ROB501 Project # 2

Keenan Burnett

November 11, 2017

1 Global Optimization with Dynamic Programming

I took inspiration for my dynamic programmig approach from Szeliski's book: *Computer Vision: Algorithms and Applications* and some lecture slides from Oxford Robotics:
`http://www.robots.ox.ac.uk/~az/lectures/opt/lect2.pdf`

1. First we want to compute the aggregrated disparity space image. I used Sum of Squared Difference, and a simple box filter for aggregation.
$C(x, y, d) = \omega(x, y) * C_0(x, y, d)$

2. Next, we want to use dynamic programming to find the globally optimal set of disparities for a single row. They key idea is that the optimization can be broken down into n suboptimizations.

Let x_k and x_{k-1} be functions which take on values of disparity. They correspond to adjacent column indices along a row we are trying to optimize. Using a slice of the disparity space image at row y, we can always compute the cost $C_{data}(x_k)$ of x_k taking on a particular disparity value. I then used $\phi(x_k, x_{k-1}) = (x_k - x_{k-1})^2$ as the smoothness penalty. $S_k(x_k)$ represents the minimum cost for each value of x_k. The DP algorithm then works as follows:

Initialize:
$S_1(x_1) = C_{data}(x_1)$

For k = 2:n
$S_k(x_k) = C_{data}(x_k) + min_{x_{k-1}}\{S_{k-1}(x_{k-1}) + \lambda\phi(x_k, x_{k-1})\}$
$b_k(x_k) = argmin_{x_{k-1}}\{S_{k-1}(x_{k-1}) + \lambda\phi(x_k, x_{k-1})\}$

Terminate:
$x_n^* = argmin_{x_n} S_n(x_n)$

Backtrack:
$x_{i-1} = b_i(x_i)$

I found that box filter of size of 3 combined with a regularization parameter λ of 225 worked best for the test images.