Klasyfikacja biomów Minecraft

Gleb Grinchik, Hleb Badzeika

June 2024

Abstract

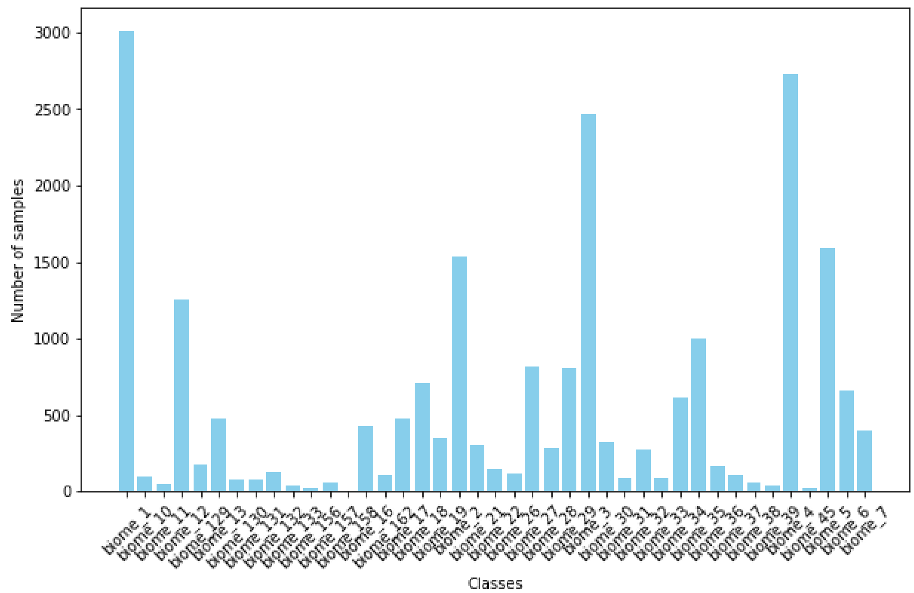
This paper presents a classification of biomes in Minecraft using various machine learning techniques, including convolutional neural networks (CNN), color histograms, Local Binary Patterns (LBP), and XGBoost. The goal was to build models capable of accurately identifying and classifying various biomes within the game. The Kaggle dataset was used to train and evaluate the models, and the results demonstrate the effectiveness of the approaches.

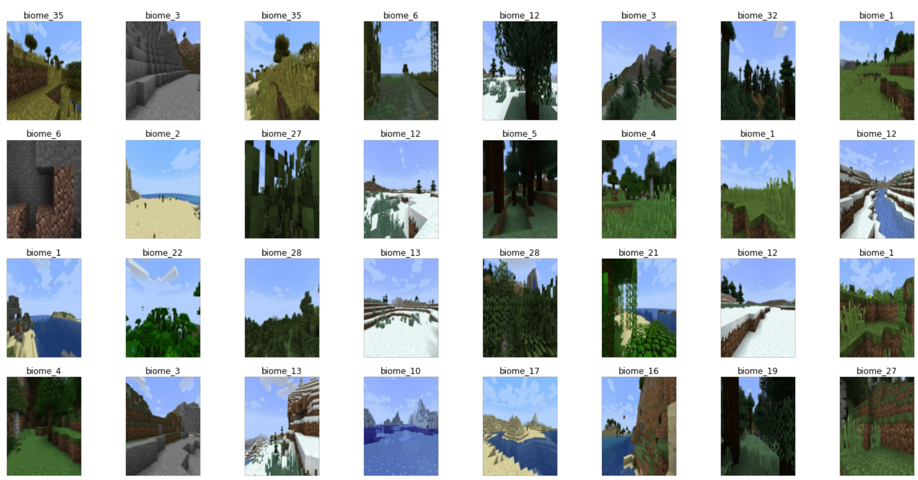
1 Introduction/Background

The objective of this project is to classify biomes in Minecraft using machine learning techniques. Minecraft is a popular sandbox game with a variety of biomes, each characterized by unique environmental features. Accurately classifying these biomes can enhance game-related data analysis and mod development. The dataset used for this project was obtained from Kaggle, containing images labeled by biome type.

2 Data Exploration

Dataset contained 22169 images of 41 Minecraft biomes. Each image had size of 320 x 180 pixels, which later was converted to 240 x 180. On the graphs below you can see labels distribution and examples of images:





3 Model Architecture

Several approaches were explored in this project:

1. **Convolutional Neural Networks (CNNs):**

   First Model: Consisted of a convolutional layer with 64 filters, a max-pooling layer, a convolutional layer with 32 filters, a flattening layer, a dense layer with 32 units, and an output layer with 41 units.

Second Model: Similar architecture with variations in filter sizes and activation functions.

Third Model: Huge model with more layers.

Also some other network architectures were tested but didn't show good results.

2. **Color Histograms and Local Binary Patterns (LBP) + XGBoost:**

   - Color features were extracted using histograms in the HSV color space.
   - Texture features were captured using Local Binary Patterns (LBP) from grayscale images.
   - Features from both methods were combined and fed into an XGBoost model for classification.
   - Hyperparameters of the XGBoost model were tuned using GridSearchCV.

4 Results

We evaluated the performance of the models using precision, recall, F1-score, and accuracy metrics. Here are some key takeaways from the results tables:

Class	Precision	Recall	F1-Score	Support
badlands	0.89	0.90	0.89	620
badlands_plateau	0.47	1.00	0.64	25
beach	0.33	0.73	0.46	11
birch_forest	0.70	0.64	0.67	254
birch_forest_hills	0.50	0.03	0.06	29
dark_forest	0.25	0.15	0.18	96
dark_forest_hills	0.00	0.00	0.00	12
desert	0.29	0.20	0.24	20
desert_hills	0.50	0.09	0.15	23
desert_lakes	0.00	0.00	0.00	4
flower_forest	0.00	0.00	0.00	5
forest	0.00	0.00	0.00	12
frozen_ocean	0.00	0.00	0.00	1
frozen_river	0.57	0.69	0.63	84
giant_tree_taiga	0.08	0.08	0.08	13
giant_tree_taiga_hills	0.68	0.32	0.44	106
gravelly_mountains	0.18	0.11	0.14	147
jungle	0.06	0.03	0.04	63
jungle_hills	0.74	0.87	0.80	310
lukewarm_ocean	0.61	0.87	0.72	60
modified_gravelly_mountains	0.00	0.00	0.00	28
mountains	0.50	0.42	0.46	26
plains	0.65	0.81	0.72	181
river	0.28	0.08	0.13	59
savanna	0.59	0.72	0.65	155
savanna_plateau	0.67	0.68	0.68	463
snowy_beach	0.60	0.47	0.53	83
snowy_mountains	0.18	0.26	0.21	19
snowy_taiga	0.51	0.47	0.49	60
snowy_taiga_hills	0.07	0.04	0.05	28
snowy_taiga_mountains	0.30	0.34	0.32	116
snowy_tundra	0.86	0.92	0.89	188
sunflower_plains	0.50	0.17	0.26	23
swamp	0.71	0.85	0.77	20
taiga	0.43	0.23	0.30	13
taiga_hills	0.80	0.57	0.67	7
taiga_mountains	0.68	0.79	0.73	539
tall_birch_hills	0.25	1.00	0.40	1
wooded_badlands_plateau	0.63	0.79	0.71	307
wooded_hills	0.82	0.82	0.82	147
wooded_mountains	0.34	0.35	0.34	75
Accuracy			0.66	4433
Macro avg	0.42	0.43	0.40	4433
Weighted avg	0.63	0.66	0.64	4433

Table 1: Report for Model v3

Class	Precision	Recall	F1-Score	Support
badlands	0.81	0.90	0.85	29
badlands_plateau	0.75	0.60	0.67	10
beach	0.64	0.65	0.65	89
birch_forest	0.65	0.66	0.66	166
birch_forest_hills	0.53	0.19	0.28	48
dark_forest	0.53	0.48	0.50	148
dark_forest_hills	0.00	0.00	0.00	13
desert	0.78	0.90	0.84	334
desert_hills	0.71	0.42	0.53	85
desert_lakes	0.50	0.07	0.12	14
flower_forest	0.87	0.57	0.68	23
forest	0.61	0.77	0.68	576
frozen_ocean	0.93	0.93	0.93	14
frozen_river	0.50	0.12	0.20	8
giant_tree_taiga	0.59	0.43	0.49	54
giant_tree_taiga_hills	0.50	0.07	0.12	15
gravelly_mountains	0.40	0.12	0.19	16
jungle	0.72	0.64	0.68	61
jungle_hills	0.47	0.30	0.37	23
lukewarm_ocean	0.00	0.00	0.00	5
modified_gravelly_mountains	0.64	0.43	0.51	21
mountains	0.67	0.87	0.76	464
plains	0.83	0.91	0.87	588
river	0.51	0.42	0.46	72
savanna	0.91	0.93	0.92	212
savanna_plateau	0.87	0.51	0.65	39
snowy_beach	0.95	0.86	0.90	22
snowy_mountains	0.51	0.26	0.34	100
snowy_taiga	0.83	0.97	0.89	64
snowy_taiga_hills	0.50	0.06	0.11	16
snowy_taiga_mountains	0.00	0.00	0.00	1
snowy_tundra	0.77	0.88	0.82	250
sunflower_plains	1.00	0.15	0.26	33
swamp	0.74	0.71	0.73	129
taiga	0.66	0.78	0.72	358
taiga_hills	0.35	0.15	0.20	62
taiga_mountains	0.00	0.00	0.00	6
tall_birch_hills	0.00	0.00	0.00	6
wooded_badlands_plateau	0.50	0.44	0.47	9
wooded_hills	0.22	0.08	0.11	142
wooded_mountains	0.41	0.23	0.29	121
Accuracy			0.70	4446
Macro avg	0.57	0.45	0.47	4446
Weighted avg	0.67	0.70	0.67	4446

Table 2: Report for Color Histogram + LBP + XGBoost

5 Discussion/Conclusion

The results demonstrate the effectiveness of machine learning models, for classifying large and well-defined biomes in Minecraft. The models achieved promising accuracy in identifying these biomes. However, the same models yielded lower performance for smaller biomes within the dataset.

6 Training Details

- **CNNs:** Trained for 10 epochs using the Adam optimizer and sparse categorical cross-entropy loss.
- **XGBoost:** Features were extracted from the images, and the model was trained with GridSearchCV used for hyperparameter optimization.

7 Data/Code availability

- **GitHub:** https://github.com/glebbadzeika/Minecraft_Biomes_Detection
- **Kaggle:** https://www.kaggle.com/datasets/willowc/minecraft-biomes