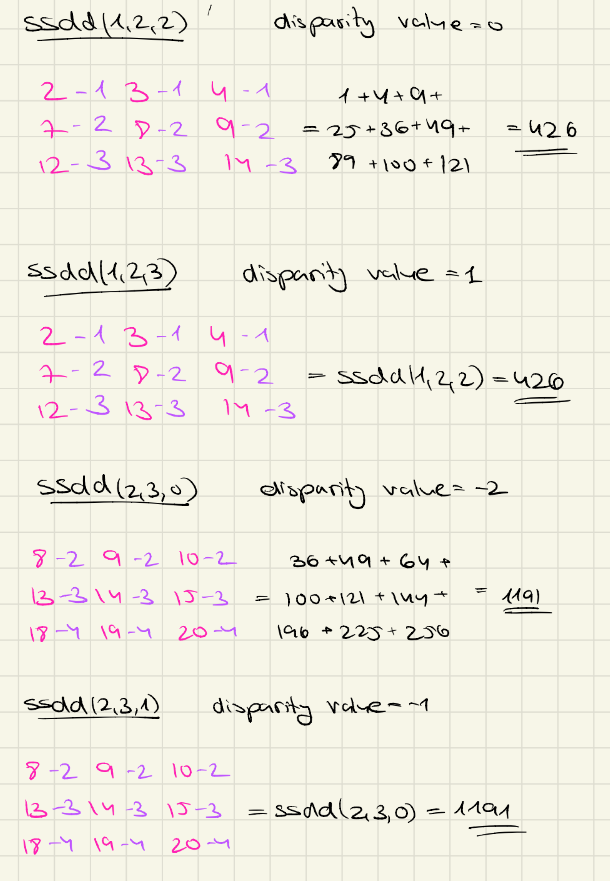
# Computer Vision – EX2

Lior Soffer 203135058

Idan Daniel 308088624

# Part A: Distance Tensor Computation

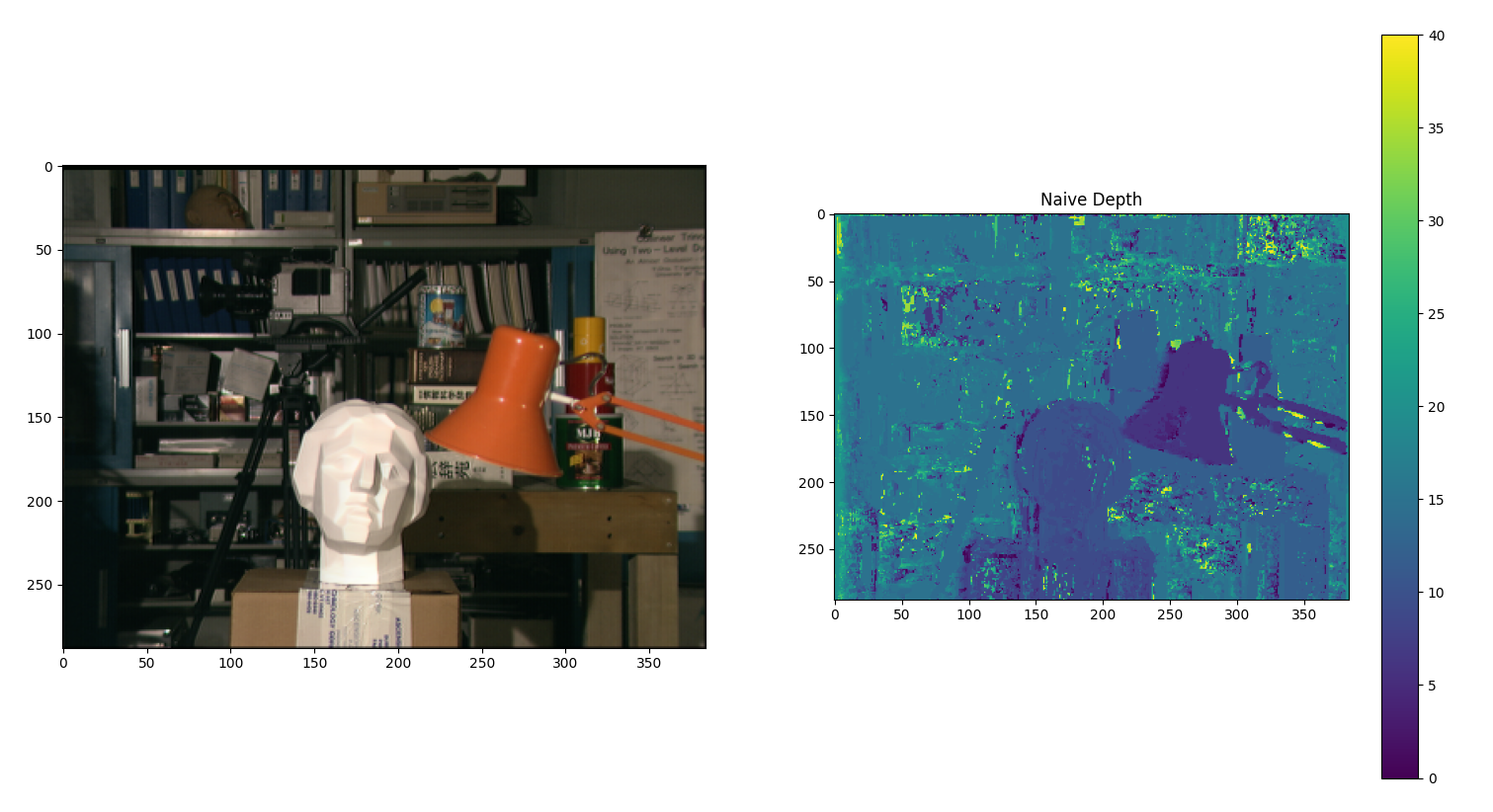
1. Compute ssdd



1. Coding

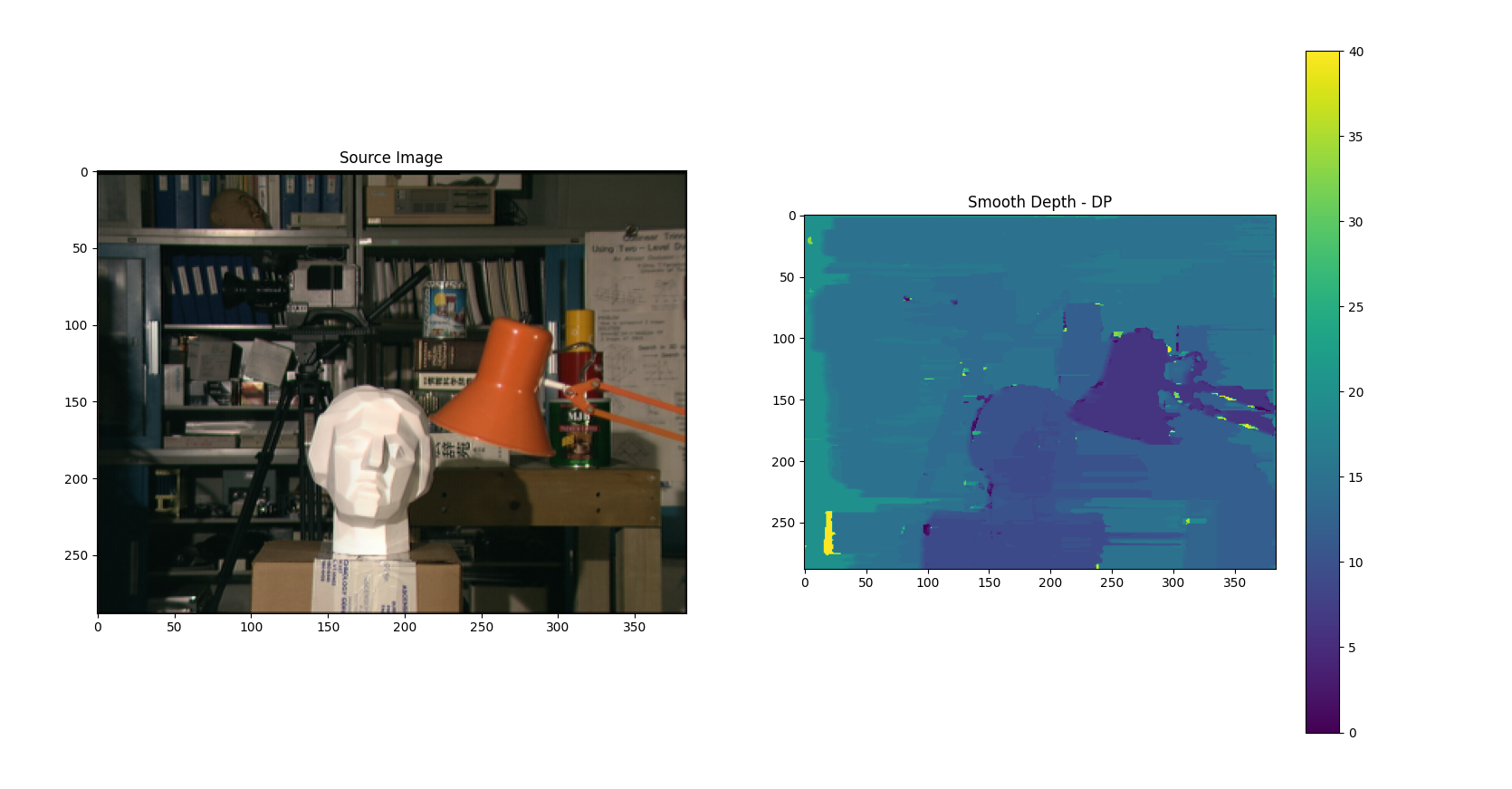
# Part B: Naive Depth Map

1. **SSDD** – In file.  
   *SDDD calculation done in 3.1696 [seconds]*
2. **Naive labeling** - The results present a noisy depth map, this is due to the naïve approach to calculating the depth map in which for each pixel, the disparity value is the minimum that it can obtain, without considering its surrounding pixels.  
   *Naive Labeling done in 0.0104 [seconds]*



# Part C: Depth Map Smoothing using Dynamic Programming

1. **Dynamic programming** – In file
2. **Depth map using dynamic programing**



Graphical user interface

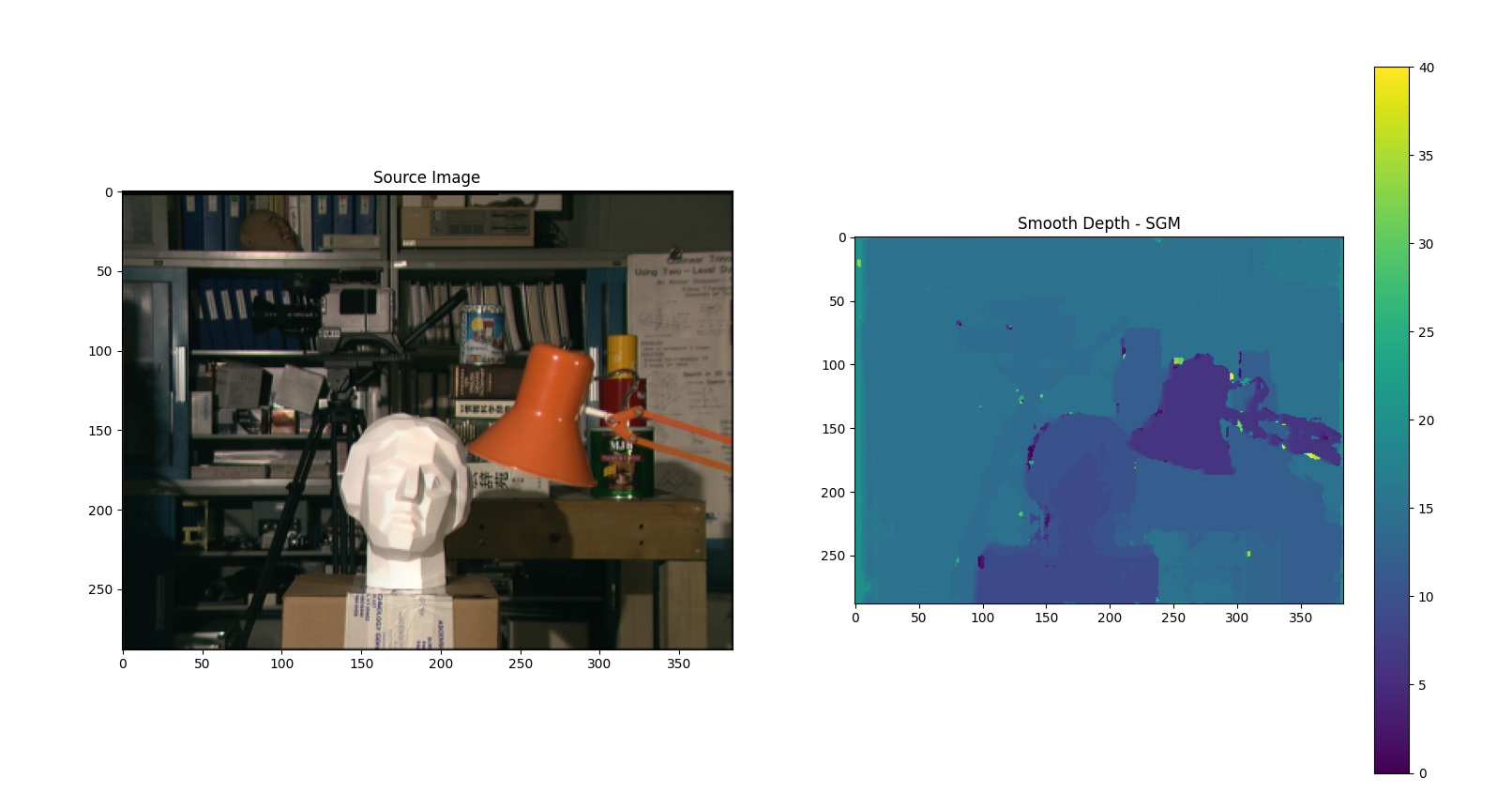
Description automatically generated

*Dynamic programming done in 114.2130 [seconds]*

1. The dynamic programming performs the depth map while looking examining the nearby pixels in the same raw. This creates a smoother, more continuous map, which is more like the reality of no quick changes. This method still suffers from streaking artifacts between rows.

# Part D: Depth Image Smoothing using Semi-Global Mapping

1. Coding
2. Coding
3. Coding
4. **SGM labeling** - *done in 751.4644 [seconds]*



Graphical user interface

Description automatically generated with low confidence

In comparison to the previous methods, the SGM method results in the smoothest depth map, which has the highest similarity to reality. The change for this method is the fact that for each pixel we obtain the disparity value from 8 different pixels in its surrounding. The SGM depth map presents the edges in the image, which were harder to detect in the previous methods.

1. Coding – *done in 1001.3185 [seconds]*

Graphical user interface, application

Description automatically generated

(\*) Please note that creating the map for each direction is done similary for both sgm\_labeling and dp\_labeling\_per\_direction. The code is duplicated since we didn’t know if we could create a helper function.

# Part E: Your own images

The following pair of images were extracted from the link provided in the assignment file. The images were resized (decrease in factor 4) to improve the running time.  

We have changed the disparity range to be 60

Show results