

---

# Lecture 13

Mapping - continued

# MIP map – another example

---



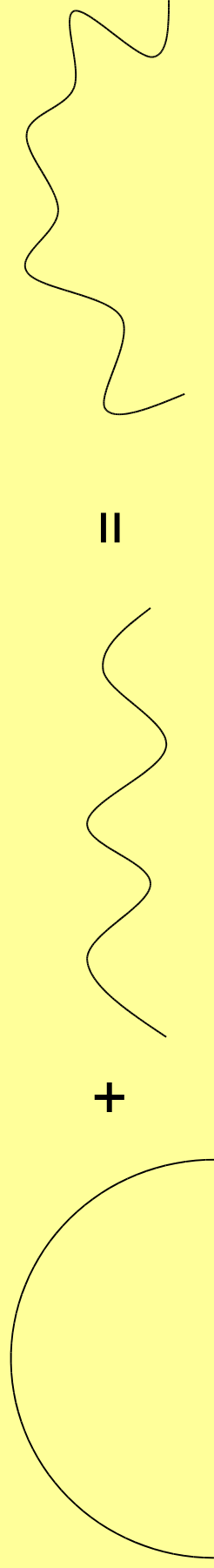
Compared to original RGB, how much more space?

# Bump Mapping

---

Use texture to perturb normals

- creates a bump-like effect



original surface

$O(u, v)$

bump map

$B(u, v)$

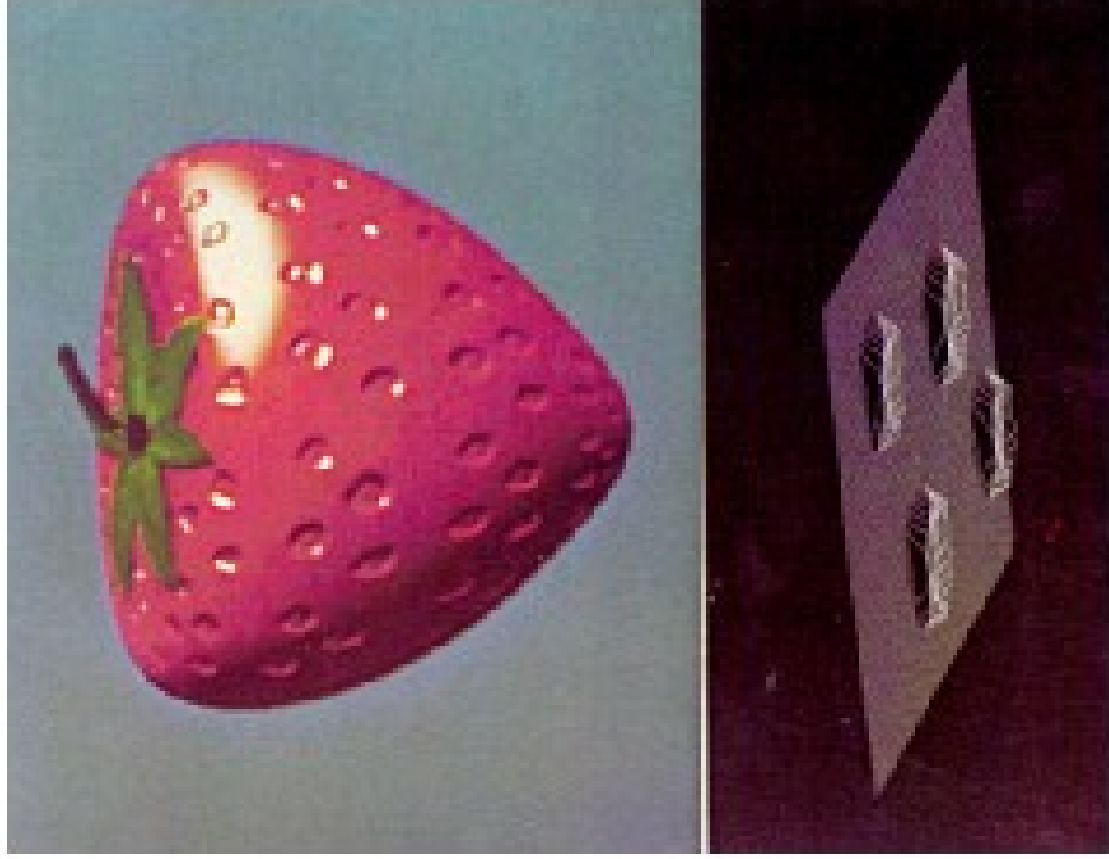
modified surface

$O'(u, v)$

Does not change silhouette edges

# Bump Mapping

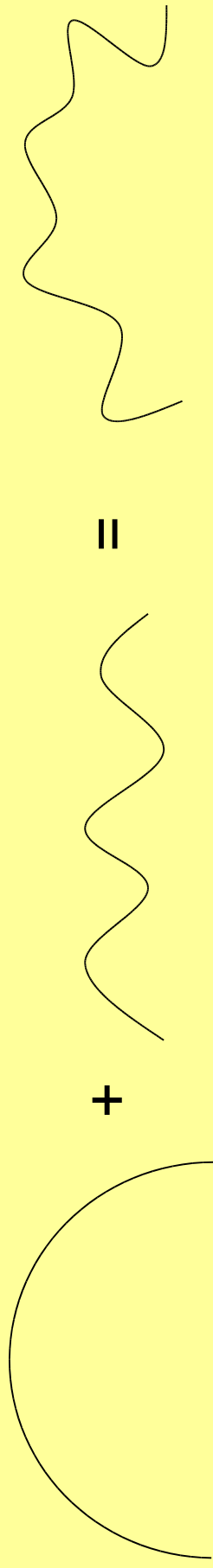
---



# Displacement Mapping

---

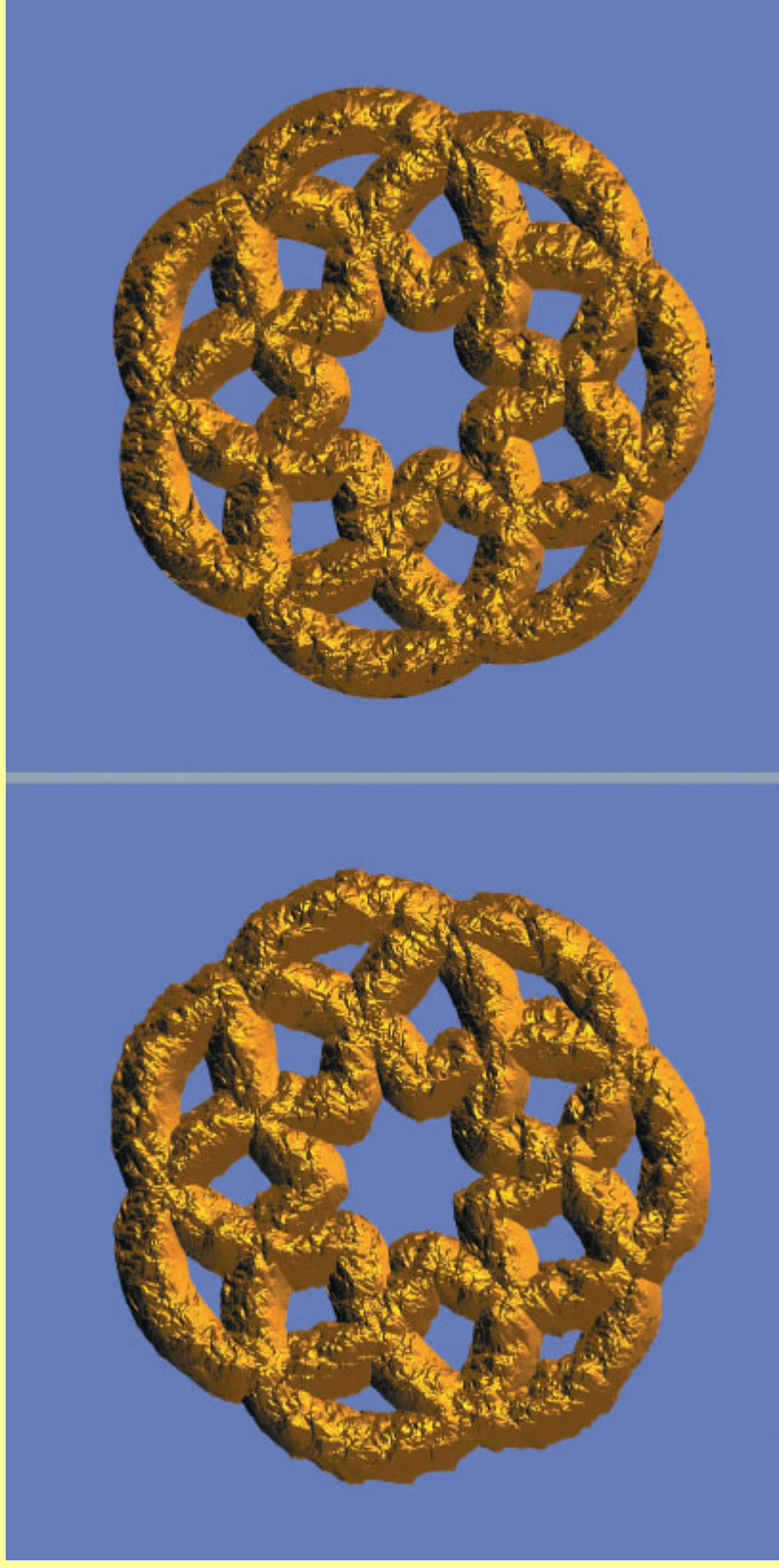
Use texture to **displace** the surface geometry



Bump mapping only affects the normals,  
Displacement mapping changes the entire  
surface (including the silhouette)

# Bump vs. displacement

---



Note the silhouettes! On the left is displacement mapping, on the right is bump.

# 3D Textures - Noise

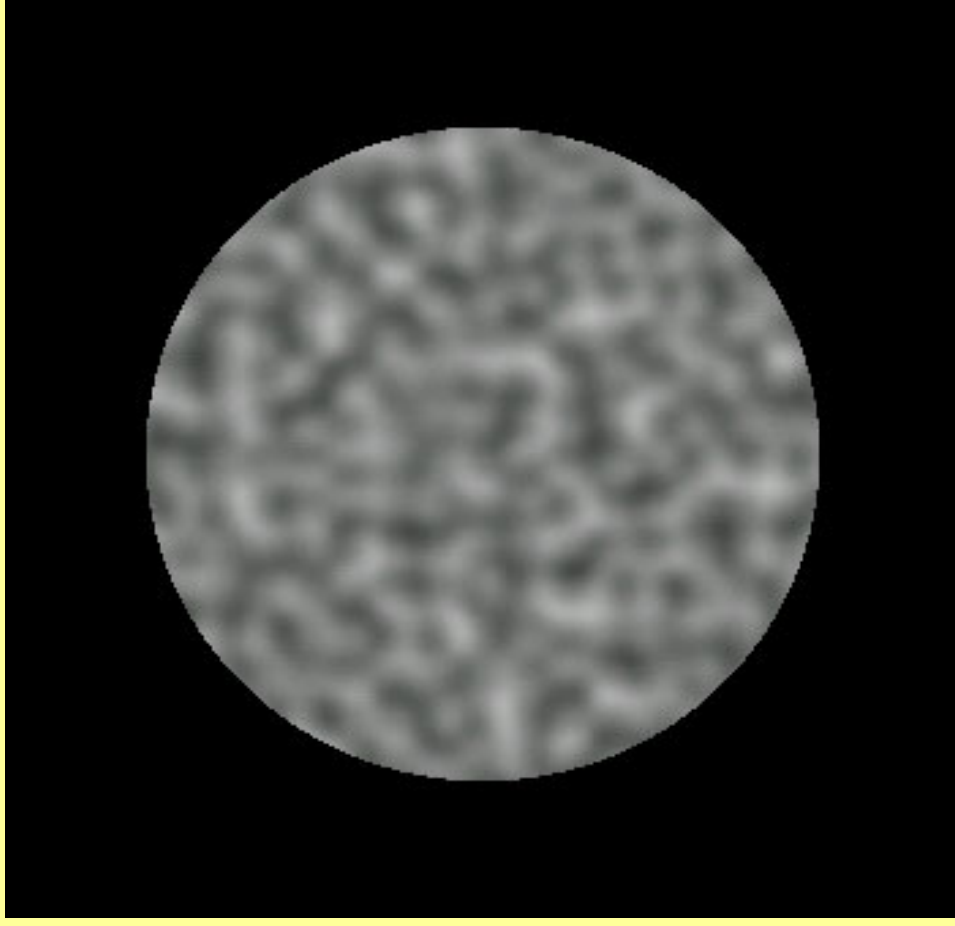
---

Pseudo-random

Bandlimited

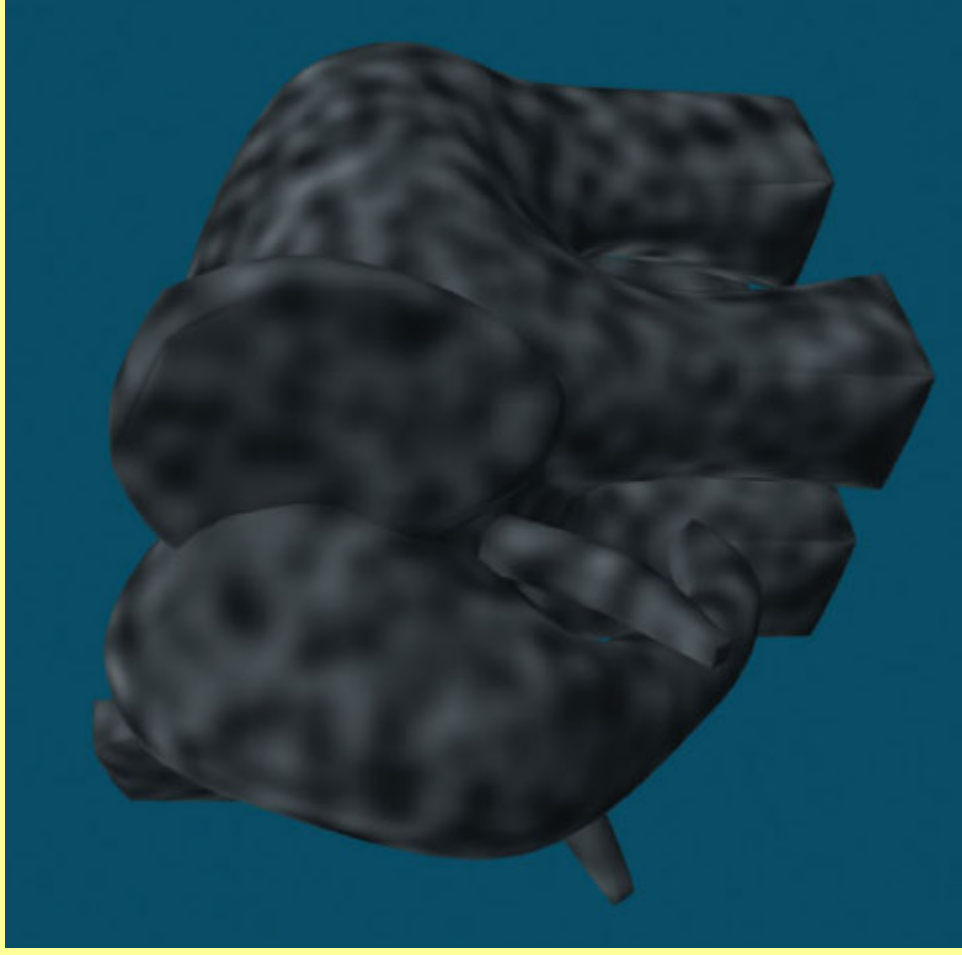
few high or low frequencies

Controllable



# 3D Textures - Noise

---



Noise by itself is not very exciting (too much variation)



# 3D Textures - Noise

---

Create a nD integer-aligned lattice of random numbers

For any nDPoint p, noise is defined as:

noise(nDPoint p)

Find 2 neighbors of p

Linearly interpolate neighbors' table values

Return interpolated value

# 3D Textures - Noise

---



# Turbulence

---

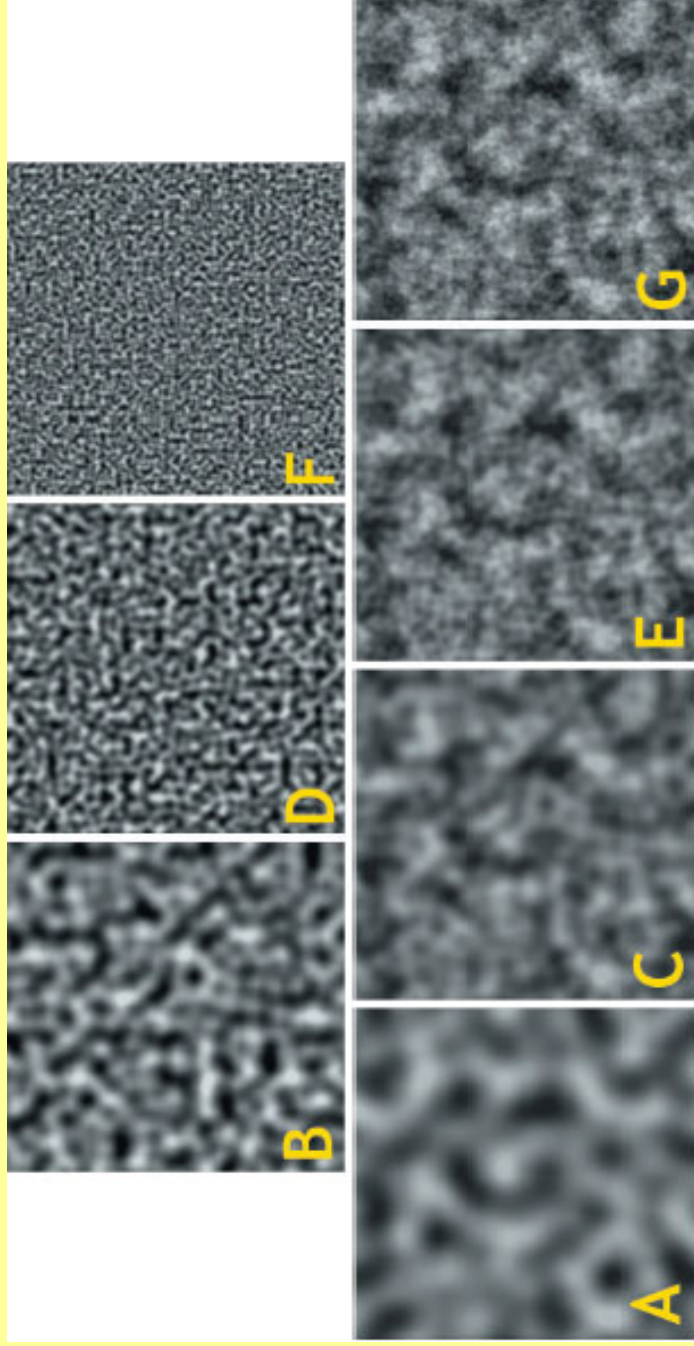
Noise with self-similarity

Add together many octaves of noise

$$\mathit{turbulence}(x) = \sum_{i=0}^k \frac{1}{2^i} \left| \mathit{noise}(2^i x) \right|$$

# Turbulence

---



$A + B + C + D + E + F$  gives G

# Turbulence – scale & add noise

---

Use a 3D mapping

$$(x_0, y_0, z_0) \rightarrow (r, s, t)$$

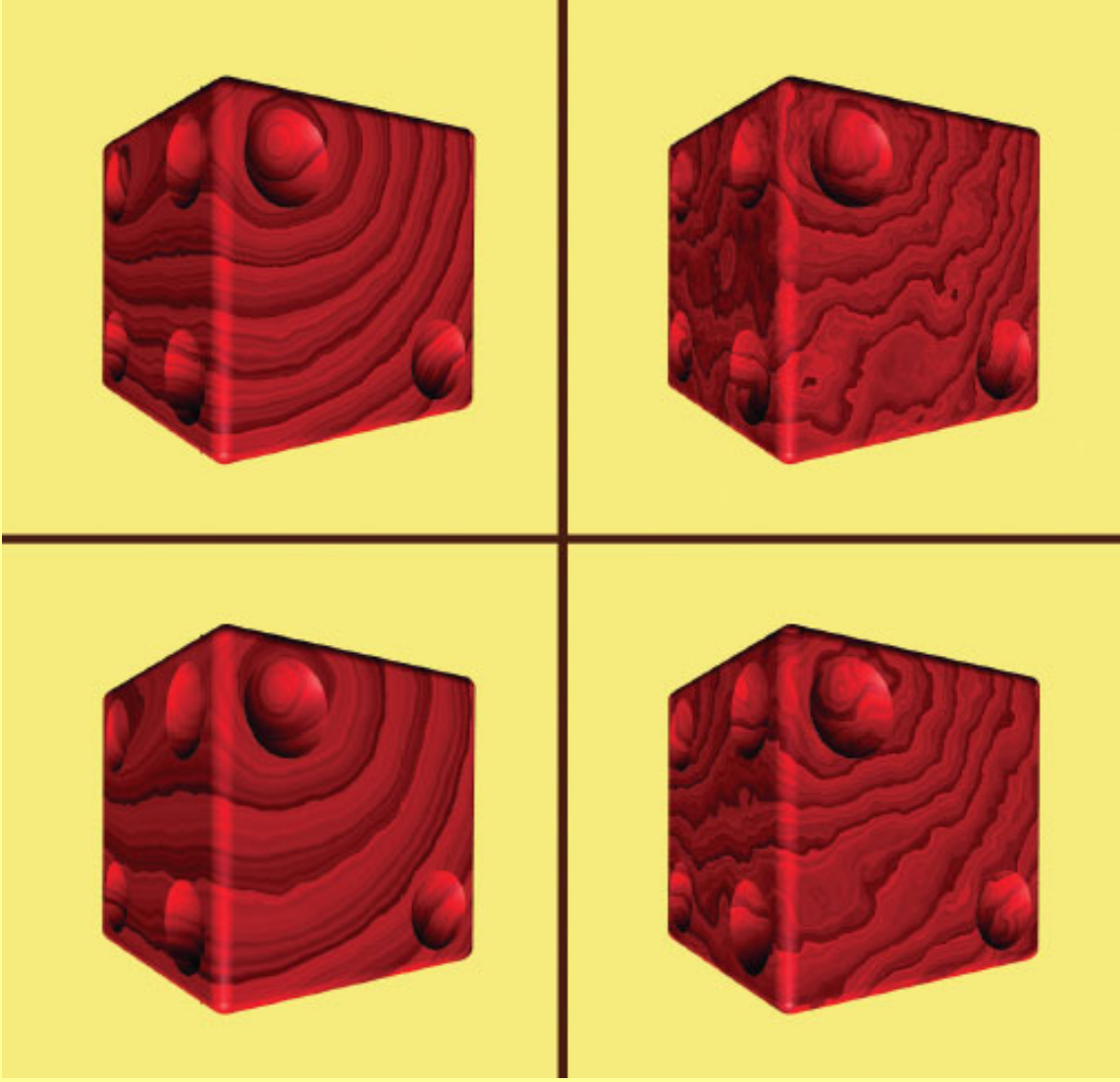
“Procedural texture synthesis”



Can simulate an object carved from a material

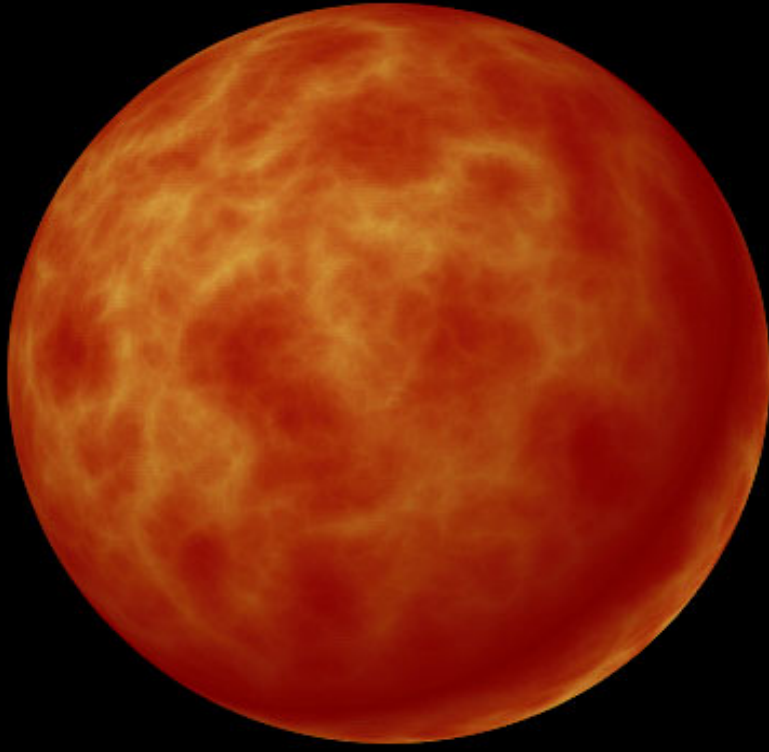
# Turbulence

---

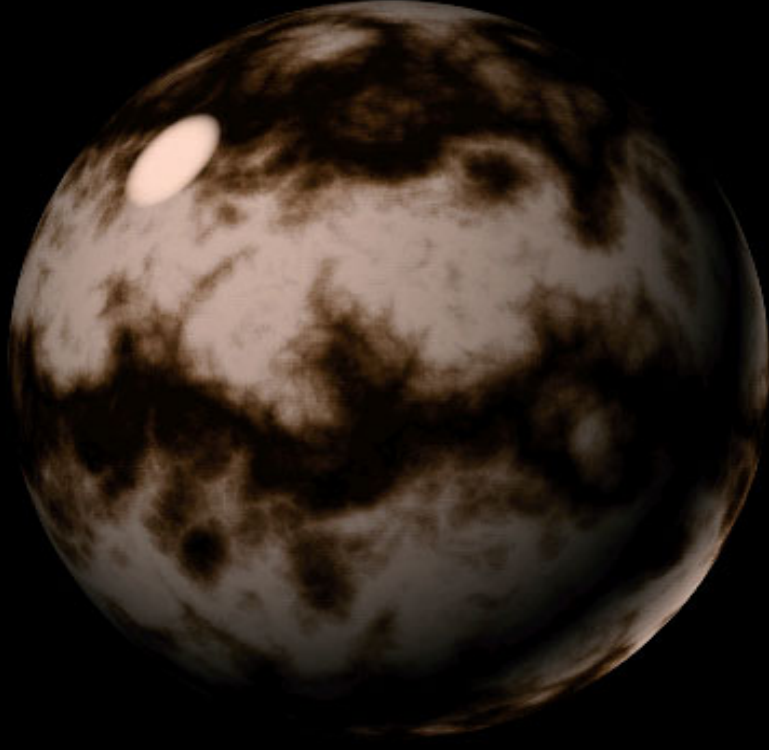


# Turbulence

---



*turbulence(x)*



*$\sin(x + turbulence(x))$*

# Animating Turbulence

---

Use time as an extra dimension (4D lattice)

