

/*

Copyright (C) 2009 Scott Grauer-Gray, Chandra Kambhammettu, and Kannappan Palaniappan

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

*/

GPU Implementation of Belief Propagation for stereo using CUDA

S. Grauer-Gray, C. Kambhammettu, and K. Palaniappan

This README describes an implementation of the CUDA belief propagation algorithm described in our paper "GPU implementation of belief propagation using CUDA for cloud tracking and reconstruction" which was published at the 2008 IAPR Workshop on Pattern Recognition in Remote Sensing (PRRS 2008) and can be found at the following address: <http://www.cis.udel.edu/~grauerg/cudaBeliefProp.pdf>. Please cite this work if using any of this code as part of a research paper or other work.

In addition, the code is distributed using the GNU General Public License, so any derivative work which is distributed must also contain this license.

Please email comments and bug reports to: sgrauerg@gmail.com (or sgrauerg@udel.edu)

Usage:

This code runs belief propagation for stereo vision on the GPU using CUDA on Linux-based systems (and the code can be modified to work on Windows/Macs). It is expected that the user has installed the CUDA SDK examples from the nVidia CUDA website <http://developer.nvidia.com/cuda-downloads> and compiled them before attempting to compile/run this program...it is needed for the cutil header/definitions used in this program.

The parameters are defined in the file `bpStereoCudaParameters.cuh` and are documented in that file and the paper. In order to compile/re-compile the program, navigate to the directory

with the files (this one...) and run "make clean" and "make". Note that the CUDA_DIR and CUDA_SDK_ROOT in the Makefile must be set to the location of the CUDA installation and location of the NVIDIA_GPU_COMPUTING_SDK, respectively.

The Tsukuba stereo set is included as part of this folder as a sample program. In order to test the implementation on this stereo set using the initial parameters as defined in bpStereoCudaParameters.cuh, perform the following steps:

1. Set the CUDA_DIR and CUDA_SDK_ROOT in the Makefile to the location of the CUDA installation and location of the NVIDIA_GPU_COMPUTING_SDK, respectively.
2. Set the PATH and LD_LIBRARY_PATH to the necessary paths needed to run CUDA programs (usually PATH is appended with \$CUDA_DIR/bin and LD_LIBRARY_PATH=\$CUDA_DIR/lib or LD_LIBRARY_PATH=\$CUDA_DIR/lib64).
3. Navigate to the folder with all the implementation files (including this README...)
4. Execute the commands "make clean" and "make" on the command line.
5. The executable named "impDriver" should now be created, so call ./driverCudaBp to run the implementation on the Tsukuba sets.
6. The output running time and accuracy should be given on the command line, with the output disparity map using the settings as defined in bpStereoCudaParameters.cuh stored in the current folder with the file name "computedDisparityMapTsukuba.pgm".
7. In order to adjust the parameters to run on different input or different parameters, adjust the desired parameters in bpStereoCudaParameters.cuh, run "make clean" and "make" to compile to the program with the adjusted parameters, then call ./driverCudaBp to run the program with the adjusted parameters.