

# First Person Shooter level generation using Generative Adversarial Networks

Edoardo Giacomello

22 gennaio 2018



# Abstract



**Estratto in lingua Italiana**



---

*To someone...*





# Acknowledgments



# Contents

<b>Abstract</b>	<b>3</b>
<b>Estratto in lingua Italiana</b>	<b>5</b>
<b>Acknowledgments</b>	<b>9</b>
<b>1 Introduction</b>	<b>17</b>
1.1 Background: Level Design . . . . .	17
1.2 State of the Art . . . . .	17
1.2.1 Procedurally Generated Content . . . . .	17
1.2.2 Procedural Content Generation via Machine Learning (PCGML) . . . . .	17
1.3 Scope . . . . .	17
1.4 Thesis Structure . . . . .	17
1.5 Summary . . . . .	17
<b>2 Towards learn-based level generation</b>	<b>19</b>
2.1 Generative Adversarial Networks . . . . .	19
2.1.1 Overview . . . . .	19
2.1.2 Deep Convolutional GAN . . . . .	19
2.1.3 Wasserstein GAN . . . . .	19
2.1.4 Wasserstein GAN with Gradient Penalty . . . . .	19
2.1.5 Recent results . . . . .	19
2.2 Game of choice: DOOM . . . . .	19
2.2.1 Description . . . . .	19
2.2.2 Motivation . . . . .	19
2.2.3 Level Data Format . . . . .	19
2.3 Summary . . . . .	19
<b>3 Dataset and Data Representation</b>	<b>21</b>
3.1 Data Sources . . . . .	21
3.2 Native Data Format: WAD Files . . . . .	21
3.3 Target Data Format: Feature Maps and Vectors . . . . .	21
3.3.1 Overview and Motivation . . . . .	21
3.3.2 Feature Maps . . . . .	21
3.3.3 Graph Representation . . . . .	21
3.3.4 Scalar Features . . . . .	21
3.3.5 Data Encoding . . . . .	21
3.4 Input and Output Pipelines . . . . .	21
3.5 Summary . . . . .	21
<b>4 System Design and Overview</b>	<b>23</b>
4.1 System Overview . . . . .	23

<b>5</b>	<b>System Architecture</b>	<b>25</b>
5.1	Component View . . . . .	25
5.2	Neural Network Architecture . . . . .	25
<b>6</b>	<b>Experiment Design and Results</b>	<b>27</b>
6.1	Parameter Tweaking and Training Phase . . . . .	27
6.1.1	Techniques and "GAN Tricks" used . . . . .	27
6.1.2	Resulting Model . . . . .	27
6.2	Sampling the network . . . . .	27
6.3	Generated Samples . . . . .	27
6.4	In-Game Demonstration . . . . .	27
6.5	Summary . . . . .	27
<b>7</b>	<b>Results Evaluation and Conclusions</b>	<b>29</b>
7.1	Results Evaluation . . . . .	29
7.1.1	Evaluation metric . . . . .	29
7.1.2	Samples Evaluation . . . . .	29
7.1.3	Loss of accuracy . . . . .	29
7.2	Summary . . . . .	29
<b>8</b>	<b>Future Work</b>	<b>31</b>
8.1	Open Problems . . . . .	31
8.2	Possible Applications and future develops . . . . .	31

# List of Figures



# List of Tables





# Chapter 1

## Introduction

### 1.1 Background: Level Design

### 1.2 State of the Art

#### 1.2.1 Procedurally Generated Content

#### 1.2.2 Procedural Content Generation via Machine Learning (PCGML)

### 1.3 Scope

### 1.4 Thesis Structure

### 1.5 Summary



## Chapter 2

# Towards learn-based level generation

### 2.1 Generative Adversarial Networks

#### 2.1.1 Overview

#### 2.1.2 Deep Convolutional GAN

#### 2.1.3 Wasserstein GAN

#### 2.1.4 Wasserstein GAN with Gradient Penalty

#### 2.1.5 Recent results

### 2.2 Game of choice: DOOM

#### 2.2.1 Description

#### 2.2.2 Motivation

#### 2.2.3 Level Data Format

### 2.3 Summary



## Chapter 3

# Dataset and Data Representation

**Overview** This chapter aims to be an overview of the processes that led to the creation of the dataset the model is trained and evaluated with. In section 3.1 a reference to the data sources is given, then the focus of section 3.2 will be on how data is natively encoded for the game engine in order to give some hints on what are the difficulties to face in converting to and from that format in an automatic way. Section 3.3 will describe in detail what data is provided the dataset, that is how levels are converted from the native format and what features are extracted in order to provide an input for the neural network. Lastly, section 3.4 will give a brief overview of data formats used and transformation steps that have been done in order to give the possibility to replicate the dataset generation.

### 3.1 Data Sources

### 3.2 Native Data Format: WAD Files

### 3.3 Target Data Format: Feature Maps and Vectors

#### 3.3.1 Overview and Motivation

#### 3.3.2 Feature Maps

#### 3.3.3 Graph Representation

#### 3.3.4 Scalar Features

#### 3.3.5 Data Encoding

### 3.4 Input and Output Pipelines

### 3.5 Summary



## Chapter 4

# System Design and Overview

### 4.1 System Overview





## Chapter 5

# System Architecture

### 5.1 Component View

### 5.2 Neural Network Architecture



## Chapter 6

# Experiment Design and Results

### 6.1 Parameter Tweaking and Training Phase

#### 6.1.1 Techniques and "GAN Tricks" used

#### 6.1.2 Resulting Model

### 6.2 Sampling the network

### 6.3 Generated Samples

### 6.4 In-Game Demonstration

### 6.5 Summary



## Chapter 7

# Results Evaluation and Conclusions

### 7.1 Results Evaluation

#### 7.1.1 Evaluation metric

#### 7.1.2 Samples Evaluation

#### 7.1.3 Loss of accuracy

### 7.2 Summary



## Chapter 8

# Future Work

### 8.1 Open Problems

### 8.2 Possible Applications and future develops

