

# 画像分析

米田 幸生



図1 検索対象の画像

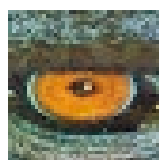
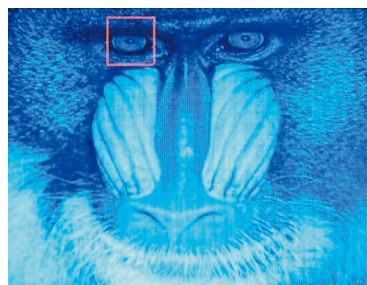
図2 検索に使う  
テンプレート画像

図3 テンプレート・マッチングの実行結果

## 分析1…似ている画像を探し出す テンプレート・マッチング

テンプレート・マッチングは、画像の中から指定した画像（テンプレート）と似ている画像（の位置）を探す処理です。物体検出などで使われます。

最近の物体検出には人工知能アルゴリズムであるR-CNN (Region-based CNN) やYOLO (You Only Look Once) といったものがよく使われますが、テンプレート・マッチングは基本的な処理と言えます。

テンプレート・マッチングの基本のアルゴリズムを以下に示します。

全体画像の左上から右下まで1ピクセルずつ右にずらしながら、テンプレート画像との類似度を計算し、類似度が最大または最小の位置をマッチング位置とする

この左上から右下まで順に見ていくことをラスタ・スキャンと言います。

テンプレート・マッチングにおいて、類似度を計算する方法として以下があります。

### ● 方法1…SSD

SSD (Sum of Squared Difference) は、「画素値の差分の2乗値の和 (2乗誤差)」を類似度とする手法です。誤差値が最小になる位置が、類似度が最も高いこととなります。

### ▶ 数式

以下の式で定義できます。

$$S(dx, dy) = \sum_{x=0}^{w-1} \sum_{y=0}^{h-1} (I(dx+x, dy+y) - T(x, y))^2 \quad \dots\dots(1)$$

ただし、入力画像の画素値： $I(x, y)$ 、テンプレート画像の画素値： $T(x, y)$ 、テンプレート画像の幅： $w$ 、高さ： $h$ 、走査位置： $dx, dy$ とする

### ▶ プログラム

プログラムをリスト1に示します。入力画像およびテンプレート画像は、展開後のメモリ容量を考慮して、グレイ・スケールに変換して比較検証しています。しかし、テンプレートに大きなサイズの画像を使うとM5Stackでpanicが発生します。テンプレート画像のみをデコードしてメモリに展開しておき、入力画像をテンプレート画像のサイズ分のみデコードしながら比較していくともう少し耐えられると思いますが、今回はできていません。ここについては、少し悩んだのですが、全体の作りとしてJPEGデコーダを1つしか使わないようにしていたのと、プログラムを作る目的をテンプレート・マッチング自体の実行に置いてしまったため、この形にしています。

画像内に複数のマッチング箇所があるケースには対応できていません。例えば複数の全く同じ画像のマッチングや、類似マッチングの順位を出すなど、まだまだ改造の余地があります。