

Cologne University of Applied Sciences  
Faculty of Computer Science and Engineering Science

---

# M A S T E R T H E S I S

## Texture Asset generation through Transformer Models

Cologne University of Applied Sciences  
Campus Gummersbach  
Master Digital Sciences

written by:  
DENNIS GOSSLER  
11140150

**First examiner:** Prof. Dr. Olaf Mersmann  
**Second examiner:** Prof. Dr. Boris Naujoks

## **Abstract**

Blob

# Contents

## List of Figures

## Tabellenverzeichnis

<b>1</b>	<b>Introduction</b>	<b>1</b>
1.1	Related work . . . . .	1
1.2	Data . . . . .	1
1.2.1	Data Retrieval . . . . .	1
1.2.2	Data Cleaning . . . . .	1
1.2.3	Patterns in the data . . . . .	2
1.2.4	Data Synchronization . . . . .	2
1.3	Training process . . . . .	2
1.4	Models . . . . .	2
<b>2</b>	<b>Experiment</b>	<b>3</b>
2.1	Transformer behind the models . . . . .	3
2.1.1	evaluate the results . . . . .	3
2.2	Roll model . . . . .	3
2.2.1	Classification or Regression . . . . .	3
2.2.2	. . . . .	3
2.2.3	Discriminator . . . . .	3
2.3	Spiral model . . . . .	3
2.4	Problems . . . . .	3
<b>3</b>	<b>Conclusion</b>	<b>4</b>
3.1	Roll model . . . . .	4
3.2	Spiral model . . . . .	4
3.3	LLM Scaling Laws . . . . .	4
3.4	Stable diffusion/ GANs with convolutional neural network . . . . .	4
3.5	Further research . . . . .	4
<b>A</b>	<b>Appendix</b>	<b>i</b>
A.1	Unterabschnitt von Appendix . . . . .	i
<b>B</b>	<b>Eidesstattliche Erklärung</b>	

## List of Figures

## List of Tables

# 1 Introduction

In this thesis we will investigate the use of traditional Transformer model to generate texture asset to use as floor texture in e.g in video games. Transformers are usually used to ... . But in this thesis the focus is to use them to generate the next Pixel in a texture. Multiple different dataset of textures will be used from the internet to train the model. The final developed models will be trained on a GPU cluster in göttingen. The models will be evaluated on a set of metrics and the results will be compared to other models.

## 1.1 Related work

## 1.2 Data

### 1.2.1 Data Retrieval

In the internet, a wide variety of textures can be found, but not all of them are suitable for this task. The textures should be seamless, devoid of shadows, and free from any objects. Textures of floors, such as carpets, tiles, wood, concrete, and more, were utilized. Two approaches were employed to acquire the data for this thesis.

- Web Data Collection

The data for this project was obtained from various online sources. Numerous free texture providers, such as textures.com, texturehaven.com, and others, were utilized for data acquisition. Due to the limitation of downloading one texture at a time from most websites, a series of scripts were developed to compile a list of suitable textures and automate the downloading process. These scripts were created using UiPath and Python.

- Video Game Textures

The second approach involved using textures from video games. The advantage of this approach is that these textures are already seamless and often of high quality and quantity. However, a drawback is that these textures can be very repetitive. To obtain these textures, downward-facing recordings of the game were made, and the textures were extracted from the video. The major challenge with this approach is the need to disable shadows and all UI elements (HUD elements) in the game, which is not always possible.

### 1.2.2 Data Cleaning

To ensure that the data is consistent and free from elements that could corrupt the model, various cleaning steps were applied. For example, all images containing 3D objects were removed, especially those gathered from video games. During the recording of the floor,

unwanted debris or pieces of wood were often present, and all extracted frames were manually checked.

In the case of web-gathered textures, there were different folder structures, and it was necessary to standardize them across all data folders. Additionally, some of them had associated files that were irrelevant for this use case and needed to be discarded.

All the images were in high-definition (HD) quality, with a height of approximately 1080 pixels.

### **1.2.3 Patterns in the data**

(color, size, ...)

### **1.2.4 Data Synchronization**

In the thesis, a manually data synchronization routine is established to maintain data consistency between the supercomputer located in Berlin/Göttingen and the local workstation.

## **1.3 Training process**

(gpu cluster göttingen, my GPU, ...)

## **1.4 Models**

(LLMs, basic idea, roll model, spiral model)

## 2 Experiment

### 2.1 Transformer behind the models

(Illustration, Tensor board)

#### 2.1.1 evaluate the results

### 2.2 Roll model

(explanation, generating new content, , , )

#### 2.2.1 Classification or Regression

#### 2.2.2

#### 2.2.3 Discriminator

### 2.3 Spiral model

(explanation, Data to Spiral form, positional embedding, )

### 2.4 Problems

(layer norm(sigmoid vs clamp), color shift to gray (illustrations of average color), Text tokens vs imgs tokens)



## **3 Conclusion**

### **3.1 Roll model**

(strength, weaknesses)

### **3.2 Spiral model**

(strength, weaknesses)

### **3.3 LLM Scaling Laws**

### **3.4 Stable diffusion/ GANs with convolutional neural network**

### **3.5 Further research**

## A Appendix

### A.1 Unterabschnitt von Appendix

ABCDE

ABC

ABC

## **B Eidesstattliche Erklärung**

Ich versichere, die von mir vorgelegte Arbeit selbständig verfasst zu haben. Alle Stellen, die wörtlich oder sinngemäß aus veröffentlichten oder nicht veröffentlichten Arbeiten anderer entnommen sind, habe ich als entnommen kenntlich gemacht.

Sämtliche Quellen und Hilfsmittel, die ich für die Arbeit benutzt habe, sind angegeben. Die Arbeit hat mit gleichem Inhalt bzw. in wesentlichen Teilen noch keiner anderen Prüfungsbehörde vorgelegen.

---

(Ort, Datum, Unterschrift)