texture()

Sets the texture that will be used on shapes.

A texture is like a skin that wraps around a shape. `texture()` works with built-in shapes, such as square() and sphere(), and custom shapes created with functions such as buildGeometry(). To texture a geometry created with beginShape(), uv coordinates must be passed to each vertex() call.

The parameter, `tex`, is the texture to apply. `texture()` can use a range of sources including images, videos, and offscreen renderers such as p5.Graphics and p5.Framebuffer objects.

To texture a geometry created with beginShape(), you will need to specify uv coordinates in vertex().

Note: `texture()` can only be used in WebGL mode.

Examples

```
let img;

// Load an image and create a p5.Image object.
function preload() {
  img = loadImage('/assets/laDefense.jpg');
}

function setup() {
  createCanvas(100, 100, WEBGL);

  describe('A spinning cube with an image of a ceiling on each face.');
}

function draw() {
  background(0);

  // Rotate around the x-, y-, and z-axes.
  rotateZ(frameCount * 0.01);
  rotateX(frameCount * 0.01);
  rotateY(frameCount * 0.01);

  // Apply the image as a texture.
  texture(img);

  // Draw the box.
  box(50);
}
```

```
let pg;

function setup() {
  createCanvas(100, 100, WEBGL);

  // Create a p5.Graphics object.
  pg = createGraphics(100, 100);

  // Draw a circle to the p5.Graphics object.
  pg.background(200);
  pg.circle(50, 50, 30);

  describe('A spinning cube with circle at the center of each face.');
}

function draw() {
  background(0);

  // Rotate around the x-, y-, and z-axes.
  rotateZ(frameCount * 0.01);
  rotateX(frameCount * 0.01);
  rotateY(frameCount * 0.01);

  // Apply the p5.Graphics object as a texture.
  texture(pg);

  // Draw the box.
  box(50);
}
```

```
let vid;

// Load a video and create a p5.MediaElement object.
function preload() {
  vid = createVideo('/assets/fingers.mov');
}

function setup() {
  createCanvas(100, 100, WEBGL);

  // Hide the video.
  vid.hide();

  // Set the video to loop.
  vid.loop();

  describe('A rectangle with video as texture');
}

function draw() {
  background(0);

  // Rotate around the y-axis.
  rotateY(frameCount * 0.01);

  // Apply the video as a texture.
  texture(vid);

  // Draw the rectangle.
  rect(-40, -40, 80, 80);
}
```

```
let vid;

// Load a video and create a p5.MediaElement object.
function preload() {
  vid = createVideo('/assets/fingers.mov');
}

function setup() {
  createCanvas(100, 100, WEBGL);

  // Hide the video.
  vid.hide();

  // Set the video to loop.
  vid.loop();

  describe('A rectangle with video as texture');
}

function draw() {
  background(0);

  // Rotate around the y-axis.
  rotateY(frameCount * 0.01);

  // Set the texture mode.
  textureMode(NORMAL);

  // Apply the video as a texture.
  texture(vid);

  // Draw a custom shape using uv coordinates.
  beginShape();
```

Syntax

```
texture(tex)
```

Parameters

tex	p5.Image|p5.MediaElement|p5.Graphics|p5.Texture|p5.Framebuffer|p5.FramebufferTexture: media to use as the texture.

This page is generated from the comments in src/webgl/material.js . Please feel free to edit it and submit a pull request!

Related References

copyToContext
Copies the shader from one drawing context to another.

inspectHooks
Logs the hooks available in this shader, and their current implementation.

modify
Returns a new shader, based on the original, but with custom snippets of shader code replacing default behaviour.

setUniform
Sets the shader's uniform (global) variables.