VISION - TME 3
Disparity map computation by propagation of seeds

Marie Diez

1 Introduction

The objective of this work is to calculate the disparity map associated with a pair of images. To do so, we will start from high confidence points (seeds) maximizing the correlation between a patch of image 1 and a patch of image 2 depending on the disparity. Then extend these points to points without seeds, assuming that the map is regular.

2 Disparity map

The disparity map represents the apparent movement between 2 images (previously transformed with epipolar rectifications). To calculate it, we can look for the disparity that gives the best correlation between the 2 patches P = (i,j) and Q = (i+d,j). The disparity is applied on the X axis because the epipolar rectification has allowed to place the images on the same y plane.

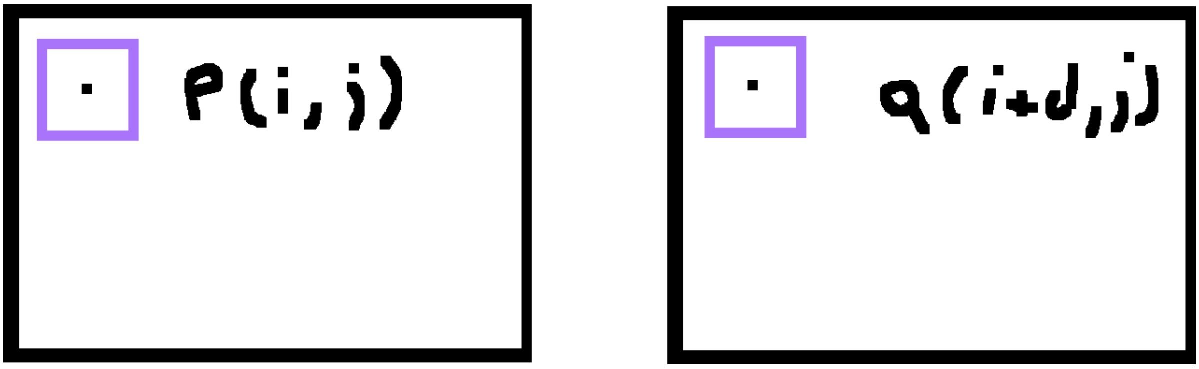

Figure 1: Patches P and Q

3 Dense disparity

Dense disparity map corresponds to the disparity map by not setting a minimum correlation threshold between 2 patches.
Here are the results obtained:

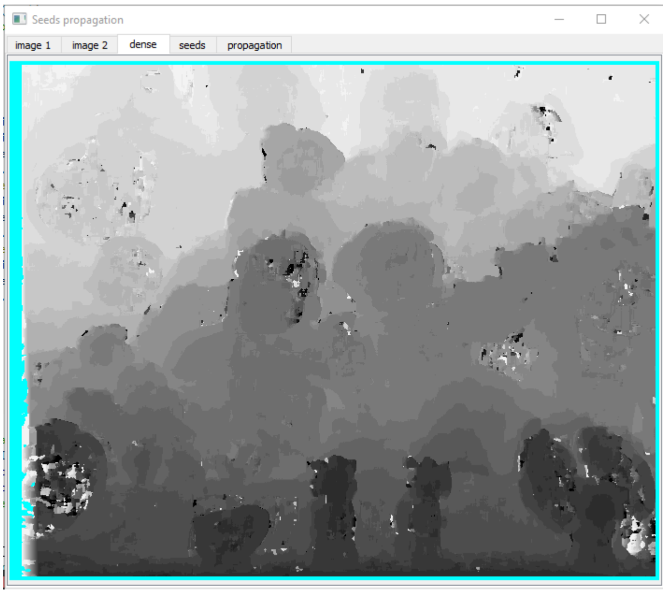

Figure 2: Dense map

4 Seeds

When a minimum correlation threshold is set (0.95 here) some points that do not meet this condition will be considered as doubtful points (for example points where the patch does not have enough texture to hang anywhere) and will therefore not be assigned a seed.
Here is the result:

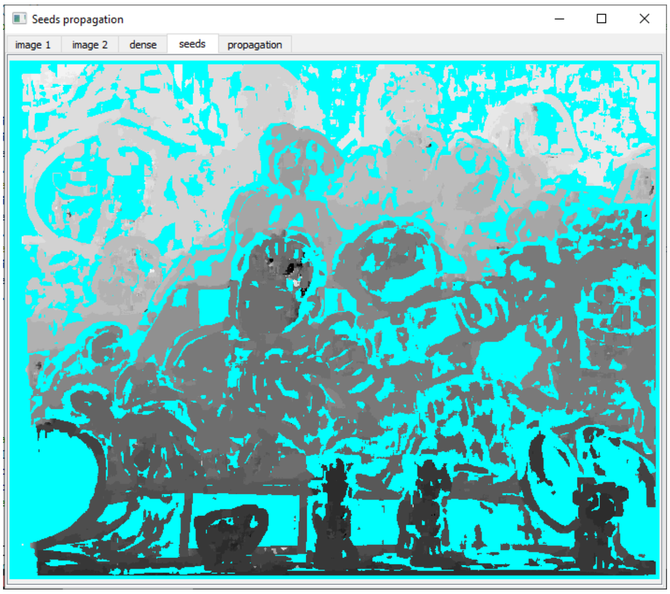

Figure 3: Seeds(x,y)=False in blue

5 Propagation of seeds

Results without a seed will be proposed the disparity d-1, d or d+1 of one of the 4 neighbours. Here are the results obtained:

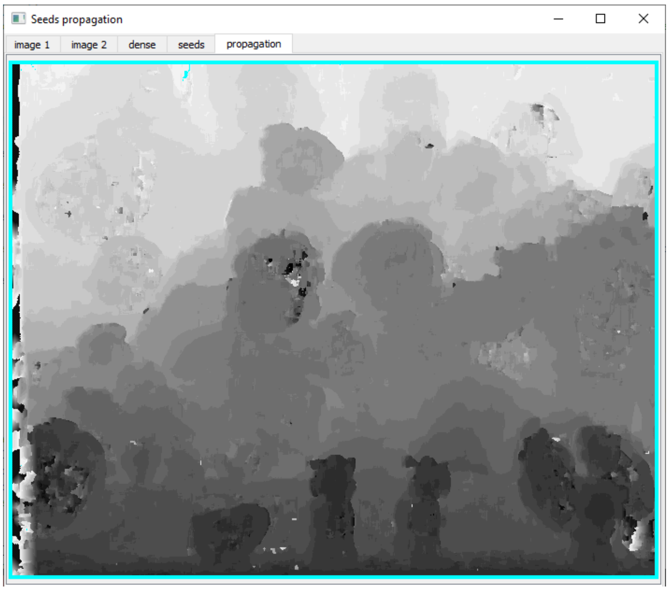

Figure 4: Propagations of seeds from neighboors

Remark : If the pixel without seed takes the value of a neighbor whose patch is not in the image, we may have problems by giving a correlation of 0, because in the calculation of the correalation (correl function) I check that the patches are well in the image. This can be seen above with 0 values on the left edge. This will also be seen in the 3D visualization. I could have not propagated values in this case, a solution adapted to the needs should be put in place.

```
float dp_m1 = ccorrel(im1, x, y, im2, x + disp(s.x, s.y) - 1, y);
float dp = ccorrel(im1, x, y, im2, x + disp(s.x, s.y), y);
float dp_p1 = ccorrel(im1, x, y, im2, x + disp(s.x, s.y) + 1, y);
```

6 3D

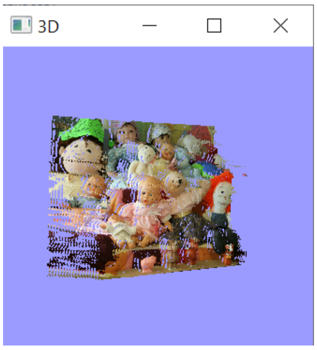

Figure 5: 3D map