

モバイルシステム演習

Processing

- 画像の表示 -

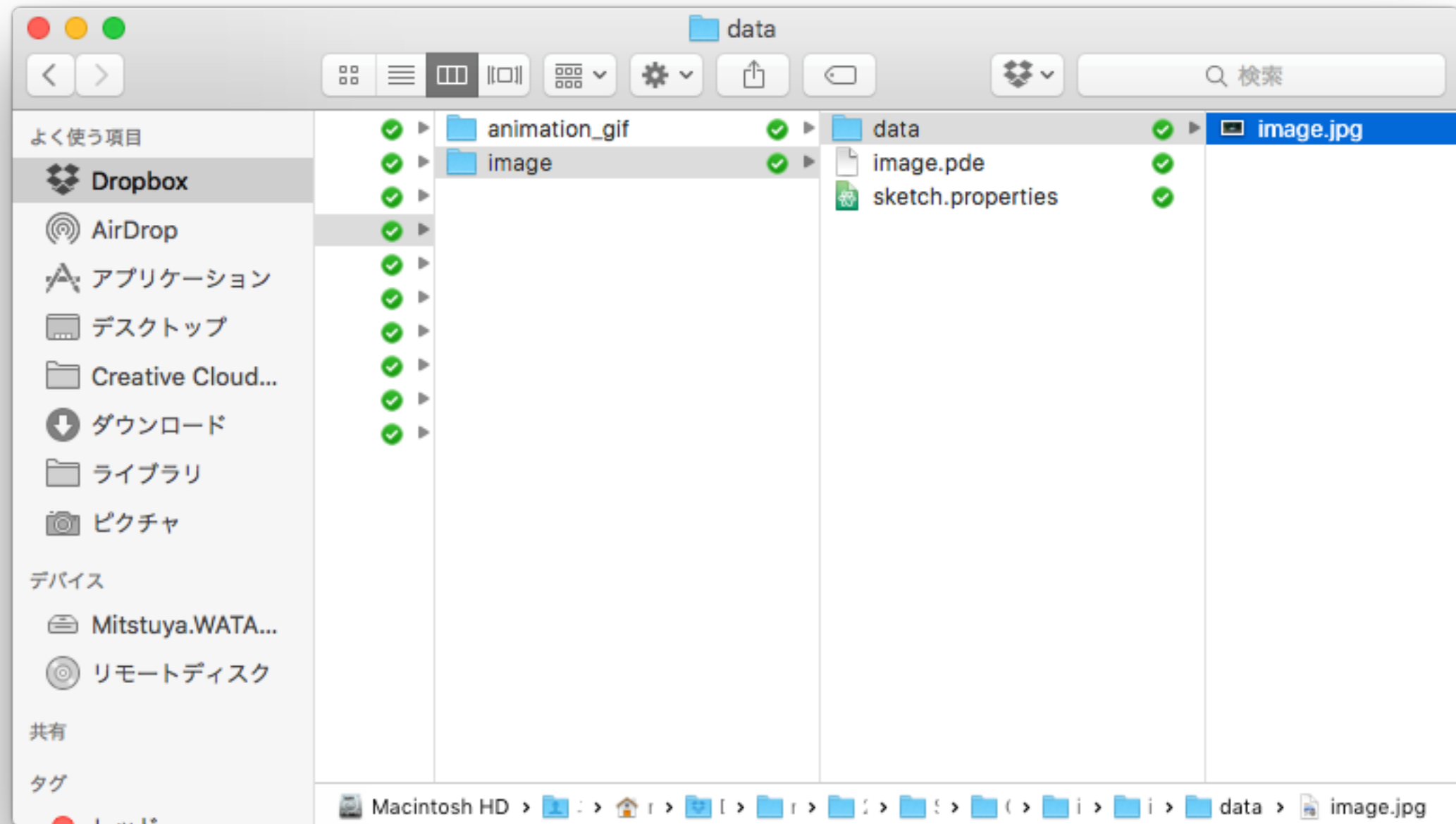
画像の表示

まずは画像を用意する

<https://unsplash.com/>



dataフォルダを作成し、 用意した画像をdataフォルダに格納

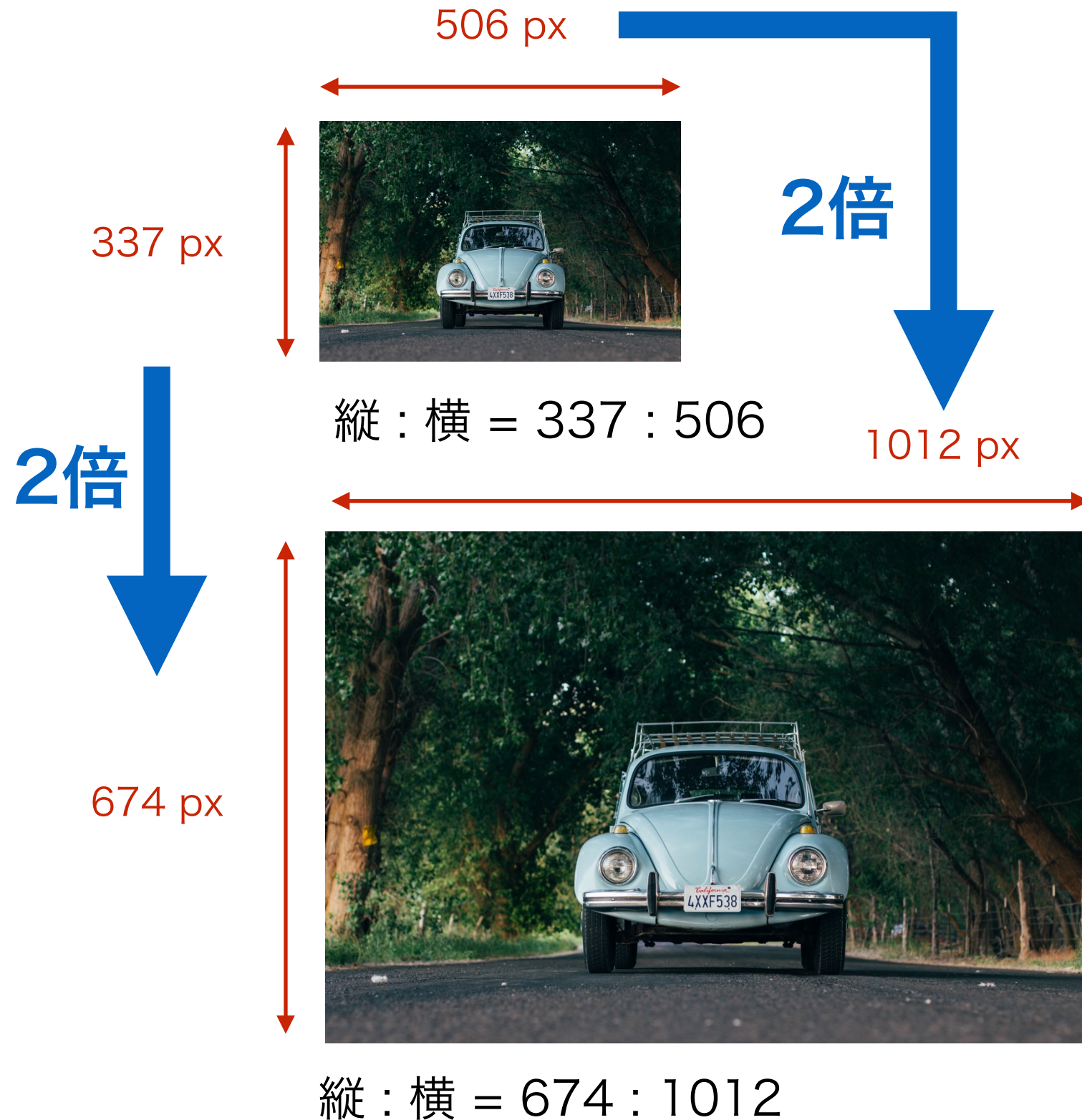


PIimage ...画像を扱うクラス

image ... (PIimageオブジェクト, x座標, y座標, 幅, 高さ);

**ピクセルアスペクト比(画素の幅と高さの数値比率)を
合わせて表示する。**

アスペクト比



アスペクト比
縦 : 横 = 1 : 1.5

```
PImage img; //画像を格納するための変数 (PImage型)
```

```
//画面のサイズ (アスペクト比を合わせる)
```

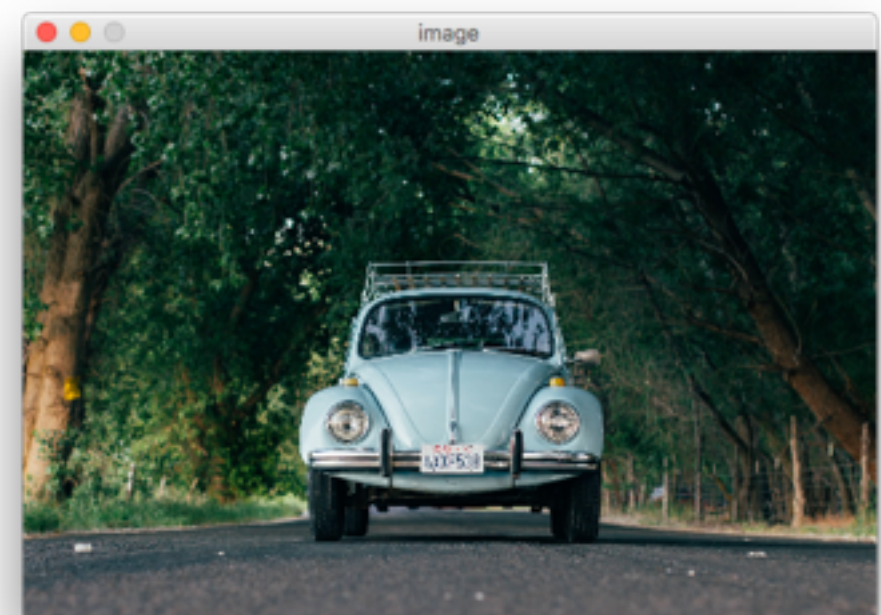
```
size(506, 337);
```

```
img = loadImage("image.jpg");
```

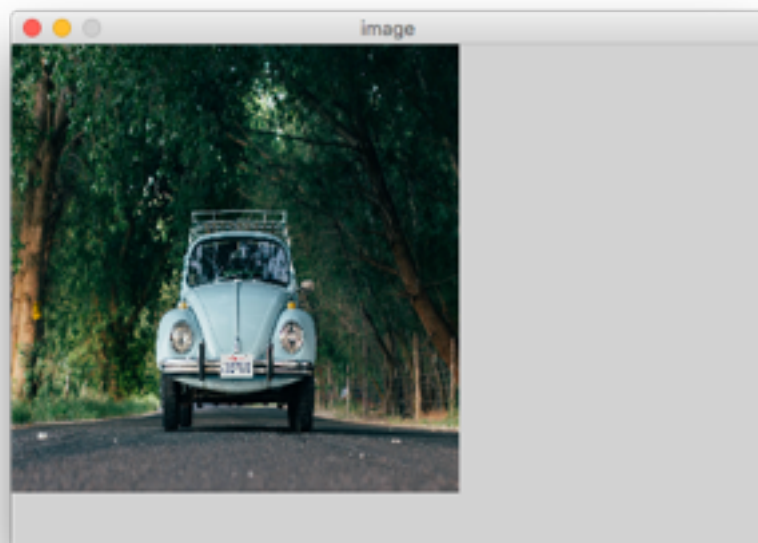
```
//ファイルから画像を読んで、imgに格納
```

```
image(img, 0, 0, width, height);
```

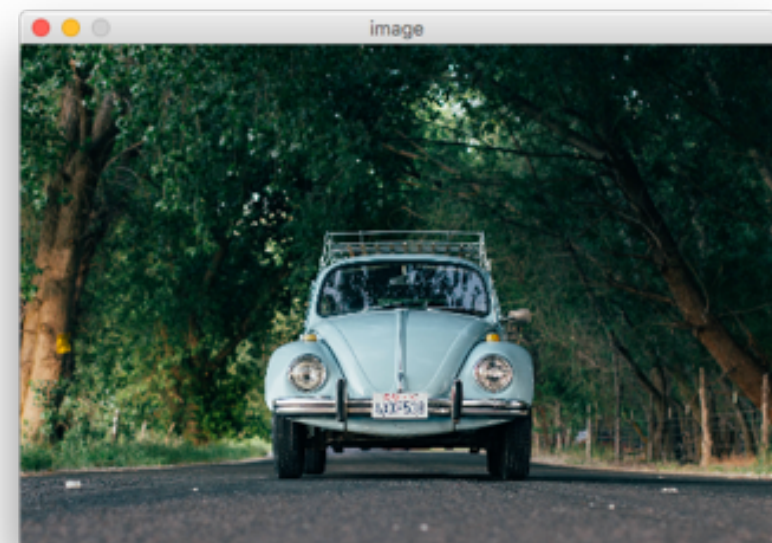
dataフォルダに
格納したファイル名



アスペクト比を合わせないと 画像が歪む



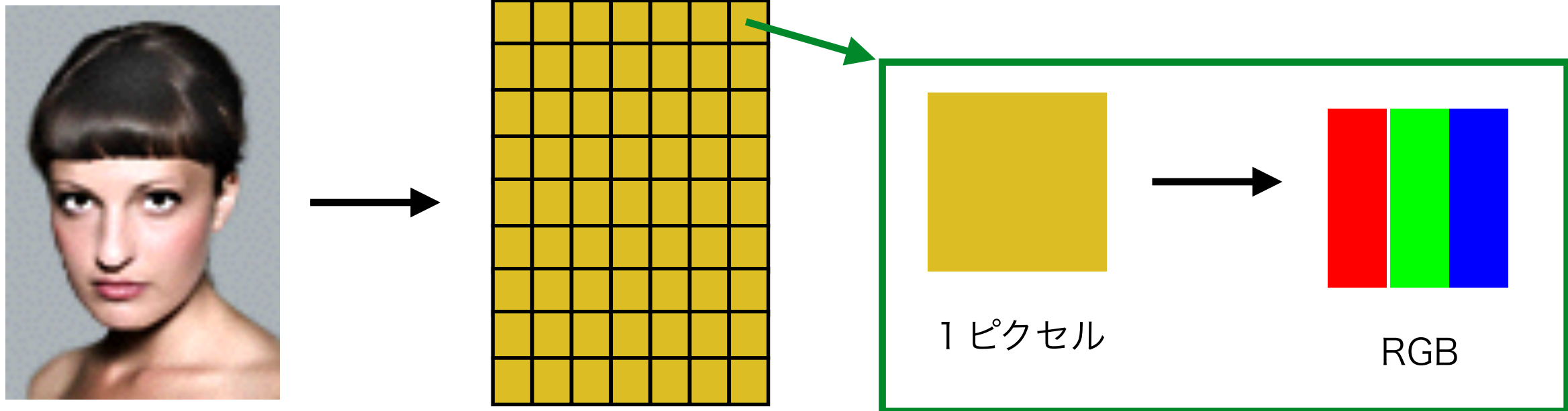
アスペクト比が正しくない



アスペクト比が正しい

画像の加工

画像はピクセルの集合体

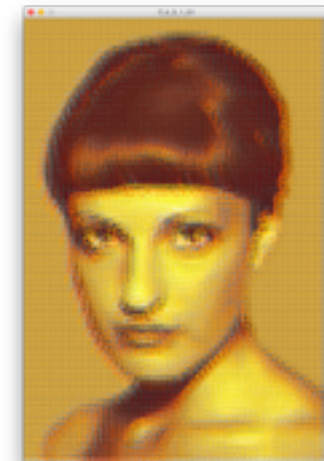
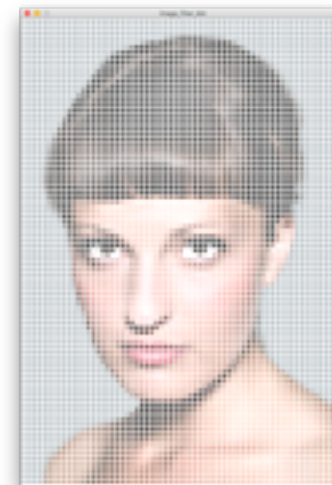
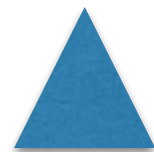
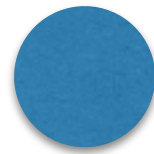
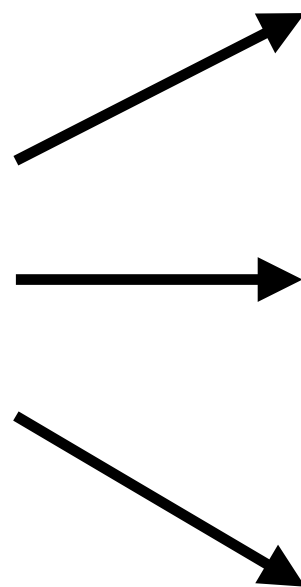
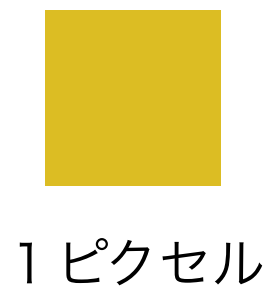


1ピクセル毎に色情報を持っている

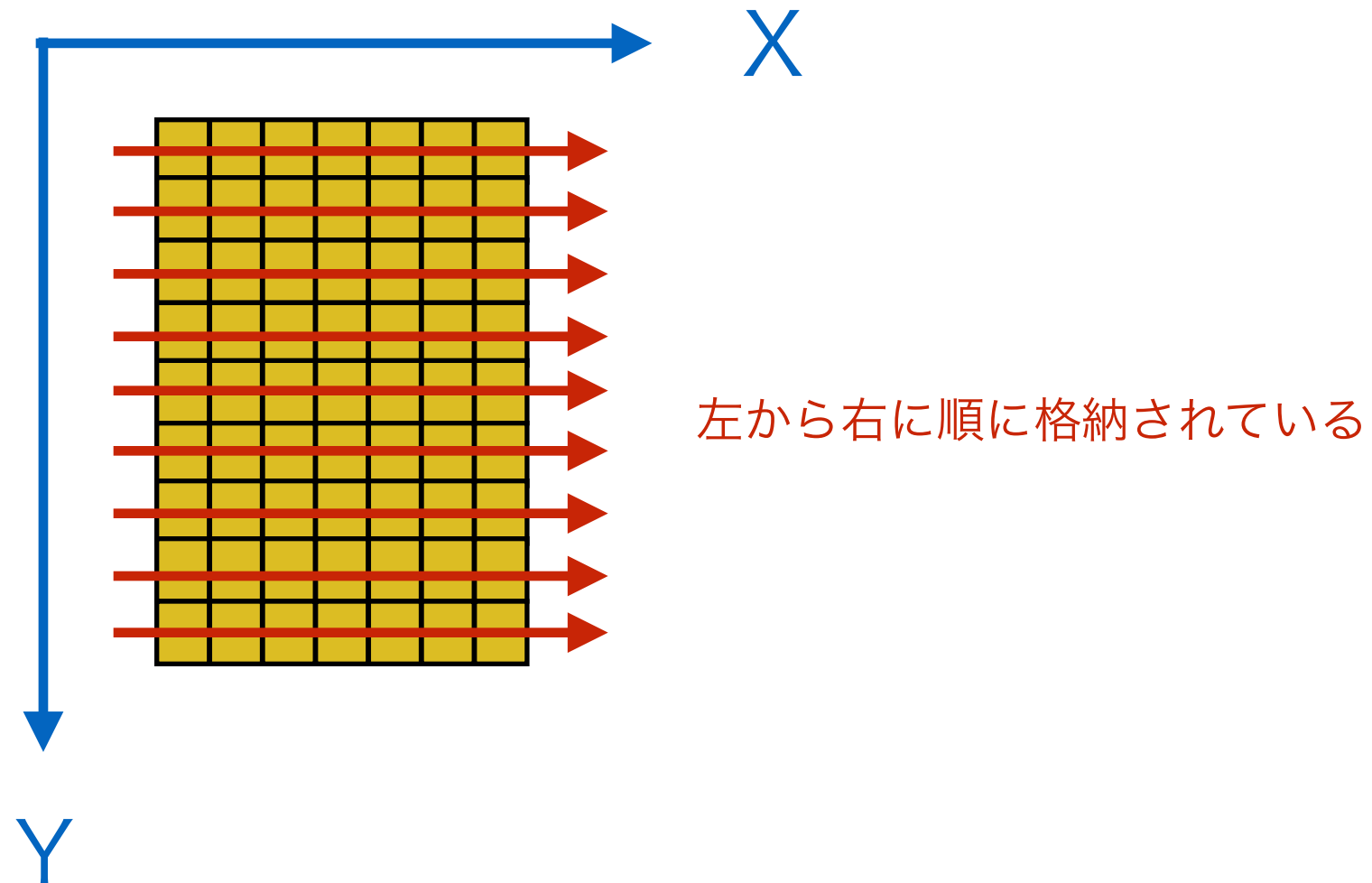
ピクセルの色情報を利用して、
新しく図形を描画する



画像が加工される



ピクセルデータは配列で情報を持つ



```
PImage img;
```

```
void setup() {
```

```
    // 画像サイズのアスペクト比と合わせる
```

```
    size(670, 970);
```

```
    img = loadImage("pic.png");
```

```
}
```

```
void draw() {
```

```
    background(255);
```

```
    float mouseXFactor = map(mouseX, 0,width, 1,15);
```

1ピクセル毎の情報を取得する

```
    for (int gridX = 0; gridX < img.width; gridX++) {  
        for (int gridY = 0; gridY < img.height; gridY++) {
```

```
            // grid position + tile size
```

```
            float tileWidth = width / (float)img.width;
```

```
            float tileHeight = height / (float)img.height;
```

```
            float posX = tileWidth * gridX;
```

```
            float posY = tileHeight * gridY;
```

```
            // get current color
```

```
            color c = img.pixels[gridY*img.width+gridX];
```

```
            fill(c);
```

```
            noStroke();
```

```
            //rect(posX, posY, mouseXFactor, mouseXFactor);
```

```
            ellipse(posX, posY, mouseXFactor, mouseXFactor);
```

```
        }
```

```
    }
```

```
}
```

1ピクセルの色情報を取得