

```
//pixelaion
//Lucian Novosel - 2014
//Open sourcerer on the internet of things
// WWW: luciannovosel.com GH: github.com/luciannovo
```

```
import processing.video.*;
Capture cam;
boolean clicked = false;
boolean loaded = false;

PImage img;
import processing.pdf.*;
PGraphics pdf;
String inputPath = "inputImages/";
String outputPath = "outputImages/";
String imageName = "selfie";
int increment = 20;
int pixel = 0;
int horz_pixels;
int vert_pixels;
int lastPixel;

void setup(){
    //Start the camera
    size(320*2, 240*2);
    cam = new Capture(this, 320*2, 240*2 , 24);
    cam.start();
}

void draw(){
    if(!clicked && !loaded){
        if(cam.available()) {
            cam.read();
        }
        image(cam, 0, 0);
    }
    else if(!clicked && !loaded){
        //Load the image and define the resolution
        horz_pixels = (img.width - img.width % increment)/increment;
        vert_pixels = (img.height - img.height % increment)/increment;
        lastPixel = img.height * img.width; //the last pixel from the image t
        println("Max horizontal pixels(rendered): ", horz_pixels);
        println("Max vertical pixels(rendered):", vert_pixels);
        loaded = true;
    }
}
```

```

        scalablePDF();
    }
}

void mouseClicked(){
    clicked = true;
    img = cam.get();
    cam.stop();
}

void scalablePDF(){
    //Create the scalable PDF
    pdf = createGraphics(horz_pixels*increment, vert_pixels*increment, PDF, out
    pdf.beginDraw();
    pdf.background(255);

    //
    int firstPixel = int(increment/2 - 1);
    println("The first sampling pixel is(picture): ", firstPixel); //
    for(int vp = 0; vp < vert_pixels; vp = vp + 1){
        pixel = firstPixel + ((vp *increment) * img.width);
        for(int hp = 0; hp < horz_pixels; hp = hp + 1){
            loadPixels();
            println("Horizontal Pixels: ", hp, "Vertical Pixels: ", vp );
            println("Pixel Location is thus: ", pixel);
            float r = red(img.pixels[pixel]);
            float g = green(img.pixels[pixel]);
            float b = blue(img.pixels[pixel]);
            renderPixel( hp, vp, r, g, b); //
            pixel = pixel + increment;//update pixel here
        }
    }

    pdf.dispose();
    pdf.endDraw();
}

void renderPixel( int horizontal_pixel, int vertical_pixel, float inpR, float
    //write to the pdf and sketch
    noStroke();
    pdf.noStroke();
    fill(inpR,inpG,inpB, 255);
    pdf.fill(inpR,inpG,inpB, 255);
    rect(horizontal_pixel*10, vertical_pixel*10, increment, increment);
    pdf.rect(horizontal_pixel*10, vertical_pixel*10, increment, increment);

```

