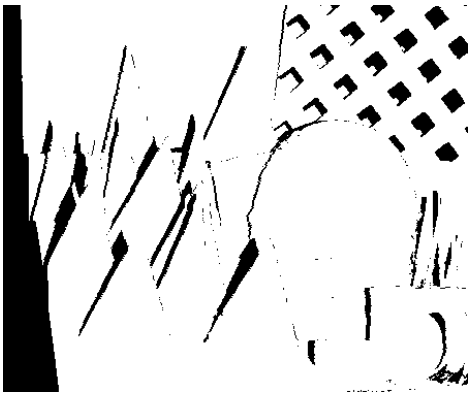
Code documentation

In this code several methods for estimating depth in a stereo image (image 1) are implemented. For this purpose a *matching cost volume* is calculated by means of sum of squared differences (SSD), sum of absolute differences (SAD) and normalised cross-correlation (NCC) and then the most appropriate match chosen either by the simple *winner-takes-it-all* approach (WTA) or *semi-global matching* (SGM). For this purpose the given images have to be converted to grayscale.

a) left

b) right

c) ground-truth



d) mask

Figure 1: Stereo images: left (a) and right (b), the corresponding ground-truth (c) and the mask (d) needed for the accX evaluation

1 Local matching

The following error-measures and correlations will be used for evaluating a corresponding matching cost between two image patches p and q of equal size $W \times H$.

1.1 Sum of absolute differences

In case of the sum of absolute differences the matching of two patches p and q is penalised depending on the sum of absolute differences of the two windows according to

$$SAD(p,q) = \sum_{x=1}^{W} \sum_{y=1}^{H} |p(x,y) - q(x,y)| \tag{1}$$

This means very similar image patches lead to a low SAD while non-matching patches result in a high SAD.

1.2 Sum of squared differences

In case of the sum of squared differences the matching process is penalised quadratically instead of linearly making use of the squared difference instead

$$SSD(p,q) = \sum_{x=1}^{W}\sum_{y=1}^{H}(p(x,y) - q(x,y))^2 \tag{2}$$

1.3 Normalised cross-correlation

In the case of the more sophisticated normalised cross-correlation the patches are normalised by substracting the means to account for slight deviations in lighting between the two pictures

$$\overline{p} = \frac{1}{H\,W}\sum_{x=1}^{W}\sum_{y=1}^{H}p(x,y) \qquad\qquad \overline{q} = \frac{1}{H\,W}\sum_{x=1}^{W}\sum_{y=1}^{H}q(x,y) \tag{3}$$

and calculating the a correlation measure for local matching according to

$$NCC(p,q) = \frac{\sum\limits_{x=1}^{W}\sum\limits_{y=1}^{H}(p(x,y)-\overline{p})(q(x,y)-\overline{q})}{\sqrt{\left[\sum\limits_{x=1}^{W}\sum\limits_{y=1}^{H}(p(x,y)-\overline{p})^2\right]\cdot\left[\sum\limits_{x=1}^{W}\sum\limits_{y=1}^{H}(q(x,y)-\overline{q})^2\right]}} \tag{4}$$

where in this case a high similarity between the two patches contrary to SAD and SSD is characterised by a high NCC. This means for our cost volume we either have to reverse the sign multiplying the NCC by -1.

2 Cost volume

We use these similarity measures to compute a cost-volume CV for a pre-defined range of disparities D

$$CV(x,y,d) = S(I_0(x,y)\,I_1(x-d,y)) \tag{5}$$

where the parameter $d \in \mathcal{D}$ and $\mathcal{D} = \{0,\dots,D-1\}$ are all valid disparities and S is any of the aforementioned error-measures.

This basically means that we take the left picture and translate the right picture trying to overlap the objects in the two pictures taken from different views. The points at a certain depth have a certain disparity and thus the optimal shift can be used to determine the correct depth. In order to account for a certain deviation we use a certain search window (W,H) rather than trying to match the points directly.

3 Matching algorithm

3.1 Winner-takes-it-all solution

One fast way of obtaining then the best disparity for each image point would be taking the point with the highest value in the cost volume along the disparity axis according to

$$\overline{d}(x,y) \in \arg\min_{d} CV(x,y,d) \tag{6}$$

This though leads to noisy results as this approach doesn't penalise label changes at all.

3.2 Semi-global matching

In semi-global matching a different approach is taken, rather than looking for the best fit on a scanline, a sort of global optimisation is used. Each pixel with a corresponding unary cost given by the cost volume is assigned an additional pairwise cost that depends on wherever the neighbouring pixels have a similar depth value or deviate significantly. This energy can be written as

$$\min_{z} \left[\sum_{i \in \mathcal{V}} g_i(z_i) + \sum_{ij \in \mathcal{E}} f_{i,j}(z_i, z_j) \right] \tag{7}$$

where $\mathcal{V}$ are the image pixels, $\mathcal{E}$ the edges, the connections between two pixels. The g_i are given by the cost volume and the pairwise cost $f_{i,j}$ defines a penalty for jumps between neighbouring pixels.

$$f_{i,j}(z_i, z_j) = \begin{cases} 0, & \text{if } z_i = z_j \\ L_1, & \text{if } |z_i - z_j| = 1 \\ L_2 & \text{else} \end{cases} \tag{8}$$

This is done as following: First messages for all four disparity directions are calculated where the first message in each direction is initialised with $\vec{0}$.

$$m^a_{i+1}(t) = \min_{s \in \mathcal{D}} \left[m^a_i(s) + f_{i,i+1}(s,t) + g_i(s) \right] \tag{9}$$

This can be done for every direction by a combination of mirroring and transposing the cost volume. Then the beliefs are computed

$$b_i(s) = g_i(s) \sum_{a \in \{L,R,U,D\}} m^a_i(s) \tag{10}$$

The correct disparity is then calculated from the believes as follows

$$\hat{d}(x,y) \in \arg\min_{d} b(x,y,d) \tag{11}$$

The last formula contains is intentionally given as $\in$ as the solutions might not be unique.

4 Evaluation: compare to ground-truth

The performance of the stereo workflow is evaluated by comparing it with a ground-truth disparity map, in this case with the *accX* measure

$$accX(z, z*) = \frac{1}{Z} \sum_x \sum_y m(x,y) \cdot \begin{cases} 1, & \text{if } |(z(x,y) - z*(x,y)| \leq X \\ 0 & \text{else} \end{cases} \tag{12}$$

This measure characterises errors less than or equal to X disparities, between the prediction z and the ground truth disparity map $z*$ with a mask m that contains 1 for the Z valid pixels and 0 for the invalid pixels.

The mask basically excludes pixels that should not be evaluated e.g. because they are occluded in either of the two pictures. The average of the remaining pixels that were estimated correctly is determined. All pixels that guessed the depth correctly (threshold X) are set to 1, all pixels that did not estimate it correctly do not contribute. In this way the *accX* measures the amount of pixels that were matched correctly to those that could possibly be matched. An *accX* of 1 would correspond to the ground truth.