

画像応用数学特論
12月11日出題レポート：
階層グラフカットでステレオ

奥下拓真

学籍番号：■■■■■

メールアドレス：okuge@ime.info.hiroshima-cu.ac.jp

提出日：1/14

環境

OS	CentOS 5.5
CPU	Intel® Core2 Duo
メモリ	2059128kB
開発環境	Emacs 21.4.1
プログラミング言語	C++
ライブラリ	OpenCV 2.4.6,MAXFLOW
コンパイラ	Gcc4.1.2

●階層グラフカットでステレオマッチング

2 枚のステレオマッチング画像に α 拡張アルゴリズムと階層グラフカットアルゴリズムを用いてステレオマッチングを行い、視差画像を生成し比較する。使用した画像は図 1 のとおりで、画像解像度は 313×277 ピクセル、最大視差を 72 とする。

データコストとスムーズコストは以下のとおりである。

- ・データコスト関数

$$\text{SAD } D(fx,y) = \sum W_{x,y} | |I(x,y) - I_r(x-fx,y,y) | |$$

- ・スムーズコスト関数

$$V(fp,fq) = c |fp-fq| \quad c \text{ は定数}$$

◆アルゴリズムの流れ

- 1.左画像，右画像を読み込む
- 2.出力画像を用意
- 3.グラフのノードの数とエッジの数を求める
- 4.ラベルの初期化
- 5.階層構造の配列 A を設定
- 6.E=とても大きな値
- 7.for ループ=0~とても大きな値
- 8.success=0
- 9.for i=0~階層構造の配列数
- 10.グラフの初期化
- 11.for p = 全てのピクセル
- 12.A[i]のうち、 β_p に最も近い値を α_p に設定
- 13.ノードのソース側に $D(\beta_p)$ 、シンク側に $D(\alpha_p)$ を設定
- 14.end for
- 15.for(p,q)=全ての隣接点
- 16.A[i]のうち、 β_p に最も近い値を α_p に設定

17. $A[i]$ のうち、 β_q に最も近い値を α_q に設定
18. ノード a のソース側に $V(\beta_p, \beta_q)$ 、シンク側に (α_p, α_q)
19. もし、 $V(\beta_p, \beta_q) \leq V(\alpha_p, \alpha_q)$ の場合
20. ノード a からノード p のエッジの重みに大きな値を設定
21. ノード p からノード a のエッジの重みに、 $V(\alpha_p, \beta_q) - V(\beta_p, \beta_q)$ が 0 のうち大きいほうを設定
22. ノード a からノード q のエッジの重みに大きな値を設定
23. ノード q からノード a のエッジの重みに、 $V(\beta_p, \alpha_q) - V(\beta_p, \beta_q)$ が 0 のうち大きいほうを設定
24. もし、 $V(\beta_p, \beta_q) > V(\alpha_p, \alpha_q)$ の場合
25. ノード p からノード a のエッジの重みに大きな値を設定
26. ノード a からノード p のエッジの重みに、 $V(\alpha_p, \beta_q) - V(\beta_p, \beta_q)$ が 0 のうち大きいほうを設定
27. ノード q からノード a のエッジの重みに大きな値を設定
28. ノード a からノード q のエッジの重みに、 $V(\beta_p, \alpha_q) - V(\beta_p, \beta_q)$ が 0 のうち大きいほうを設定
29. end for
30. 最大流・最小カットアルゴリズムの適用
31. E' = 求めたラベルで計算した総コスト関数
32. $E' < E$ なら、現在のラベルを求めたラベルにし、 $E = E'$ にし、 $\text{success} = 1$ にする
33. グラフの消去
34. end for
35. もし $\text{success} == 0$ ならループ脱出
36. end for
37. スケール変換を行う

●実験結果

図1のような ppm 形式の画像を使って、スムーズコストの定数 c を変化させながら α 拡張と階層グラフカットでのステレオマッチングを行い、その結果の比較を行った。図2~7にその結果と表1に計算処理時間を示す。



(a) 左画像



(b) 右画像

図1 入力画像

・スムーズコスト1



図2 階層グラフカット



図3 α 拡張

・スムーズコスト 3

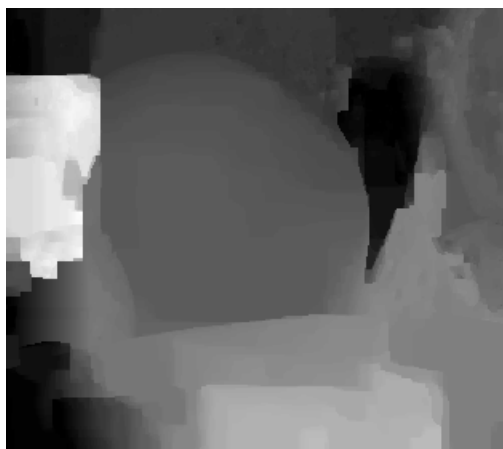


図 4 階層グラフカット



図 5 α 拡張

・スムーズコスト 5



図 6 階層グラフカット



図 7 α 拡張

表 1 計算時間

スムーズコスト	階層グラフカット	α 拡張
1	84 秒	138 秒
3	73 秒	189 秒
5	63 秒	254 秒

●考察

α 拡張よりも階層グラフカットの方が実行時間が短くなっていることが確認できた。しかし、階層グラフカットにおいて、出力視差画像の左側の一部がうまく出力できていないことも確認された。またスムーズコストを調整することで結果が変化するため、画像によって最適なコストを設定することが重要だと考える。