

Remove background

CODE

```
Capture video; // Variable for capture device
PImage backgroundImage; // Saved background

float threshold = 10; // How different must a pixel be to be a foreground pixel

void setup() {
  size(320,240);
  video = new Capture(this, width, height, 30);
  video.start();
  // Create an empty image the same size as the video
  backgroundImage = createImage(video.width,video.height,RGB);
}

void draw() {
  // Capture video
  if (video.available()) {
    video.read();
  }

  // We are looking at the video's pixels, the memorized backgroundImage's pixels,
  // as well as accessing the display pixels.
  // So we must loadPixels() for all!
  loadPixels();
  video.loadPixels();
  backgroundImage.loadPixels();

  // Begin loop to walk through every pixel
  for (int x = 0; x < video.width; x ++ ) {
    for (int y = 0; y < video.height; y ++ ) {
      int loc = x + y*video.width; // Step 1, what is the 1D pixel location
      color fgColor = video.pixels[loc]; // Step 2, what is the foreground color

      // Step 3, what is the background color
      color bgColor = backgroundImage.pixels[loc];

      // Step 4, compare the foreground and background color
      float r1 = red(fgColor);
      float g1 = green(fgColor);
      float b1 = blue(fgColor);
      float r2 = red(bgColor);
      float g2 = green(bgColor);
      float b2 = blue(bgColor);
      float diff = dist(r1,g1,b1,r2,g2,b2);
```

```
// Learning Processing
// Daniel Shiffman
// http://www.learningprocessing.com
```

```
// Example 16-12: Simple background
removal
```

```
// Click the mouse to memorize a
current background image
import processing.video.*;
```

```

// Step 5, Is the foreground color different from the background color
if (diff > threshold) {
    // If so, display the foreground color
    pixels[loc] = fgColor;
} else {
    // If not, display green
    pixels[loc] = color(0,255,0);
    // We could choose to replace the background pixels with something other
    than the color green!
}
}
}
updatePixels();
}

void mousePressed() {
    // Copying the current frame of video into the backgroundImage object
    // Note copy takes 5 arguments:
    // The source image
    // x,y,width, and height of region to be copied from the source
    // x,y,width, and height of copy destination
    backgroundImage.copy(video,0,0,video.width,video.height,0,0,video.
width,video.height);
    backgroundImage.updatePixels();
}

```