Brian Fulkerson > Software

GCMex - MATLAB wrapper for graph cuts multi-label energy minimization

This package provides source and precompiled Matlab mex files for the graph cuts based multi-label energy minimization techniques described in the thesis of Olga Veksler (GCoptimization Version 2.3). The wrapper can be used to solve energy minimizations in the form of:

```
E(l) = sum_p D(p,l_p)  + sum_{p,q} Vpq(l_p,l_q)
```

Download the compiled mex files (Windows, Mac, Linux) and source code here.

This wrapper is used in the following paper to implement a multi-label CRF on superpixels for class segmentation:

Class Segmentation and Object Localization with Superpixel Neighborhoods, B. Fulkerson, A. Vedaldi, and S. Soatto. In Proceedings of the International Conference on Computer Vision, 2009.

Please cite this paper (in addition to the citations listed on Olga Veksler's page) if you use this in a publication. Click here for the citations in .bib format.

Matlab Help

```
 GCMEX An efficient graph-cut based energy minimization

 [LABELS ENERGY ENERGYAFTER] =
   GCMEX(CLASS, UNARY, PAIRWISE, LABELCOST,EXPANSION)

   Runs a minimization starting with the labels for each node defined
   by CLASS, with unary potentials UNARY and the structure of the
   graph and pairwise potentials defined by PAIRWISE. LABELCOST
   determines data costs in terms of the labels of adjacent nodes.

 Parameters:
   CLASS:: An 1xN vector which specifies the initial labels of each
     of the N nodes in the graph
   UNARY:: A CxN matrix specifying the potentials (data term) for
     each of the C possible classes at each of the N nodes.
   PAIRWISE:: An NxN sparse matrix specifying the graph structure and
     cost for each link between nodes in the graph.
   LABELCOST:: A CxC matrix specifying the fixed label cost for the
     labels of each adjacent node in the graph.
   EXPANSION:: A 0-1 flag which determines if the swap or expansion
     method is used to solve the minimization. 0 == swap,
     1 == expansion. If ommitted, defaults to swap.

 Outputs:
   LABELS:: A 1xN vector of the final labels.
   ENERGY:: The energy of the initial labeling contained in CLASS
   ENERGYAFTER:: The energy of the final labels LABELS

 How do I know if I should use swap or expansion? From GC_README.txt:
   The expansion algorithm for energy minimization can be used
   whenever for any 3 labels a,b,c V(a,a) + V(b,c) <= V(a,c)+V(b,a).
   In other words, expansion algorithm can be used if the binary
   energy for the expansion algorithm step is regular, using V.
   Kolmogorov's terminology.
```

```
The swap algorithm for energy minimization can be used whenever
for any 2 labels a,b V(a,a) + V(b,b) <= V(a,b)+V(b,a). In other
words, swap algorithm can be used if the binary energy for the
swap algorithm step is regular, using V. Kolmogorov's terminology.
```

Changelog

- **Version 2.3.0** - Changed version number to match Veksler's. Switched to adjacency list max-flow implementation, fixing intermittent crash. Improved error reporting.
- **Version 1.0** - Initial release

Note: This software can be used only for research purposes. If you wish to use this software for commercial purposes, you should be aware that there is a US patent: R. Zabih, Y. Boykov, O. Veksler, "System and method for fast approximate energy minimization via graph cuts ", United Stated Patent 6,744,923, June 1, 2004