

blend()

Copies a region of pixels from another image into this one.

The first parameter, `srcImage`, is the `p5.Image` object to blend.

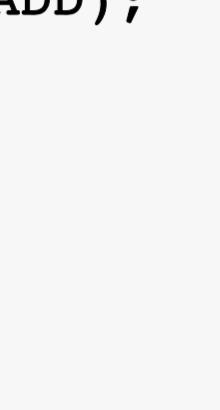
The next four parameters, `sx`, `sy`, `sw`, and `sh` determine the region to blend from the source image. (`sx`, `sy`) is the top-left corner of the region. `sw` and `sh` are the regions width and height.

The next four parameters, `dx`, `dy`, `dw`, and `dh` determine the region of the canvas to blend into. (`dx`, `dy`) is the top-left corner of the region. `dw` and `dh` are the regions width and height.

The tenth parameter, `blendMode`, sets the effect used to blend the images' colors.

The options are `BLEND`, `DARKEST`, `LIGHTEST`, `DIFFERENCE`, `MULTIPLY`, `EXCLUSION`, `SCREEN`, `REPLACE`, `OVERLAY`, `HARD_LIGHT`, `SOFT_LIGHT`, `DODGE`, `BURN`, `ADD`, or `NORMAL`.

Examples



```
let mountains;
let bricks;

// Load the images.
function preload() {
  mountains = loadImage('/assets/rockies.jpg');
  bricks = loadImage('/assets/bricks_third.jpg');
}

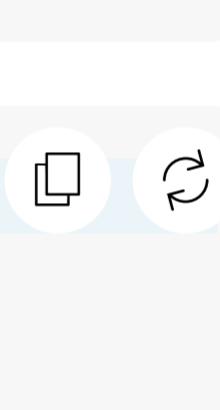
function setup() {
  createCanvas(100, 100);

  // Blend the bricks image into the mountains.
  mountains.blend(bricks, 0, 0, 33, 100, 67, 0, 33, 100, ADD);

  // Display the mountains image.
  image(mountains, 0, 0);

  // Display the bricks image.
  image(bricks, 0, 0);

  describe('A wall of bricks in front of a mountain landscape.
The same wall of bricks appears faded on the right of the
image.');
}
```



```
let mountains;
let bricks;

// Load the images.
function preload() {
  mountains = loadImage('/assets/rockies.jpg');
  bricks = loadImage('/assets/bricks_third.jpg');
}

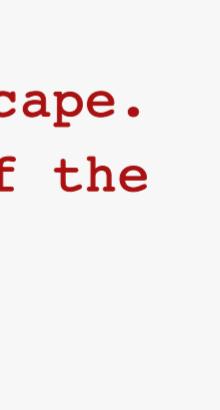
function setup() {
  createCanvas(100, 100);

  // Blend the bricks image into the mountains.
  mountains.blend(bricks, 0, 0, 33, 100, 67, 0, 33, 100,
DARKEST);

  // Display the mountains image.
  image(mountains, 0, 0);

  // Display the bricks image.
  image(bricks, 0, 0);

  describe('A wall of bricks in front of a mountain landscape.
The same wall of bricks appears transparent on the right of the
image.');
}
```



```
let mountains;
let bricks;

// Load the images.
function preload() {
  mountains = loadImage('/assets/rockies.jpg');
  bricks = loadImage('/assets/bricks_third.jpg');
}

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

  // Blend the bricks image into the mountains.
  mountains.blend(bricks, 0, 0, 33, 100, 67, 0, 33, 100,
LIGHTEST);

  // Display the mountains image.
  image(mountains, 0, 0);

  // Display the bricks image.
  image(bricks, 0, 0);

  describe('A wall of bricks in front of a mountain landscape.
The same wall of bricks appears washed out on the right of the
image.');
}
```

Syntax

```
blend(srcImage, sx, sy, sw, sh, dx, dy, dw, dh, blendMode)
```



```
blend(sx, sy, sw, sh, dx, dy, dw, dh, blendMode)
```



Parameters

<code>srcImage</code>	<code>p5.Image</code> : source image
<code>sx</code>	Integer: x-coordinate of the source's upper-left corner.
<code>sy</code>	Integer: y-coordinate of the source's upper-left corner.
<code>sw</code>	Integer: source image width.
<code>sh</code>	Integer: source image height.
<code>dx</code>	Integer: x-coordinate of the destination's upper-left corner.
<code>dy</code>	Integer: y-coordinate of the destination's upper-left corner.
<code>dw</code>	Integer: destination image width.
<code>dh</code>	Integer: destination image height.
<code>blendMode</code>	Constant: the blend mode, either <code>BLEND</code> , <code>DARKEST</code> , <code>LIGHTEST</code> , <code>DIFFERENCE</code> , <code>MULTIPLY</code> , <code>EXCLUSION</code> , <code>SCREEN</code> , <code>REPLACE</code> , <code>OVERLAY</code> , <code>HARD_LIGHT</code> , <code>SOFT_LIGHT</code> , <code>DODGE</code> , <code>BURN</code> , <code>ADD</code> or <code>NORMAL</code> . Available blend modes are: <code>normal</code> <code>multiply</code> <code>screen</code> <code>overlay</code> <code>darker</code> <code>lighter</code> <code>color-dodge</code> <code>color-burn</code> <code>hard-light</code> <code>soft-light</code> <code>difference</code> <code>exclusion</code> <code>hue</code> <code>saturation</code> <code>color</code> <code>luminosity</code> http://blogs.adobe.com/webplatform/2013/01/28/blending-features-in-canvas/

This page is generated from the comments in `src/image/p5.Image.js`. Please feel free to edit it and submit a pull request!

Related References

`blend`
Copies a region of pixels from another image into this one.

`copy`
Copies pixels from a source image to this image.

`delay`
Changes the delay between frames in an animated GIF.

`filter`
Applies an image filter to the image.



Donate Today! Support p5.js and the Processing Foundation.



p5.js	Resources	Information	Socials
	Reference Tutorials Examples Contribute Community About Start Coding Donate	Download Contact Copyright Privacy Policy Terms of Use	GitHub ↗ Instagram ↗ X ↗ YouTube ↗ Discord ↗ Forum ↗