

Design Programming

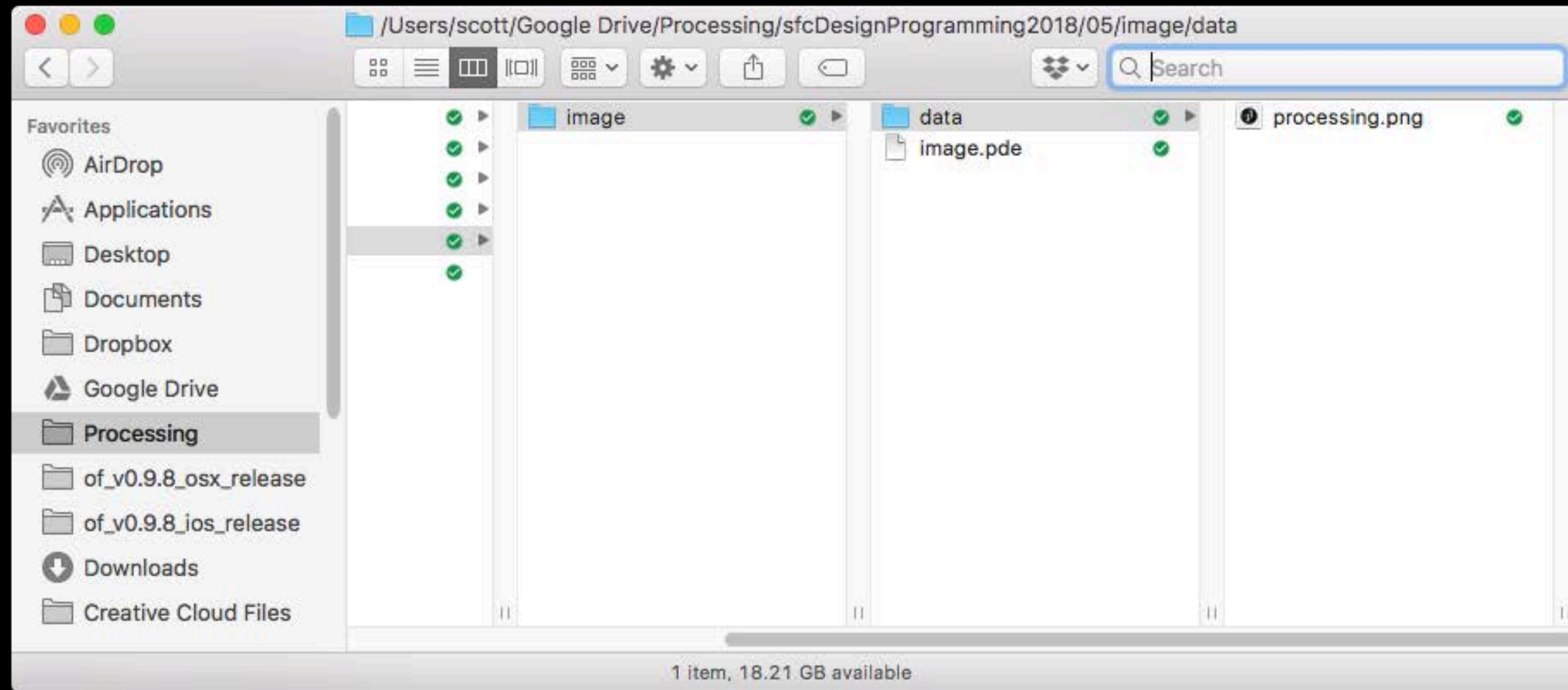
Spring 2018 - 5

2018.5.16 Keio University, SFC

画像について

拡張子	色深度	透明度
<code>.gif</code>	1 – 8 bit	1 bit
<code>.jpg</code>	24 bit	—
<code>.png</code>	1 – 24 bit	8 bit

画像データの格納場所



スケッチフォルダ/data/ 内

PImage

```
PImage img;
```

```
void setup() {  
  size(500, 500);  
  img = loadImage("processing.png");  
}
```

```
void draw() {  
  image(img, 0, 0, width, height);  
}
```

PImageという独自のデータ型の変数を宣言

loadImage関数を使って画像を読み込む。
さきほどのフォルダ内であればファイル名のみ

image関数で用意した画像を描画する。

パラメータは

- PImageの変数
- 画像の左上角の横座標
- 画像の左上角の縦座標
- (幅)
- (高さ)



tint関数

```
PImage img;
```

```
void setup() {  
  size(500, 500);  
  img = loadImage("processing.png");  
}
```

```
void draw() {  
  tint(mouseX, 0, mouseY);  
  image(img, 0, 0, width, height);  
  noTint();  
  image(img, 0, 0, width / 4, height / 4);  
}
```

tint関数で色を指定することで、指定の色合いをつけることができる

noTint関数でtint関数を無効化



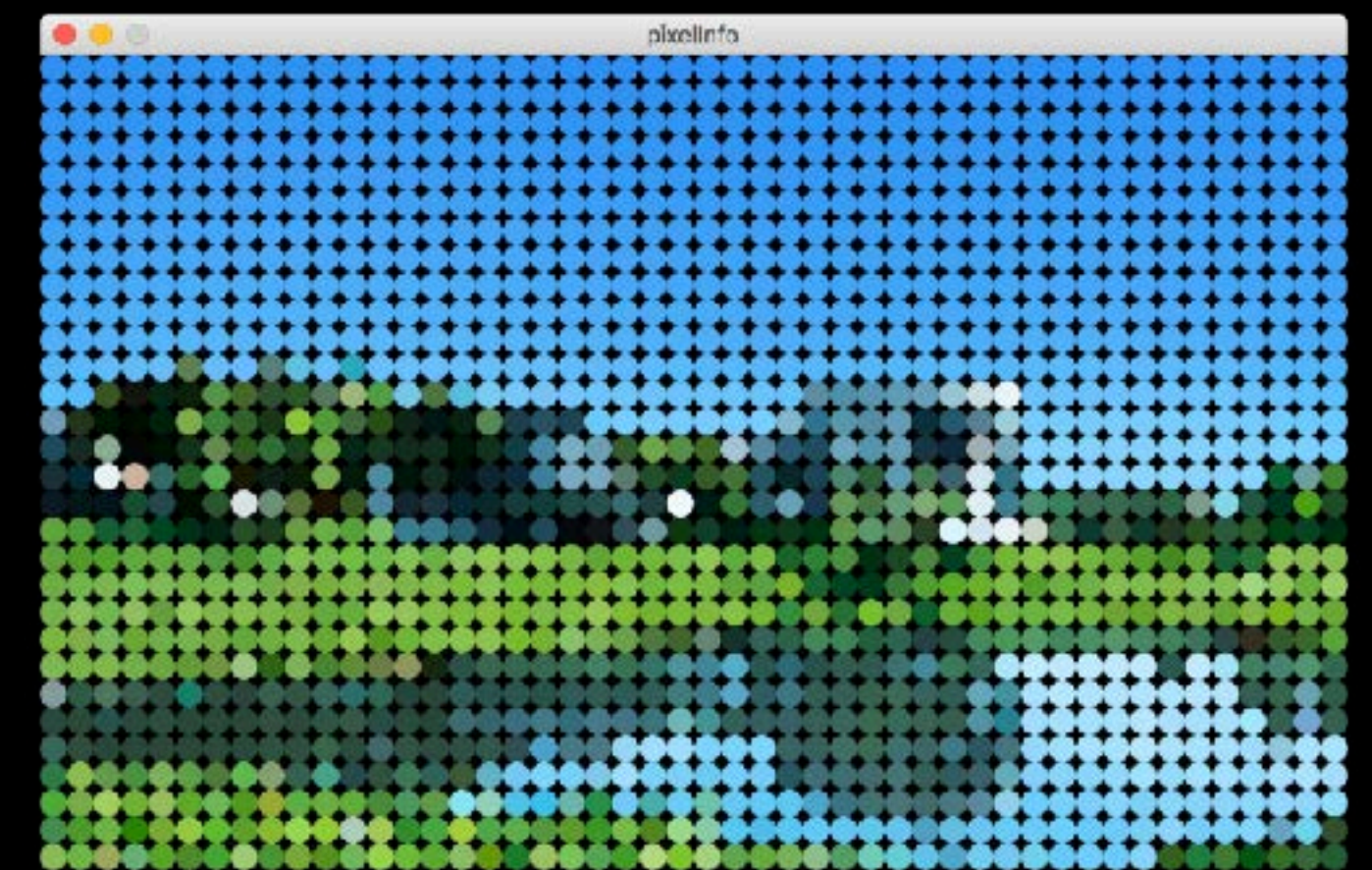
画像の情報を取り出す

```
PImage img;  
int p = 30;
```

```
void setup() {  
  img = loadImage("sfc.jpg");  
  size(720, 450);  
  ellipseMode(LEFT);  
  noStroke();  
}
```

```
void draw() {  
  background(0);  
  for (int y = 0; y < img.height; y+=p) {  
    for (int x = 0; x < img.width; x+=p) {  
      color c = img.get(x, y);  
      fill(c);  
      ellipse(x / 2, y / 2, p / 2, p / 2);  
    }  
  }  
}
```

get(x, y)で画像の左上角からの指定の座標の色情報を取得できる。



演習1

さきほどの例を利用して、色や描画の命令を変更し
OpenProcessingにアップロードしなさい。

《Toposcan / Ireland 2013》



高谷史郎による作品.

ディスプレイの一部のピクセル
の情報から横一直線の線で埋め
尽くされていく.

http://www.ntticc.or.jp/ja/feature/2015/Openspace2015/Works/Toposcan_Ireland_2013_j.html

(ICCウェブサイト) より.


```
PImage img;  
int p = 30;
```

画像の情報を取り出す

```
void setup() {  
  img = loadImage("sfc.jpg");  
  size(720, 450);  
  noFill();  
}
```

red(c), green(c), blue(c)により指定したcolor変数の各色の値を取り出せる。

```
void draw() {  
  background(0);  
  for (int y = 0; y < img.height; y+=p) {  
    for (int x = 0; x < img.width; x+=p) {  
      color c = img.get(x, y);  
      stroke(255);  
      int r = (int)(red(c) + green(c) + blue(c)) / 100;  
      ellipse(x / 2 + p / 4, y / 2 + p / 4, r, r);  
    }  
  }  
}
```

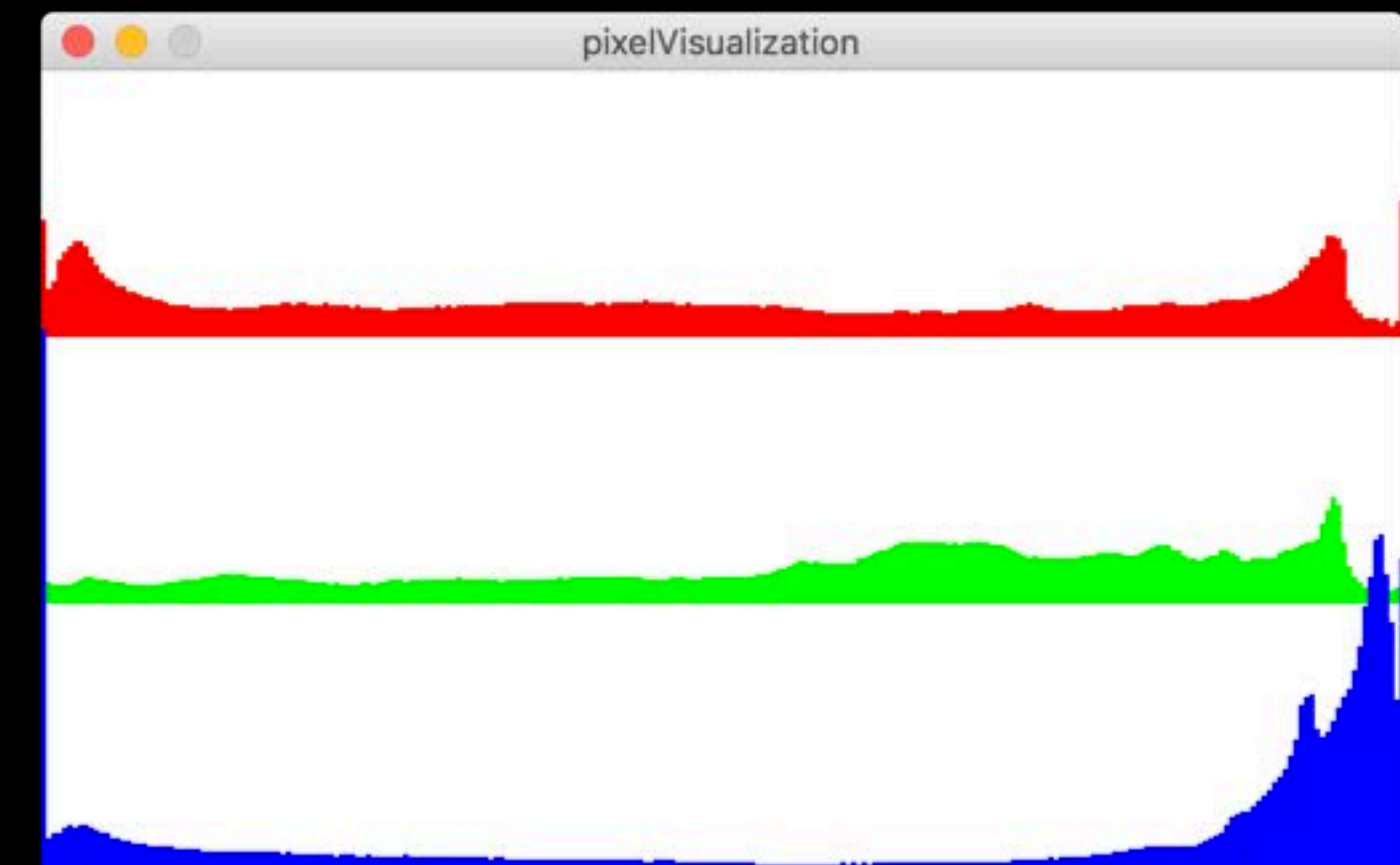


網点プロジェクトクション



IAMASの「車輪の再発明プロジェクト」
内でのクワクボリョウタによる技法。
RGBそれぞれの色を画像から読み込み、
値に応じた針穴をあけ、RGBそれぞれの
光を透かし合成する。

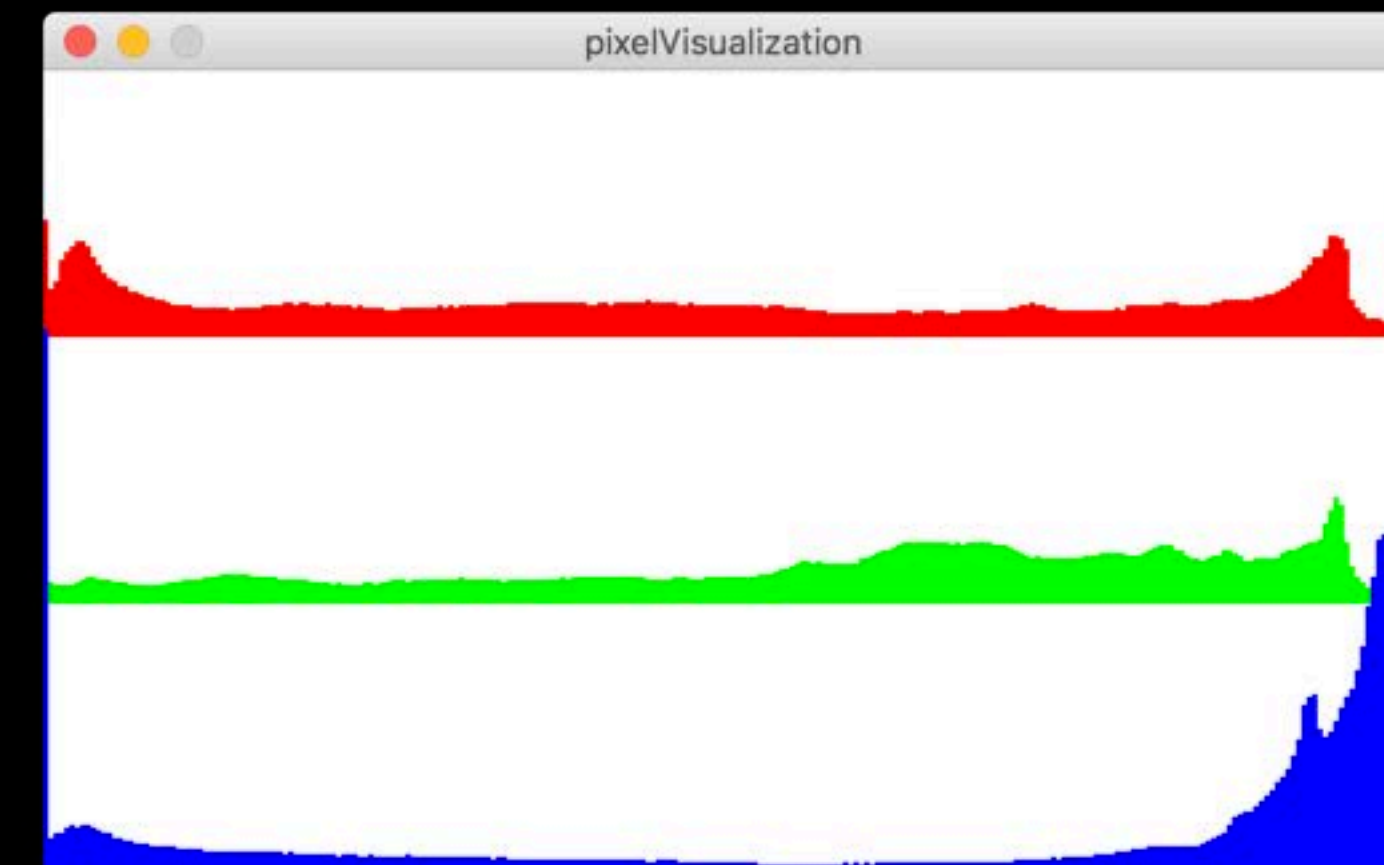
画像の情報を取り出す



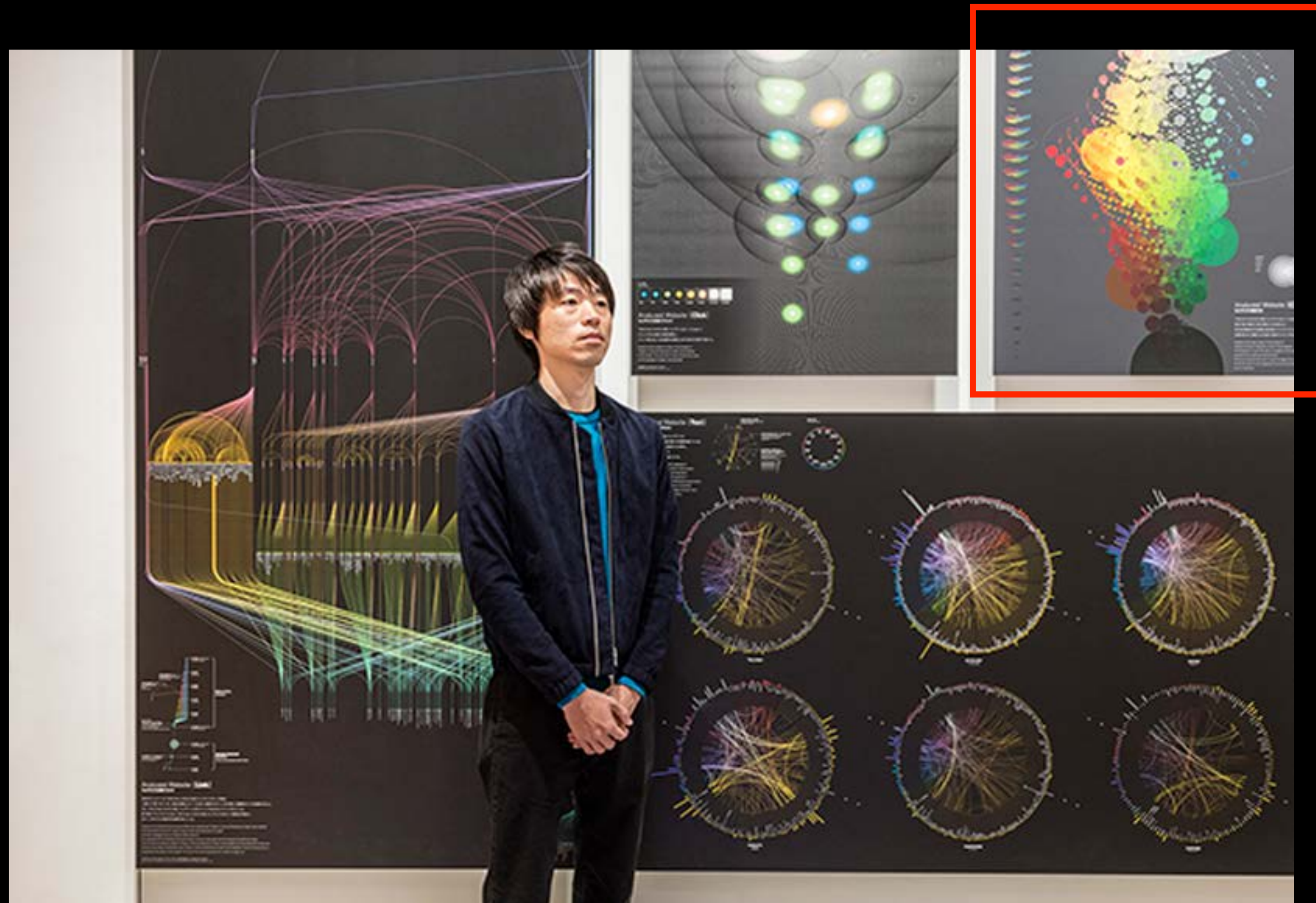
```
PImage img;  
int norm = 30000;  
int h;
```

画像の情報を取り出す

```
void setup() {  
  img = loadImage("sfc2.jpg");  
  size(512, 300);  
  noStroke();  
  h = height / 3;  
}  
  
void draw() {  
  background(255);  
  for (int i = 0; i <= 255; i++) {  
    int r = 0, g = 0, b = 0;  
    for (int y = 0; y < img.height; y++) {  
      for (int x = 0; x < img.width; x++) {  
        color c = img.get(x, y);  
        if (red(c) == i) r++;  
        if (green(c) == i) g++;  
        if (blue(c) == i) b++;  
      }  
    }  
    fill(255, 0, 0);  
    rect(i * 2, h, 2, - h * r / norm);  
    fill(0, 255, 0);  
    rect(i * 2, h * 2, 2, - h * g / norm);  
    fill(0, 0, 255);  
    rect(i * 2, h * 3, 2, - h * b / norm);  
  }  
}
```



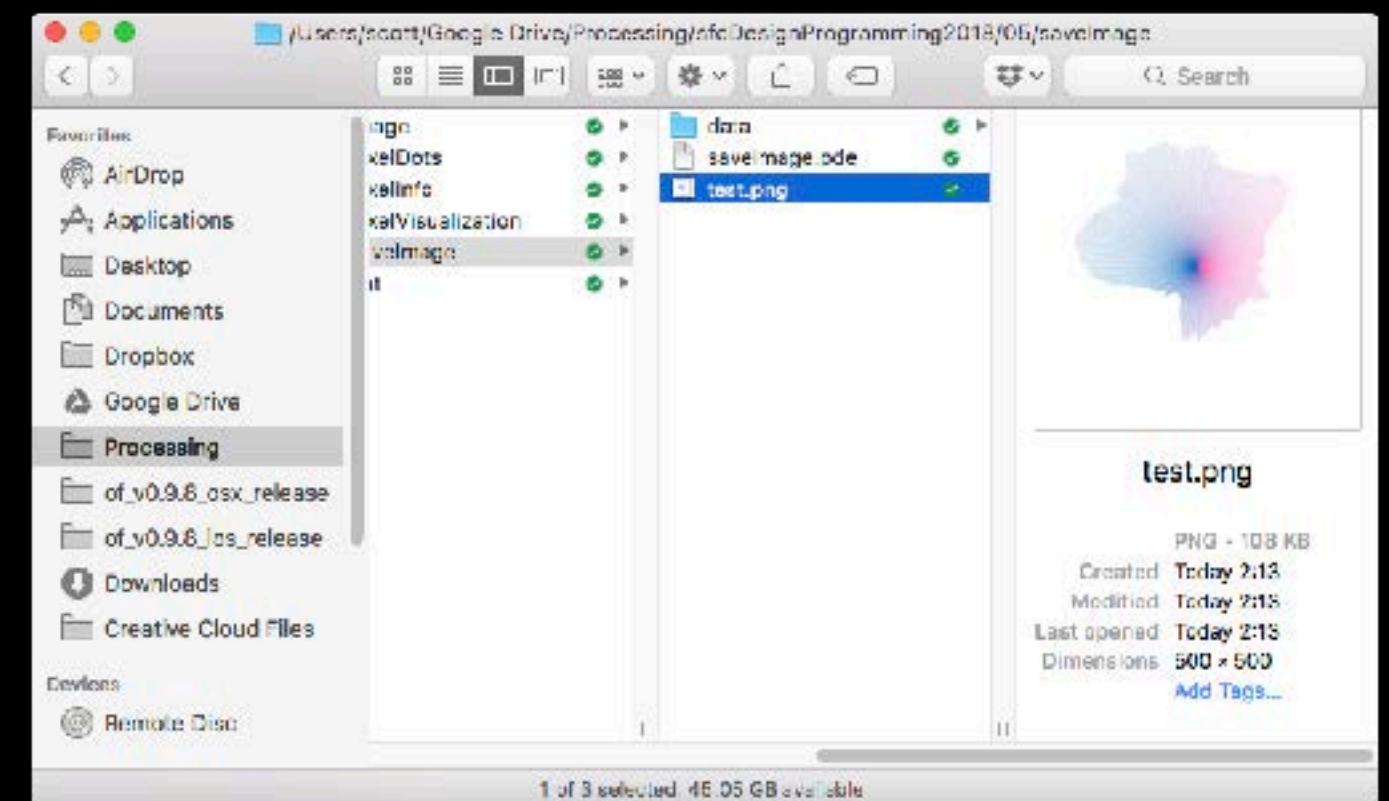
ウェブサイトの解析



21_21_DESIGN_SIGHTで行われた「デザインの解剖展」の中野豪雄さんの作品. ウェブサイトの色のデータビジュアライゼーション.

画像を書き出す

```
size(500, 500);  
background(255);  
  
for (int i = 0; i < 360; i++) {  
  float x = width / 2 + 300 * sin(radians(i)) * noise(float(i) / 50);  
  float y = height / 2 + 300 * cos(radians(i)) * noise(float(i) / 50);  
  stroke(127 * (1 + sin(radians(i))), 102, 194, 100);  
  
  line((int)x, (int)y, width / 2, height / 2);  
}  
  
save("test.png");
```



演習2

save関数を利用して、これまでの自身が気に入っているスケッチにキーボードの「s」を押したときだけ画像が保存される機能をつけなさい。

また、OpenProcessingでアイコン未設定の場合は書き出した画像を設定しなさい。

課題

- 今日学習した「画像を描画する, 画像の情報を抽出する, 画像を書き出す」を使用してスケッチを1つアップロードしなさい.

アップロード先は「04 image」とする.

※今日学習していない分野でもすでに自分が知っている技術は使用可能とする.

〆切: 5月22日24時まで

次回

- 始まりと終わりの定義
- 時間軸の設計