



Monte Carlo

BY

Lukas Momberg	11141259	AI
Jona Siebel	11141394	AI
Dennis Goßler	11140150	AI
Thomas Donst	11138843	AI
Patrick Schuster	11126452	AI

Wie funktioniert euer Partikelfilter?

Cut-Off-Werte

- "OutOfBounds" Außerhalb der Map
- Kollision in eine Wand
- Unrealistische Werte "zu weit weg"

Warum genau diese Werte?

- Es waren die offensichtlichsten & essentiellsten.
- Andere werte mögen möglich gewesen sein aber waren eher luxus als wirklich ein Must Have

Was hat gut funktioniert, was nicht und (zumindest spekulativ) warum?

- Cutoff Filter (Löscht manche guten Werte zu stark oder schwach raus)[verwechslung]

```
private float RateFromData(float[] target, float[] data, int dataSize)
{
    float rating = 0;

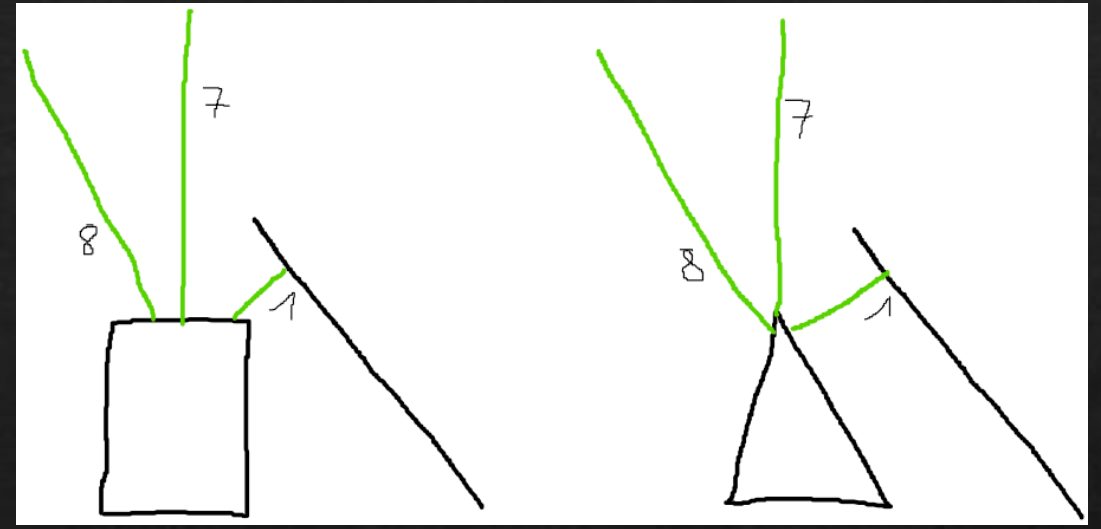
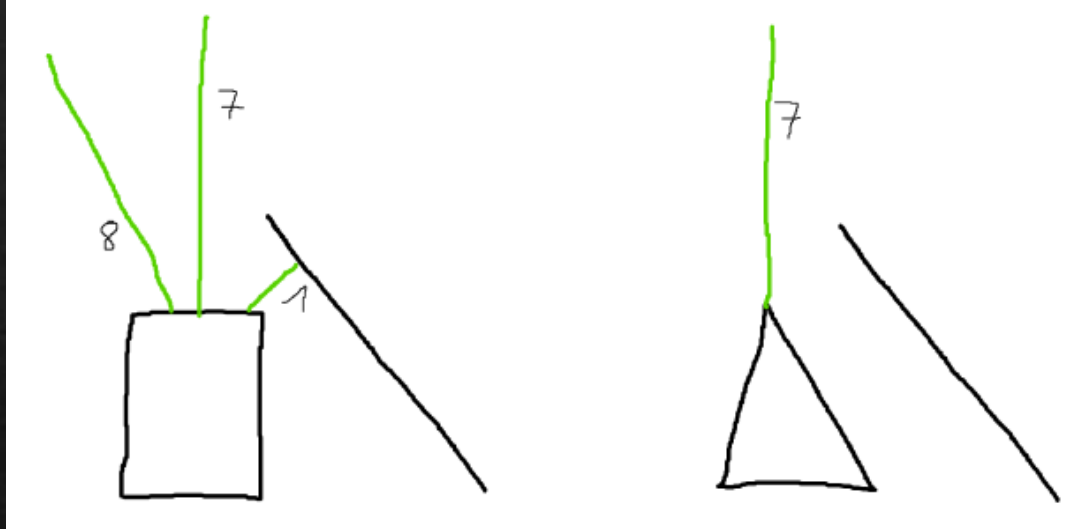
    for (int v = 0; v < dataSize; v++)
    {
        float value = data[v];
        bool impossibleValue = value == float.PositiveInfinity;

        if (impossibleValue)
        {
            rating = 0; // impossible values are valed as 0
            break;
        }
        else
        {
            float distanceToWall = data[v] - target[v];
            float distanceToWallAbsolute = Mathf.Abs(distanceToWall);

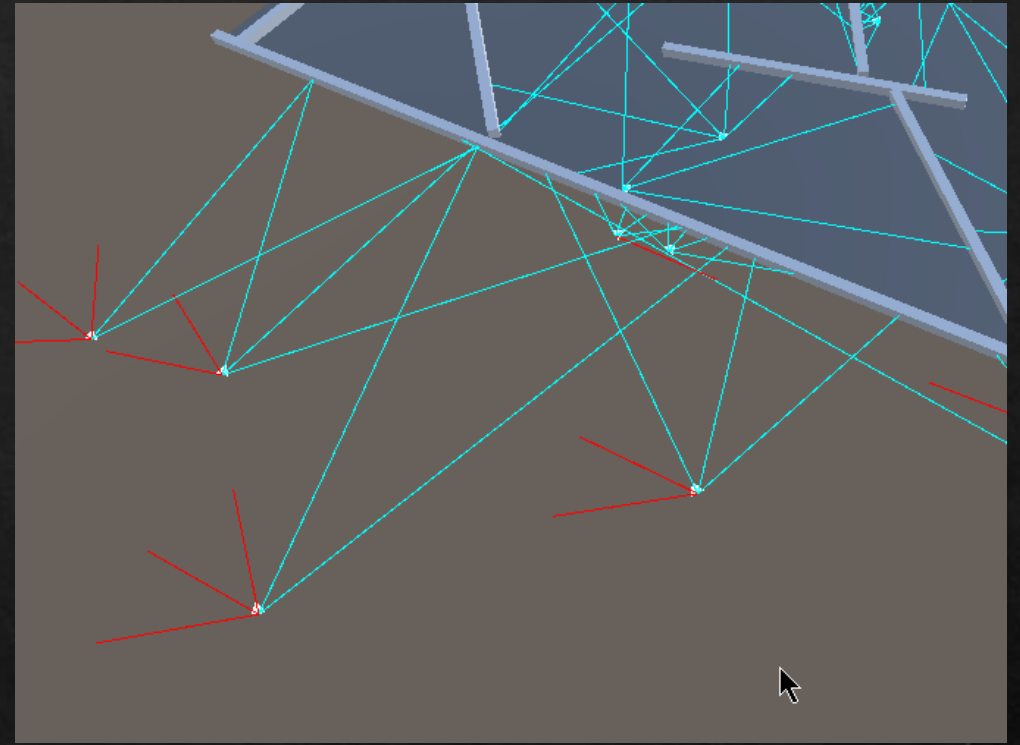
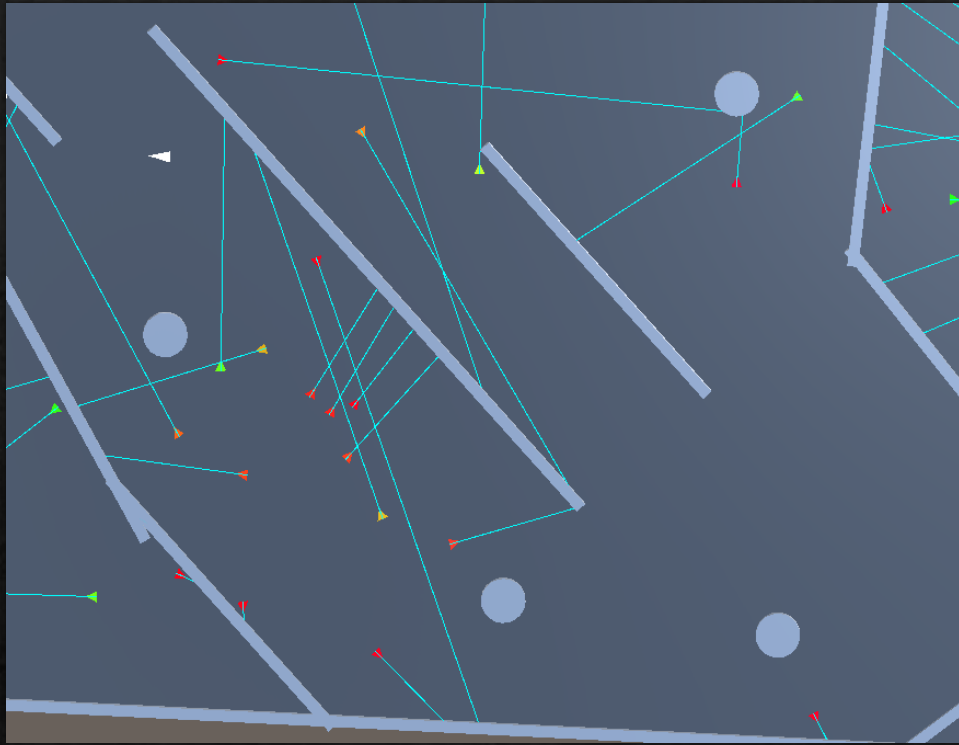
            if (distanceToWallAbsolute > CutOfRange)
            {
                rating = 0;
            }
            else
            {
                rating += distanceToWallAbsolute;
            }
        }
    }

    rating /= (float)dataSize;

    return rating;
}
```



Messunterschied - Problematik



Messunterschied - Lösung

Reseed / ReShuffle

Wann Reseedet ihr?

- Wenn die Distanz einen unterschied höher als 5 hat oder Out of Bounds ist

Auf welche Art(en) Reseedet ihr?

- Wir sammeln die schlechten und positionieren sie bei die guten Partikel in einem Radius von Maximal 0.5

Warum genau diese Art(en)?

- Einfach zu verstehen und die Umsetzung lag auf der Hand

```

1 Verweis
private void GhostReposition()
{
    List<int> spawnCandidates = new List<int>();
    List<int> reposCandidates = new List<int>();
    SensorDataSet sensorDataSet = _sensorDataHistory.GetCurrent();

    TotalScore = 0;

    for (int i = 0; i < sensorDataSet.SensorDataVieldOfViewListSize; i++)
    {
        SensorDataVieldOfView sensorDataVieldOfView = sensorDataSet.SensorDataVi;
        float rating = sensorDataVieldOfView.Rating;
        bool reposition = rating == 0;
        bool goodScore = rating >= 1f;

        TotalScore += rating;

        if (reposition)
        {
            reposCandidates.Add(i);
        }

        if (goodScore)
        {
            spawnCandidates.Add(i);
        }
    }
}

```

```

if (spawnRange > 0)
{
    foreach (int index in reposCandidates)
    {
        int spawnIndex = Random.Range(0, spawnRange - 1);
        int spawnID = spawnCandidates[spawnIndex];

        GhostController ghostNonHit = cs.ghosts[index];
        GhostController spawnCadidatPos = cs.ghosts[spawnID];

        SpawnAroundGhost(ghostNonHit, spawnCadidatPos, RepositionRadius);
    }

    spawnCandidates.Clear();
    reposCandidates.Clear();
}

```

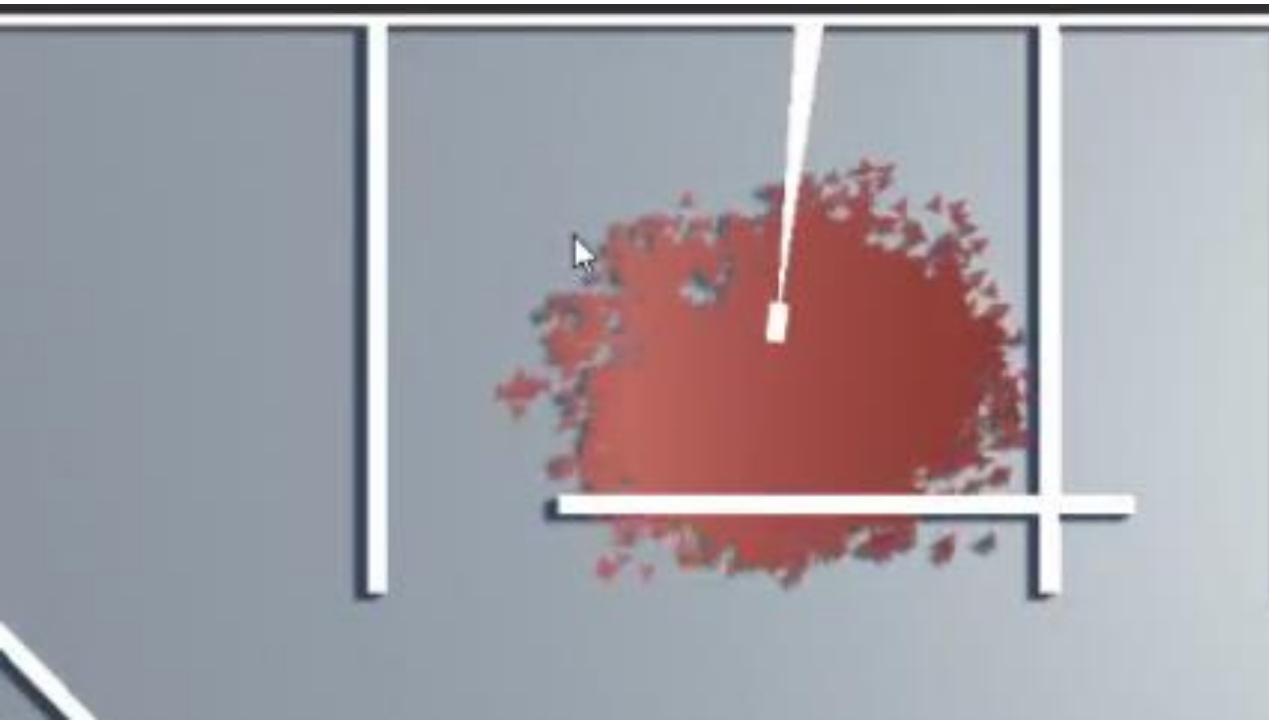
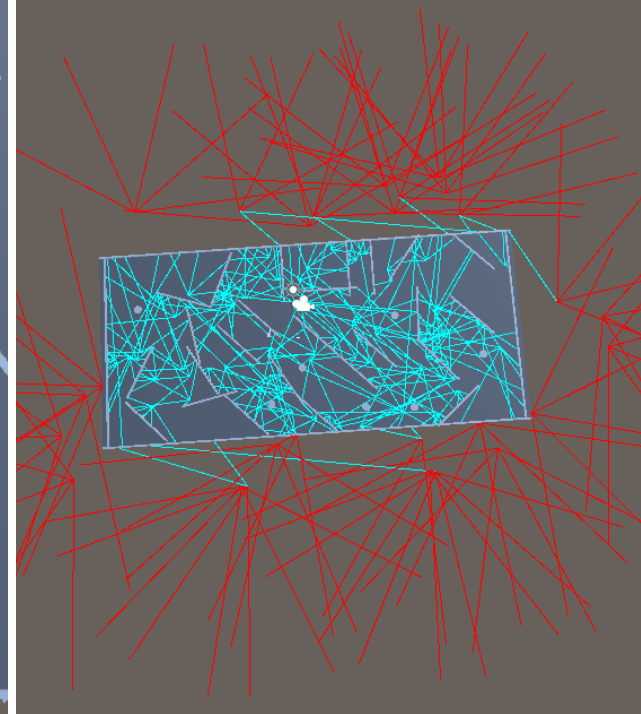
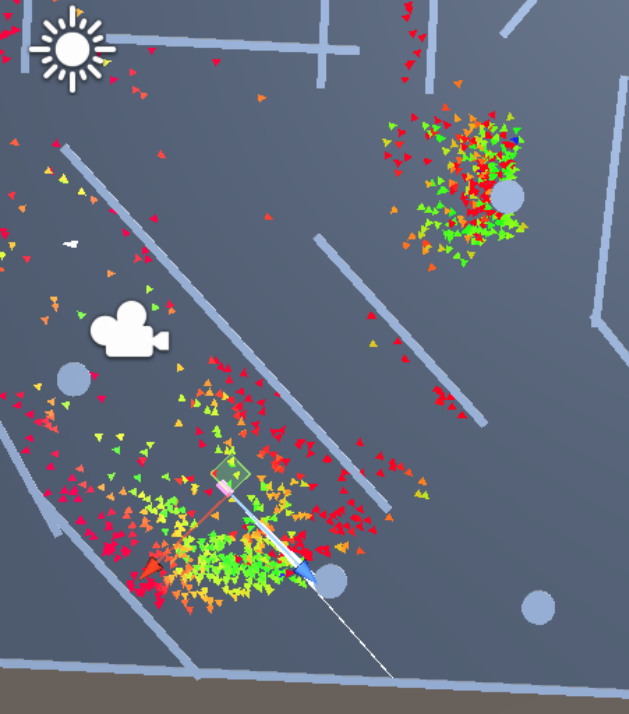

Video

Abbruchbedingung

Was ist eure
Abbruchbedingung?

- Wenn der Durchschnitt aller Partikel einen Score von 80% erreicht

```
Debug.Log("Score : " + TotalScore);  
  
if (TotalScore >= 0.80f)  
{  
    Debug.Log("Finsihed");  
    Debug.Break();  
}
```



Ende