Homework 3.B: Stereo matching with Graph Cut
Ye Ji and Davi Geiger

Due November 19th, 2014

Introduction
The Maximum Flow algorithm will be used in the following direct graph, figure 1.

Let us refer to the set \( M \) as all the pixels matches in the image, i.e., \( M = \{ (e,l,r); e,l,r = 1, ..., N \} \). The graph \( G(V,E) \) shown in 1 has a set of vertices \( V \) doubling the number of pixels (plus (s,t)), i.e., \( V = \{ u_{i,r}^l; (e,l,r) \in M \} \cup \{ v_{i,r}^l; (e,l,r) \in M \} \cup \{ s,t \} \) and three set of edges \( E = E_M \cup E_C \cup E_P \).

1. \( E_M = \{ (u_{i,r}^l, v_{i,r}^l) | (e,l,r) \in M \} \) stores the matching of pixel \((e,l)\) with \((e,r)\). If this edge is cut, we interpret it as a match.

2. \( E_C = \{ (u_{i,r}^l, u_{i+1,r}^l) \cup (u_{i,r}^l, u_{i,r-1}^l) \cup (v_{i,r}^l, v_{i+1,r}^l) \cup (v_{i,r}^l, v_{i,r-1}^l) | (e,l,r) \in M \} \) are edges set to \( \infty \) so that they are never cut. Thus, the solution is always weakly monotonic (never decreases in \( l \) nor \( r \)).

3. \( E_P = \{ (s, u_{i,l+d_{true}^l+\Delta d_{min}^l}) \cup (v_{i,r}^l, u_{i-1,r}^l) \cup (u_{i-1,r}^l, v_{i,r}^l) \cup (u_{i,r}^l, u_{i+1}^l) \cup (u_{i+1}^l, v_{i}^l) \cup (v_{i+r}^l, u_{i+d_{true}^l+\Delta d_{max}^l}, t) | (e,l,r) \in M \} \) control the smoothness, discontinuities, occlusions. First, let us set all of them to the same value, a constant \( \mu \). Later we can elaborate further on this.

Question 1: Creating a Disparity Range

Our first step is to identify a range of disparities for each pixel. We do this as follows

1. Disparity list and cost: Loop over every 8 pixels \((2^3)\) in the left image. Each template is \(32 \times 32\) pixels, so there is overlap in the covered left image. The centers of the template are \((x^C_{t}, y^C_{t})\) in the left image, where \( x^C_{t}, y^C_{t} \in \{16, 24, 32, 40, 48, ...\} \). Compute the Scattered Network. Do the same on the right image, i.e., compute the Scattered Network for templates \(32 \times 32\) pixels displaced by \(8 = 2^3\) pixels.
Figure 1:

Compute the cost of the matches between left and right templates. Since we do know the disparities for this image vary between 40 pixels and 210 pixels, we may just consider disparities between $d_{\text{min}} = 32 = 4 \times 2^3$ and $d_{\text{max}} = 224 = 28 \times 2^3$, in
intervals of $2^3$, i.e., $d \in \{d_{\text{min}}, d_{\text{min}} + 8, d_{\text{min}} + 16, \ldots, d_{\text{max}} - 8, d_{\text{max}}\}$. Clearly, there is a total of 25 different disparity values. Make sure to only search for valid pixels, not beyond the image boundary. Note that the scattered network is to be computed on the left image and right image before matching takes place.

2. **First Stereo Algorithm**  Show the result of assigning for each pixel the disparity that has the best cost. Since the assignment is for each $2^3$ locations, one can simply display the result at a grid that is subsampled by a factor $8 = 2^3$.

**Question 2: Max Flow**

Build the graph and run the max flow algorithm only considering the disparity range obtained in Question 1. Note that the disparity at each left center template is the one given by the label cut.

Note that as we build the graph, the only nodes and edges needed are the nodes and set $E_M$ accounting for the disparity range $(d_{\text{min}}, d_{\text{max}})$. Try different values for the parameter $\mu$, where the value should be about the same as the threshold used in part 1.