

グレースケール画像の生成

- 変数の宣言時に領域（サイズ・型）の確保

```
cv::Mat gray_img = cv::Mat(src_img.size(), CV_8UC1);
```

- 幅と高さ: 入力画像src_imgと同じ
 - ➡ `src_img.size()` で高さ と 幅 の指定
 - 型: `CV_8UC1`
 - ➡ 8bit unsigned, channel=1（輝度のみ）
- （create関数を使う方法もある）

グレースケール変換

```
//画像の走査
for (y=0; y<src_img.rows; y++) {
    for (x=0; x<src_img.cols; x++) {
        //画素値の取得
        cv::Vec3b s = src_img.at<cv::Vec3b>(y, x);
        uchar val = 0.11448*s[0] //B
                    + 0.58661*s[1] //G
                    + 0.29891*s[2]; //R
        gray_img.at<uchar>(y, x) = val;
    }
}
```

uchar :

unsigned char と同じ (画素1チャンネルの型)

グレースケール変換

- 輝度を以下の式で算出

$$V(x, y) = 0.11448B(x, y) + 0.58661G(x, y) + 0.29891R(x, y)$$

- (x, y) : 画素の座標
- $B(x, y), G(x, y), R(x, y)$: 指定画素のRGB値
- $V(x, y)$: 指定画素の輝度