

ガンマ補正と変動画素数のしきい値設定で
検知もれや誤検知を防ぐ

フレーム間差分と $+\alpha$ の技で 確実に異常検知

吉田 大海



写真1 閉店後の倉庫…基本の画像処理で見守れないか検討する

閉店後の貴金属店に強盗が押し入り、翌日になって事態を把握した店員が警察に通報する…。そんな監視カメラ映像が報道番組のビデオで最近流れるようになりました。カメラ映像から侵入者などの店内異常を検知するには、高度なAI技術が必要になると諦めがちです。しかし、環境を限定すれば簡単かつ安価に閉店後の見守りシステムを構築できます(写真1)。ここでは、そのソフトウェア部分となる画像処理を中心に解説していきます。

基本…フレーム間差分

● 動きをフレーム間差分で捉える

監視カメラの映像は動画であり、動画はフレームという静止画の集まりでできています。アニメ映像は、1秒間に何枚もの絵(フレーム)を高速で切り替えることで実現していますが、それと同様です。例えば、動きの滑らかな映像は一般的に1秒間に60枚のフレーム

リスト1 フレーム間差分で動きを検知する

```
int FDD(cv::Mat img, cv::Mat img2, int t)
{
    int x, y;
    // 画素値のx座標、y座標

    int X = img.cols, Y = img.rows;
    // 画像の横画素数X, 画像の縦画素数Y

    int BF, F, P; // 1フレーム前と現フレーム

    int count = 0; // 検知画素をカウント

    for (y = 0; y < Y; y++) {
        for (x = 0; x < X; x++) {

            // ----< 入力画像から画素値を読み込む >----
            BF = img.at<cv::Vec3b>(y, x)[1];
            F = img2.at<cv::Vec3b>(y, x)[1];

            P = abs(BF - F); // フレーム間差分

            if (P > t) { P = 255; count++; }
            else { P=0; }

            img.at<cv::Vec3b>(y, x)[0] = P;
            img.at<cv::Vec3b>(y, x)[1] = P;
            img.at<cv::Vec3b>(y, x)[2] = P;

        }
    }

    return count;
}
```

数で構成されており、単位にはfps(frames per secondの略)が用いられます。つまり、この場合は60fpsです。動きはフレーム間で生じた変化と捉えることができるため、フレーム間差分によって検出できます。プログラムをリスト1に示します。

● 無人かつ固定カメラという条件設定で強みを生かす

フレーム間差分だけで不審者侵入などの異常を検知するのは至難というより無謀と思われる方が大半だと思います。しかし、撮影条件を限定し、入念にパラメータを設定すれば十分に機能します。まず、本システムの運用条件は次の2つです。

- 1, 閉店後かつ無人である