

画像応用数学特論

12/12 課題

2014/01/14

知能システム研究室 水野良祐

メールアドレス [\[REDACTED\]@wm.hiroshima-cu.ac.jp](mailto: [REDACTED]@wm.hiroshima-cu.ac.jp)

○ソースコードに関して

・作成環境

プログラム作成の環境は以下のようにしている。

使用した OS : ubuntu 12.04 LTS

コンパイラ : g++ バージョン 4.6

プログラミング言語 : C++

メモリ : 2GB

CPU : Intel® Xeon(R) CPU X5460 @ 3.16GHz × 4

・コンパイル方法

以下のコマンドでコンパイルを行うことが可能である。

```
g++ -c graph.cpp
```

```
g++ -c ImgEx.cpp
```

```
g++ -c maxflow.cpp
```

```
g++ -o stereo_graph stereo_graph.cpp graph.cpp maxflow.cpp ImgEx.cpp `pkg-config --cflags opencv` `pkg-config --libs opencv`
```

・画像形式、ライブラリのインストール方法

画像の形式は png ファイルとなっている。res フォルダ内に right.png、left.png と置いておくと認識される。

必要なライブラリは opencv のみとなっている。

・実行方法

./stereo_graph と入力することで実行を行うことができる。

```
please enter rabel value...
```

```
please enter graph or alpha...
```

と出力されるので、上はラベルの初期値を 1~256 の範囲で入力する。下は α 拡張アルゴリズムと階層グラフカットのどちらを選択するかというものであり、graph を入力すると階層グラフカットを選択し、alpha と入力すると α 拡張アルゴリズムを選択する。

それらを入力したら計算が自動的に開始される。

○アルゴリズム

・ α 拡張アルゴリズムによるステレオマッチング

```
for p=すべてのピクセル
     $f_p = \text{初期値}$ 
end for
for ループ = 0 ~ 反復回数 iter
    success = 0
    for  $\alpha = 0 \sim \text{DISPMAX}$ 
        グラフの初期化
        すべてのノードの追加
        for p = すべてのピクセル
            ノードのソース側に  $D(f_p)$ 、シンク側に  $D(\alpha)$  を設定
        end for
        for(p,q) = すべての隣接点
            ノード a のソース側に  $V(f_p, f_q)$ 、シンク側に  $V(\alpha, \alpha)$  を設定
            ノード p とノード a のエッジの重み(双方向)に  $V(f_p, \alpha)$  を設定
            ノード a とノード q のエッジの重み(双方向)に  $V(\alpha, f_q)$  を設定
        end for
        最大、最小流カットアルゴリズムの適用
        E' = 求めたラベルで計算した総コスト関数
        E' < E なら、原罪のラベルを求めたラベルにし、E = E'にし、success = 1 にする
        グラフの消去
    end for
    もし success = 0 ならループを脱出する。
End for
```

・階層グラフカットでステレオマッチング

```
for p=すべてのピクセル
     $B_p = \text{初期値}$ 
end for
ラベルの設定 (A[i]の計算)
for ループ = 0 ~ 反復回数 iter
    success = 0
    for i = 0 ~ A[i] == NULL, i++
        グラフの初期化
        すべてのノードの追加
        for p = すべてのピクセル
            A[i]のうち、 $B_p$  にもっとも近い値を  $a_p$  に設定
            ノードのソース側に  $D(b_p)$ 、シンク側に  $D(a_p)$  を設定
        end for
        for(p,q) = すべての隣接点
            A[i]のうち、 $B_p$  にもっとも近い値を  $a_p$  に設定
            A[i]のうち、 $B_q$  にもっとも近い値を  $a_q$  に設定
            ノード a のソース側に  $V(b_p, b_q)$ 、シンク側に  $V(a_p, a_q)$  を設定
            もし  $V(b_p, b_q) \leq V(a_p, a_q)$  の場合
                ノード a からノード p へのエッジの重みに10000を設定
                ノード a からノード q へのエッジの重みに10000を設定
                ノード p からノード a へのエッジの重みに  $V(a_p, b_q) - V(b_p, b_q)$  が0のうち大き
```

```

    一方を設定
    ノード q からノード a へのエッジの重みに  $V(b_p, a_q) - V(b_p, b_q)$  か 0 のうち大きい方を設定
    一方を設定
    もし  $V(b_p, b_q) \geq V(a_p, a_q)$  の場合
        ノード p からノード q へのエッジの重みに 10000 を設定
        ノード a からノード q へのエッジの重みに 10000 を設定
        ノード a からノード q へのエッジの重みに  $V(a_p, b_q) - V(a_p, a_q)$  か 0 のうち大きい方を設定
        一方を設定
        ノード a からノード p へのエッジの重みに  $V(b_p, a_q) - V(a_p, a_q)$  か 0 のうち大きい方を設定
    end for
    最大、最小流カットアルゴリズムの適用
    E' = 求めたラベルで計算した総コスト関数
    E' < E なら、原罪のラベルを求めたラベルにし、E = E' にし、success = 1 にする
    グラフの消去
end for
もし success = 0 ならループを脱出する。
End for

```

表の計算方法は SAD を採用した。

$$\text{SAD} : \sum_w |I_l(x, y) - I_r(x - d, y)|$$

データコスト、スムーズコストは以下のようになる。

$$D(f_{x,y}) = \sum_w |I_l(x, y) - I_r(x - f_{x,y}, y)|$$

$$V(f_p, f_q) = c |f_p - f_q|$$

$$c = \begin{cases} c_0 & (\text{if } d > T) \\ c_1 & (\text{if } d \leq T) \end{cases}$$

$$d = |I_l(p) - I_l(q)|$$

c は $c_0 = 1$ 、 $c_1 = 3$ とした。

○実行結果

図 1、図 2 のような画像を用意し、それらを縮小処理し、グレースケールへと変換したものを入力へと用いる画像とした。それぞれ left.png、right.png とし、図 3、図 4 として示す。これらはそれぞれ 200*179 ピクセルである。



図1:用意した左の画像



図2:用意した右の画像



図3:left.png



図4:right.png

以下に図3、4を使用して得られた実行結果を図5、6として示す。ラベル数は $N = 16$ とした。



図5:階層グラフカットによるステレオマッチング



図6: α 拡張によるステレオマッチング

○それぞれの手法における速度、精度の比較

それぞれの手法の実行時間は、階層グラフカットがおよそ23～25秒、 α 拡張が45～47秒となった。階層グラフカットの方が α 拡張よりも概ね 1/2 ほど実行時間を短縮できていることが分かる。

作成した α 拡張のアルゴリズムは入力されたラベル数の分、つまり上のアルゴリズムの説明では DISPMAX の分だけ繰り返す必要がある。それに対して階層グラフカットの方は入力されたラベル数をもとに計算し、格納されている配列の数の分だけ繰り返すため、繰り返す回数が少なくて済むためであると考えられる。

精度の比較であるが、 α 拡張では赤ん坊の右手と頭の間隔がきちんと出ているのに対して、階層グラフカットのほうでは埋まってしまっている。またグラフカットの方では所々に黒い点が存在するのが見て取れる。以上より精度の点では α 拡張を用いたステレオカットの方が良いのではないかと考えられる。

以下に階層グラフカットによるステレオマッチングと α 拡張によるステレオマッチングによる相違点を示しておく。

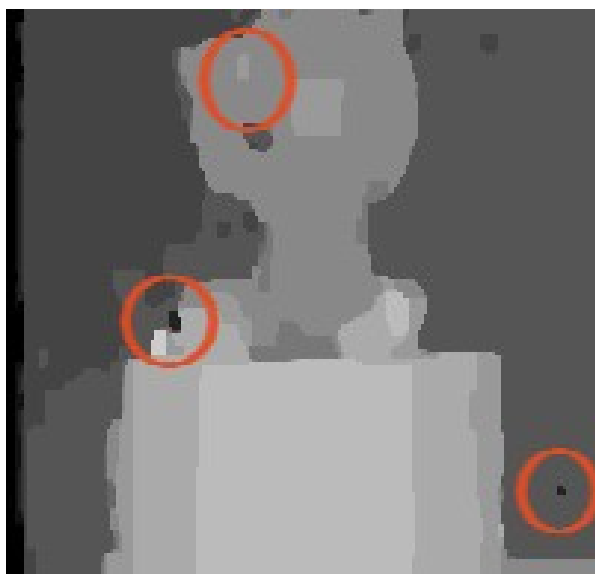


図7:相違点(階層グラフカット)

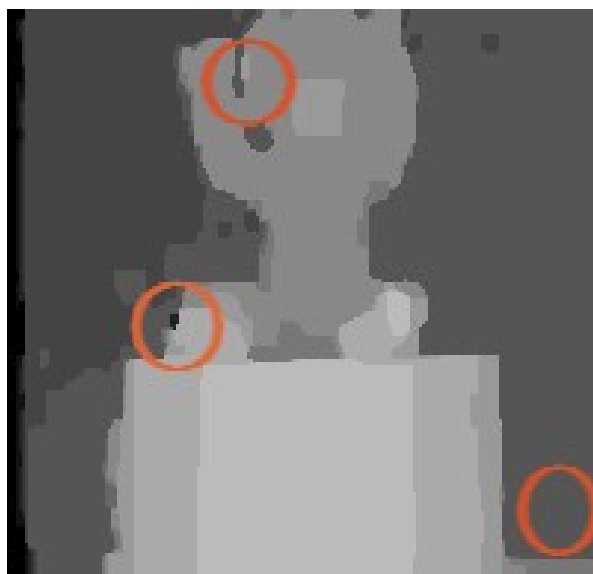


図8:相違点(α 拡張)