Game Development Class

3. Static 2D and 3D art

par Panagiotis Tsiapkolis (Panthavma)

le March 10th 2024

– Visual Arts

- We saw how to program and design a prototype
- Today: visuals, without animations or VFX.

– QUESTION:

What node structure can we use to make a platformer character?

- Sprite → KinematicBody, CollisionShape
- Sprite → KinematicBody → CollisionShape
- KinematicBody → CollisionShape, Sprite
- Spatial → KinematicBody, CollisionShape, Sprite

– ANSWER:

KinematicBody → CollisionShape, Sprite. We want the physics body as a root, to which we add the collider. The sprite stays attached to the root.

– QUESTION:

How do we add acceleration to an entity?

- $\ast$ $position+ = accel$
- $\ast$ $position+ = accel \times \delta_t$
- $\ast$ $vitesse+ = accel$
- $\ast$ $vitesse+ = accel \times \delta_t$

– ANSWER:

$vitesse = accel \times \delta_t$. We can check it with dimensional analysis:

$$m/s = m/s^2 \times s$$

2D

– Artstyle and Artistic Direction

* Artistic Direction: Coherence and global style
  - Target feelings
  - Technical limitations
  - Production limitations

– Color Theory

- Color theory: the esthetic and communicative qualities of colors
  - Definition: Hue, Saturation, Value
  - Esthetics aspects: Cold/Hot, etc.
- Combination: Relationship between colors (complementaries...), composition

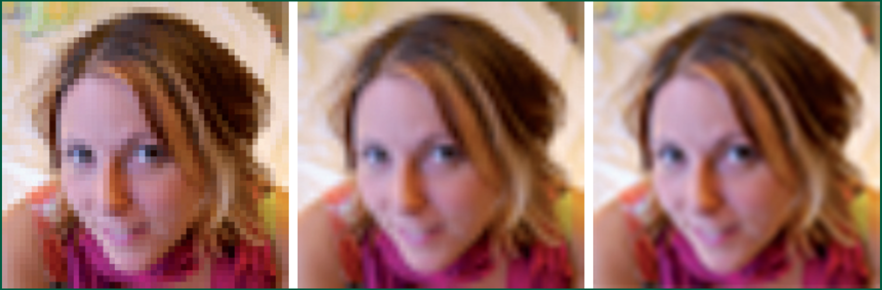
– Color Theory, Tips

- Simple Palettes: ”Split Complementary”, etc
- Other option: Choose three colors and their mixed hues
- Tool: Go into grayscale to see where the eye is going

Sprites

- Sprite: 2D array of pixels
- Pixel: RGB(A) or an identified (indexed palettes)
- Blit: Copy pixels on the screen relative to a fixed point (anchor/origin)
- Sizes often powers of two for performance
- Transformations: Rotation, Scale, etc.
  - Can affect results (signal theory)

– Scale and algorithms



Source : Real Time Rendering

– Pixel Art

- Style based on platform limitations (example: Game Boy Color)
- More limited resolution (GBC: 160x144)
- Limited colors (56 of 32768) and per tile (4 on an 8x8 tile)
- Sprites (8x8 / 8x16) vs Backgrounds, and their limits
- Other restrictions (nb sprites per line / frame, memory...)

– Vector vs Raster

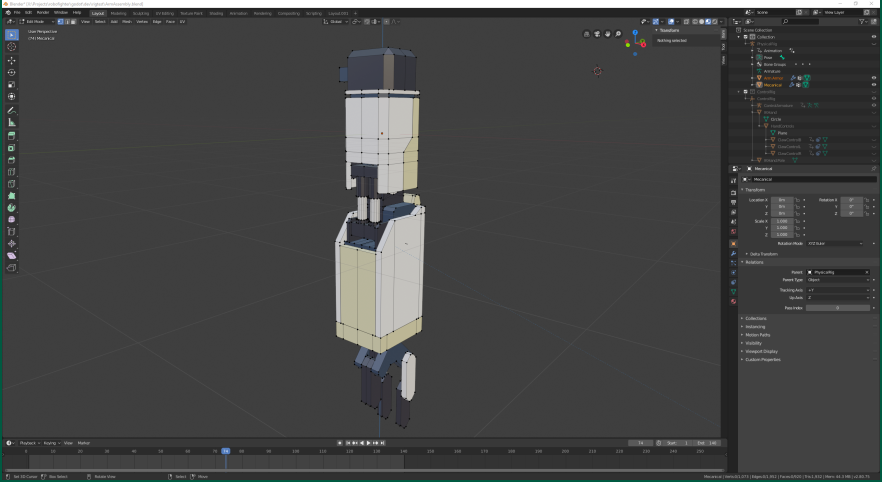
- Vector: Images defined by mathematical shapes (primitives)
- Advantage: Perfect resolution
- Disadvantage: Computation time

3D

– 3D Modelling

- Mesh: Graph of vertices assembled as faces (triangles for the computer)
- Material: Surface rendering parameters and associated data (textures, etc.)
- 3D Model: Mesh + materials + associated data (rigs, animations...)
- 3D Scene: 3D Models + Lights + Camera
- We're gonna use a DCC software like blender to make them.

– 3D Modelling



Blender* [X:\Projects\robofighter\godot\dev\rigtest\ArmAssembly.blend]

User Perspective
(74) Mecanical

Transform
Nothing selected

Mecanical

Transform

Location X 0m
Y 0m
Z 0m

Rotation X 0°
Y 0°
Z 0°

Scale X 1.000
Y 1.000
Z 1.000

Rotation Mode XYZ Euler

Delta Transform

Relations

Parent PhysicalRig
Parent Type Object
Tracking Axis +Y
Up Axis Z
Pass Index 0

Collections
Instancing
Motion Paths
Visibility
Viewport Display
Custom Properties

74 Start: 1 End: 140

Mecanical | Verts:0/1,073 | Edges:0/1,952 | Faces:0/920 | Tris:1,932 | Mem: 44.3 MB | v2.80.75

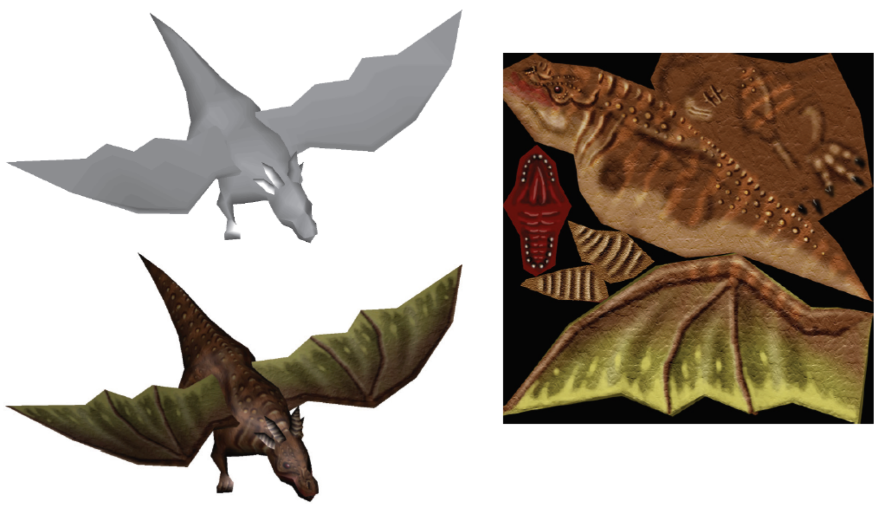
– Materials and Shaders

- The computation of a pixel's color is done by a shader (gpu program) depending on material (shader + parameters)
- PBR (Physically Based Rendering): Based on physics and measured parameters
  - Albedo: Color of the object under white light
  - Other parameters: Metalness, Roughness...
- Some materials can be complex (reflections, subsurface scattering...)

Textures

- Texture: An image applied to the mesh
- Three scales of detail
  - Macro: Several pixels = Mesh
  - Micro: Sub 1px = Shader
  - Meso: 1-4px = Texture
- Apply via UV Mapping: every vertex of the mesh points to a point of the texture
- Can hold colors or parameters
- A lot more uses of textures exist: Tech art

– Textures

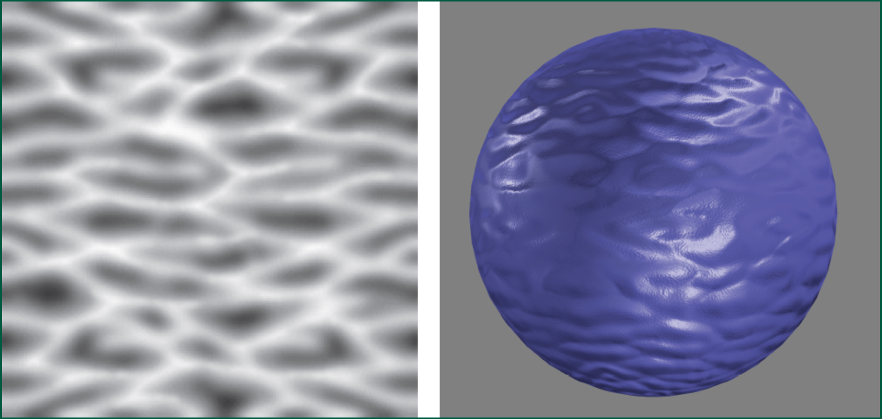


Source : Real Time Rendering

– Bump mapping

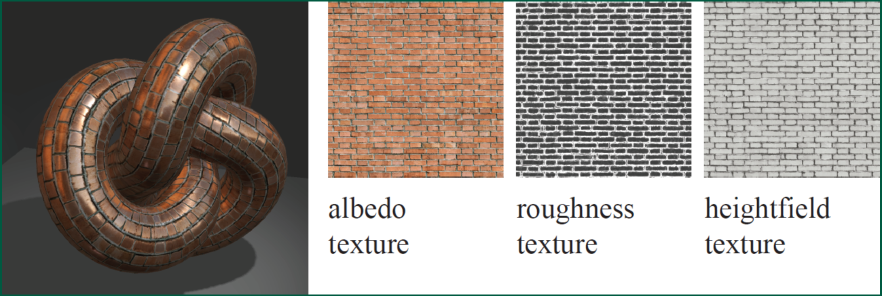
- Example: Bump / Normal Mapping
- Adds fake geometric detail for lighting
- Using height or normal textures (surface direction)
- Limited at grazing angles: prefer displacement mapping.
- Often created by projecting from a more detailed model.

— Heightmap



Source : Real Time Rendering

– More complex materials

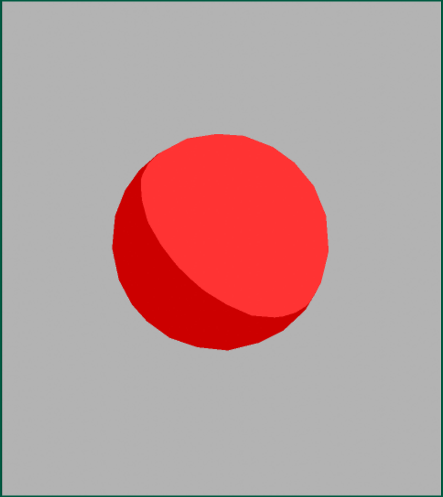


Source : Real Time Rendering

– Toon Shading

- Stylized shading: reduce shading to limited tones
- Need a function to do so (often with a lighting coefficient)
- Toon Ramp: Way of storing this function in a texture
- We'll talk more about this for our complex subjects

– More complex materials

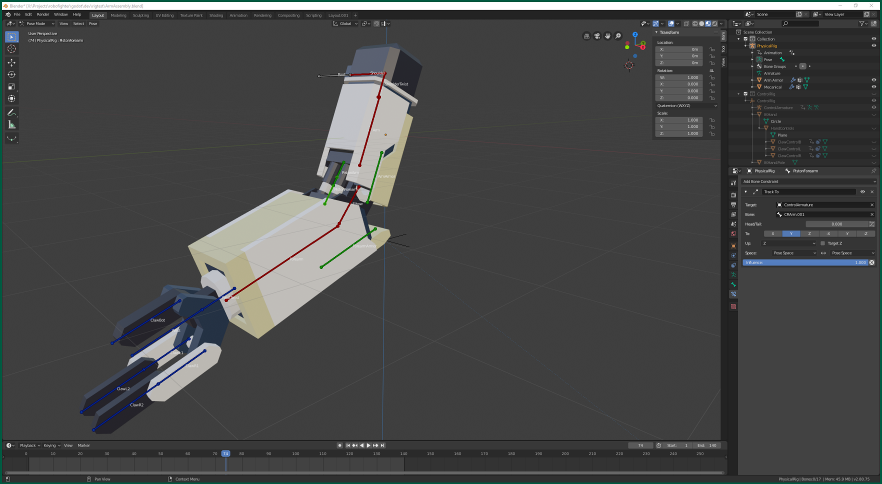


Source : Panthavma

– Skeleton / Rigging

- Additional note: skeleton based animation (exists in 2D too)
- Skeleton: bone hierarchy that allows moving the model like a doll
- Weight Painting: Associating vertices with bones
- Rig: Skeleton + Controls

– Skeleton / Rigging



Blender* [X:\Projects\robofighter\godot\dev\rigtest\ArmAssembly.blend]

User Perspective
(74) PhysicalRig : PistonForearm

PhysicalRig › PistonForearm

Add Bone Constraint

Track To

Target: ControlArmature
Bone: CRArm.001
Head/Tail: 0.000
To: X [✓ Y] Z -X -Y -Z
Up: Z □ Target Z
Space: Pose Space ↔ Pose Space
Influence: 1.000

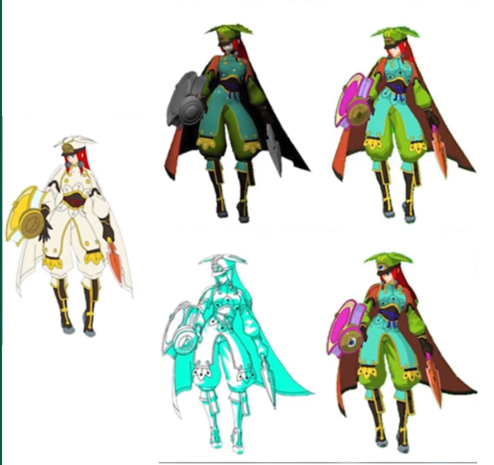
PhysicalRig | Bones:0/17 | Mem: 45.9 MB | v2.80.75

Integration into games

– Pipeline

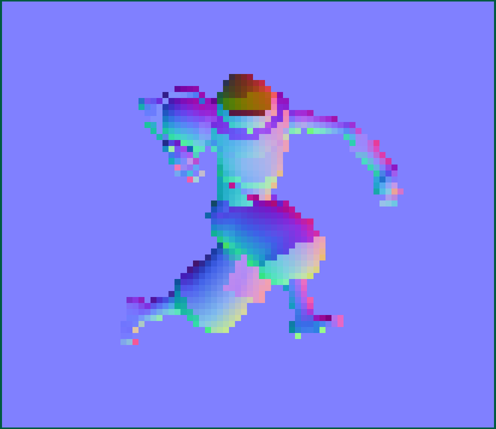
- List of steps and tools to make a character / object / etc.
  - Concept artist makes drawings (esp. reference drawings)
  - Create high poly mesh through sculpting
  - Create low poly mesh and project via baking
  - Create materials and textures
  - Create rig and weight paint
  - Create animations

– Pipeline 2D



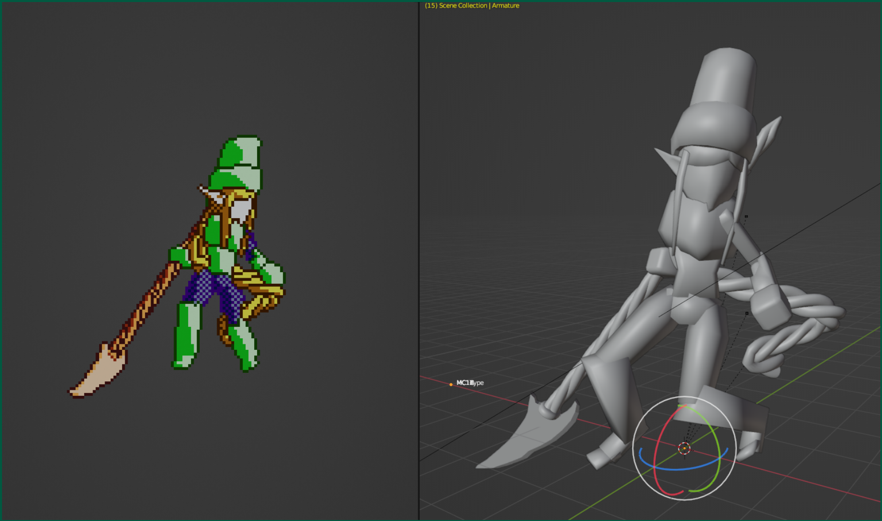
Source: BlazBlue

– Pipeline 3D 1



Source: Dead Cells

– Pipeline 3D 2



Source: Molten Winds

– Art packs, low intensity styles

- There's way to avoid it if it's not your thing
- Using an art pack (KennyNL, OpenGameArt...): Coherence may be a problem, but can use it as a base to edit
- Less complex styles: Terminal (ncurse), low res pixel art, low poly

Questions?

- Discord : https://discord.gg/CWjWfC9K9T
- Website : https://panthavma.com/gamedevclass/
- Next Time : Game design, Mechanics and emotions (March 17th 2024)
- On the side : Try out 2D and 3D art! Define an artistic direction!
- Software: Blender, GIMP, Inkscape, Krita