

画像処理及び演習

最終課題

森本，塚田，澤野

注意事項 (その1)

- 複数の問題が提示されます．どの問題から解いても構いません．ファイル名だけ気をつけてください．
- ファイル名の例
 - 問題1の場合
 - ➡ final¹_src_学籍番号.cpp
(アルゴリズム説明ファイルに対応したコメント文を書く)
 - ➡ final¹_output_学籍番号.jpg
(結果の画像．printfの結果・出力画像のキャプチャなど．tiff/gif/pngなどでもよい．)
 - ➡ final¹_explain_学籍番号.txt
(アルゴリズム説明ファイル．プログラムのコメント文と対応させて，簡単な説明をしてください．コメント文で採点者が理解できるなら，そのままでもよいです．)

注意事項 (その2)

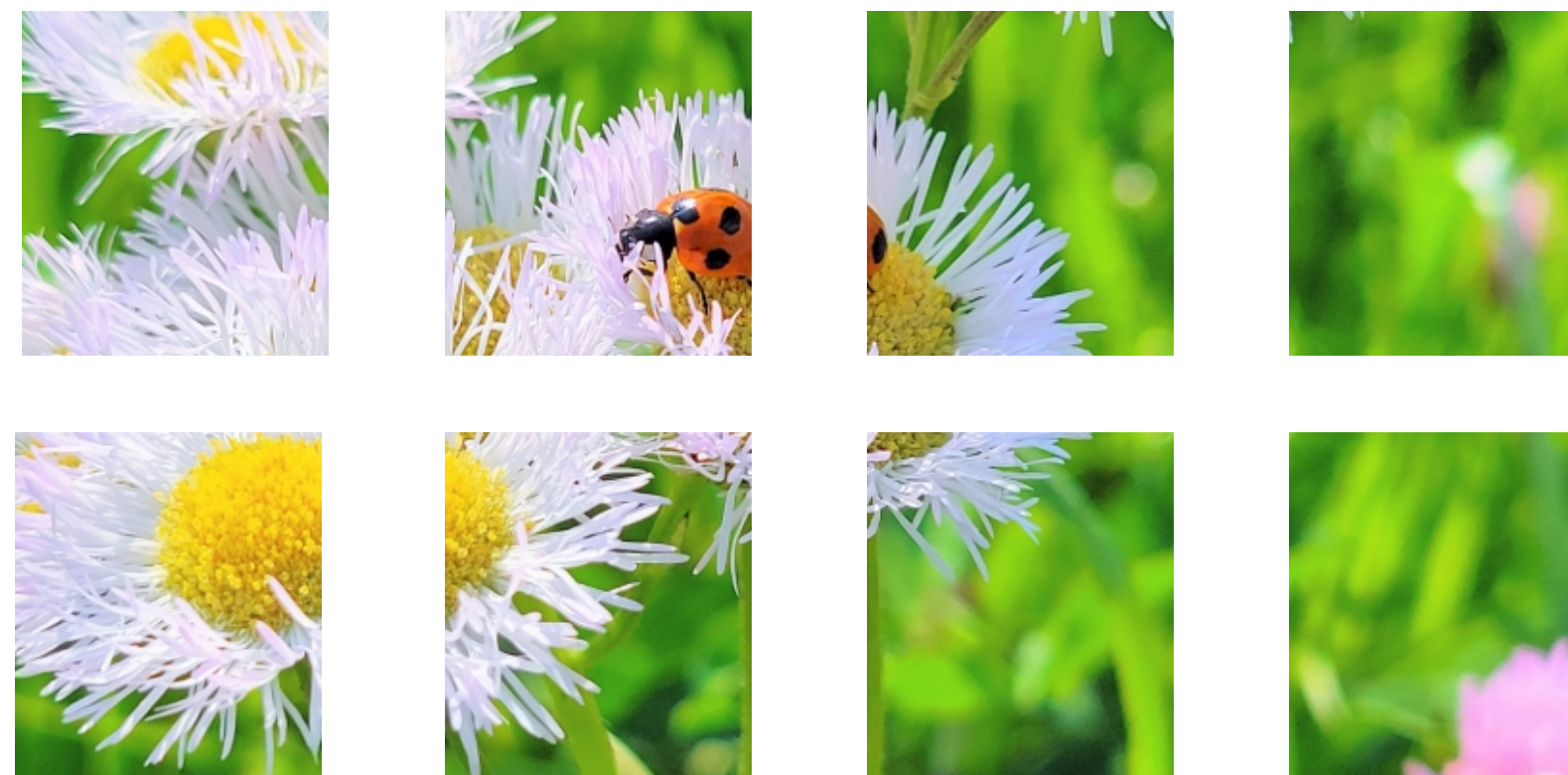
- 質問サイト，オープンなSNSにアップロードして，クラウド上で不特定多数に質問してはいけません。
- Webの参照，友人・TA・教員と相談した場合は，ソースコードに相談した旨を記載してください。得点は減りません。窃盗防止です。
- 他人のPCを盗み見は禁止です。自己防衛してください。
- 期限の延長はありません。投稿ミスがないようにしてください。

問題1

- 入力画像(input1.jpg)とテンプレート8個(template-0.jpg, template-2.jpg, ..., template-7.jpg)が与えられます. テンプレートは入力画像を4x2で分割した画像です. 各テンプレートを入力画像に当てはめたときの左上の座標を出力してください. .



入力画像 (input1.jpg)



テンプレート

テンプレートの番号	座標(x, y)
0:	160, 180
1:	0, 0
2:	320, 180
3:	480, 0
4:	0, 180
5:	480, 180
6:	320, 0
7:	160, 0

出力例

分割されている大きさが既知ですので, キリのいい数字になりますが, 誤差があってもOKとします.

問題2

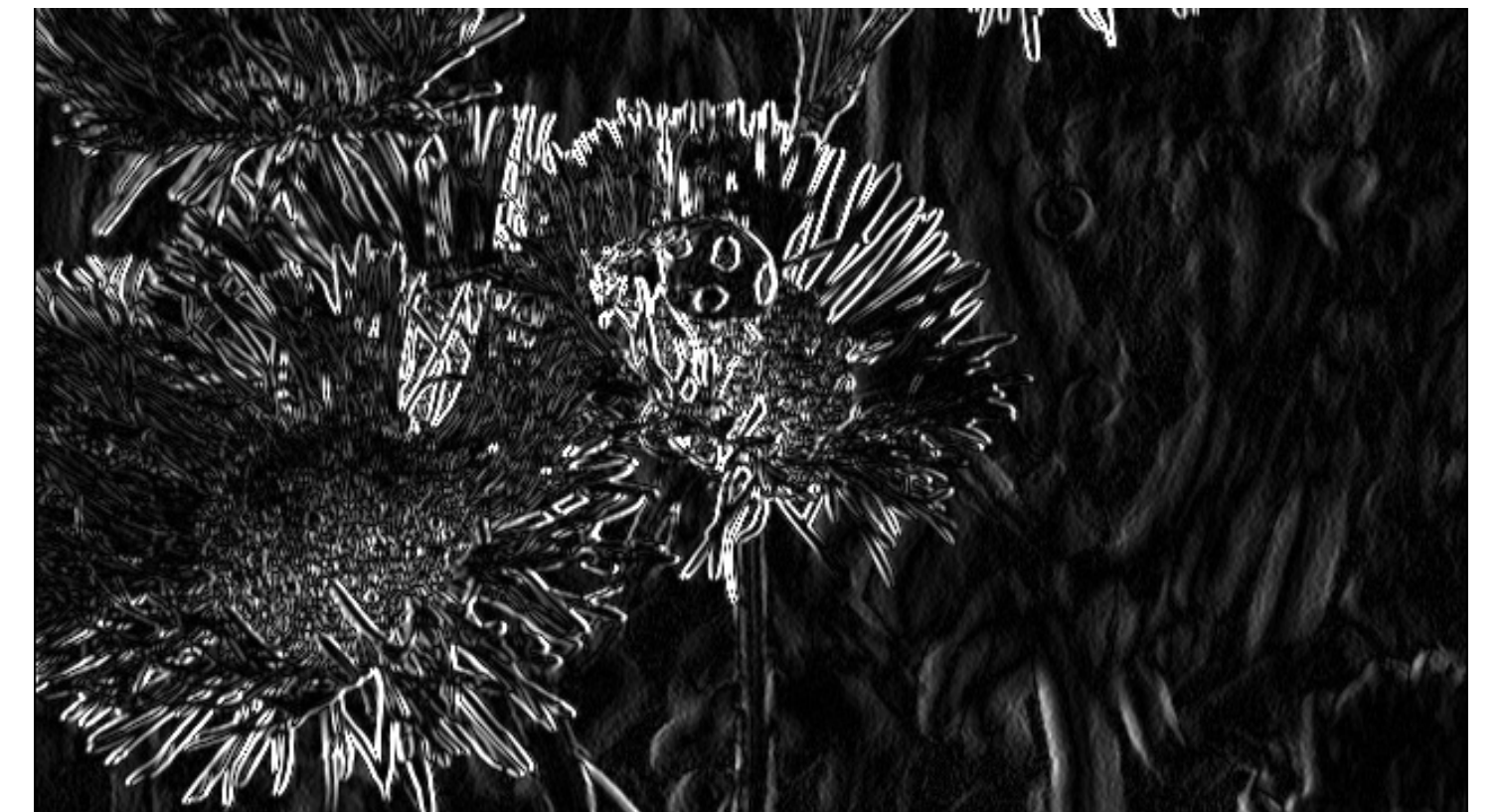
- 入力画像(input1.jpg)が与えられます．フィルタサイズ3x3のSobelフィルタ(横方向)を施した結果を求めよ



入力画像 (input1.jpg)

-1	0	1
-2	0	2
-1	0	1

Sobelフィルタ



出力結果

問題3

- てんとう虫が存在する入力画像(input1.jpg)が与えられます.
赤色と黒色という特徴を用いて、てんとう虫のみを抽出して、
てんとう虫を黒色のバウンディングボックス(幅3)を描画する
プログラムを作成せよ
- 面積閾値を使用してもよい



入力画像 (input1.jpg)




出力例

問題4

- 画像が8個(template-0.jpg, template-1.jpg, ..., template-7.jpg)が与えられます．右隣の画像を出力せよ．存在しない場合は-1を出力すること
- ヒント
 - 画像の右端と左端を走査して，差分が小さい値が隣になります．
 - 差分の絶対値の和の閾値を使うとよい

基準のテンプレートの番号

右隣のテンプレートの番号

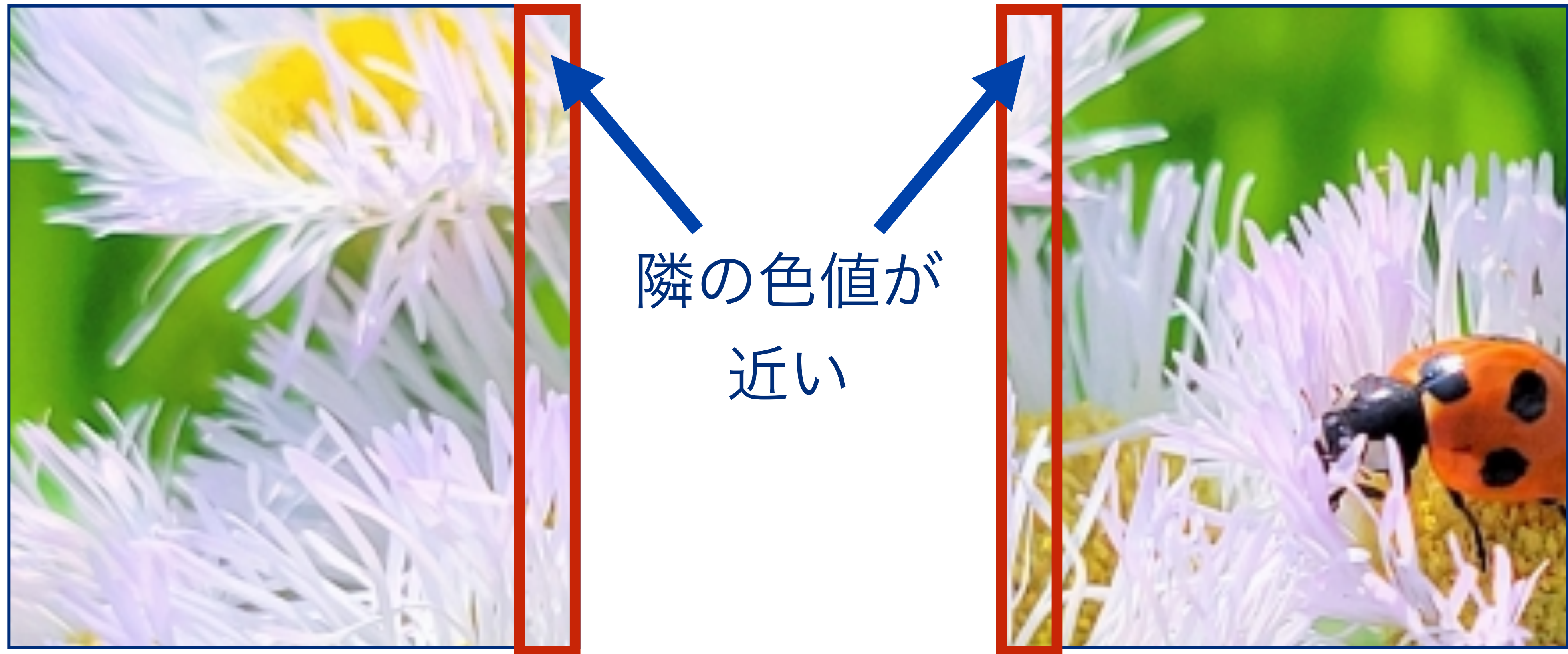


0	=>	2
1	=>	7
2	=>	5
3	=>	-1
4	=>	0
5	=>	-1
6	=>	3
7	=>	6

出力例

問題4のヒント

- 隣り合うテンプレートの右端と左端の画素値を比較して、差の絶対値の総和が最も近い(はず)



問題5

- パズルのピースの画像が8個(template-0.jpg, template-1.jpg, ..., template-7.jpg)が与えられます．このピースの画像を組み合わせて，完成されたパズルの画像を出力してください．パズルのサイズは，ピースのサイズを基準として，4x2になります．与えられた画像のみで作成してください．



出力例

付録 (テンプレートの一括読み込み)

```
#define TEMPLATE_NUM (8)
```

(略)

```
//テンプレートの入力
```

```
cv::Mat template_img[TEMPLATE_NUM];
```

```
char buf[256];
```

```
for (int i=0; i<TEMPLATE_NUM; i++) {  
    sprintf(buf, "template-%d.jpg", i);
```

```
    fprintf(stderr, "%s\n", buf);
```

```
    template_img[i] = cv::imread(buf);
```

```
    if (template_img[i].empty()) {
```

```
        fprintf(stderr, "%s cannot be opened.", buf);
```

```
        return 0;
```

```
    }
```

```
}
```