the general affordances of GANs and the domain-specific requirements of Islamic calligraphy and demonstrate the potential synergies between the two fields. This includes a hands-on approach to use a GAN architecture for creating calligraphy pieces with a new mode of expression that shows how using GANs can expand the creative space of artists and designers who use Islamic calligraphy. Through the exploration, I outline different conceptual connections between GANs and Islamic calligraphy and speculates about possible future applications of these connections. As having a well-designed dataset is a necessity to work with GANs purposefully, and because no such regularized dataset is available for

Islamic calligraphy, I undertake the process of making regularized datasets from scratch that fit into my design and purpose of working with GANs.

My hope is that this project can work as a proof of concept to show that strong connections exist between GANs and Islamic calligraphy. These connections can encourage further research not only in GANs but also in Islamic calligraphy itself, to better understand the knowledge structure of this domain. The learning here is in the form of a comparison of the knowledge that the GANs extract from the dataset as compared to our intuitive knowledge of calligraphy. Hopefully, these two in parallel can lead to a reconciliation between Islamic calligraphy and new technologies in this critical turning point.

1.1 The Path to This Study

There has not been a single path that led me to this study. But the most important motive behind it has been my strong passion for Islamic calligraphy. I started learning *nasta'līq* script in middle school and continued practicing it along with *shekaste-nasta'līq* script for more than seven years under the supervision of great calligraphy masters in The Society of Iranian Calligraphers (*anjuman-e khoshnevisan-e iran*) in my hometown. Since then, I have been greatly fascinated by this artform and this passion has been the driver that led to this project.

I continued my career as a graphic designer in the years that followed. With a desire to incorporate calligraphy in my works, I experienced the shortcomings of digital technology in supporting some of the most fundamental features of Islamic calligraphy and how my ideas were not supported by the available tools. I always questioned the role of technology in Islamic calligraphy and how digital tools were not adding to my creative space. There was always a gap between my imagination and what I was able to create.

My questions lingered and the gap persisted. In the first term of Digital Futures at OCAD, when I was working on my project "Invisible Kelk", I had a vivid idea about exploring the latent continuum between forms in calligraphy. I failed to make the visuals based on my imagination and subsequently changed the idea to basic transformations that I could achieve with the conventional tools (Figure 1.1). However, I was left with a strong vision that motivated me to find a new tool that can support my imagination.

Figure 1.1 Interactive installation "Invisible Kelk", 2019

I had the first sparks when I started to learn about neural networks and deep learning that this new technology is related to my questions and concerns. Hence, it started to broaden my knowledge, specifically in GANs, through various resources to get a deep understanding of how they work and what their dimensions are. My intuitions were only reinforced the deeper I delved into the subject and learned about the continuous nature of the latent space in GANs. Similar to classification neural networks, GANs can estimate the probability distribution of a dataset if trained with a tuned dataset. But GANs also have an extra feature of generating new samples from the continuous learned distribution. This made me determined to explore the connections between GANs and Islamic calligraphy.

Despite the strong intuitions, I was not able to explicitly express these connections or to contemplate the synergies of the two domains. That created the urge to find a mediative conceptual framework to translate concepts across the two fields to see the connections more clearly. I explored different frameworks in an independent study during the Spring-Summer term that led to considering knowledge as the connector. I also went back to hand practice using different tools to investigate my intuitive visual and practical knowledge of calligraphy in light of my relationship with the tool (Figure 1.2). This helped me better understand my relationship with GANs as a tool and how it can change my space of creativity.

Figure 1.2 Results from my hand practice to create calligraphy using unconventional tools.

All in all, this project is a result of a long period of being consumed with the gaps between Islamic calligraphy and digital tools, and the accumulation of visions about creating calligraphy in a new way that reveals the unseen potentials of this artform. Looking back, I have always had a great fondness for Islamic calligraphy and have been concerned with the future of it as a designer and as a calligrapher. I feel lucky to be driven by my passion and hope to contribute to a flourishing future for this domain.

1.2 Scope

In this project, I stayed with the aspects that I felt I can contribute to best regarding my background and skills. I narrowed down my study to the analytical aspects of exploring the potentials of GANs for Islamic calligraphy with a focus on understanding the technology. It is my fervently held belief that the socio-cultural aspects of using modern technology in Islamic calligraphy, as well as socio-political aspects of the development of new technologies, are of utmost importance with this regard and should be explored by people in the related disciplines if a fruitful interplay of the two fields is intended.

Regarding Islamic calligraphy, although the framework used to connect the concepts between the two fields can be used for any script or style in Islamic calligraphy, I have mostly focused on *nasta'līq* script in this project. This is the script that I have practiced myself and therefore I have the most knowledge about compared to other scripts. This makes the process of visual evaluation of the results possible for me using my knowledge of this script. In addition, preparing datasets for other scripts was not practical due to the time-consuming process of the dataset creation and therefore did not fit into the time frame of this study.

1.3 My Position

I undertake this project with a fervent belief that there are strong connections between GANs and Islamic calligraphy. Although the roots of this belief are intuitive, I dedicate this project to deepen my understanding of GANs to depict these connections and demonstrate the potentials of using GANs in Islamic calligraphy. I believe meaningful connections can be demonstrated through a conscious and purposeful approach towards GANs that focuses on fundamental aspects of Islamic calligraphy.

An important aspect of the purposefulness of the approach, I believe, is to be aware of the over-excitement state that surrounds the artistic use of GANs. There is still a newness and fascination about what GANs create and how they unleash new modes of creativity in visual domains. Hence, any random training using arbitrary datasets of Islamic calligraphy would most probably lead to the generation of very exciting results. Such a passive use of GANs in Islamic calligraphy would be interesting, but not essentially consequential.

Rather, an informed approach should rely on a deep understanding of GANs to design experiments that aim for specific aspects of Islamic calligraphy instead of passively adopting the technology. This way we can investigate how the basic affordances of GANs can support salient features of Islamic calligraphy that are hardly accessible in digital tools. The results of this approach would hopefully be significant contributions the scope of which is more than a personal experience and can be used on a greater scale.

Although this study pursues the idea of utilizing new technology in Islamic calligraphy, it is not to suggest that we change the traditional practice, nor to automate the tasks and processes of traditional calligraphy. I believe Islamic calligraphy can, and will, live in its most traditional and original form. What this study focuses on is the fact that our lives are intertwined with modern technology and the technology has seriously affected the presence – and the quality of what is presented – of Islamic calligraphy. As the influence of technological advancements on Islamic calligraphy cannot be ignored, this study promotes a proactive and conscious mindset towards these technological advancements so that the delicate aspects of this artform are taken into consideration in the development of the technologies used in it. Consequently, such an informed approach can lead to the expansion of the creative space for the domain of Islamic calligraphy and can help to unleash the hidden potentials of this artform.

1.4 Methods and Techniques

A major source of inquiry in the topics related to Islamic calligraphy has been my interviews with professional calligraphers who have modern approaches towards Islamic calligraphy. The topic of these discussions has been mostly around the use of technology in Islamic calligraphy and also about our implicit knowledge of calligraphy; the ways we acquire the knowledge and how visual and practical knowledge of calligraphy affect creation and creativity in this domain.

In my thought process to see the connections between Islamic calligraphy and GANs, I use knowledge as the bridge that overlaps with both fields through which concepts can be translated across the two domains. This translation takes place mostly through modelling calligraphy in terms of data, information and knowledge. This makes it possible to think about concepts of Islamic calligraphy in the light of the task-based nature of GANs and leads to the design of my experiment of using a GAN architecture for Islamic calligraphy.

For the experiment, I design and create calligraphy datasets that fit into the purpose of my GAN training. I use a method that makes the creation of regularized datasets possible in a feasible way. This method includes the making of letter combinations in *nasta'līq* script using a calligraphy font with a reduced alphabet to create maximum diversity and minimum redundancy of the forms. In this method, human supervision is used to make necessary adjustments according to the design of the datasets, and to compensate for the shortcomings of the font.

I use the calligraphy datasets for training a GAN architecture. In the process of training, I iterate over the hyper-parameters of the network to achieve stable training conditions and to achieve desirable results. Both in the process of stabilizing the networks and in final training sessions, I train the network from scratch as well as using a technique called transfer learning. In this technique, the network is trained with the custom dataset in a previously trained network. This technique helps with achieving results more quickly and also is useful when achieving stable training conditions is troublous.

The results of the final trained networks are analyzed in the light of conceptual explorations to demonstrate the connections between GANs and Islamic calligraphy. These results are also used as building blocks to create digital calligraphy pieces that reveal a new mode of expression in Islamic calligraphy. The pieces are digitally composed using video editing techniques and displayed both virtually and physically using video projection.

I also speculate about the future implications of using GANs in Islamic calligraphy. This is in the form of extrapolating GAN applications, given the connections to Islamic calligraphy, to picture how GANs can be used in various functional and artistic cases to augment the use of Islamic calligraphy. My use of speculation in this project is close to the notion of speculative design by James Auger and is to speculate based on logical iterations of GANs to meet the complex and subtle requirements of Islamic calligraphy [7].

1.5 Challenges

There is a lack of resources both in artistic uses of GANs and in analytical aspects of Islamic calligraphy. Most of the works in Islamic calligraphy are descriptive and, as valuable as they are, they do not help much in understanding the structural aesthetics of this art form [8]. Also, in GANs, although artists have been increasingly using GANs in the past few years, still the main source of acquiring knowledge about GANs and working with them is computer science literature.

To obtain an accurate knowledge of GANs, I have investigated original GAN papers and related literature in computer science. This has been a steep learning curve for me to understand the basic concepts of GANs through its technical language of statistics and probability theory. I have also relied on watching talks and lectures by pioneer figures in these fields since content presented in these forms provides high-level knowledge without getting too deeply into the mathematical aspects and focuses on main ideas.

Regarding practical aspects, one serious challenge is the availability of GANs and the methods of working with them. There are some tools available that provide limited end-user access for working with GANs. However, a more serious hands-on use of GANs essentially requires working directly with the code and needs a level of programming skill. Even with the code itself, although some famous GAN architectures are publicly available, working with these networks is not equally easy and depends on the implementation of the code.

Finally, training GANs requires high computational power. This fact makes GAN training both costly and time-consuming. Training sessions are usually performed on powerful GPUs that I do not have access to locally. Inevitably, I use cloud computing services that give access to such GPUs and charge per hour of usage. This takes away the option of iterating over different networks and different training configurations and limits the scope of my project.

2 Background and Context

2.1 Islamic Calligraphy

The development of Islamic calligraphy has its roots deep in the religion and the practice of writing the Quran[2]. However, the realm of Islamic calligraphy as an artform is not limited to religious subjects and encompasses a wide range of works in various contexts and is practiced in different countries with languages written in the Arabic script. Here in the title, Islamic means related to the culture of the lands in which Islam has had a major presence over the past fourteen hundred years [9].

Over centuries, Islamic calligraphy was developed into well-established scripts – also referred to as styles[3]. These scripts are commonly known as the Six Pens: *naskh*, *thuluth*, *tawqī'*, *ruqā'*, *muhaqqaq*, and *rayḥān*. The regularization of these scripts is mostly credited to Ibn Muqla and his follower, Ibn al-Bawwab, using a system of rhomboidal dots that is used to measure the proportions of letters and words which is still being used in teaching Islamic calligraphy. In addition to the Six Pens, regional scripts were also created in different areas, such as *ta'līq*, *nasta'līq* and *shekaste-nasta'līq* in Iran, and *dīvanī* in the Ottoman empire (Figure 2.1).

[2] Quran is the holly book of Islam.

[3] To avoid confusion, in this text I refer to "scripts" as different styles of Islamic calligraphy that have different rules and letter proportions, such as *naskh*, *thuluth*, etc., and I refer to "styles" as style differences in each script. For example, works of Mirza Gholamreza Esfahani and Mir Emad are considered to have different "styles" in *nasta'līq* "script."

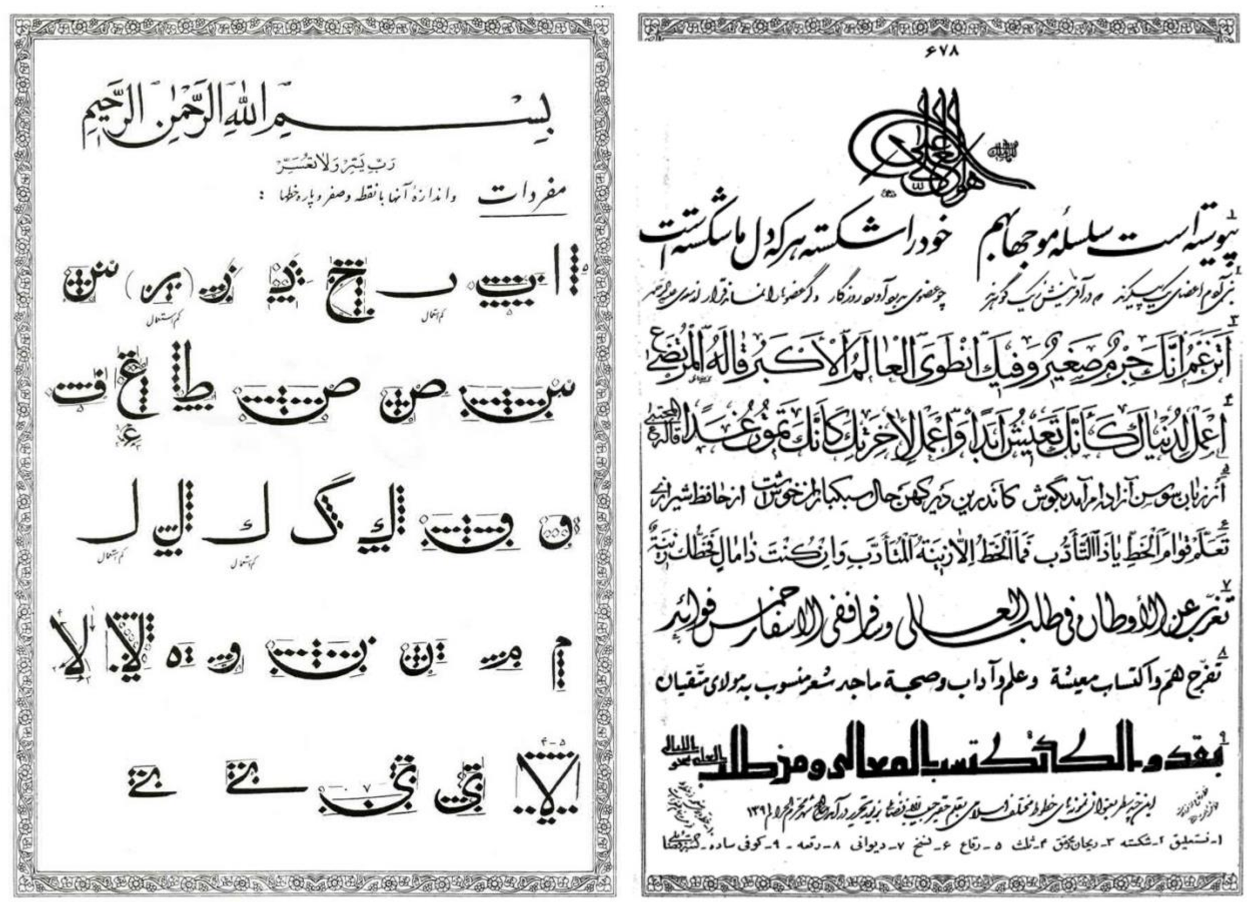

Figure 2.1 Right: A demonstration of different Islamic calligraphy scripts. Left: The measurement system using dots demonstrated in naskh script. Works by Habibollah Fazaeli (1922-1997).

These scripts are different in terms of popularity, prevalence and usage. For example, *nasta'līq* is the predominant script in Iran and Pakistan, whereas *naskh* and *thuluth* are more popular in Arab countries. Some scripts, such as *Ta'līq*, are less popular today and not many calligraphy masters practice them. Also, in terms of usage, a script like *naskh* had been mostly used for regular transcriptions where legibility is a priority, but *thuluth* had been regularly used for architectural inscriptions due to its decorative trait.

2.1.1 The Tool

The implement used in all the aforementioned scripts is traditionally a reed pen or *qalam* (Figure 2.2). Although many different types of reeds have been used, the basic structure of the reed pen is the same. The dried reed is cut to form a slender surface and the nib is cut at an angle. Different angles are used for different scripts and calligraphers may cut the nib in slightly different angles to fit their hand and how they hold the reed pen. Different sizes of reed pens are used to make calligraphy on different scales. As mentioned before, the rhomboidal dots created with the same reed pen are used to measure letters' proportions.

Figure 2.2 The reed pen used in Islamic calligraphy. Photo by Aieman Khimji. Source: Wikimedia Commons.

2.1.2 How Islamic Calligraphy Is Practiced

The very basic method through which Islamic calligraphy is taught and learned is practice [9][10]. This act of practicing is mostly in the form of copying samples that a master calligrapher gives to the students. These samples start from basic forms and isolated letters, then letter combinations, then word combinations as sentences or verses, and eventually, it proceeds to more complex compositions. In a typical calligraphy session, a master calligrapher teaches students either one-on-one or in small groups. The master writes a calligraphy sample (*mashq*) while the student(s) watch to learn how it is done using the reed pen. Students then take the sample home to practice it by trying to copy it as closely as possible through repetition. The next session starts with checking the results by the master calligrapher and identifying issues. This is usually done by writing on the student sample, using a different colour, to clearly show the differences between what the created form is and what it should be.

Although different masters use slightly different methods, and these methods might vary in different regions, the fundamental element in all of them is still practicing through repetition and copying so that eventually the student can start writing new pieces with an acceptable quality without having a reference sample. Through this process, students look at many pieces of their masters and other famous masters, they make many pieces and they are corrected many times before they can make new pieces from their own imagination.

This process builds different levels of knowledge and competence in a calligrapher. Baba Shah Esfahani[4] mentions three levels of competence in calligraphic practice [10]. The first is visual practice (*mashq-i naẓarī*), in which the student studies masters' calligraphy pieces. The second is pen practice (*mashq-i qalamī*), which entails practicing by copying from master's writing, from single letters to words, sentences and other compositions. These two steps usually take a long time. But eventually, the student is ready to proceed to the third stage, imaginative practice (*mashq-i khayālī*.) Imaginative practice goes beyond copying and requires the student to create new pieces. using the visual and practical knowledge acquired in the first two stages, the student is able to generalize that knowledge to create new pieces.

2.1.3 Different Faces of Islamic Calligraphy in Modern Times

The practice of traditional scripts and styles of Islamic calligraphy is still very popular in many countries. In Iran, for example, calligraphy masters and students are at work in the Society of Iranian Calligraphers (*anjuman-e khoshnevisan-e iran*) with branches all over the country where traditional scripts are practiced. Also in Turkey, despite the adoption of Roman script in 1928, Islamic calligraphy is still practiced under the master-apprentice system [9, pp. 596–597]. The popularity of Islamic calligraphy goes even beyond the Arab world, Iran, Turkey and other regions that have historically been directly influenced by Islam in the past fourteen centuries, and reaches Europe, the United States, Western China, and Indonesia [9, pp. 601–603].

In addition to well-established traditional scripts, a few new scripts have also emerged during the past years. An example is *moalla* script created by the Iranian calligrapher Hamid Ajami. This new script has visual similarities to thuluth and is mostly used as a display script. The creation of a new script is a challenging creative task and is a rare instance. Ajami believes the innovation of a new

[4] Baba Shah Esfahani is one of the most famous Iranian calligraphers in the 16th century.

script is an intuitive process and not intentionally possible [11]. However, these contemporary scripts have not gained the same popularity compared to traditional ones.

There have also been new artistic approaches toward Islamic calligraphy with the emergence of *hurufiyya* movement and *saghakhaneh* school [8]. Modern artists started using calligraphy as a compositional element and not essentially as its traditional function as the carrier of content. These movements were developed into various branches and styles of using Islamic calligraphy such as freeform calligraphy, abstract calligraphy and calligraffiti.

2.1.4 Knowledge of Islamic Calligraphy

We can think of two types of knowledge in Islamic calligraphy that enable us to do different tasks; visual knowledge and practical knowledge. While the practical knowledge enables calligraphers to use the traditional reed pen (or any other tool with a similar structure), the visual knowledge covers a larger set of tasks and is not restricted to calligraphers only. Some of the abilities related to the practical knowledge of calligraphy are controlling pen movements, the use of different reed widths, controlling ink intake, controlling pen pressure, etc. Acquiring this type of knowledge needs training and practice and people can gain different levels of it that means different levels of proficiency in terms of precision and consistency.

The visual knowledge of calligraphy also requires training. However, this type of training is in the form of seeing multiple samples. Typically, different sources of information are involved in this process. You see some calligraphy samples and you also get the information that, for example, this is calligraphy in *nasta'līq* script. After being trained with enough samples, you can generalize and identify other calligraphy samples in *nasta'līq* script that you have never seen before. In other words, now you "know" what *nasta'līq* script looks like and you can classify that script among others.

The range of tasks that depend on visual knowledge is wide and people can have radically different levels of this knowledge. If you have been exposed to calligraphy samples just through reading books or seeing signs written in calligraphy, you probably can loosely differentiate between calligraphy and other forms of writing. This means that you can simply classify what is calligraphy and what is not calligraphy. But if you have been exposed to more samples, you might be able to differentiate between different calligraphy scripts as well. If you are even more familiar with calligraphy, you can also classify samples based on their calligraphic quality, saying this is good calligraphy and this is normal or poor calligraphy. Now, if you are a professional, you can even distinguish between different styles in one script.

The knowledge of calligraphy can also be viewed from a task-based perspective. Knowledge in general enables us to perform certain tasks, and in the light of these tasks, it can be interpreted as separate pieces of knowledge. Some examples are the knowledge regarding the correct form of letters and words, the knowledge of placing dots of letters in correct positions, the knowledge of correct spacing between words, the knowledge of different compositional forms, etc. All these pieces also include visual and practical forms of knowledge.

Knowledge of calligraphy is also mostly implicit and therefore could not be verbally transferred or codified. Gaining a level of practical proficiency in Islamic calligraphy is only possible through training with the reed pen, and visual knowledge is gained only through visual engagement with many calligraphy pieces in different forms.

2.2 Generative Adversarial Networks

Generative Adversarial Networks (GANs) are a family of deep neural networks that work based on adversarial training of two networks against each other; a generator and a discriminator. Through training, GANs can estimate the probability density distribution of a dataset and also can generate new samples from the estimated probability density distribution.

The field of GANs is a new and rapidly advancing branch of research in deep learning. Since the breakthrough paper by Ian Goodfellow et al. in 2014 [12], there have been significant improvements in the performance of the GANs and a massive number of papers published introducing many different GAN architectures for different purposes in various domains.

2.2.1 Applications of GANs

Many different GANs have been developed for different purposes to date. Although GANs can be used for various types of data, there have been many impressive applications of them in visual domains. The quality of generated samples has been considerably improved in networks like StyleGAN [13][14] compared to older architectures such as DCGAN [15] to a point that some generated samples are indistinguishable from real images [16]. Some GANs also perform well at representing disentangled and interpretable features of the original dataset. In addition to merely generating photo realistic new samples from a dataset, GANs have also been used for text-to-image translation [17], cross-domain image generation [18], semantic photo editing [19][20], image reconstruction [21][22], increasing image resolution [23], video prediction [24], 3D object generation [25], dentistry [26], and augmented reality [27].

2.2.2 Artistic Use of GANs

The generative aspect of GANs has created an increasing interest in the artistic use of GANs. Many artists are using different GANs in their works as a new tool that equips them with new modes of expression. One major difference in using GANs for technical purposes compared to artistic uses is that in technical cases, the performance of a network is usually evaluated based on quantitative metrics that are commonly used across the research community, whereas in artistic cases, the results are mostly visually evaluated. In many cases, the training is stopped at early stages, when the artist is happy with the results, and the results are usually cherry-picked. Also, in many cases, artists create their specific datasets for their unique artistic intentions.[5]

Although the use of GANs is still not widespread among artists, some artists are sharply focused on using GANs in their works. The German artist Mario Klingemann[6] is one of the artists who uses GANs in his works, mostly by exploring the continuous nature of the latent space. Refik Anadol[7] has also explored this continuous nature by generating results from moving in the latent space, what he interprets as “dreams” or “hallucinations” of a network. Helena Sarin uses StyleGAN [13] and CycleGAN [28] with datasets of her own drawings, sketches, and photographs, and incorporates manual post-production processes on GAN-generated results to create her final artworks [29].

While many artists use GANs as a creative tool in their workflow, some people look at GANs from a machine creativity perspective (or AI creativity). An interesting work, by Elgammal et al., is CAN (Creative Adversarial Networks) [30] in which modifications are introduced to the objective function of GANs so that the network keeps generating novel results by learning artistic style.

[5] An example is “Myriad” by Anna Ridler (http://annaridler.com/myriad-tulips).

[6] https://underdestruction.com

[7] https://refikanadol.com

2.2.3 Working With GANs

Despite the diverse and interesting applications of GANs, the accessibility of GANs is still seriously limited for end-users. Presently, a few software tools such as RunwayML[8] and Playform[9] are available that provide an interface for people with no experience in coding so that they can work with different GANs. However, working with GANs in these tools is mostly limited to working with pre-trained networks or transfer learning with minimal control over the hyper-parameters of the network and training configurations.

Training most GANs, like other machine learning systems, usually requires a large number of data samples. There is an ongoing line of research that focuses on the concept of learning from a limited number of samples [31] and even learning from a single sample [32]. Some recent GAN papers also address the same issue [33]. However, with current GANs we are still talking about a dataset of at least a few thousand samples if exploring the representation of the domain is intended rather than merely generating interesting visuals.

Training GANs also requires high computational power. GANs are usually trained on powerful GPUs either locally or through cloud platforms. The training time of GANs depends on the network architecture, the number and size of data samples, and the computational power available. But generally, training GANs is a time-consuming process and depending on the purpose of the training, it can take from hours to weeks to achieve desirable results.

[8] https://runwayml.com

[9] https://www.playform.io

3 Making a Calligraphy Dataset

As GANs are still data-hungry, it is crucial to think about methods of creating consistent and regularized calligraphy datasets for GAN training that aim for knowledge extraction. Different datasets are required for training that intend for different aspects of calligraphy in different scripts, and it creates the urge to find feasible methods of creating diverse datasets. It should also be considered what each dataset represents and what the purpose of the experiment is.

In the absence of a regularized dataset of Islamic calligraphy, many different approaches can be considered toward making a calligraphy dataset of thousands of data samples. These approaches introduce interesting conceptual and practical challenges with regard to what the dataset represents and how it should be prepared. In this chapter, I explore some of the possible approaches and their challenges and explain the concept and process of making my datasets that follow the general discussion about the challenges.

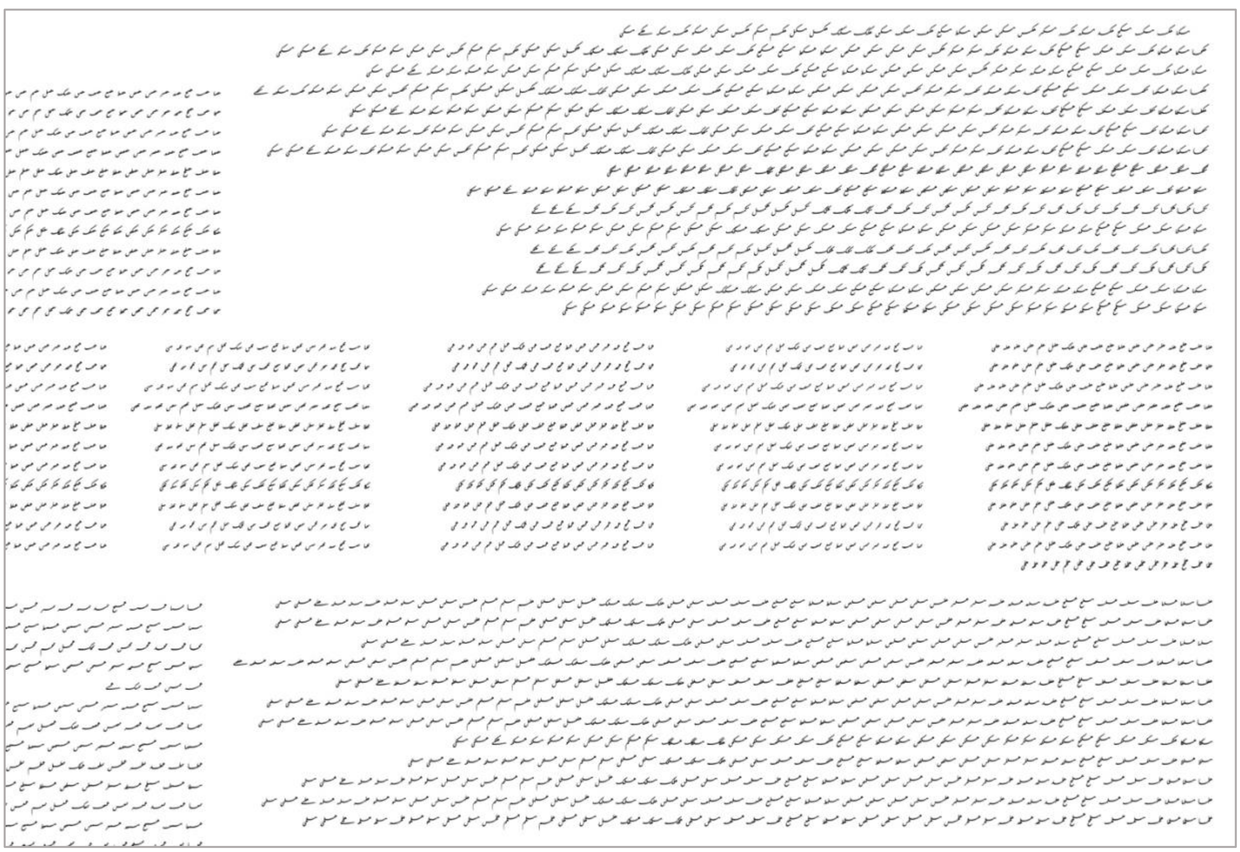

Figure 3.1 A cropped view of one of the 12 batches of data samples being prepared for Nas4-60 dataset, explained in 3.5.

3.1 Conceptual Challenge of Creating a Calligraphy Dataset

While preparing a large dataset is quite straightforward in many domains, it introduces interesting conceptual challenges in the domain of Islamic calligraphy. To clarify these challenges, let's first consider datasets of natural images and see what some of the basic considerations are in making such datasets.

For example, image datasets of human faces are common to use as a benchmark for GANs. One of the considerations in creating a dataset of human faces, as obvious as it may be, is that each sample should contain an image of a human face. The images should be properly cropped around the face so that they include all the elements of the face, but not much of other body parts such as shoulders and arms. Also, the presence of other objects, such as glasses or hats, in the samples depends on the design of the dataset. Moreover, the dataset is supposed to provide a wide diversity of features in that domain. In the case of human faces, we expect to see samples of different races, genders and ages, diverse backgrounds, different facial gestures and head positions, various lighting conditions, different hair length and styles, skin and eye colours, etc. At the end of the day, the generated results would be as diverse as the dataset in terms of the features.

Considering this example, we can see some of the differences more clearly. When we talk about a dataset of faces, or similarly birds, flowers, etc., each sample contains an object that itself contains certain elements. However, Islamic calligraphy as a domain can refer to many different things. For simplicity, let's consider one script only. Are we talking about a dataset of letters? Or words? Or lines? or other forms of composition? In a dataset of human faces, there are specific elements in a certain structure, but each sample with different features. But what are those fixed, common elements in a dataset of calligraphy? Even in a dataset of handwritten digits, there are clear clusters of content for each digit, each of which has the same fundamental visual structure. In a calligraphy dataset of words, elements are letters, and the presence of letters is a binary choice; you either have a letter in that sample or you do not. So, in a dataset of words, you have many samples that are mutually exclusive in terms of such elements. Then, what is it that they have in common that makes them belong to the same domain?

It is also very important to focus on the notion of diversity in a dataset and the sources of visual diversity in the samples. Some examples of natural images were discussed above in which the diversity of samples comes from changes in visual features. As another example of natural images, consider an image dataset of horses. In this case, diversity simply means images of different horses in different places and situations. Depending on the decisions around making the dataset, this might be limited to standing horses only, or it may include horses in motion as well. It may also include riders or exclude them in the images. The same general concept applies to a quite different dataset of Roman characters [34]. In this dataset, the diversity comes from different fonts with a wide range of visual components such as the thickness of lines, decoration, serifs, etc.

Now, let's see what diversity means and where it comes from in a dataset of Islamic calligraphy samples. A very simple way of thinking about it is that collecting as many calligraphy pieces as possible will automatically guarantee diversity, as it can consist of different scripts, different styles, different compositions, etc. But is this a meaningful diversity? We should note that the diversity under the general category of Islamic calligraphy can be enormous on many different levels. Different scripts are included, and in each script, there are many different forms of composition. Some pieces comprise only a couple of words, some have many different lines, some have overlapping words, some have sloped baselines, etc. If we look at this diversity from the lens of calligraphic features, what is intuitively obvious is that such a wide range of samples lacks enough common high-level features that persist in the dataset (Figure 3.2). An analogy that I can think of, that is too wide to be considered a single domain, is a dataset of animals. In a dataset of animals, it is hard to find meaningful high-level features that are common in all the samples. Instead, we can identify clusters, in which there are clear common high-level features, such as a cluster of dog image samples or bird image samples.

Figure 3.2 Examples of calligraphy pieces in different scripts and different compositions. Works by: Abu al-Ma'ali (d. 1209), Kamal al-Din (d. 1567), Mir Emad (1554-1615), Ala al-Din Tabrizi (1524-1593). Source: Wikimedia Commons.

Similarly, in a dataset of Islamic calligraphy, we can identify clusters based on different high-level features, such as scripts or compositional forms. But, even in these clusters, the meaning of diversity is not essentially clear. To clarify this, let's consider a dataset of single words, which is the simplest possible compositional element in this domain. However, we still have widely different approaches to visually diversify samples. One approach is to diversify letter combinations to create different words. Another approach is to diversify based on different scripts or styles in a fixed collection of words. Another way could be having the samples written by calligraphers to incorporate the inevitable differences that happen in different samples as a source of diversity.

There are numerous ways to think about and incorporate diversity into the design of a calligraphy dataset. However, it makes sense if such decisions follow the specific purpose of the experiments for which the datasets are designed for. This way, diversity can have many different meanings in different cases and might be radically different from dataset to dataset.

3.2 Technical Challenges of Creating a Calligraphy Dataset

To explore technical challenges that creating a calligraphy dataset can pose, let's consider some of the possible approaches that we can take to create one with large amounts of data samples. The technical difficulties in the following options demonstrate that gathering purposeful data samples for specific experiments is not straightforward. Such technical challenges can be the topic of separate research.

Probably one of the first approaches that come to mind is collecting as many available calligraphy pieces as possible. Depending on the experiment, one might be interested in all scripts, certain scripts, or even more specifically, one style in a script. Let's forget about the accessibility of pieces for now – although this itself can be one of the most important issues – and concentrate on the technical issues of using such a dataset. The following are some of the issues that I can think of.

One of the first things that need further consideration is the fact that according to the convolutional architecture of most GANs, data samples of a dataset should be exactly the same size, and in most cases, they should be square. However, almost all calligraphy pieces are rectangular with different aspect ratios. To fit these samples into a square frame, they can be cropped, scaled down to fit into the frame, or disproportionally scaled to fill the frame. Cropping would result in losing parts of data, and fitting into the frame by scaling down would leave some blank parts holding no information. Finally, a disproportionate downscaling would result in deformed calligraphy forms that affect both what the network learns from the dataset and what it regenerates (Figure 3.3).

Figure 3.3 A demonstration of the issues arising from size regularization of calligraphy pieces. Calligrapher: Mirza Gholamreza Esfahani (1830-1887).

In addition to the aforementioned issue, if we want to use entire pieces of calligraphy as our data samples, we should note that most of such samples are extensively illuminated. There might be experiments in which these illuminations do not cause any difficulty, but in general, we can anticipate that the achieved representation of such a dataset would be a combination of calligraphy and illumination at the same time. A similar phenomenon was demonstrated by Mike Tyka in a visualization technique that helped to understand how neural networks carried out difficult visual classification tasks [35]. In one of the presented samples, we can see how a classification network trained on images of saxophones also captures human parts like hands that hold the instrument (Figure 3.4).

Figure 3.4 Saxophone Dreams by Mike Tyka. Image source: https://miketyka.com

We also need to be aware of the variety of different styles of backgrounds and illuminations that are present in any large dataset of calligraphy pieces. Overall, it is very interesting to think about the relationship of the isolated calligraphy forms and the background or illuminations of the pieces and how they are similar to or different from what we see as background when using natural images of a certain domain, like images of animals in different environments and hence with different backgrounds.

Another issue is the abundance of overlapping forms in calligraphy pieces. In some scripts, such as *thuluth*, it is part of the compositional nature of the script even in writing single lines. Also, in *nasta'līq* script, the popular style *siyah-mashq* is the repetition of words and letters written on top of each other (Figure 3.5).

Figure 3.5 Samples of overlapping forms in calligraphy pieces in nasta'līq script (left) and thuluth script (right). Works by Fathollah Khan Jalali (d. 1918), Ali Badavi (d. 1940), Aziz al-Rafaei (d. 1934).

Finally, regularizing samples from available pieces can be also challenging. There are two types of regularization that need to be done. First is the regularization of the baseline since in many calligraphy pieces words are written on an angled baseline. Second, and the more challenging one, is the regularization of the reed width to achieve the same stroke thickness proportional to the sample size throughout the dataset[10]. Both of these regularizations are possible but require image processing algorithms to be applied to raw data. According to the diversity of samples, these algorithms might not be very easy to make and can be the topic of separate research projects. However, the need for such regularizations and the effects of using an irregular dataset on the training and the type of representations captured require further discussion. We can see examples of such data regularizations in datasets of other domains as well. For instance, human face datasets such as FFHQ [36] and UTKFace [37] are regularized in a way that the face is positioned in the middle of the image and fills most of it.

[10] Why this regularization is needed is something intuitive that is obvious from a calligraphy perspective but it is challenging to explicitly justify.

It worth mentioning that the above challenges and concerns that arise from the option of creating a calligraphy dataset from available pieces should be addressed if one wants to pursue controlled experiments. For artistic purposes only, using any dataset can be a viable option. Using any raw calligraphy datasets of any script or style in GAN training would probably result in very interesting visual outcomes, but the scope would be limited to those works.

The other approach that can resolve many of the aforementioned issues, is making a dataset with controlled parameters. It can be assigned to a group of calligraphers to make calligraphy samples based on a certain plan which is essentially informed by the design of the experiment. Depending on the purpose of the experiment, style consistency might be a concern or not. But maintaining a high level of quality seems to be necessary in any case, except for cases that the experiment itself is about exploring different levels of quality. Considering the fact that most GAN architectures still require datasets of at least tens of thousands of samples, using this approach can be extremely expensive and time-consuming. However, as long as required financial and human resources are available, this can be considered a viable approach.

3.3 Concept and Process

As discussed in 3.1 and 3.2, creating a calligraphy dataset is subject to different challenges. The scope of these challenges may vary depending on the design of the intended dataset, but even in a comparatively simple case, such as creating a dataset of single words, these challenges make it significantly time-consuming and costly to create a dataset from archival pieces or to commission calligraphy masters to create one. An alternative method is suggested here using calligraphy fonts as a fairly quick and affordable way of creating controlled and consistent data samples in large quantities. Two datasets in *nasta'līq* script, namely Nas3-10k and Nas4-60k, are introduced using this method which include samples of isolated letter combinations.

Although calligraphy fonts suffer from serious issues in supporting some of the basic calligraphic features, they can still be a good estimate for standard calligraphy samples under certain conditions. For this experiment, data samples of single words (up to four-letter combinations) in *nasta'līq* script were created using a calligraphy font with manual human modifications and adjustments as the process could not be fully automated and many calligraphic decisions had to be made in person. Furthermore, for the specific design of these datasets, a reduced alphabet was used to eliminate redundancies of forms.

A limited number of *nasta'līq* fonts are available and they widely vary in style and quality of the results. Some of the available fonts in OpenType formats are IranNastaliq and Noto Nastaliq Urdu. However, the complex positioning of the glyphs and the multiplicity of possible forms for letter combinations make it problematic to properly implement calligraphic features in these fonts. There are also calligraphy fonts available that offer a wider range of options regarding alternative letterforms and elongations with fewer issues, but these fonts are not available in OpenType format and require a middleware to be used in other software such as Adobe Illustrator. In the design of my datasets, Mirza font [38] designed by Amir Mahdi Moslehi was used that is created based on the style of the great Iranian calligrapher Mirza Gholamreza Esfahani (1830-1887). This font is accessible through the middleware QalamBartar[11] that provides features to toggle between different letterforms and to choose different elongation options.

Figure 3.6 A comparison of the word "nasta'līq" written in different calligraphy fonts.

To have a controlled set of visual features in my dataset, data samples were limited to single isolated letter combinations. These letter combinations are the simplest building blocks that are used in different compositional forms. This also makes it possible to control the source of diversity in the dataset which is the different combination of letterforms in each data sample. Because the font used is created based on the style of an individual calligrapher, the style can be considered consistent. Also, using a font eliminates any diversity that stems from minor differences of the same letters that happen if the letter combinations are created by hand.

[11] https://maryamsoft.com/qalambartar

3.4 Nas3-10k Dataset

The basic concept of creating Nas3-10k dataset was to create a dataset of all single letters, two-letter and three-letter combinations that includes all possible elongations and alternative letterforms added to the basic forms. To avoid repetition of forms, a reduced alphabet was used to eliminate letters that share the same basic shape and are different only in the position and number of the dots. For example, letters *be*, *pe*, *te* and *s̲e* are similar (in all initial, medial and final forms) except for the number and position of the dots. Also, letters *nun* and *ye* have similar initial and medial forms to the mentioned four letters. To avoid repeating the same form, similar letters were grouped and replaced by the simplest letter in the group. Considering the Persian alphabet of 32 letters as the base, this reduces the alphabet to 18 letters that have at least one distinct initial, medial or final form. Partial similarities were also taken into account when creating letter combinations (Figure 3.7).

Reduced alphabet				Original alphabet	Name (transliterated)
Final	Medial	Initial	Isolated		
ـا	N/A	N/A	ا	ا	ʾalef
ـب	ـبـ	بـ	ب	ب	be
				پ	pe
				ت	te
				ث	s̱e
ـح	ـحـ	حـ	ح	ج	jim
				چ	če
				ح	ḥe
				خ	xe
ـد	N/A	N/A	د	د	dâl
				ذ	ẕâl
ـر	N/A	N/A	ر	ر	re
				ز	ze
				ژ	že
ـس	ـسـ	سـ	س	س	sin
				ش	šin
ـص	ـصـ	صـ	ص	ص	ṣâd
				ض	zâd
ـط	ـطـ	طـ	ط	ط	tâ
				ظ	ẓâ
ـع	ـعـ	عـ	ع	ع	ʿayn
				غ	ġayn
ـف	ـفـ	فـ	ف	ف	fe
ـق	ـقـ	قـ	ق	ق	q̈âf
ـک	ـکـ	کـ	ک	ک	kâf
				گ	gâf
ـل	ـلـ	لـ	ل	ل	lâm
ـم	ـمـ	مـ	م	م	mim
ـن	ـنـ	نـ	ن	ن	nun
ـو	N/A	N/A	و	و	vâv
ـه	ـهـ	هـ	ه	ه	he
ـی	ـیـ	یـ	ی	ی	ye

Figure 3.7 In the reduced alphabet, letters are grouped based on their visual similarities and the simplest letter is kept. Also, letters with partial similarities of the initial and medial forms are grouped (highlighted with the same colour).

Additionally, all the dots of the letters from the reduced alphabet were deleted. The reasons for this decision are twofold. Firstly, it is based on the fact that the main visual structure of letterforms is not the dots, and when the visual aspect of the main letterforms is intended, we can ignore the dots. This can be further supported by the fact that even in traditional calligraphy, sometimes putting dots in the exact correct number and position is not important and follows the overall composition of the piece. Secondly, this decision was informed by the serious issues of all calligraphy fonts in correctly positioning the dots, which makes it extremely time-consuming to correct manually[12].

The process of creating Nas3-10k started with creating all letter combinations in a text file using the reduced alphabet. These letter combinations were converted to *nasta'līq* using the calligraphy font in Adobe Illustrator and possible elongations and alternative letterforms were added (Error! Reference source not found.). Then, they were converted to shapes and dots were deleted. Also, in the process of reviewing all the words to delete the dots, samples with any issues were deleted. These defects are usually the result of the font's shortcomings or bugs. Finalized samples were then exported to JPEG files with a multiplication factor depending on the final required size of the data samples and the pixel dimension of the largest letter combination. These considerations were taken into account to minimize the whitespace around words in the final square-size data samples. This process also maintains the same stroke width across the dataset.

[12] The only exception made is letter nun in Nas3-10k in which the dot was kept. The reason for that was the fact that this is the only letter with a strictly fixed dot position and I considered that as the main structure of the letter.

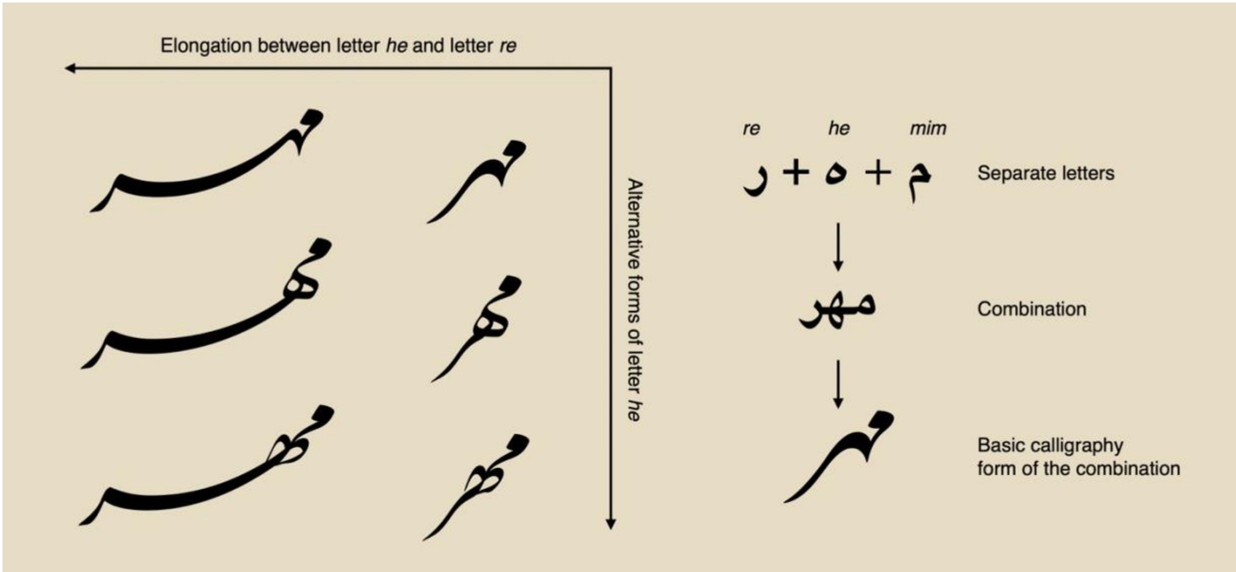

Figure 3.8 An example of creating alternative letterforms and possible elongations in the combination of letters mim, he and re.

In samples that consist of separate parts – due to having letters that do not have a connected form in the first or second position – manual spacing adjustments were also required because the font is incompetent in creating correct letter spacings (Figure 3.9). This step specifically added a significant time to the whole process as all the samples had to be checked and adjusted individually.

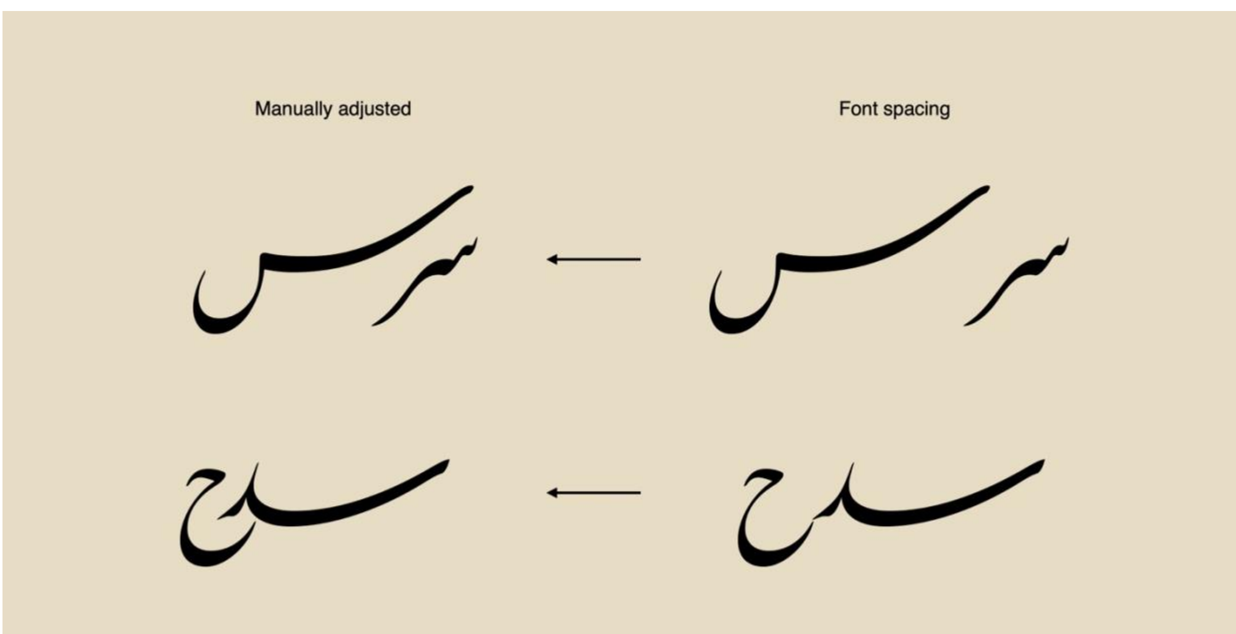

Figure 3.9 Examples of manual spacing adjustments in Nas3-10k

With 15 unique initial and medial letterforms, and 18 unique final letterforms, this dataset consists of around 4300 unique basic letter combinations. After adding possible elongations and alternative letterforms, and final reviews, the number of data samples in this dataset reaches 9,372 (Figure 3.10).

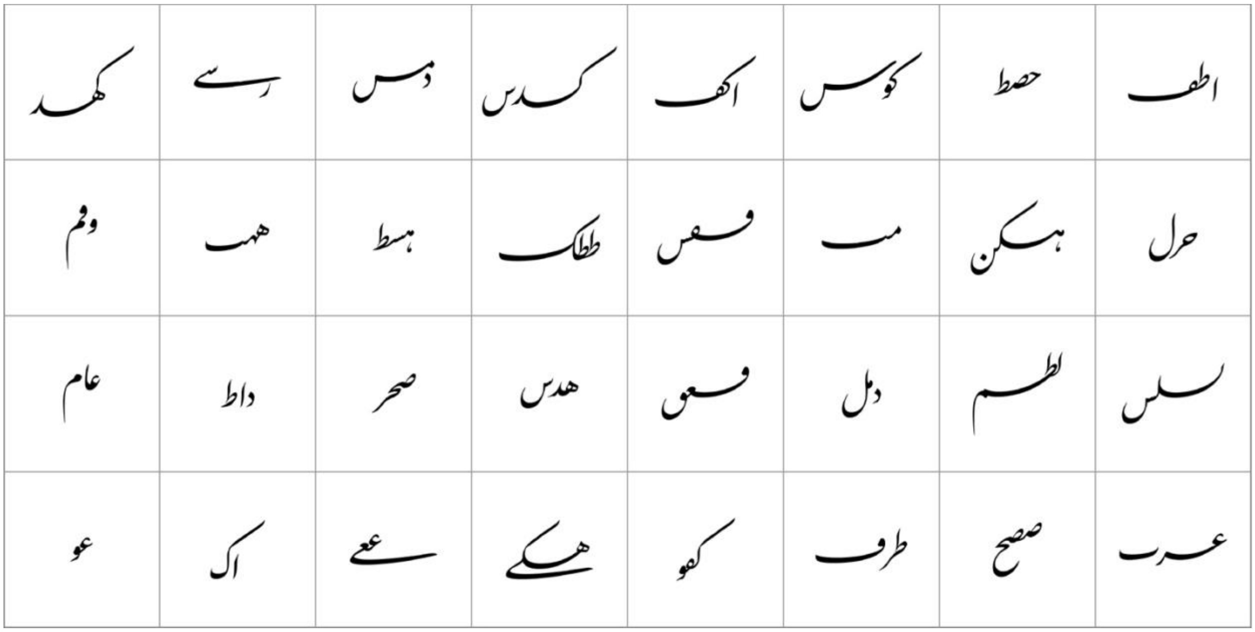

Figure 3.10 Random samples from Nas3-10k dataset

3.5 Nas4-60k Dataset

In an effort to create a larger and more consistent dataset of *nasta'līq* letter combinations, Nas4-60k dataset was created to include up to four-letter combinations. The general process of making Nas4-60k is similar to the one in Nas3-10k. However, some new decisions were made in the design of Nas4-60k to make the process less subjective and also possible in a limited time frame.

In Nas4-60k, letter combinations are limited to those that are fully-connected. To do this, those letters in the reduced alphabet without a connected initial or medial form were used only in the last position of the letter combinations. This includes letters *'alef*, *dâl*, *re*, and *vâv*, and reduces the letters used in initial and medial positions from 15 in Nas3-10 to 11 in Nas4-60k (Figure 3.7).

This decision in the design of Nas4-60k is important with regard to the pieces of calligraphy knowledge that the dataset represents. Although both datasets represent the basic building blocks of calligraphy, having separate pieces in data samples of Nas3-10k inevitably creates the need to use a type of compositional knowledge to place the pieces together correctly. The decision to keep fully-connected letter combinations limits the represented knowledge in the dataset to basic calligraphic features, such as the visual knowledge of what letters look like and how they are connected. Thereby, in training GANs with this dataset, the same set of features should be checked to see if the network manages to represent or not. This difference between the two does not make one dataset superior to the other, but it is important to be aware of what each dataset represents.

In addition, discarding disconnected samples eliminated the need to manually adjusting the spaces between letters in disconnected forms. Thus, Nas4-60k is less subject to personal decision-making. Also, this decision significantly reduced the production time. Given the exponential growth of sample numbers when adding four-letter combinations, this change made the production process possible within the available timeframe.

The process of adding possible elongations and alternative letterforms was also more selective in Nas4-60k dataset to keep superior options and to avoid the exponential surge in the number of elongated samples. This prevents the undesirable abundance of similar elongated forms and also helps to keep an acceptable balance between elongated and non-elongated samples throughout the dataset.

With 11 letters having connected initial and medial forms, and all 18 letters having connected final forms, 26,352 basic letter combinations were created. After adding elongations and alternative forms and final reviews, the total number of data samples reaches 59,890 unique samples (Figure 3.11).

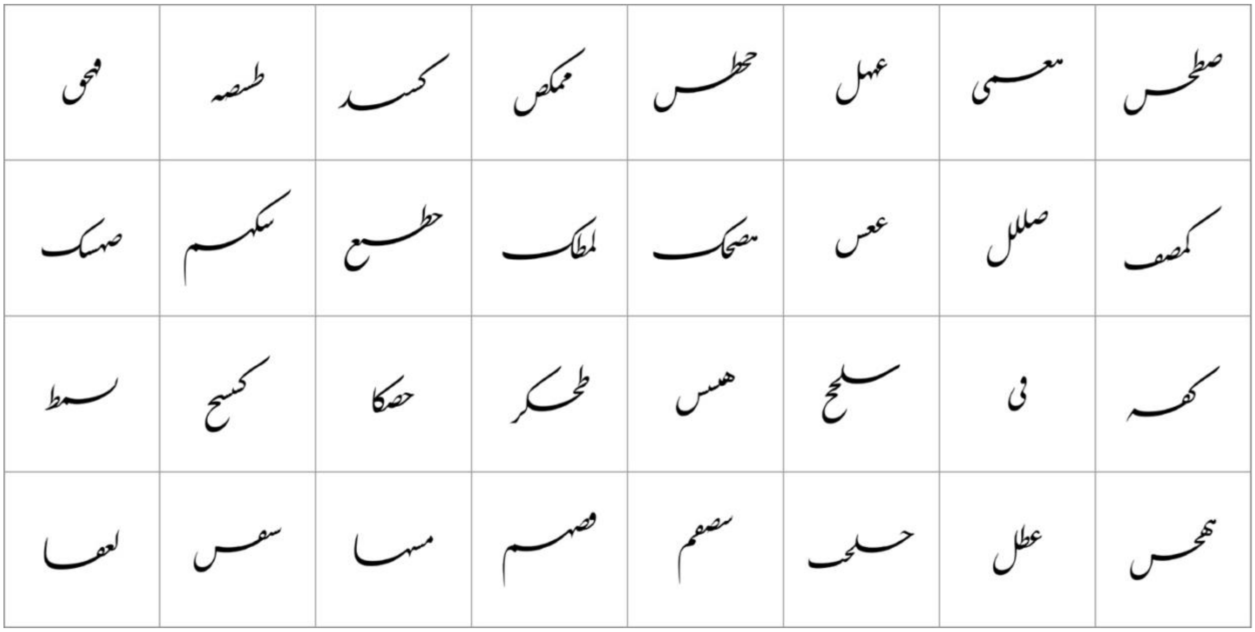

Figure 3.11 Random samples from Nas4-60k dataset

3.6 Summary

Designing a purposeful dataset of Islamic calligraphy is subject to different technical and conceptual considerations depending on the purpose of the dataset and the method of creation. Conceptually, there are different aspects such as what the dataset represents in terms of the calligraphic knowledge that it holds, and what the sources of diversity in the dataset are, that should be taken into consideration in the design of the dataset. Practically, there can be various methods considered to either create a dataset originally or to collect data samples from processing existing calligraphy pieces. However, these methods are either prone to technical complexities due to the need for various algorithms to pre-process existing data, or costly due to the extensive need for professional calligraphy work.

As GAN training requires large datasets and there is no regularized dataset of Islamic calligraphy publicly available, a feasible method was proposed to use calligraphy fonts to create controlled datasets for purposeful GAN training. In this method, a semi-automated generative process is used to create glyph combinations using calligraphy fonts combined with manual human review and adjustment of the samples.

Using this method, two datasets, namely Nas3-10k and Nas4-60k, of single letter combinations in *nasta'līq* script were created with around 10,000 and 60,000 unique data samples respectively. These datasets consist of samples of up to four-letter combinations and were created with a focus on the visual qualities of the basic letterforms. In the design of these datasets, a reduced alphabet was used to eliminate the visual redundancies of the letters that share the same basic structure. The creation of these datasets mostly follows the same concept but there are minor differences in the design of the two in terms of the type of calligraphy knowledge that they represent. The creation of these datasets was an essential step for a purposeful GAN training that is explained in the next chapter.

In addition to the concerns about collecting a large number of calligraphy samples, what is also important is choosing the representation space in which each data sample is represented. My decision on choosing pixel representation space for my datasets is informed by firstly, the technical and practical challenges that using other representation spaced pose on the creation of the dataset, and secondly, the facts around choosing the GAN architecture for my training sessions and the representation space that it uses (see 4.1 on choosing the network).

4 Training Process

I undertook a gradual up-scaling process for training my networks. I started with a commercial end-user software tool with a minimal dataset, and then I proceeded to more professional platforms and tools to be able to use larger datasets and to have more control over the training parameters.

This process involved parallel activities such as dataset preparation, learning about platforms, and educating myself on the network itself. At each step, these activities converged to a set of training and getting outputs. Next steps were undertaken accordingly based on the analysis of the outcomes to move forward. Each step was larger in terms of required time, resources, and knowledge. But this upscaling was necessary to get to an acceptable level of visual fidelity so that the generated results would show the points clearly and the results used in the final works would speak for themselves.

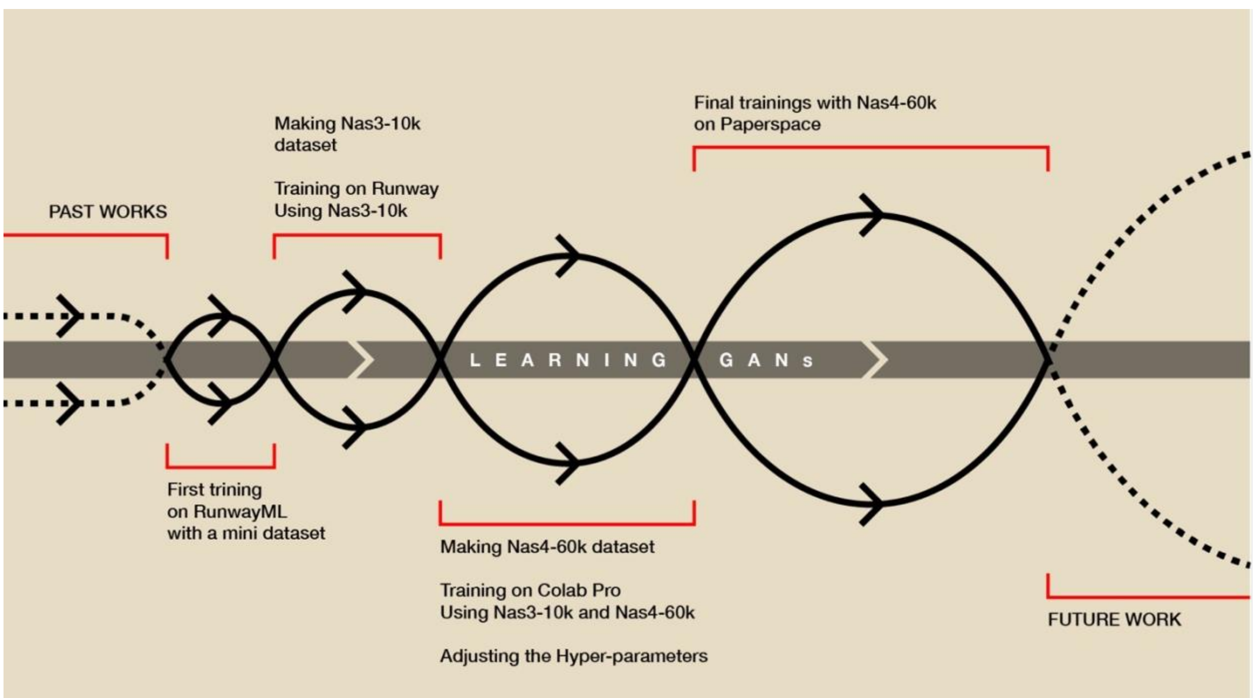

Figure 4.1 A schematic representation of different phases of network training and related activities.

4.1 Choosing the Network

In a search for finding a GAN architecture that can be used for Islamic calligraphy, I have considered those networks that have been created for domains that can be considered closer to Islamic calligraphy compared to natural images. An example of these networks is GlyphGAN [34] that has shown promising results on Roman characters. However, the network architecture of GlyphGAN is designed specifically for single letters in that script and it does not suit the structure of an Islamic calligraphy dataset. Any modification to the network architecture requires advanced computer science and programming skills and does not fit into the scope of this project.

What also mattered in choosing the network architecture was the availability and usability of the code regarding my basic programming skills. Some of the GAN architectures have gained significant popularity among a large group of people in different domains, and therefore, those networks come with simpler implementations of the code in different platforms and also tutorials and guidelines provided by people who use them. This makes them more practical options to stay away from too much technical complexity.

One of these general-purpose GAN architectures that has shown impressive results in various domains is StyleGAN [13] that was first introduced in 2018 and later significantly improved in 2019 into StyleGAN2 [14]. Although this network has been created mainly for datasets of natural images, what made this network the final candidate in my project was the fact that StyleGAN network has proved to be one of the best networks in extracting disentangled and interpretable features of training datasets.

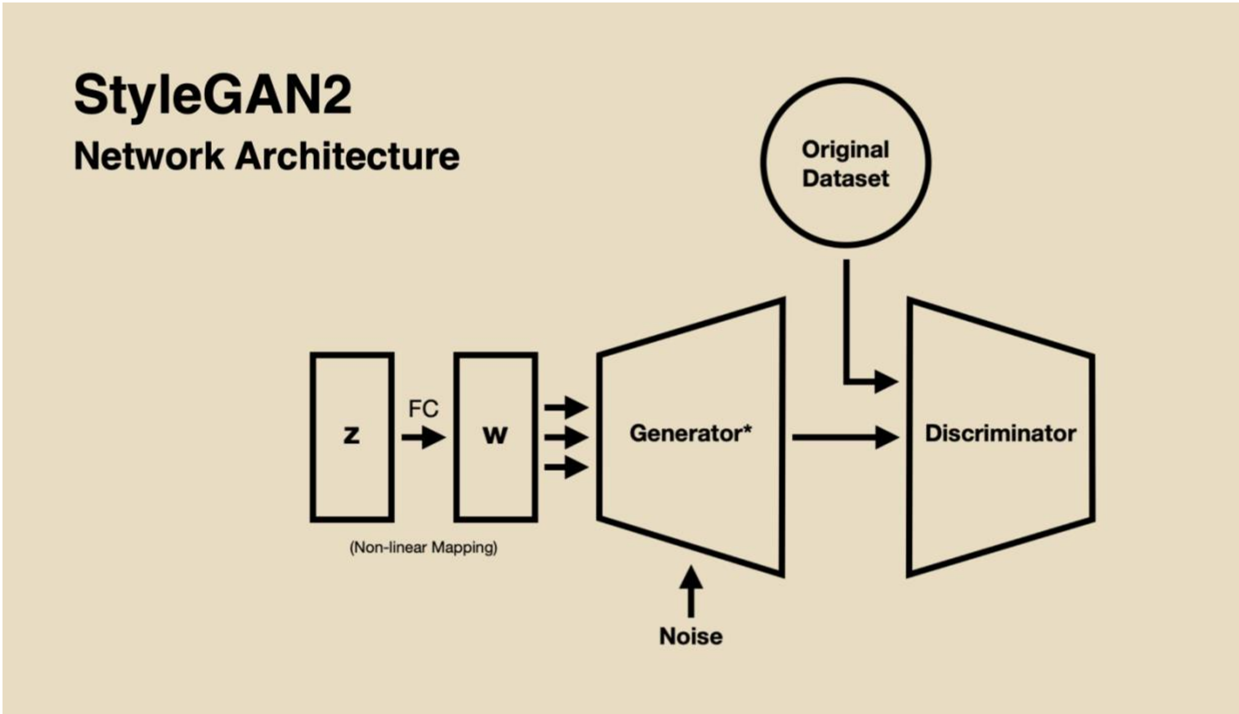

Figure 4.2 StyleGAN2 network architecture

The latest version of this network is StyleGAN2-ada [33] which is used in my training sessions and has additional adaptive augmentation features added to the Discriminator's structure to facilitate training with smaller datasets. These augmentation tools have not been used in my training sessions due to the fact they are in the form of mirroring, parametric deformations and colour changes, and are either irrelevant or not appropriate for a dataset of calligraphy. Though, the code is much simpler and also there are new built-in tools added that facilitate the process of generating samples from a trained network, therefore, exploring the latent space is relatively easier.

4.2 Early Attempts in RunwayML

During the "Introduction to Artificial Intelligence" course last year, I was introduced to RunwayML [39]. This software tool provides a user-friendly environment to experiment working with GANs at an entry-level. The basic feature of the software is to run networks pre-trained on different datasets provided by a community of collaborators. It also provides the feature to train some existing networks through transfer learning, including StyleGAN and StyleGAN2.

In the first attempt to train a GAN on Islamic calligraphy, I prepared a dataset of around 900 calligraphy samples consisting of random words, using the same *nasta'līq* font that I used in creating Nas3-10k and Nas4-60k (Figure 4.3). Unlike the final datasets, the test dataset was not regularized with the considerations mentioned in the design of my final datasets.

Figure 4.3 Random samples from the first dataset used for training in RunwayML

Using this dataset, a StyleGAN network, pre-trained on a dataset of human faces, was trained in RunwayML. The generated results did not show much diversity and mostly collapsed on one general form with some resemblance to data samples in the dataset. The residues of features from the previous training also persisted in the results. However, some general calligraphic features could be seen in the generated samples (Figure 4.4).

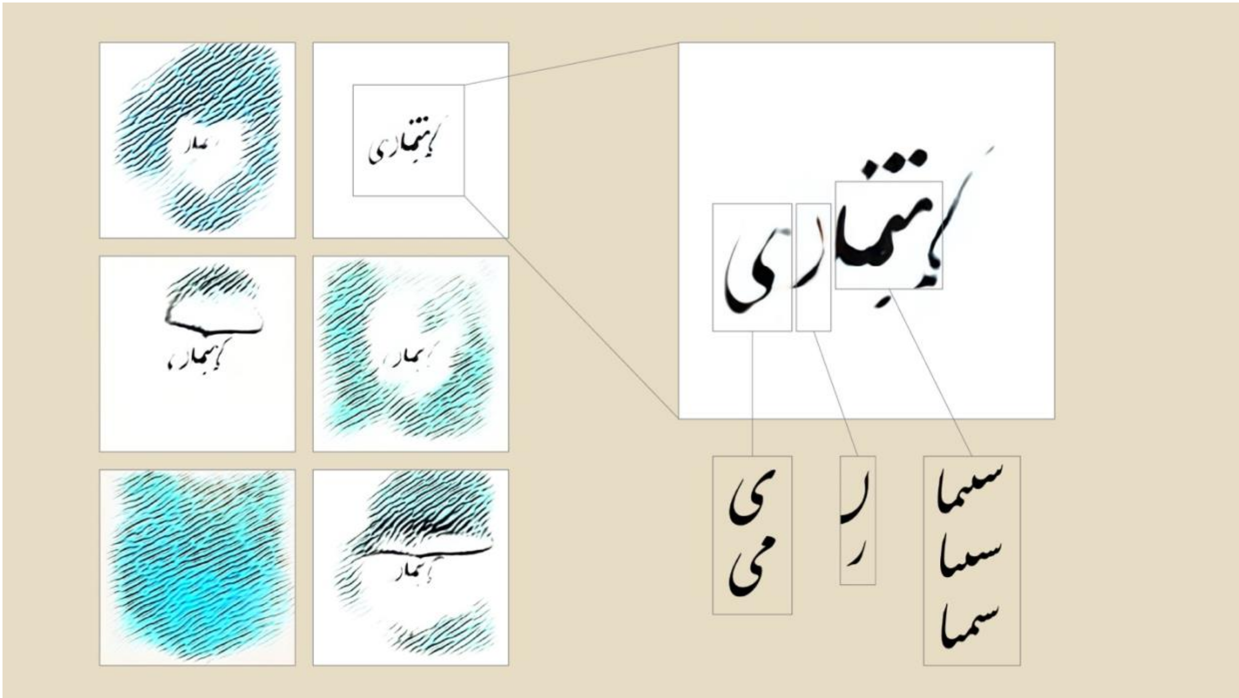

Figure 4.4 Generated samples from the first training in RunwayML (left) and extracted calligraphic features (right).

With some positive signs in the generated results, I proceeded with training another network with Nas3-10k dataset in RunwayML. The second network I used was a StyleGAN2, pre-trained on an image dataset of nebulas. The reason for choosing a different pre-trained network was because of the more diverse feature structure of the original training dataset that was considered more similar to my dataset compared to the dataset of faces.

The network was trained for 10,000 epochs and the results showed significant improvements. Although the network collapsed again into generating a limited number of distinct forms, in each cluster there were clear calligraphic features observed (Figure 4.5). The network diverged after a certain point and the results deteriorated towards the end but the results after 6000 epochs showed promising feature extraction. Also, interpolating between points created quite smooth transitions between letterforms (Figure 4.6). This was a sign that the network is finding a good estimate of the data distribution.

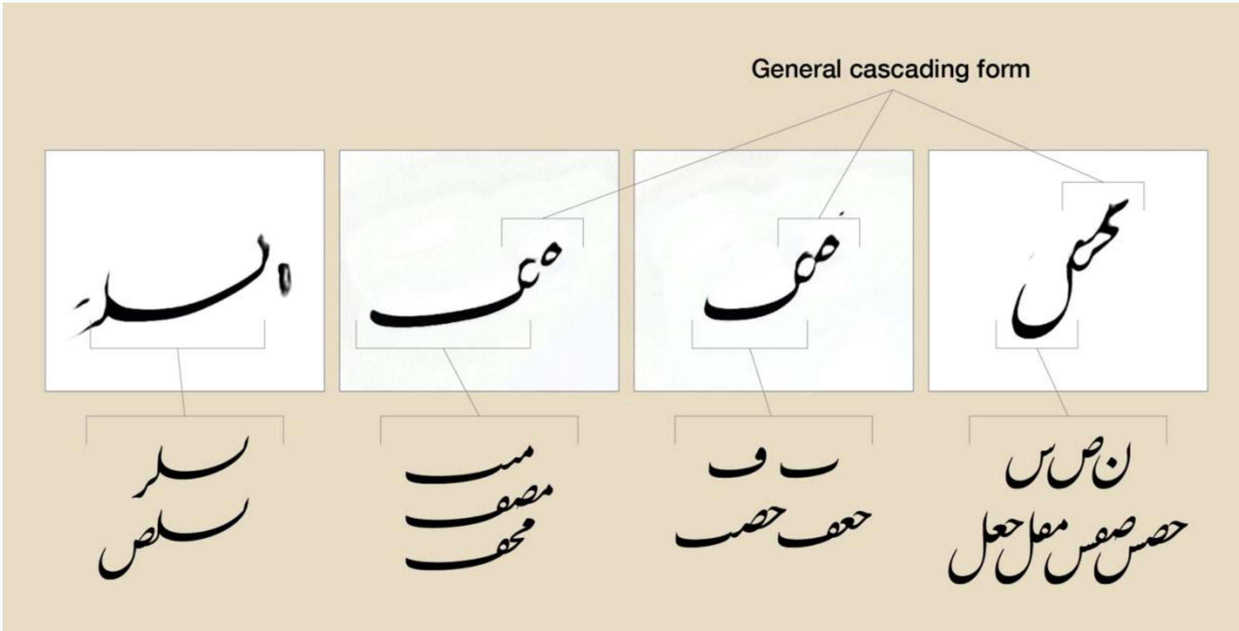

Figure 4.5 Generated samples in RunwayML using Nas3-10k dataset and extracted calligraphic features.

Figure 4.6 Smooth transitions of letterforms in interpolation between points using the trained network in RunwayML.

RunwayML provides a convenient way of getting familiar with GANs and facilitates the initial steps of training with custom datasets. However, there is almost no control over the parameters of the network and also very limited tools available to explore the latent space of the trained network.

Moreover, working with larger datasets is not possible in RunwayML and training for a long time is not cost-efficient. These initial rounds of training in RunwayML worked as proof of concept that StyleGAN model can capture calligraphic features if trained with a large and tuned dataset for long enough time. Therefore, to proceed further with more serious training, it was inevitable to use other platforms for more purposeful control over training and for better evaluation of the generated samples.

4.3 Adjusting Hyper-Parameters

Training GANs is a delicate process and obtaining a stable training session can be difficult. This is often done by using different hyper-parameters as the adjustable parameters in the training process to find the combination that results in the best performance. There are different techniques to search for the right hyper-parameters, such as grid search and random search. However, these methods were not affordable for my project regarding the fact that it requires many training sessions with the use of quantitative metrics. Instead, I had to rely on my visual evaluation of the results to decide which set of hyper-parameters were more suitable for my training sessions.

Moving forward from RunwayML, I considered a couple of options for heavier training. Regarding the need for high processing power for GAN training, local training with normal computers is not an option. Among the options to have access to powerful GPUs that can handle the computational load, Google Colab, Amazon Web Services (AWS), Google Cloud Platform, and Paperspace were considered.

As the most affordable and the easiest to use, Google Colab was selected. To access more powerful GPUs and having more notebook runtime, I upgraded my account to access Google Colab Pro. An implementation of the original StyleGAN2-ada code along with tutorials by Derrick Schultz[13] were used to start training on Google Colab.

To test the performance of the network, I started my training with Nas3-10k dataset. In the early rounds of training by using the default hyper-parameters, the network showed some progress in early stages, but two major issues happened pretty quickly during the training. The first issue was a serious mode collapse. After generating results with some diversity, the network quickly collapsed

[13] https://artificial-images.com

into generating only one mode. Another issue was divergence, which also happened in the early stages of the first few training sessions. The network captured some visual features of the dataset, but after a certain point, started generating random results without any relevance to the original dataset (Figure 4.7).

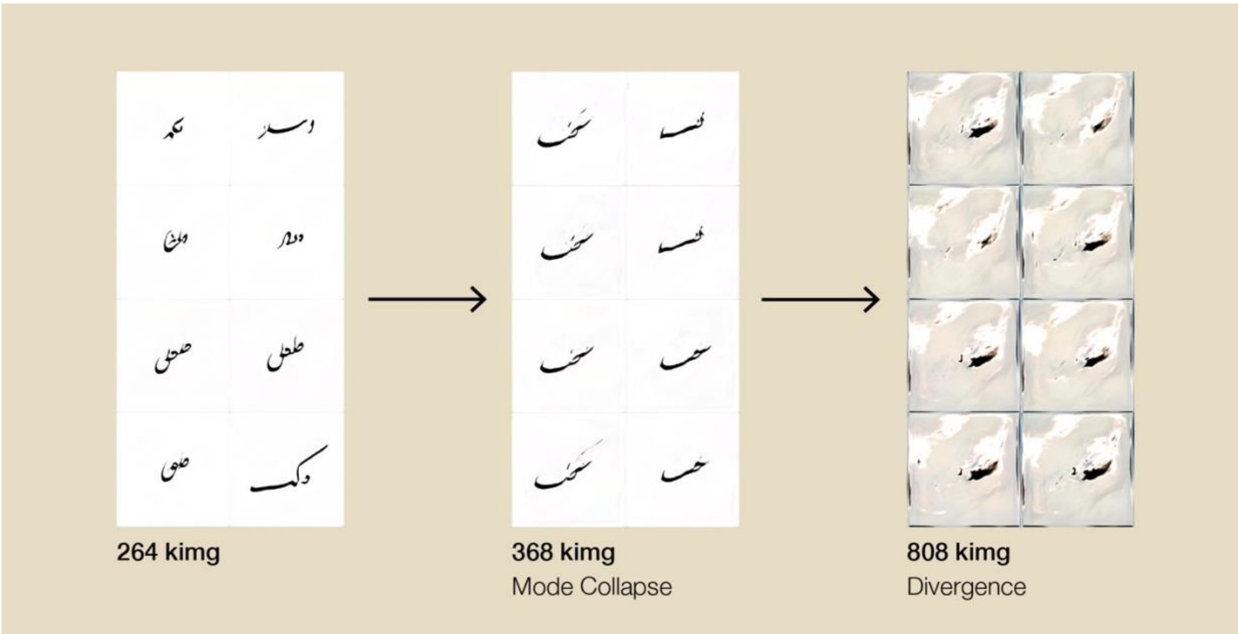

Figure 4.7 An example of mode collapse and divergence in a training session

To resolve these issues, I started playing with the hyper-parameters. Using the suggestions of creators of the network and other references, I changed certain parameters to check the impact on the results. Eventually, I had to fork my own repository of the original code to implement custom configurations in the code to change the hyper-parameters that were not available in the arguments of the training command.

After more than a dozen training sessions and more than 200 hours of training time, I found a combination of hyper-parameters that led to stable training. With more stable networks during this process, I started using Nas4-60k dataset, but I had to use the 512x512 pixels version due to extremely large files that the original 1024x1024 pixels version of the dataset created when converted to the format needed to be used in training.

Google Colab was a feasible option to get started with the code and to train multiple sessions to achieve a stable training configuration. But there were certain issues with Colab that created the urge to move to a more stable and scalable platform. Although powerful GPUs, including NVIDIA

P100 and V100, are accessible in Colab, there is no guarantee that V100 GPUs are available at all times. Training with V100 GPUs is more than three times faster than on P100 GPUs based on benchmarks and also based on my own experience using both in my training. Additionally, as the data and training results are being stored on Google Drive, working with large datasets created issues for training sessions. Nas4-60k had to be used in 512x512 pixels and even with that change, there were many issues regarding the connection between Colab and Drive when training networks on that dataset.

4.4 Final Training Sessions

The set of hyper-parameters achieved in test training done in Google Colab were used to run final training sessions in Paperspace.[14] This platform was chosen considering the price, the availability of required GPUs, and ease of use. In the final pieces of training, Nas4-60k dataset was used in both 512x512 and 1024x1024 pixels, and networks were trained for more than 1,000 kimg. Networks were both trained from scratch and using transfer learning from a pre-trained network on FFHQ dataset provided by NVIDIA. The results of these training sessions were constantly monitored during the process to check the progress of the networks and the training was stopped when the visual improvements seemed to plateau.

The necessity of training in 1024x1024 pixels stems from the fact that due to the regularization of Nas4-60k dataset, letter combinations cover a considerably small area of the whole frame in the majority of the samples. This causes a resolution issue when the frame is 512x512 pixels regarding the fact that the calligraphy piece in the frame is of much smaller pixel dimensions.

However, using the same hyper-parameters obtained in training with 512x512 pixels data resulted in divergence and mode collapse, and as a result, I had to further decrease the learning rate to avoid it. With readjusting the hyper-parameters, stable training was attained, but the same fidelity of previous training on smaller data samples was not fully achieved (Figure 4.8). Furthermore, this upscaling to train with 1024x1024 pixels data added a significant time and cost to the training process. Training the networks on 1024x1024 pixels data was two times slower compared to training on 512x512 pixels data on the same GPU.

[14] https://www.paperspace.com

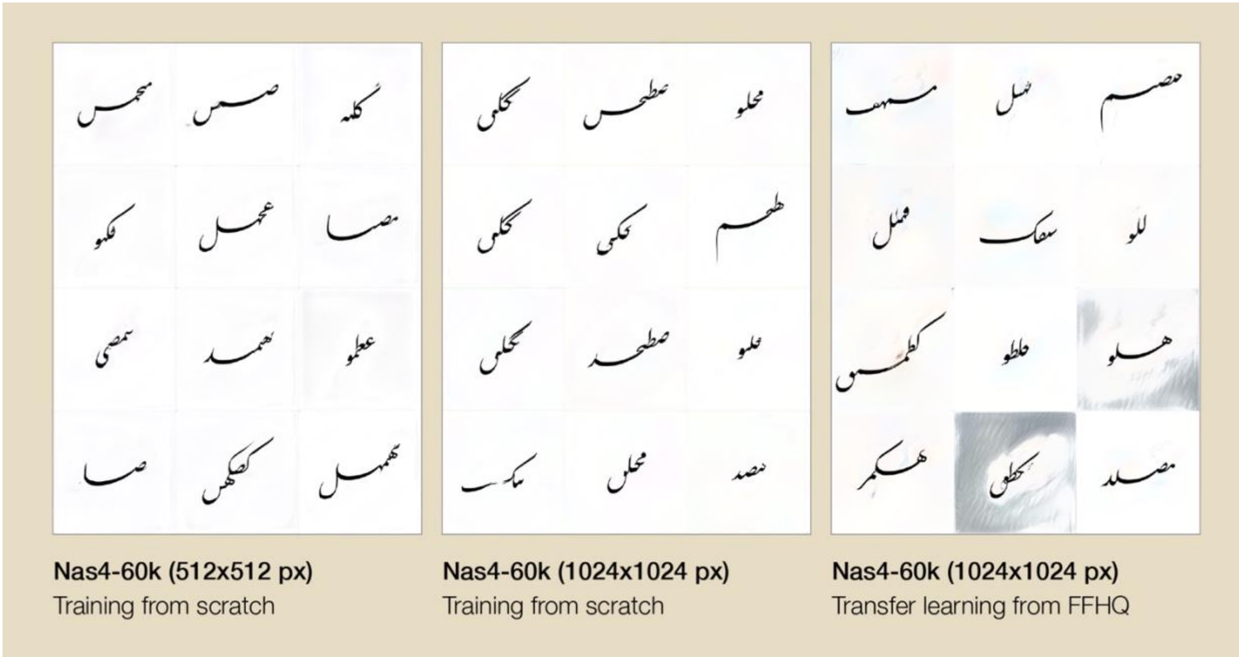

Figure 4.8 A comparison of randomly generated samples from networks trained on Nas4-60k in 512x512 px and 1024x1024 px with different training configurations.

A collection of randomly generated samples from networks with the best visual results is presented in Figure 4.8. In networks trained on the 1024x1024 px version of Nas4-60k dataset, letterforms are less accurately generated compared to the network trained on the 512x512 px dataset. Also, using transfer learning for the 1024x1024 px dataset creates better results overall. However, as can be seen in the figure, some residues from the previous training persist in parts of the latent space. For analysis and final works, the best visual results among different trained networks were used.

5 Results and Reflections

5.1 The Generation of Exact Letterforms

The trained network goes beyond capturing general calligraphic features and is able to generate distinguishable initial, medial and final forms of all letters existing in the original data samples. The connections between letters are also correctly formed. Among the samples, a good diversity of elongated and non-elongated forms is observed as well as alternative letterforms. Randomly generated samples also show a higher frequency of elongated samples that conforms with the distribution of elongated samples in the original dataset. Similarly, being less than ten percent of the original dataset, two and three-letter combinations are less frequently generated. A collection of randomly generated samples is presented in Figure 5.1.

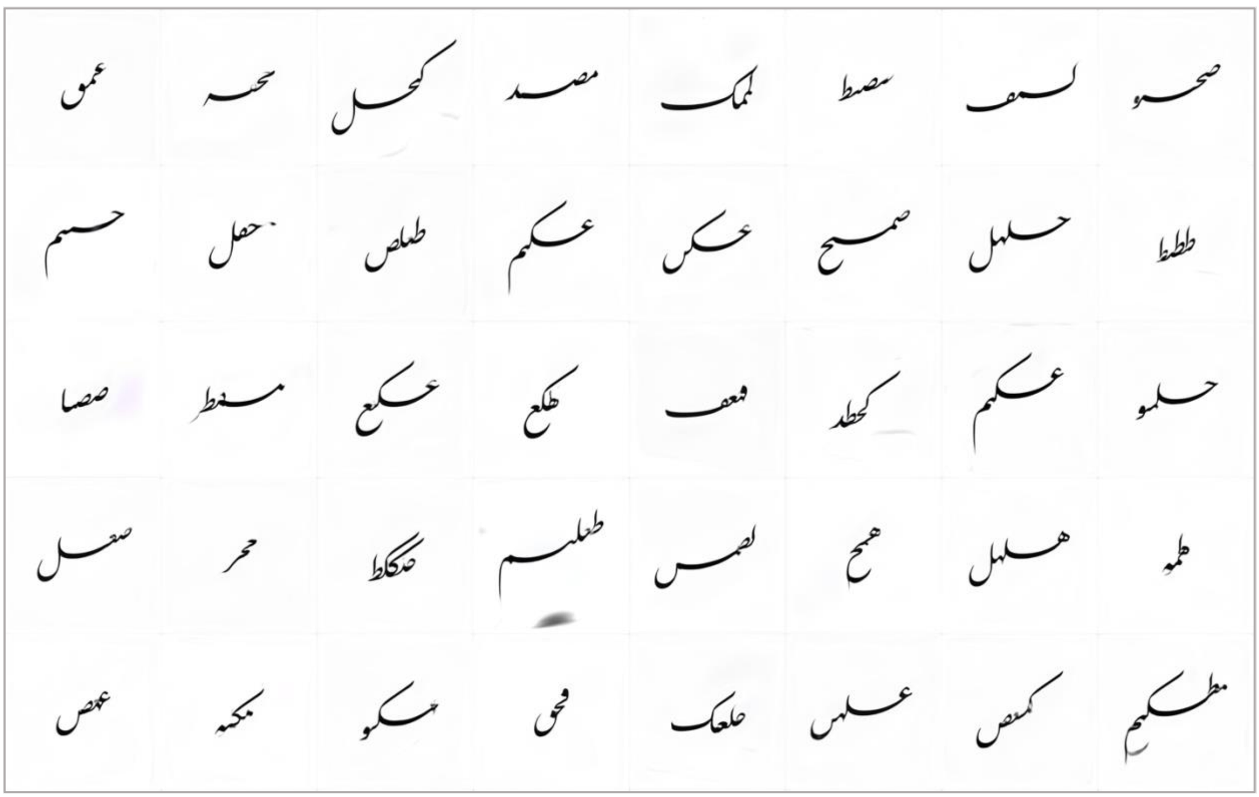

Figure 5.1 Random results from a network trained on Nas4-60k dataset for 1500 kimg, showing diversity in forms and extraction of fine calligraphic features.

5.2 The Generation of Five and Six-Letter Combinations

The dataset used in the final training sessions, Nas4-60k, comprises data samples with up to four-letter combinations. However, five and six-letter combinations are also observed among the generated samples with letters properly formed and correctly connected.

Figure 5.2 Curated examples of generated five and six-letter combinations.

5.3 Random Changes of Generated Results

One thing that was observed during training was random changes of the generated results from a fixed seed in different snapshots from the network. StyleGAN2-ada provides a collection of randomly generated samples from fixed seeds in every snapshot during the training. These samples help to evaluate the progress of the network visually and see how the network is improving a sample in the latent space. These random changes of the results made it difficult to evaluate the progress of the network based on comparing the generated results of a fixed seed. Instead, I had to collectively look at the results and decide whether the network was still improving, had plateaued, or was starting to diverge. Figure 5.3 shows a sequence of generated samples from three fixed seeds in the final stages of a training session.

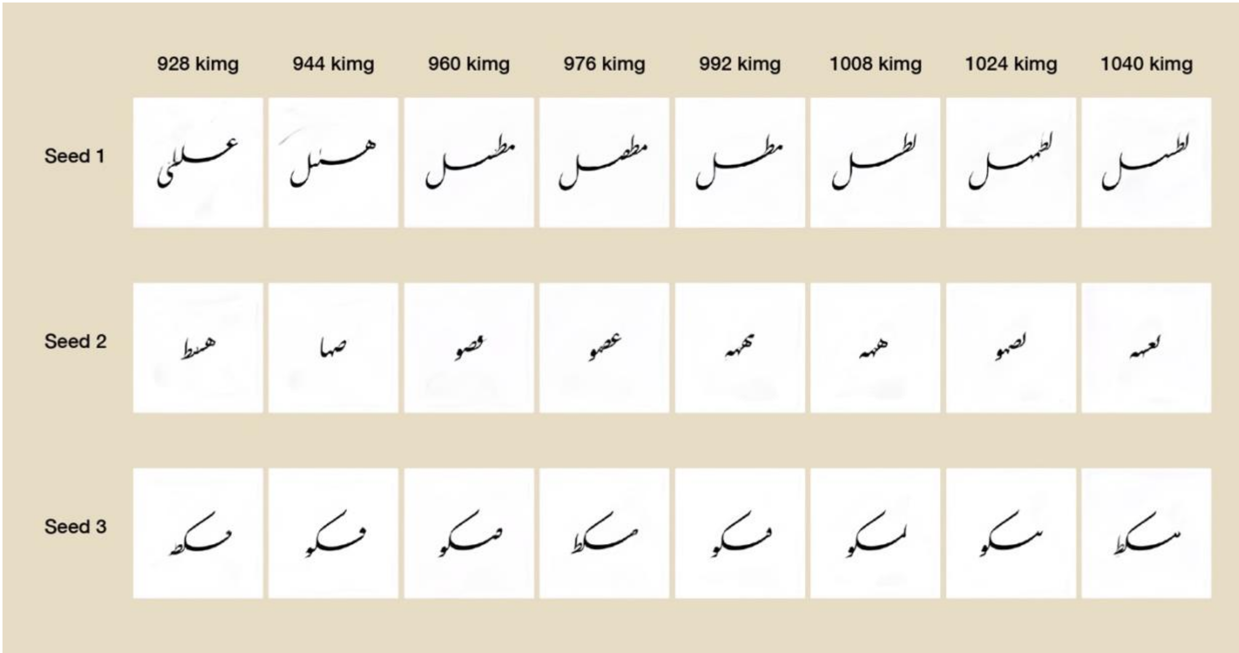

Figure 5.3 Examples of random changes in samples generated from fixed seeds.

As can be seen in Figure 5.3, generated samples from a fixed seed maintain general similarities, but random fluctuations happen in the details of the samples. A possible explanation of this can be the built-in noise inputs in StyleGAN that are added to each convolutional layer that creates stochastic variations to samples without changing high-level attributes. This feature is explained in the original paper [13] and is demonstrated in [40]. Although this feature of StyleGAN works well for natural image applications, it is disruptive in this case and I was not able to figure out how to deactivate it in the original code.

5.4 Smooth Transitions Between Forms

To further examine the qualities of the trained network, I started exploring the latent space through random interpolations to spot parts with significant and meaningful phenomena. The criteria for choosing these spots were set according to specific calligraphic features that I had in mind in relation to what I had learned from the affordances of GANs from other applications. The presented results are cherry-picked from numerous generated interpolations, but the observed qualities are mostly shared throughout the latent space.

Interpolating between different points of the latent space creates smooth morphs between forms. During the interpolations, samples retain calligraphic features and transitions happen in a meaningful way between letterforms and between elongated and non-elongated samples. Figure 5.4 demonstrates a sequence of 48 samples from a continuous interpolation between two points of the latent space.

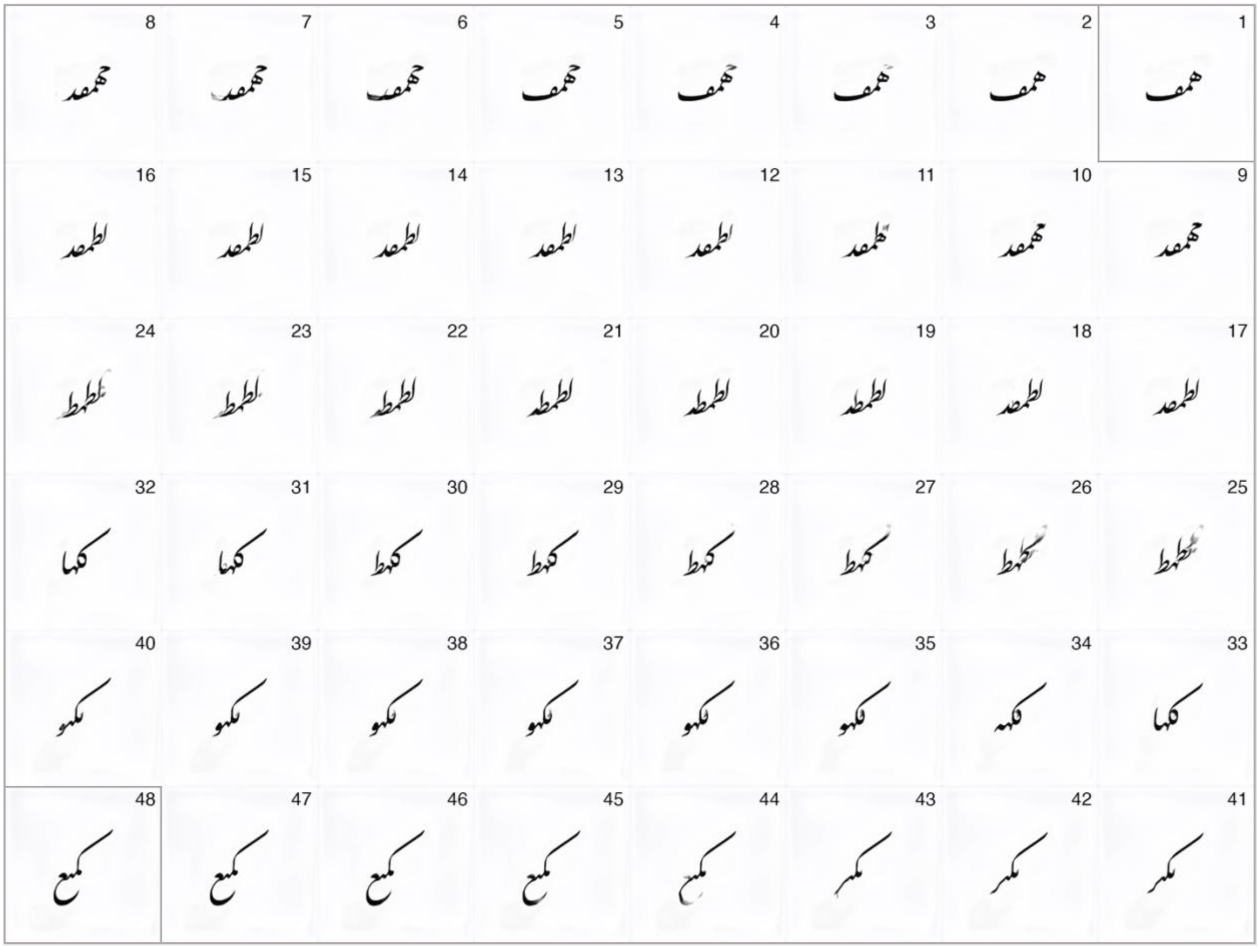

Figure 5.4 A sequence of 48 samples generated through interpolation between two points of the latent space.

Interpolating between closer points in the latent space shows more specific transitions between letterforms. Figure 5.5 demonstrates fine transitions of individual letters in a three-letter generated sample. In the first eight samples of the sequence, the first two letters transform into different letters, and in the next samples of the sequence, only the second letter morphs. Also, an interpolation between elongated and non-elongated forms is demonstrated in Figure 5.6.

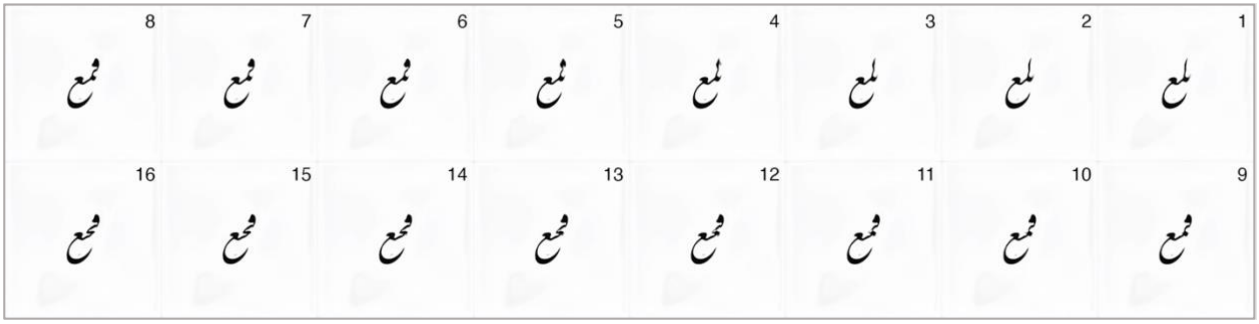

Figure 5.5 Fine transition of letterforms in interpolation between close points in the latent space.

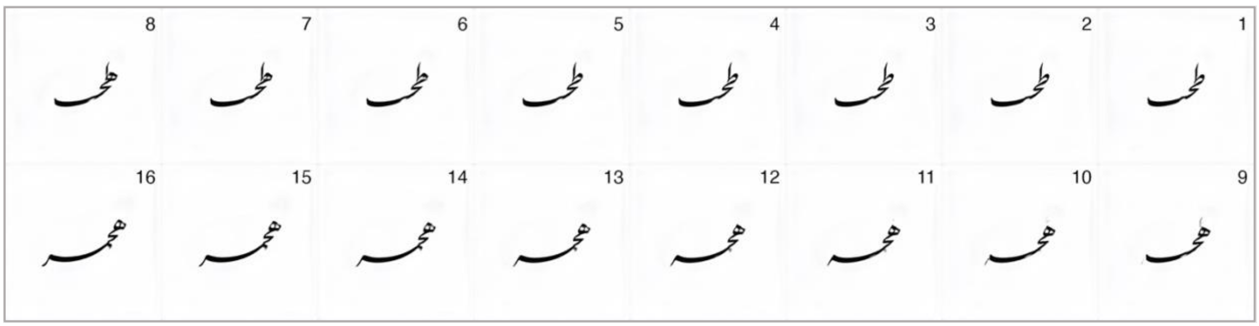

Figure 5.6 Interpolation between elongated and non-elongated forms.

Interpolating between different points of the latent space generally results in smooth transitions between forms and retains basic calligraphic features in intermediate steps. There are also some sudden changes observed in some samples that might be an indicator that mode collapse has happened in some parts of the latent space. An example of these abrupt changes can be seen between steps 5 and 8 in Figure 5.4, where the transition of the last letter in the samples cross-fade into each other instead of being smoothly morphed.

Overall, the latent space proves to have very promising qualities in terms of extracting calligraphic features, generating new samples, diversity of forms, and continuous representation of the data. This is an indicator that the network has captured a good estimate of the probability distribution of the original dataset.

5.5 What Is “New”?

GANs are able to generate new samples, in addition to estimating the probability distribution of the dataset. While the meaning of a “new” sample is intuitive in most cases, it can be defined in different ways for Islamic calligraphy depending on the network architecture and the design of the dataset. With the specific design of my dataset and the use of StyleGAN architecture in my training, it is important to inspect the newness in the generated results to get a better understanding of the experiment.

The generation of samples with more than four letters is solid evidence that the network is generating new samples that are definitely not in the original dataset (Figure 5.2). Based on the fact that letterforms and the connections between them are correctly generated, it can be comprehended that the network has captured letterforms and how they are connected as basic features from the original dataset. Given this, a new sample is a sample that has a combination of elements in a different way, and as the dataset already covers almost all possible four-letter combinations, the network is forced to generate new letter combinations that do not exist in the original dataset and creates those samples with a larger number of letters.

A question is then, why longer combinations are not generated? If the network has managed to capture individual letterforms as visual components and also the way they are connected, what keeps the network from generating combinations of more letters? This can be justified in light of the fact that the length of each sample is also a visual feature. The range of this feature in the generated samples is essentially within the range presented by the training dataset and will not exceed those limits. In other words, eight-letter samples, for example, would be much longer than any of the samples in the dataset and the network does not create those combinations as a result.

Another aspect of newness in the results is related to the fact that the network generalizes the visual features from discrete data samples and makes them accessible in a continuous way. This continuous representation creates a range of visual features in the generates samples from a discrete dataset. In addition to the generation of samples with exact letterforms, there are parts of the latent space that generate results with visual similarities to the dataset, but with less identifiable replication of exact letterforms. These aspects of the latent space are further discussed in 5.6 and 5.7.

5.6 Plasticity of Forms

The continuous representation of the dataset in the latent space reveals an important aspect of Islamic calligraphy which is the plasticity of forms. Despite the strict rules in Islamic calligraphy, there is a level of inherent plasticity in the structure of letters and words. There are certain parameters in forms that can be altered while the results remain within the standards of form and proportion. Plasticity is central to composition in Islamic calligraphy, especially in line justification. Examples of this plastic nature of forms in *nasta'līq* script are illustrated in Figure 5.7.

A closer look in the latent space of my trained network shows promising results regarding this aspect of Islamic calligraphy. Moving in close proximity of some samples results in fine changes of certain parts of the forms while other parts remain unchanged. Interestingly, these changes happen mostly in parts of the forms that are more apt to be plastic. Figure 5.8 demonstrates meaningful changes of these plastic features in some examples when moving in the vicinity of those forms in the latent space.

Figure 5.7 Examples of plasticity in nasta'līq script. A: The elongated form of letter sin. B: The final forms of letter be. C: Initial form of letter he. Works by Mir Emad (1554-1615), Mirza Gholamreza Esfahani (1830-1887).

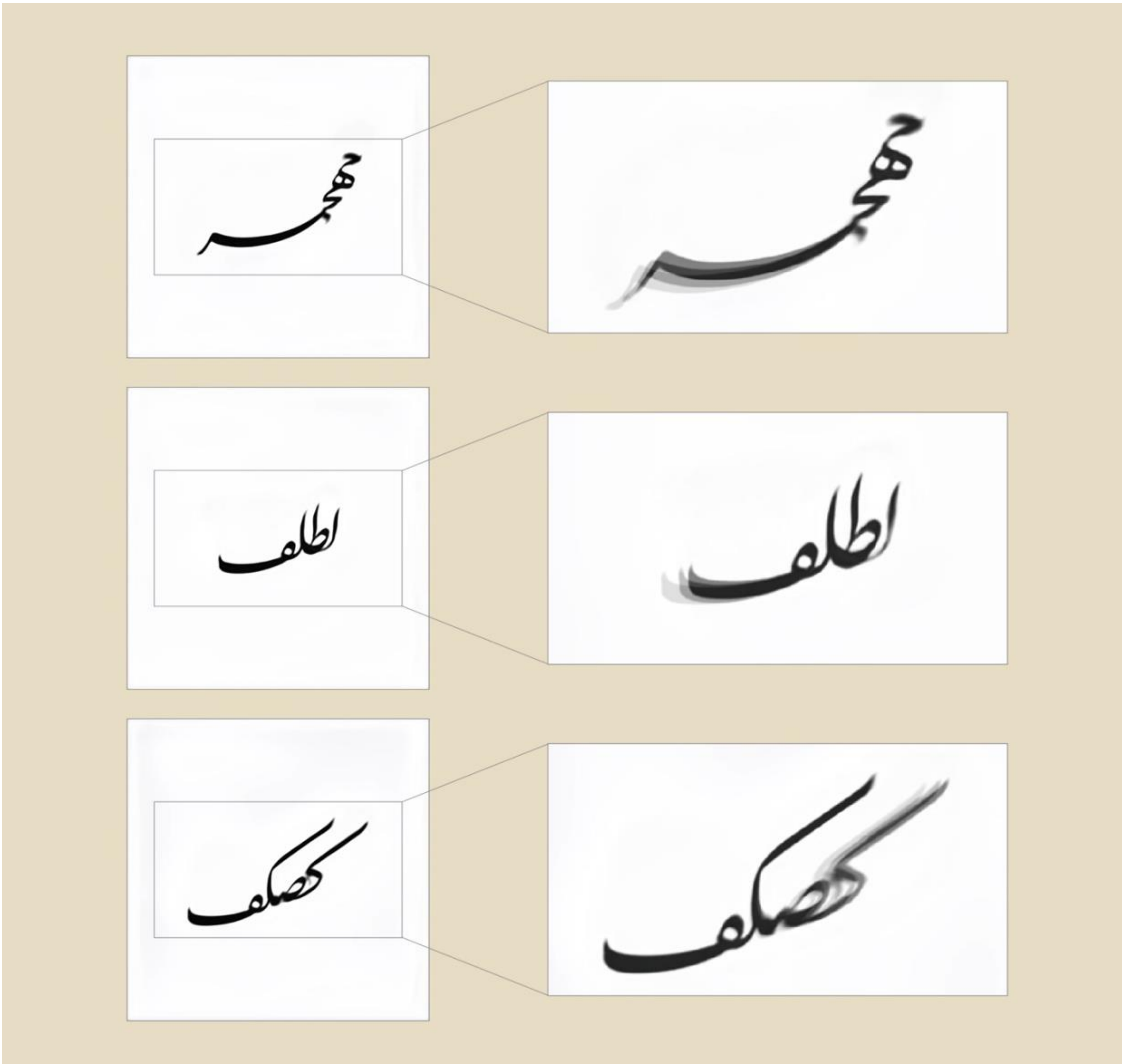

Figure 5.8 Left: Examples of generated samples. Right: The overlay of multiple samples generated from the latent space in the close proximity of the samples.

Some GAN architectures now do an impressive job in extracting interpretable features in a disentangled way. Regarding human faces, for instance, a model like StyleGAN2 can provide a disentangle feature extraction in different levels [40][41]. In GANs that provide such latent space, one can find directions in the latent space that correspond to specific features. However, as the latent space is a hyper-dimensional space (e.g., 512-dimensional in most configurations of

StyleGAN2), finding these directions might not be straightforward. A paper recently published by Härkönen et al. [42] introduces a technique that helps to identify meaningful directions in the latent space. In [43] the technique is demonstrated on GANs trained on different datasets.

It is not unreasonable to imagine a GAN can extract features from a calligraphy dataset in a similar disentangled fashion. In such a network, we can expect to find directions in the latent space that correspond to meaningful features in the dataset. Considering a dataset that is limited to one script and one style, if the data samples provide a level of diversity that is due to the plasticity of the forms, at least some of the features extracted in the network should correspond to plastic aspects of the dataset (see 3.1 for a discussion on diversity in a calligraphy dataset). As an example, we can expect to find a direction that controls the length of an elongated form without changing other features.

Considering the typographic tool that is used to create the dataset, a diversity of these features is not available in the dataset. Letter combinations in the dataset are formed of fixed glyphs of the font. At the most, there are limited options available in the font to choose from. In the case of elongations, there are only two options available with different lengths. What the network managed to do is to generalize these features and present a continuous range based on discrete and limited samples in the dataset.

5.7 Abstract Forms

In the earlier stages of training, samples show general similarities to the dataset, but without letterforms being exactly distinguishable (Figure 5.9). These samples have some high-level features of the dataset such as the stroke qualities and basic forms that are shared among different letters in elongated and non-elongated samples. However, hardly any letterform is generated in its exact form that can be legible. In this stage of training, the network extracts enough features from the dataset that creates a clear visual reference to the dataset but is not trained for long enough to capture fine details and lower-level features to generate letterforms in all details.

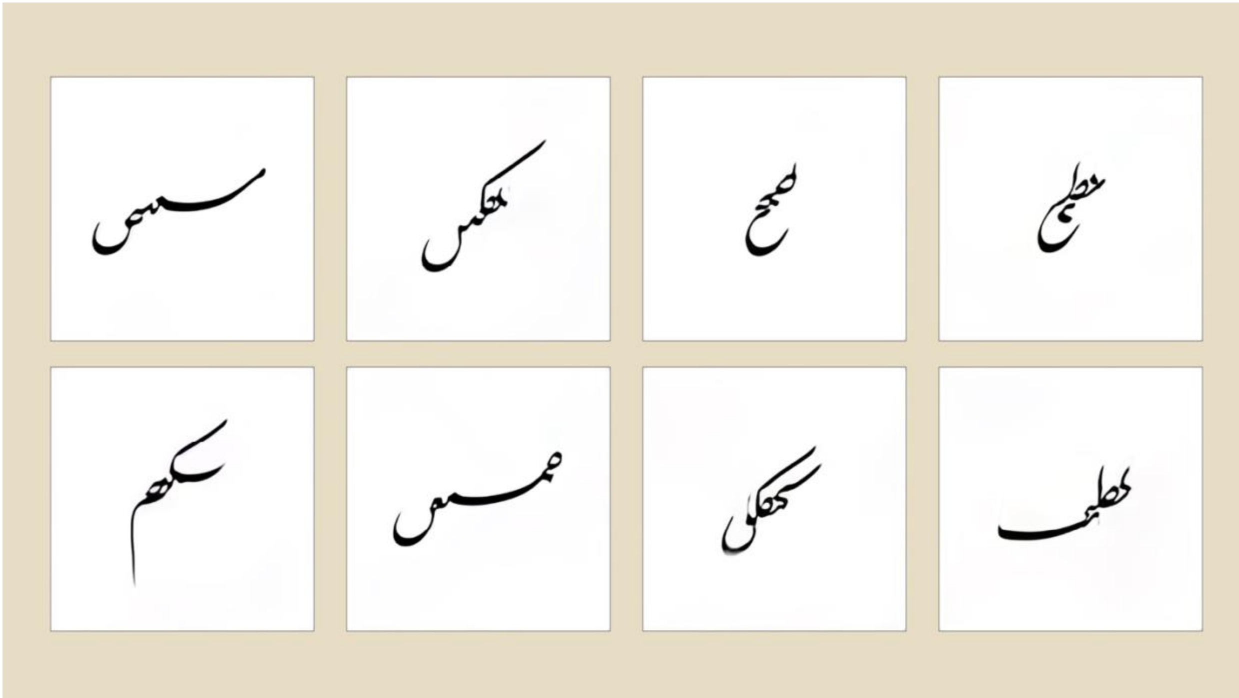

Figure 5.9 Curated results from earlier stages of training showing general visual features of the dataset without identifiable exact replication of letterforms.

The abstract qualities of the generated samples show a close relation to abstract works in Islamic calligraphy. Through the evolution of modern movements in Islamic calligraphy, some artists broke free from the traditional rules, totally deconstructed letters and used them as abstract elements in their works. These abstract works visually reference high-level features of Islamic calligraphy. In some works, one can identify the visual elements of one particular script, although no exact letterforms are used and the work is illegible. For example, the work by Faramarz Pilaram in Figure 5.10 references *nasta'līq* script. In this work, some similarities to letters can be seen, but letters have been deconstructed in a way that only general similarities can be identified and not the exact letters themselves. Also, in the work by el Seed, forms have a resemblance to scripts such as *thuluth* or *muhaqqaq*, but similarly illegible.

Figure 5.10 Examples of abstraction in Islamic calligraphy.

5.8 Incorporating the Results in the Final Exhibited Work

The generated results from the trained networks are used as compositional elements in final calligraphy pieces. These elements are created through moving in the latent space in loops to generate transitions between calligraphy forms. In my love for works of the great Iranian calligrapher Mirza Gholamreza Esfahani (1830-1887), generated morphs are composed into pieces in *siyah-mashq* form.

Siyah-mashq as an artform originates from the practice of writing letters and words repeatedly to achieve mastery. Eventually, it became a compositional style and a mode of personal artistic expression for calligraphers with a focus on visual qualities of the script rather than content (Figure 5.11). As Maryam Ekhtiar mentions: "… the technique and form of *siyah-mashq* triumph over content, the text having either negligible meaning or none whatsoever. These works have a strikingly abstract quality; the bold forms of the individual letters and their arrangement on the page are what provide the medium of communication between calligrapher and viewer. In many cases, the dots over or under letters are omitted, so as not to distract from the letters' shapes [10]."

Figure 5.11 Siyah-mashq pieces in nasta'līq script by Mirza Gholamreza Esfahani (1830-1887).

Making the final results was an effort to fill the gap between what I could imagine and what I could make. I could imagine a continuous space of possibilities in calligraphy but only static slices of this space were achievable using the conventional tools. Using GANs reveals a continuous space of visual possibilities in Islamic calligraphy and creates new modes of expression. Making these pieces was inspired by a dream; to submerge and float into an infinite space of forms in calligraphy.

Figure 5.12 Still images of the final calligraphy pieces created using the components form the latent space interpolations of the trained network. Full pieces are accessible via: https://www.arshsobhan.com/latent

6 Summary and Conclusion

Motivated by personal creative visions and informed by the gap between Islamic calligraphy and modern technology, the goal of this project was to demonstrate meaningful connections between GANs and Islamic calligraphy. For this, a purposeful approach was used to utilize a GAN architecture with a custom-made dataset to investigate the latent space's qualities of the trained network regarding fundamental features of Islamic calligraphy. The generated results were analytically explored and also were used to create calligraphy pieces with a new mode of expression.

My approach was closely informed by a deeper understanding of the latent space in GANs and its general affordances, which was acquired by studying core concepts of GANs and also by exploring GAN applications in other domains. This understanding helped me with the cross-domain translation of concepts and to extrapolate applications of GANs to Islamic calligraphy that itself led to the design of the datasets and the training of the networks.

For the training sessions, StyleGAN architecture was used with a gradual upscaling of the datasets and tools of working with the network. I started network training in a commercial software tool with limited control over training configurations and eventually moved to more direct ways of working with the network to have more control over the training process and the exploration of the results. Limiting factors such as the accessibility and the usability of GAN architectures, my limited programming skills, technical difficulties of dataset creation, and the resource-hungry nature of GANs affected the design and the scope of the experiment.

Through a custom method, two datasets were created in *nasta'līq* script for network training, namely Nas3-10k and Nas4-60k. The two datasets comprise isolated letter combinations using a reduced alphabet to create maximum visual diversity and minimum redundancy in the data samples. The employed method provides a feasible way of creating regularized datasets with different designs in Islamic calligraphy.

The results from the final training show successful extraction of different calligraphic features from the dataset. Diverse letterforms are observed in the generated samples with high fidelity and connections between letters are correctly formed. The network also manages to generalize the extracted features into the generation of five and six-letter combinations.

The latent space of the trained network provides a continuous representation of the features presented in the dataset. Samples generate through Interpolation between points in the latent space retain calligraphic features and show a smooth and meaningful transition of the forms. The results demonstrate that this continuous representation in the latent space corresponds to the plastic nature of forms which is an important feature of Islamic calligraphy which is not accessible by conventional digital tools.

Interpolations in the latent space also provide a new way of sampling from the learned representation of the dataset. The latent space provides infinite possibilities from finite samples and makes it seamlessly accessible. Continuous sampling unleashes novel aesthetic qualities of Islamic calligraphy and creates new modes of creative expression.

These results provide solid evidence that GANs can be considered for a wide range of applications in Islamic calligraphy. The latent space represents fundamental calligraphic features and unleashes new modes of creative expression that are not accessible with conventional tools. Taking into consideration that the network used in the experiment is mainly designed for natural images, and the fact that the dataset is the minimum viable option regarding the complexity and diversity of samples in Islamic calligraphy, the results encourage further research into the use of GANs for different creative and practical purposes in the domain of Islamic calligraphy.

The use of GANs seems to be a promising way of opening new horizons for Islamic calligraphy. With GANs designed for domain-specific challenges of Islamic calligraphy, the hope is that this new technology can significantly expand the creative and practical space of Islamic calligraphy for a wide spectrum of stakeholders. An informed approach toward GANs can compensate for the gap between Islamic calligraphy and modern technology and can drastically change the future of this artform.

6.1 Future Work

As the results of training GANs depend on both the design of the dataset and the network architecture, an immediate next step is designing similar task-specific experiments using different available GAN architectures and different datasets to get a better understanding of contributing factors in the richness and quality of the results in the latent space. Diversifying the design of datasets for different purposes can provide valuable perspectives regarding the performance of different GANs for different tasks in Islamic calligraphy.

The suggested method of creating Islamic calligraphy datasets can be improved in different ways. In the making of my datasets, I used a font that is originally created for end-user commercial use of *nasta'līq* script. The development of fonts specially designed to make datasets can significantly speed up the process and minimize human operator errors. Creating custom-made fonts for dataset creation purposes can help to automate the process and can also help to diversify the datasets with regard to different factors. Depending on the design of the dataset, other factors such as letters' frequency of usage and the proportion of the elongated forms to the whole data samples might be important to be taken into consideration.

A wider range of GAN architectures can be considered for various purposes. In addition to regular GANs, the use of other architectures, such as conditional GANs, can be examined for more complex tasks such as combining and customizing different styles and scripts of Islamic calligraphy. The fact that most of these networks require two datasets – either paired or unpaired – highlights the importance of improving the methods of dataset creation.

Many GANs extract a hierarchy of features from the training datasets, and the functionality of conditional GANs is directly related to this hierarchical extraction of features. Therefore, a vital next step would be a deeper look into the networks to see the structure of the extracted features. This can provide important inspiration about the visual feature structure of the dataset and can be used in designing better experiments and ultimately, better networks for Islamic calligraphy.

On a broader scale, there is no reason to put aside other branches of machine learning. Many aspects of Islamic calligraphy can be thought of in relation to Recurrent Neural Networks (RNNs) or Reinforcement Learning. There can be also important inspirations obtained from applying simple classification neural networks on datasets of Islamic calligraphy to see the feature structure that the network extracts from the domain.

An informed approach towards this new technology is not only about the technical aspects and many different factors should be taken into consideration. Different cultural impacts of implementing technology on the traditional school of calligraphy and the accessibility of the technology for artists and designers should also be considered both through an analytical and critical lens. The multifaceted factors surrounding the subject necessitates a collective approach that includes calligraphers, scientists, technologists, artists, designers, etc., to bring different perspectives and concerns to the table. But to make such a collective effort possible, a lot of work still needs to be done regarding awareness, education and accessibility of GANs.

References

[1] J. R. (Wayne) Osborn, “The type of calligraphy : writing, print, and technologies of the Arabic alphabet,” UC San Diego, 2008.

[2] B. Parhami, “Evolutionary Changes in Persian and Arabic Scripts to Accommodate the Printing Press, Typewriting, and Computerized Word Processing,” vol. 40, no. 2, p. 8, 2019.

[3] A. Bayar and S. Khalid, “How a Font Can Respect Basic Rules of Arabic Calligraphy,” *International Arab Journal of e-Technology*, vol. 1, Jan. 2009.

[4] D. M. Berry, “Stretching letter and slanted-baseline formatting for Arabic, Hebrew, and Persian with ditroff/ffortid and dynamic POSTSCRIPT Fonts,” *Softw. Pract. Exper.*, vol. 29, no. 15, pp. 1417–1457, Dec. 1999.

[5] S. Afshar, “Considering the old, designing the new - Sahar Afshar - ATypI 2017,” Sep. 16, 2017, Accessed: Nov. 24, 2020. [Online]. Available: https://www.youtube.com/watch?v=SBCPp-vxMBs.

[6] “The complex relationship between Arabic calligraphy and technology,” *Arab News*, Jun. 13, 2020. https://arab.news/wu8y9 (accessed Oct. 19, 2020).

[7] J. Auger, “Speculative design: crafting the speculation,” *Digital Creativity*, vol. 24, no. 1, pp. 11–35, Mar. 2013, doi: 10.1080/14626268.2013.767276.

[8] S. Dagher and C. Dagher, Eds., *Arabic Hurufiyya: Art and Identity*. Milano, Italy: Skira, 2017.

[9] S. Blair, *Islamic calligraphy*. Edinburgh University Press, 2006.

[10] M. Ekhtiar, “Practice Makes Perfect: The Art of Calligraphy Exercises (Siyāh Mashq) in Iran,” *Muqarnas*, vol. 23, pp. 107–130, 2006.

[11] P. Soltani, “Interview with Hamid Ajami: ‘Inventing a new script is not intentional,’” 2019. https://newspaper.hamshahrionline.ir/id/114262/%D8%A7%D8%A8%D8%AF%D8%A7%D8%B9-%D8%AC%D8%AF%DB%8C%D8%AF-%D8%A7%D8%B1%D8%A7%D8%AF%DB%8C-%D9%86%DB%8C%D8%B3%D8%AA.html (accessed Dec. 14, 2020).

[12] I. Goodfellow *et al.*, "Generative Adversarial Nets," in *Advances in Neural Information Processing Systems 27*, Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, Eds. Curran Associates, Inc., 2014, pp. 2672–2680.

[13] T. Karras, S. Laine, and T. Aila, "A Style-Based Generator Architecture for Generative Adversarial Networks," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2019, pp. 4396–4405, doi: 10.1109/CVPR.2019.00453.

[14] T. Karras, S. Laine, M. Aittala, J. Hellsten, J. Lehtinen, and T. Aila, "Analyzing and Improving the Image Quality of StyleGAN," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2020, pp. 8107–8116, doi: 10.1109/CVPR42600.2020.00813.

[15] A. Radford, L. Metz, and S. Chintala, "Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks," *arXiv:1511.06434 [cs]*, Jan. 2016, Accessed: Apr. 12, 2020. [Online]. Available: http://arxiv.org/abs/1511.06434.

[16] "This Person Does Not Exist." https://thispersondoesnotexist.com/ (accessed Jan. 26, 2021).

[17] H. Zhang *et al.*, "StackGAN: Text to Photo-Realistic Image Synthesis with Stacked Generative Adversarial Networks," in *2017 IEEE International Conference on Computer Vision (ICCV)*, Oct. 2017, pp. 5908–5916, doi: 10.1109/ICCV.2017.629.

[18] Y. Taigman, A. Polyak, and L. Wolf, "Unsupervised Cross-Domain Image Generation," *arXiv:1611.02200 [cs]*, Nov. 2016, Accessed: Jul. 30, 2020. [Online]. Available: http://arxiv.org/abs/1611.02200.

[19] G. Perarnau, J. van de Weijer, B. Raducanu, and J. M. Álvarez, "Invertible Conditional GANs for image editing," *arXiv:1611.06355 [cs]*, Nov. 2016, Accessed: Jan. 26, 2021. [Online]. Available: http://arxiv.org/abs/1611.06355.

[20] H. Zhang, V. Sindagi, and V. M. Patel, "Image De-Raining Using a Conditional Generative Adversarial Network," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 30, no. 11, pp. 3943–3956, Nov. 2020, doi: 10.1109/TCSVT.2019.2920407.

[21] Y. Li, S. Liu, J. Yang, and M. Yang, "Generative Face Completion," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jul. 2017, pp. 5892–5900, doi: 10.1109/CVPR.2017.624.

[22] R. A. Yeh, C. Chen, T. Y. Lim, A. G. Schwing, M. Hasegawa-Johnson, and M. N. Do, "Semantic Image Inpainting with Deep Generative Models," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jul. 2017, pp. 6882–6890, doi: 10.1109/CVPR.2017.728.

[23] C. Ledig *et al.*, "Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jul. 2017, pp. 105–114, doi: 10.1109/CVPR.2017.19.

[24] C. Vondrick, H. Pirsiavash, and A. Torralba, "Generating Videos with Scene Dynamics," *Advances in Neural Information Processing Systems*, vol. 29, 2016, Accessed: Mar. 16, 2021. [Online]. Available: https://proceedings.neurips.cc/paper/2016/hash/04025959b191f8f9de3f924f0940515f-Abstract.html.

[25] J. Wu, C. Zhang, T. Xue, W. T. Freeman, and J. B. Tenenbaum, "Learning a Probabilistic Latent Space of Object Shapes via 3D Generative-Adversarial Modeling," *arXiv:1610.07584 [cs]*, Jan. 2017, Accessed: Jan. 26, 2021. [Online]. Available: http://arxiv.org/abs/1610.07584.

[26] J.-J. Hwang, S. Azernikov, A. A. Efros, and S. X. Yu, "Learning Beyond Human Expertise with Generative Models for Dental Restorations," *arXiv:1804.00064 [cs]*, Mar. 2018, Accessed: Jan. 26, 2021. [Online]. Available: http://arxiv.org/abs/1804.00064.

[27] DeepLearningAI, "GANs for Good- A Virtual Expert Panel by DeepLearning.AI," Sep. 30, 2020. https://www.youtube.com/watch?v=9d4jmPmTWmc (accessed Jan. 20, 2021).

[28] "Helena Sarin." https://www.nvidia.com/en-us/gtc/ai-art-gallery/artists/helena-sarin/ (accessed Jan. 26, 2021).

[29] H. Sarin, "Eyeo 2019 - Helena Sarin," Aug. 16, 2019, Accessed: Jan. 26, 2021. [Online]. Available: https://vimeo.com/354276365.

[30] A. Elgammal, B. Liu, M. Elhoseiny, and M. Mazzone, “CAN: Creative Adversarial Networks, Generating ‘Art’ by Learning About Styles and Deviating from Style Norms,” *arXiv:1706.07068 [cs]*, Jun. 2017, Accessed: Jul. 12, 2020. [Online]. Available: http://arxiv.org/abs/1706.07068.

[31] N. Bendre, H. T. Marín, and P. Najafirad, “Learning from Few Samples: A Survey,” *arXiv:2007.15484 [cs]*, Jul. 2020, Accessed: Jan. 27, 2021. [Online]. Available: http://arxiv.org/abs/2007.15484.

[32] B. M. Lake, R. Salakhutdinov, and J. B. Tenenbaum, “Human-level concept learning through probabilistic program induction,” *Science*, vol. 350, no. 6266, pp. 1332–1338, Dec. 2015, doi: 10.1126/science.aab3050.

[33] T. Karras, M. Aittala, J. Hellsten, S. Laine, J. Lehtinen, and T. Aila, “Training Generative Adversarial Networks with Limited Data,” *arXiv:2006.06676 [cs, stat]*, Oct. 2020, Accessed: Jan. 09, 2021. [Online]. Available: http://arxiv.org/abs/2006.06676.

[34] H. Hayashi, K. Abe, and S. Uchida, “GlyphGAN: Style-consistent font generation based on generative adversarial networks,” *Knowledge-Based Systems*, vol. 186, p. 104927, Dec. 2019, doi: 10.1016/j.knosys.2019.104927.

[35] A. Mordvintsev, C. Olah, and M. Tyka, “Inceptionism: Going Deeper into Neural Networks,” *Google AI Blog*. http://ai.googleblog.com/2015/06/inceptionism-going-deeper-into-neural.html (accessed Apr. 12, 2020).

[36] T. Karras, S. Laine, and T. Aila, “Flickr-Faces-HQ Dataset (FFHQ),” *Flickr-Faces-HQ Dataset (FFHQ)*, 2018. https://github.com/NVlabs/ffhq-dataset (accessed Feb. 25, 2021).

[37] Z. Zhang, Y. Song, and H. Qi, “UTKFace,” *UTKFace*, 2017. https://susanqq.github.io/UTKFace/ (accessed Feb. 25, 2021).

[38] A. M. Moslehi, “Mirza Typeface,” *Personal portfolio website of Iranian Graphic Designer Amir Mahdi Moslehi*, 2017. http://www.amirmahdimoslehi.com/Works/mirza/ (accessed Feb. 27, 2021).

[39] “Runway | Make the Impossible,” *Runway*. https://runwayml.com/ (accessed Feb. 18, 2021).

[40] Tero Karras FI, “A Style-Based Generator Architecture for Generative Adversarial Networks,” Mar. 03, 2019. https://www.youtube.com/watch?v=kSLJriaOumA&feature=youtu.be (accessed Feb. 05, 2021).

[41] Tero Karras FI, “StyleGAN2,” Dec. 11, 2019. https://www.youtube.com/watch?v=c-NJtV9Jvp0&feature=youtu.be (accessed Feb. 05, 2021).

[42] E. Härkönen, A. Hertzmann, J. Lehtinen, and S. Paris, “GANSpace: Discovering Interpretable GAN Controls,” *Advances in Neural Information Processing Systems*, vol. 33, pp. 9841–9850, 2020.

[43] Erik Härkönen, “GANSpace: Discovering Interpretable GAN Controls,” Mar. 30, 2020. https://www.youtube.com/watch?v=jdTICDa_eAI&t=80s (accessed Feb. 05, 2021).