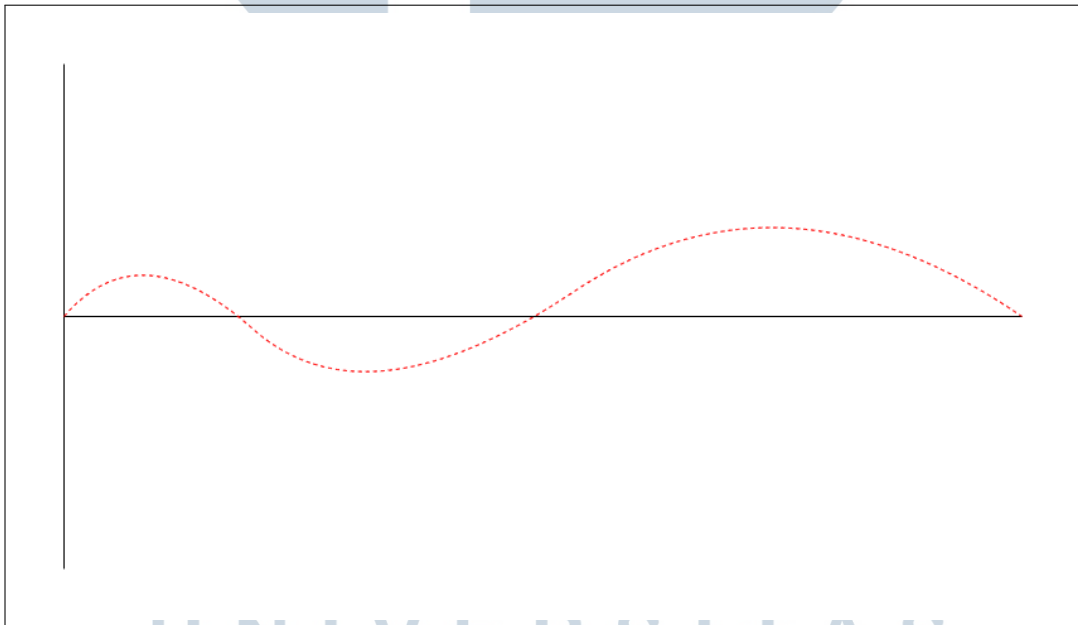


## BAB 2

### LANDASAN TEORI

#### 2.1 Perlin Noise

Perlin Noise biasanya digunakan dalam komputer grafis untuk membuat pola natural yang dapat diaplikasikan dalam satu dimensi, dua dimensi, dan tiga dimensi [7]. Pola natural pada Perlin Noise merupakan pola semi acak dimana, tingkat interval ketidak teraturannya setiap nilainya tidak jauh. Perlin Noise diterapkan dengan menggunakan gradien *pseudo random* yang disesuaikan dengan titik dalam ruang, dan dilakukan interpolasi antara titik titik tersebut [8]. Hal tersebut membuat jarak antar titik tidak terputus jauh sehingga dapat menciptakan visualisasi tonjolan yang halus. Contoh hasil nilai dari metode Perlin Noise dapat dilihat pada gambar 2.1.



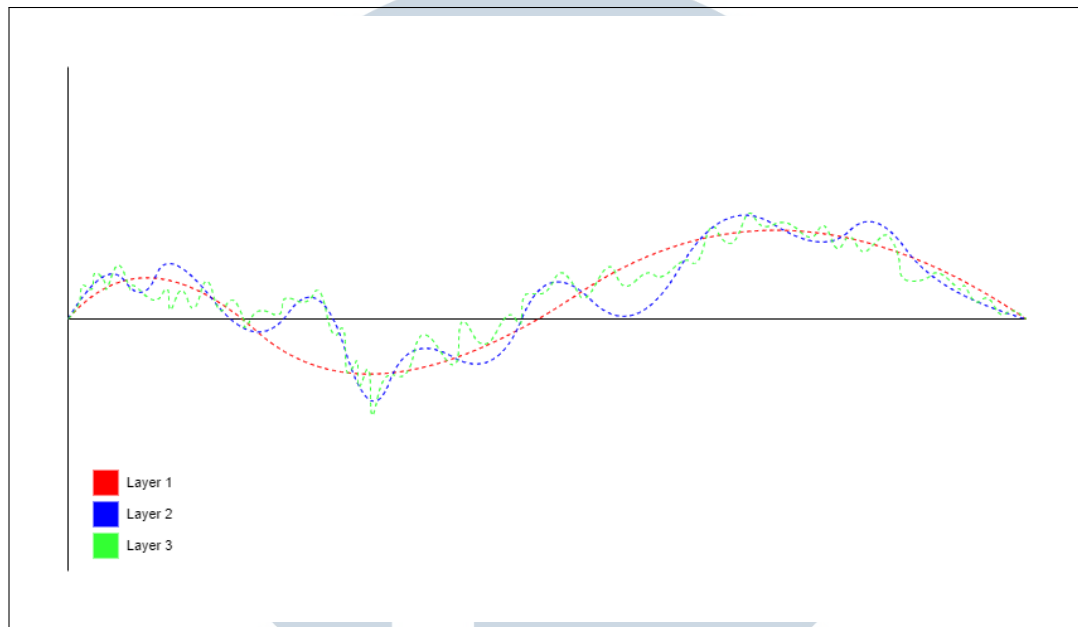
Gambar 2.1. Contoh visualisasi hasil Perlin Noise

Sumber: [8]

#### 2.2 Fractal Brownian Motion

Fractal Brownian Motion digunakan untuk menambah detail dan kompleksitas tekstur dengan melakukan beberapa iterasi pada *noise*, dimana hasil dari beberapa iterasi *noise* tersebut digabungkan/*noise layering* [9].

Gambaran/contoh dari hasil Fractal Brownian Motion atau *noise layering* dapat dilihat pada gambar 2.2.



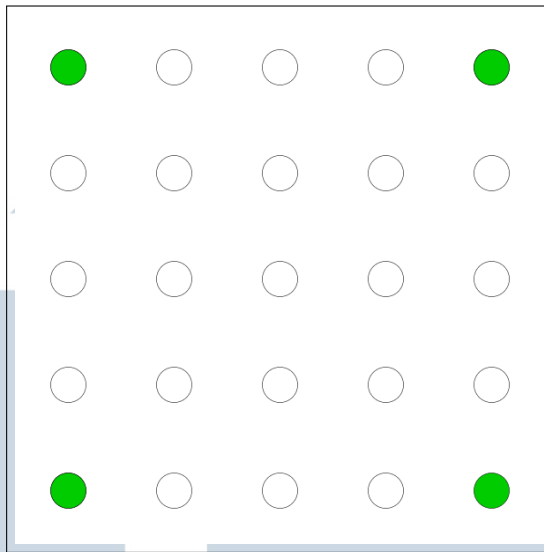
Gambar 2.2. Contoh visualisasi hasil Fractal Brownian Motion

Sumber: [9]

Dalam pengaplikasiannya Fractal Brownian Motion mempunyai tiga variabel utama yaitu **Octave** yang berarti jumlah iterasi/lapisan *noise*, **Lacunarity** yang berarti tingkatan kepadatan dari frekuensi, dan **Persistence** yaitu besaran pengaruh dari *noise* kepada jumlah seluruh *noise* [10].

### 2.3 Diamond-square

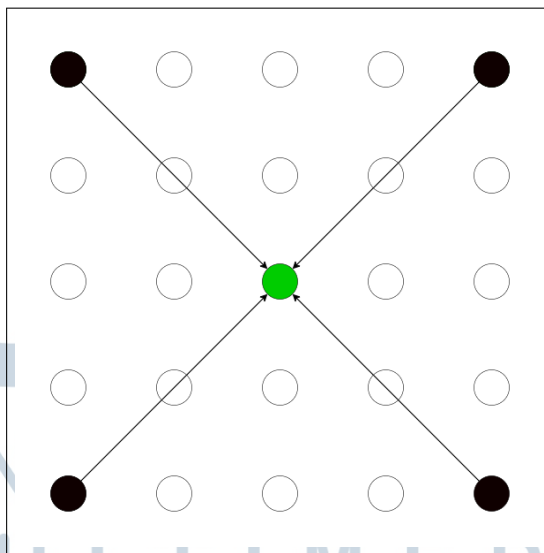
Diamond-square adalah metode perkembangan dari metode Midpoint Displacement, dimana metode Midpoint Displacement merupakan metode yang membagi keempat nilai ketinggian di setiap pojok *terrain* yang berbentuk persegi menjadi persegi yang lebih kecil dimana nilai ketinggian hanya dihitung secara vertikal dan horizontal, sedangkan metode Diamond-square memiliki tambahan langkah untuk menghitung nilai ketinggian secara diagonal [6]. Untuk implementasi metode Diamond-square dibutuhkan *terrain* dengan resolusi  $2^n + 1$ , dengan tahap pertama yaitu memberi nilai ketinggian pada keempat pojok *terrain* seperti visualisasi 2.3.



Gambar 2.3. Langkah 1: inisialisasi nilai ketinggian setiap titik ujung terrain

Sumber: [11]

Tahap kedua yang dilakukan adalah mengisi nilai tengah dengan nilai ketinggian dengan cara memberikan nilai rata-rata dari keempat ujung yang telah diinisialisasi pada tahap kedua seperti pada gambar 2.4 yaitu *square step*.



Gambar 2.4. Langkah 2: square step

Sumber: [11]

Setelah mendapatkan nilai tengah *terrain* pada *square step* maka akan dilanjutkan dengan mendapatkan nilai tengah diantara setiap nilai yang telah terisi menggunakan metode *diamond step* dimana nilai tengah dihitung dari rata-rata nilai

