Interactive Media
and
Game Development

2-D Tiles and Sprites

Outline

• Tiles
• Sprites

• More material:
  – Ari Feldman. Designing Arcade Computer Game Graphics, Online at:
    http://www.gamemaker.nl/feldman/full.zip
  – Tsugumo. So You Want to Be a Pixel Artist?, Online at:
    http://web.cs.wpi.edu/~claypool/courses/frontiers-06/samples/pixel-artist/default.html
Tiles

- A tile is a small, square 2d image for a sprite-based game
  - Needed for commonly backgrounds
- Often repeated
  - Too hard to make every pixel different!
- RPGs make heavy use
  - Grass, trees, water, sand
- Start with a grass tile to warm up

Grass is Green

- Use a basic green square
- But looks unnatural
  - Like flat, shiny metal
- No illusion of movement

SIMPLY A GREEN BOX WITH LIGHTER GREEN DOTS.
Grass has Variation

• Can do a lot with simple enhancement of color shades

SIMPLY A GREEN BOX WITH LIGHTER GREEN DOTS.

Make Random

• Use the “spray” tool

3 GREEN SHADES RANDOMLY "SPRAYED" AND A 16X16 CHUNK TAKEN FROM IT.
Make Look Random with Control

• Draw by hand for more control
  - 4 pixel line strokes

Don’t Try This at Home

• Don’t use same texture for all, else not much better than just colors
The “Grid” (1 of 4)

- Looks too much like tiles
- “Large” blank is problem, so remove

The “Grid” (2 of 4)

- Still, some “lines” are visible when repeated
- Break up with more color
The “Grid” (3 of 4)

• Much better!

The “Grid” (4 of 4)

• When rubber hits the road?
Outline

• Tiles
• Sprites (next)

Animation

• Animation → produces the illusion of movement
• Display a series of frames with small differences between them
• Done in rapid succession, eye blends to get motion
• Unit is Frames Per Second (fps). For video:
  - 24-30 fps: full-motion (Game Maker does 30)
  - 15 fps: full-motion approximation
  - 7 fps: choppy
  - 3 fps: very choppy
  - Less than 3 fps: slide show
  - 2D Sprites can get away with about about $\frac{1}{2}$ that
• To do successfully, need to keenly observe, focus on differences in movement
  - Apply basic principles (next)
Motion Line

- Invisible line created by object as moves
  - Locate in center of gravity
- Straight if flying
  - Ex: bullet
- Up and down if bounces
  - Ex: rubber ball
- Depends upon speed and desire for exaggeration
  - Ex: Human sprinting versus walking
  - Ex: Warcraft III

Key Frames

- Images at extremes in movement
  - Most noticeable to observer
  - Ex: for flight wings up and wings down
  - Ex: for walking, right leg forward, leg together
- The more the better?
  - Smoother, yes
  - But more time to develop (tradeoffs)
  - And more prone to errors, “bugs” that interfere with the animation
In-Between Frames

- Generated to get smooth motion between key-frames
  - Can be tedious and time consuming to make
  - Most software allows duplication

Frame Animation Guidelines

<table>
<thead>
<tr>
<th>Object</th>
<th>Minimum # of Frames</th>
<th>Maximum #</th>
</tr>
</thead>
<tbody>
<tr>
<td>4-legged animal running</td>
<td>4</td>
<td>16</td>
</tr>
<tr>
<td>Animal biting</td>
<td>2</td>
<td>5</td>
</tr>
<tr>
<td>Crawling</td>
<td>2</td>
<td>12</td>
</tr>
<tr>
<td>Explosions</td>
<td>5</td>
<td>16</td>
</tr>
<tr>
<td>Falling</td>
<td>3</td>
<td>5</td>
</tr>
<tr>
<td>Flying</td>
<td>2</td>
<td>12</td>
</tr>
<tr>
<td>Jumping</td>
<td>2</td>
<td>10</td>
</tr>
<tr>
<td>Kicking</td>
<td>2</td>
<td>6</td>
</tr>
<tr>
<td>Punching</td>
<td>2</td>
<td>6</td>
</tr>
<tr>
<td>Rotating/spinning</td>
<td>4</td>
<td>16</td>
</tr>
<tr>
<td>Running</td>
<td>2</td>
<td>12</td>
</tr>
<tr>
<td>Swinging (an object)</td>
<td>2</td>
<td>8</td>
</tr>
<tr>
<td>Throwing (an object)</td>
<td>2</td>
<td>6</td>
</tr>
<tr>
<td>Vehicle flying</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>Vehicle moving</td>
<td>2</td>
<td>8</td>
</tr>
<tr>
<td>Walking</td>
<td>2</td>
<td>12</td>
</tr>
</tbody>
</table>

(See GameMaker tutorial shooter for examples of Enemy Planes, Explosions)
Secondary Actions

• Animation part that does not lead movement, but follows it
  - Add extra dimension of reality
  - Ex: Hair moving in wind
  - Ex: Cape billowing backward

Steps in Creating Animation Sequences (1 of 3)

• Conceptualize - have vision (in mind or on paper) of what animation will look like
• Decide on object behavior
  1. Animated once (no looping)
  2. Animated continuously (using cycles)
     - 2nd choice means must make last key frame blend with first
• Choose an image size - will contain and constrain object
  - Test and experiment briefly to have plenty of room
• Design key-frames - drawing the motion extremes
  - Use simple shapes to represent main actions
     * Ex: stick figures or basic shapes (circles, squares)
Steps in Creating Animation Sequences (2 of 3)

• Estimate the in-betweens - think of how many you will need to complete the sequence smoothly
  - Be conservative. Easier to add additional transition frames than remove them
• Create object motion lines - trace the motion line and motion angles for the sequence. Make sure properties are consistent with object, else adjust
  - Use your painting program’s “line tool”
  - If not, make the appropriate adjustments to the sequence and repeat
• Apply secondary enhancements - Embellish to look convincing and enticing

Steps in Creating Animation Sequences (3 of 3)

• Test each movement
  - Can be done with ‘copy’ and ‘undo’ in tool
  - Others have animation rendering (ex- Game Maker)
  - Look for flaws (movement, discolored pixels …)
• Repeat - Repeat for all animations
General Animation Tips (1 of 3)

• Remember the relationship between frames and animation smoothness
  - More frames, more smoothness (but more time)
• Always account for color
  - Primary actions and secondary actions should be rendered in colors that make them easy to see.
  - Otherwise, the effectiveness of the animation can be compromised
• Use tempo wisely- Never too fast or too slow
  - Try to mimic nature. Observe yourself. Study the speed at which different types of objects move in different situations.

Based on Chapter 9, Designing Arcade Computer Game Graphics, by Ari Feldman

General Animation Tips (2 of 3)

• Try to individualize your objects
  - Unique and individualized touches make seem real. “Personality” that distinguishes it
  - Easiest, use exaggeration and embellishment (i.e., secondary actions)
• Keep it simple - Unnecessary complexity can ruin animation
  - Stick with primitives and minimal frames
  - Don’t do any more work than you have to!
General Animation Tips (3 of 3)

• Use exaggerated elements - as an animation device, adds depth
  - Especially important for short animation sequences to make convincing
• Constantly observe - study of the objects around you. Study how different things move. Study books on animation. Observe your favorite games
  - Will give insights into animation techniques, make better animations yourself

Based on Chapter 9, Designing Arcade Computer Game Graphics, by Ari Feldman

Primitives

• Used in many games. If identify, can apply primitive rules and use:
  - Cylindrical primitive
  - Rotational primitive
  - Disintegration primitive
  - Color flash primitive
  - Scissors primitive
  - Growing primitive
  - Shrinking primitive
  - Minor primitives (used less often)

Based on Chapter 9, Designing Arcade Computer Game Graphics, by Ari Feldman
Cylindrical Primitive

- Spinning, rotating objects (hulls, wheels, logs...)
- Easy to master since doesn’t require major changes
- Instead, uses markers that change
  - Show go from one end to another
- Need at least 3 frames

Rotational Primitive

- Object moving in place (gun turret, asteroid...)
- Again, easy since rotate picture fixed degrees

<table>
<thead>
<tr>
<th>Arcade Game Object</th>
<th>Degree Increments per Frame</th>
<th>Total Frames Required</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>Asteroids/meteors (coarse)</td>
<td>45°</td>
<td>8</td>
<td>Minimum required to produce convincing animation.</td>
</tr>
<tr>
<td>Asteroids/meteors (smooth)</td>
<td>225°</td>
<td>16</td>
<td>Sufficient to render convincing animation.</td>
</tr>
<tr>
<td>Gun turrets (coarse)</td>
<td>90°</td>
<td>4</td>
<td>Minimum required to produce convincing animation.</td>
</tr>
<tr>
<td>Gun turrets (smooth)</td>
<td>45°</td>
<td>8</td>
<td>Sufficient to render convincing animation.</td>
</tr>
<tr>
<td>Spinning objects (coarse)</td>
<td>90°</td>
<td>4</td>
<td>Minimum required to produce convincing animation.</td>
</tr>
<tr>
<td>Spinning objects (coarse)</td>
<td>45°</td>
<td>8</td>
<td>Sufficient to render convincing animation.</td>
</tr>
<tr>
<td>Vehicle/character facings (coarse)</td>
<td>90°</td>
<td>4</td>
<td>Minimum required to produce convincing animation.</td>
</tr>
<tr>
<td>Vehicle/character facings (smooth)</td>
<td>45°</td>
<td>8</td>
<td>Sufficient to render convincing animation.</td>
</tr>
</tbody>
</table>

Based on Chapter 9, Designing Arcade Computer Game Graphics, by Ari Feldman
Disintegration Primitive

• Remove object from screen (character dies, explosion...)
  - Melting – reduce vertical area
  - Dissolving – remove random pattern
  - Color fading – extreme color change
• Take fixed percentage out for smooth

<table>
<thead>
<tr>
<th>Selected Removal Method</th>
<th>Estimated Percent Removed per Frame</th>
<th>Total Frames</th>
</tr>
</thead>
<tbody>
<tr>
<td>Melting (coarse)</td>
<td>25</td>
<td>4</td>
</tr>
<tr>
<td>Melting (smooth)</td>
<td>10</td>
<td>10</td>
</tr>
<tr>
<td>Dissolving (coarse)</td>
<td>25</td>
<td>4</td>
</tr>
<tr>
<td>Dissolving (smooth)</td>
<td>10</td>
<td>10</td>
</tr>
<tr>
<td>Color fade (coarse)</td>
<td>12.5°</td>
<td>8°</td>
</tr>
<tr>
<td>Color fade (smooth)</td>
<td>6.25°</td>
<td>16°</td>
</tr>
</tbody>
</table>

Color Flash Primitive

• Flickering behind object (flash of jewel, sparkle of torch, pulse behind rocket...)
  - Usually intense, contrast color
  - Usually short animation (but can be complex)
Scissors Primitive

- One of most popular (walking, biting)
- Few key frames, large changes in between

Growing/Shrinking Primitive

- For explosion, growth/reduction potion
- Pay attention to scale (ex: 2 works well)
Minor Primitives (1 of 3)

FIGURE 9-18: Paton Primitive Example
FIGURE 9-19: Squeeze Primitive Example
FIGURE 9-20: Swing Primitive Example
FIGURE 9-21: Slide Primitive Example
FIGURE 9-22: Open/Close Primitive Example
FIGURE 9-23: Sounce Primitive Example
FIGURE 9-24: Stomp Primitive Example

Based on Chapter 9, Designing Arcade Computer Game Graphics, by Ari Feldman

Minor Primitives (2 of 3)

FIGURE 9-25: Slinking Example
FIGURE 9-26: Simplified Flying Sequence
FIGURE 9-28: Basic Walking Example #1
FIGURE 9-29: Basic Walking Example #2

Based on Chapter 9, Designing Arcade Computer Game Graphics, by Ari Feldman
Minor Primitives (3 of 3)

FIGURE 9-30: Complex Walking Example

FIGURE 9-33: Running Primitive Example (Humans)

FIGURE 9-38: Running Primitive Example (Animals)

Based on Chapter 9, Designing Arcade Computer Game Graphics, by Ari Feldman