Bauhaus-Universität Weimar

Photogrammetric Computer Vision
Assignment 6

Winter Semester 24/25

Submission Deadline: 26.01.25, 8 pm

VII. Stereo Image Matching

Each automated triangulation-based measuring system must contain an efficient correspondence analysis method. Therefore, in this exercise you should collect some practice with matching of homologous image points.

1. **Area-based image matching using normalized cross-correlation:**

   For the exercise a pair of normal images is taken from the Middlebury stereo vision research page (`left.png` and `right.png`). The images are provided in the Moodle section for this assignment.

   a) Read the images and convert the gray value intensities to float values (`double`). Implement a procedure in MATLAB for the *normalized cross-correlation* (`mean, sqrt, mean2`) without using the build-in functions (i.e. `std, var, cov, std2, corr2, corrcoef, xcov, xcorr`).

   - For each pixel in the left image define a reference window `img(i-r : i+r, j-r : j+r)` and search horizontally in the right image for a window position with maximum correlation. You may have to cope with the image borders (`min, max`).
   - Produce a *disparity map* for the left image by registering the horizontal coordinate difference between the reference windows and most similar search windows.

   b) Visualize the disparity map as *gray value image* (`imshow(…, [])`).

   c) Find the optimal parameters for the *window size* and for the *search range*.

   Hints: Loops are very slow in MATLAB. Pre-calculation of the mean values will accelerate the matching. In octave `mean2(A)` can be expressed as `mean(A(:))`.Optionally you might want to improve your final *disparity map* with median filtering (`medfilt2`).

Computer Vision in Engineering – Prof. Dr. Rodehorst
M.Sc. Mariya Kaisheva
mariya.kaisheva@uni-weimar.de

Basic statistics in **MATLAB**:

```
a = [3 1 6 3 4];
b = [1 5 3 4 3];

s1 = sqrt(mean((a-mean(a)).^2))                % Standard deviation of a
s2 = sqrt(mean(a.*a)-mean(a)^2)              % Standard deviation faster
s3 = std(a,1)                                % MATLAB Standard deviation

c1 = mean((a-mean(a)).*(b-mean(b)))              % Covariance of a and b
c2 = mean(a.*b) - mean(a)*mean(b)                    % Covariance faster
c3 = cov(a,b,1)                       % Covariance matrix [aa, ab; ba, bb]


v1 = std(a,1)^2                                            % Variance of a
v2 = cov(a,a,1)                              % Alternative computation

n1 = (mean(a.*b) - mean(a)*mean(b)) / sqrt(std(a,1)^2 * std(b,1)^2)
n2 = corrcoef(a,b)                    % Correlation matrix [aa, ab; ba, bb]


>>
s1 =    1.6248
s2 =    1.6248
s3 =    1.6248

c1 =   -0.8800
c2 =   -0.8800
c3 =    2.6400   -0.8800
       -0.8800    1.7600

v1 =    2.6400
v2 =    2.6400    2.6400
        2.6400    2.6400

n1 =   -0.4082
n2 =    1.0000   -0.4082
       -0.4082    1.0000
```