Block Matching

Block matching is a technique used in stereo vision to find the disparity between two images. The disparity is the difference in the horizontal position of a point in the two images. The disparity is used to construct a 3D model of the world.

Algorithm

The algorithm for block matching is as follows:

```
def block_matching_imp1(imgL, imgR, window_size, method, numDisparities = 32):
    # imgL and imgR are grayscale images
    assert imgL.shape == imgR.shape
    assert imgL.ndim == 2
    assert imgR.ndim == 2
    # window_size is the size of the window
    assert window_size % 2 == 1
    # numDisparities must be less then the width of the image
    assert numDisparities < imgL.shape[1]
    # method is the method used to compute the distance between two pixels
    # method can be 'SAD' or 'SSD'
    assert method in ['SAD', 'SSD']

    win_size_over_2 = window_size//2
    imgLP = np.pad(imgL, win_size_over_2, mode='constant')
    imgRP = np.pad(imgR, win_size_over_2, mode='constant')
    disparity = np.zeros(imgL.shape, np.uint8)

    for r in tqdm(range(win_size_over_2, imgLP.shape[0] - win_size_over_2)):
        for p in range(win_size_over_2, imgLP.shape[1] - win_size_over_2):
            winL = imgLP[r - win_size_over_2 : r + win_size_over_2 + 1 , p -
win_size_over_2 : p + win_size_over_2 + 1].astype('int32')
            min_index = win_size_over_2
            min_cost = np.Inf

            for i in range(p - numDisparities if p - numDisparities >
win_size_over_2 else win_size_over_2, p + 1):
                winR = imgRP[r - win_size_over_2 : r + win_size_over_2 + 1 , i -
```

```
win_size_over_2 : i + win_size_over_2 + 1].astype('int32')
                cost = 0
                if method == 'SAD':
                    cost = int(np.sum(np.absolute(winL - winR)))
                elif method == 'SSD':
                    cost = int(np.sum(np.square(winL - winR)))

                if cost <= min_cost:
                    min_cost = cost
                    min_index = i

            disparity[r - win_size_over_2, p - win_size_over_2] =
np.absolute(int(p) - int(min_index))

    return disparity
```

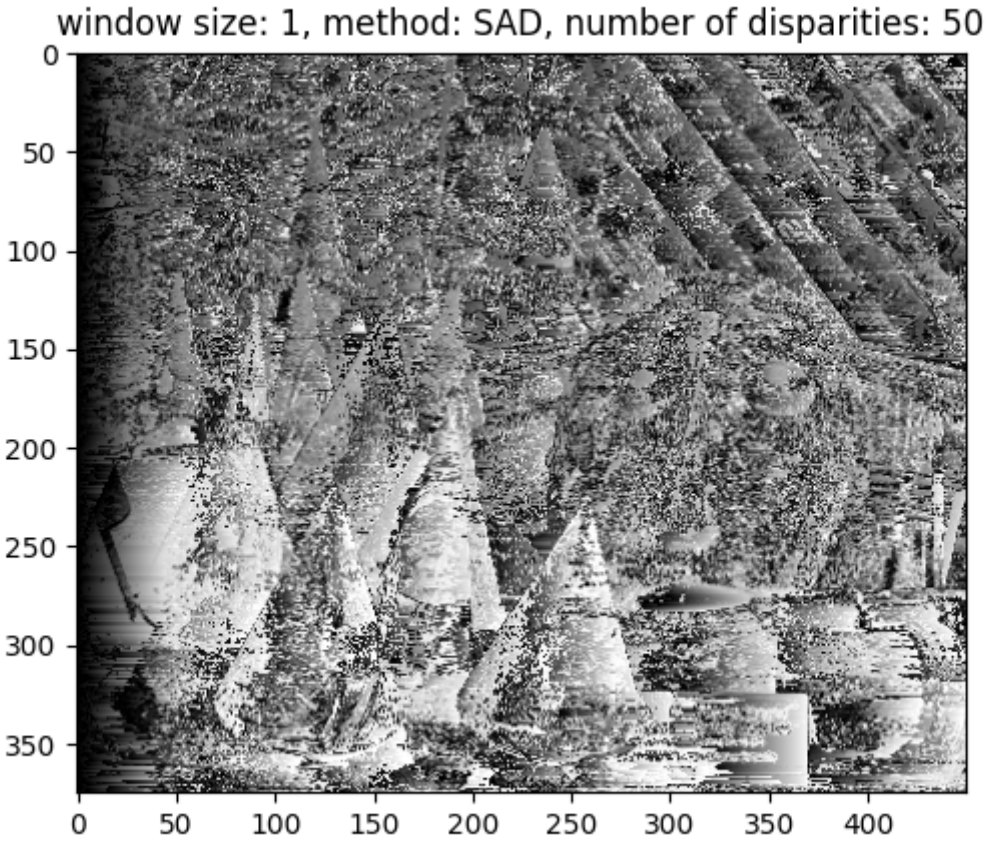
Parameters

The parameters for block matching are as follows:

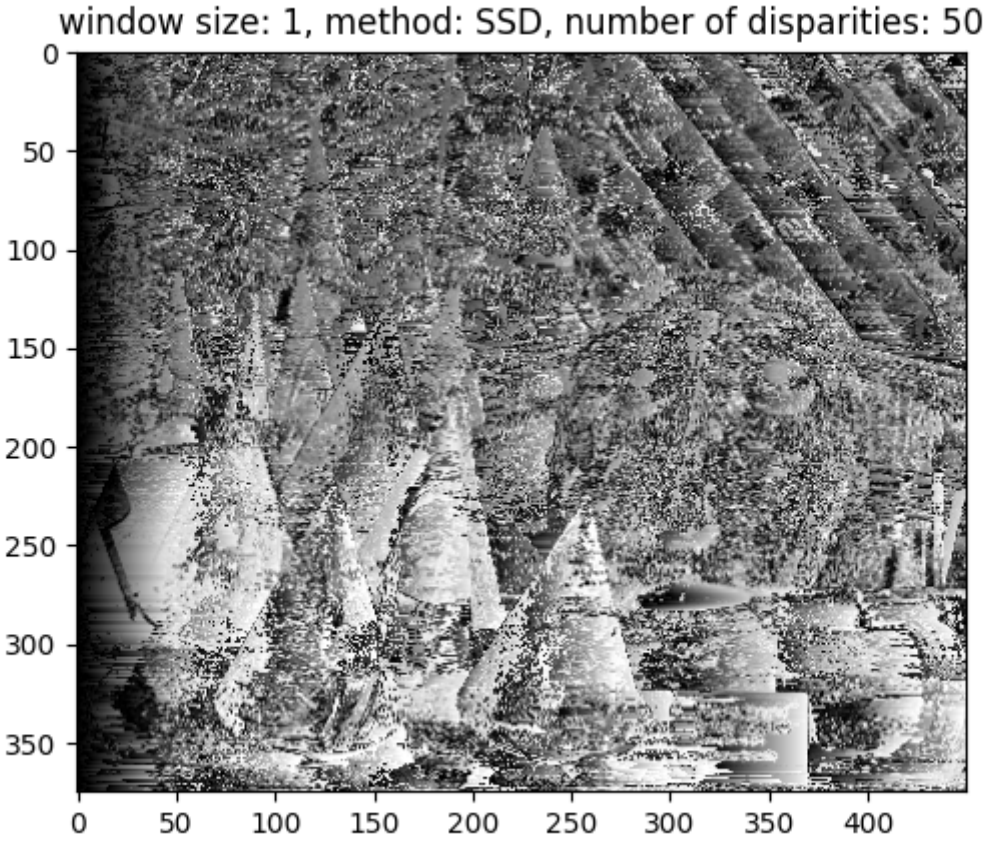
1. window size

   Window size parameter tuning is usually a trade-off between smoothness and detail. A larger window size will give smoother results, but will lose detail. A smaller window size will give more detail, but will be more noisy.

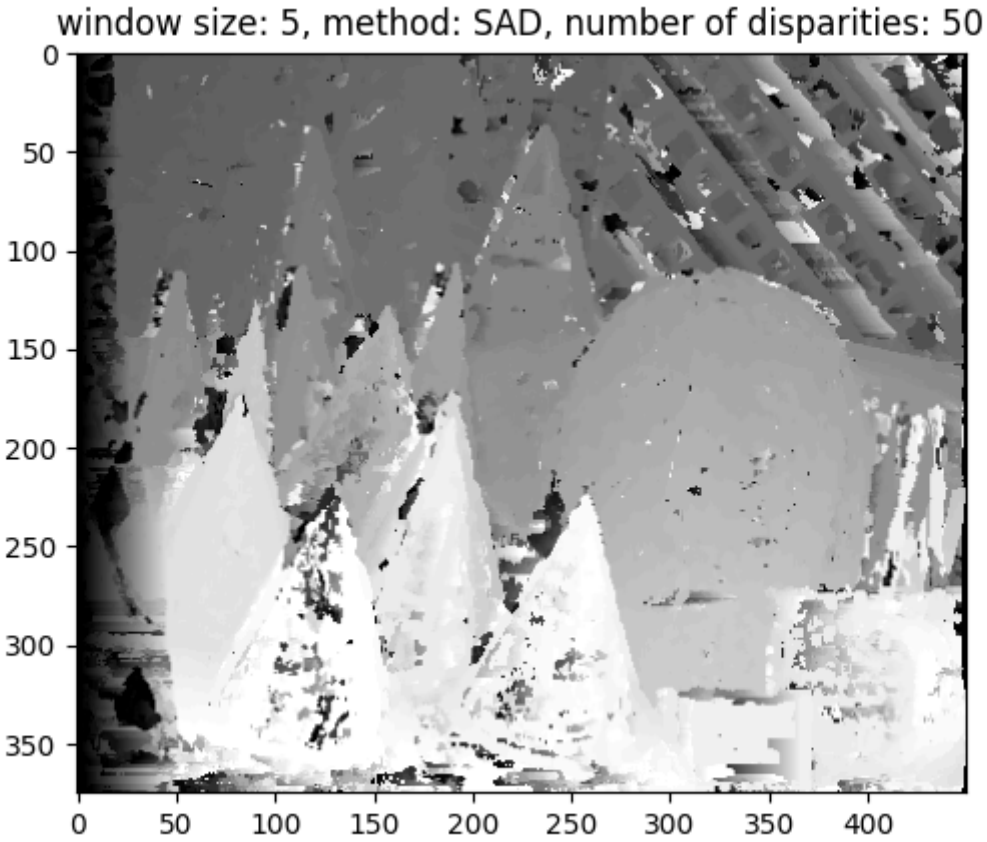
   The following results show the effect of the window size.



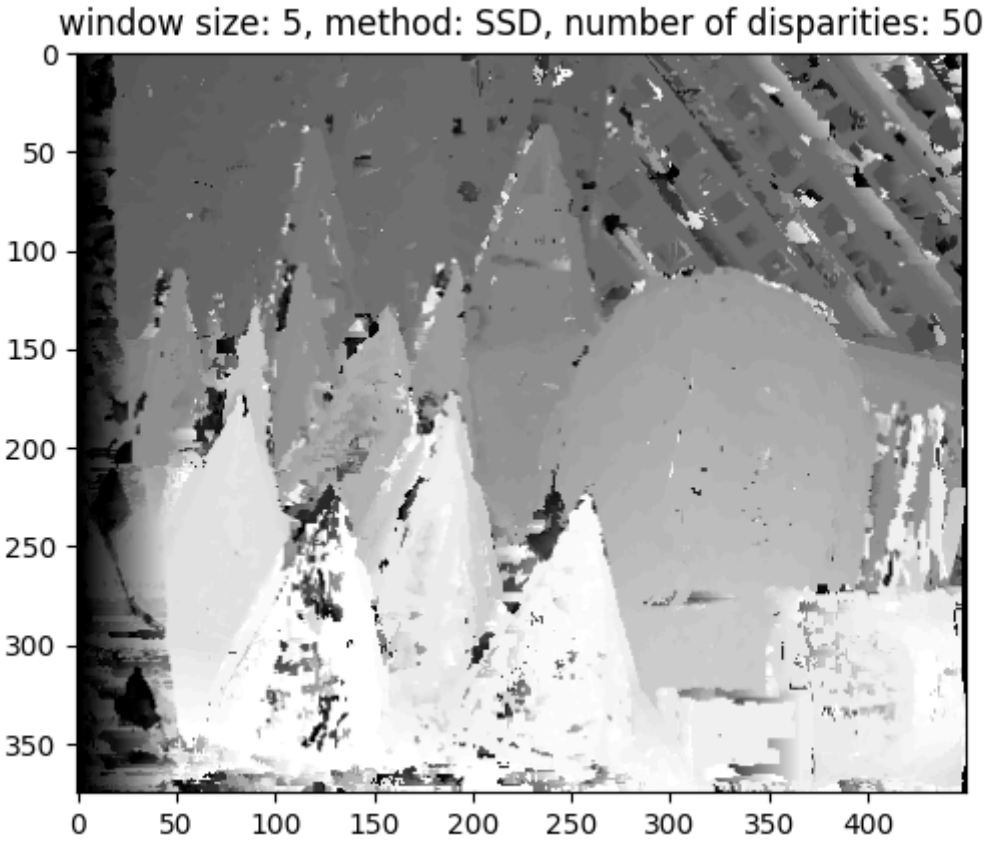
SAD, window size = 1



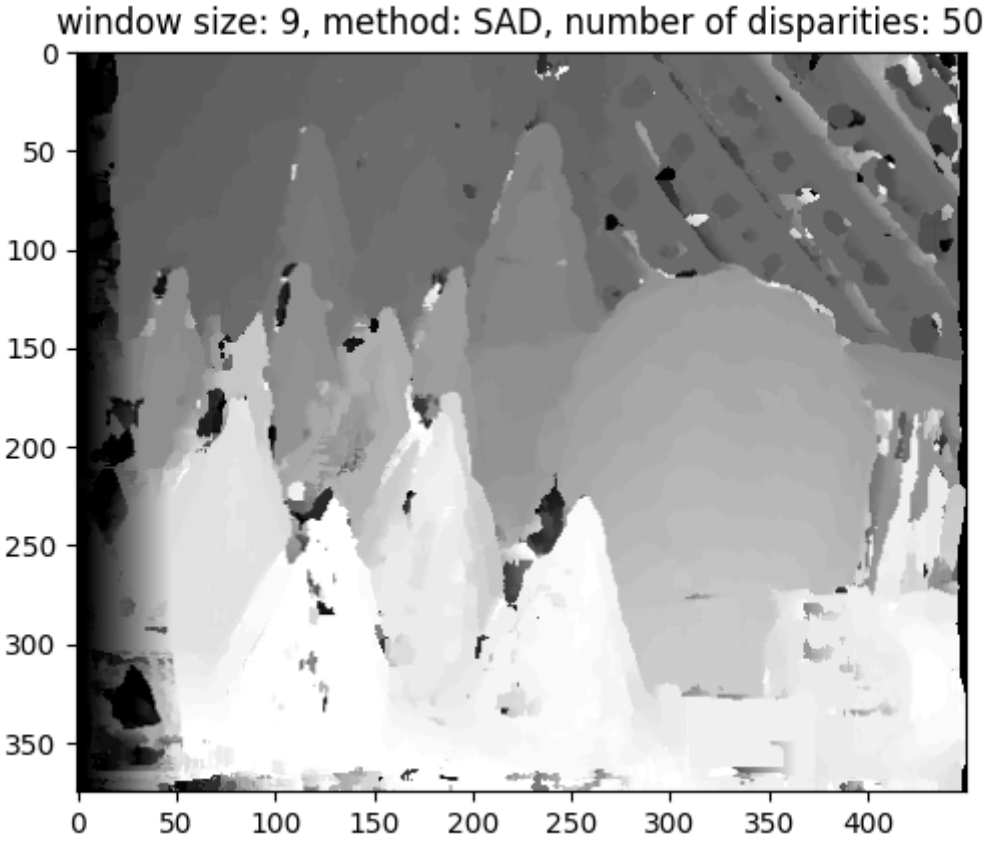
SSD, window size = 1



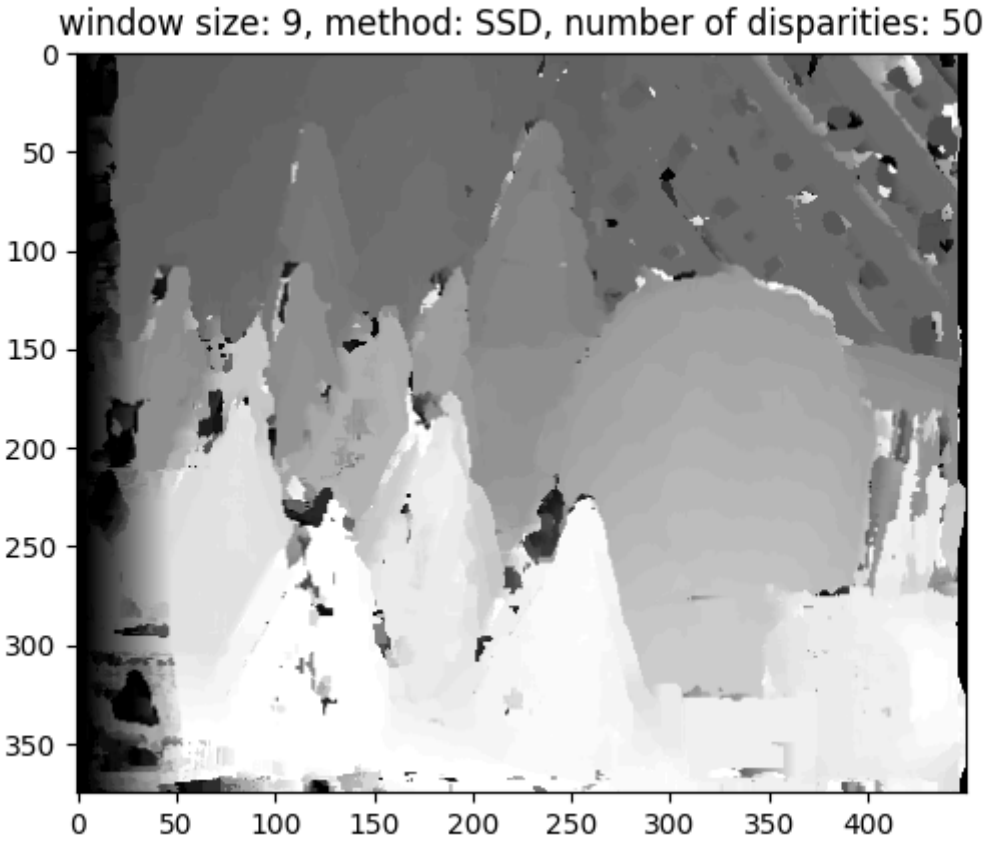
SAD, window size = 5



SSD, window size = 5



SAD, window size = 9



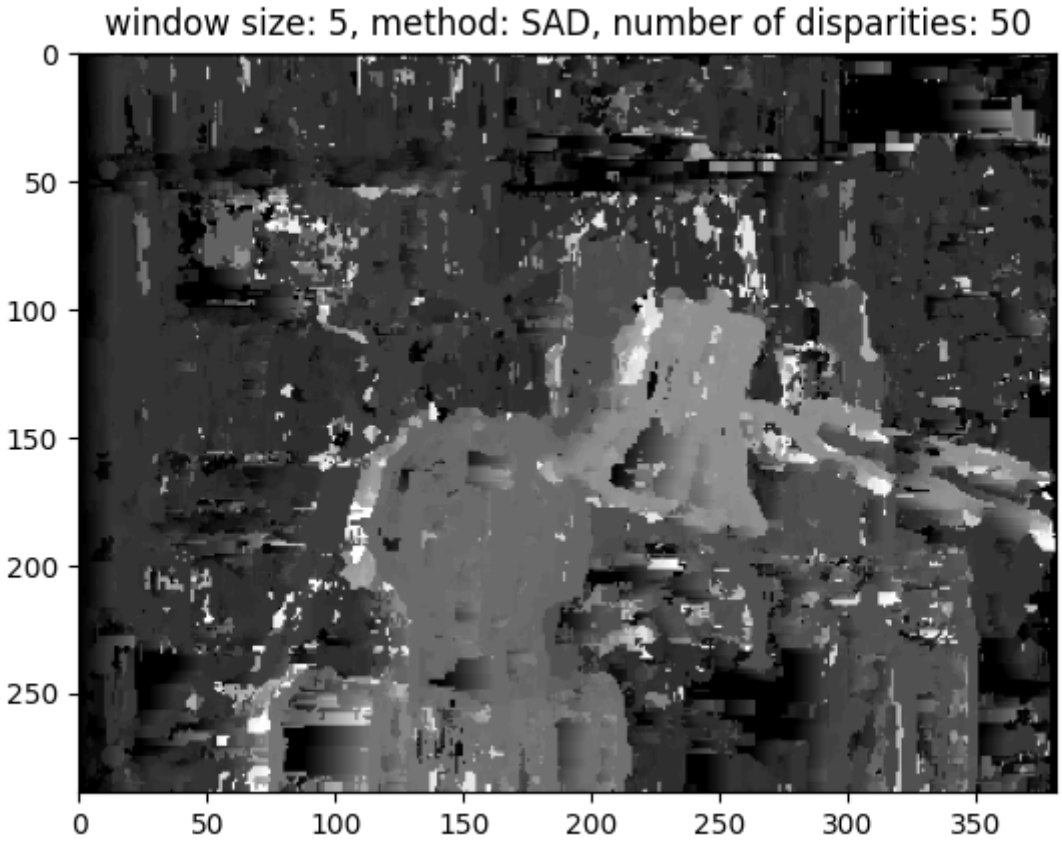
SSD, window size = 9

2. search range (`numDisparities`)

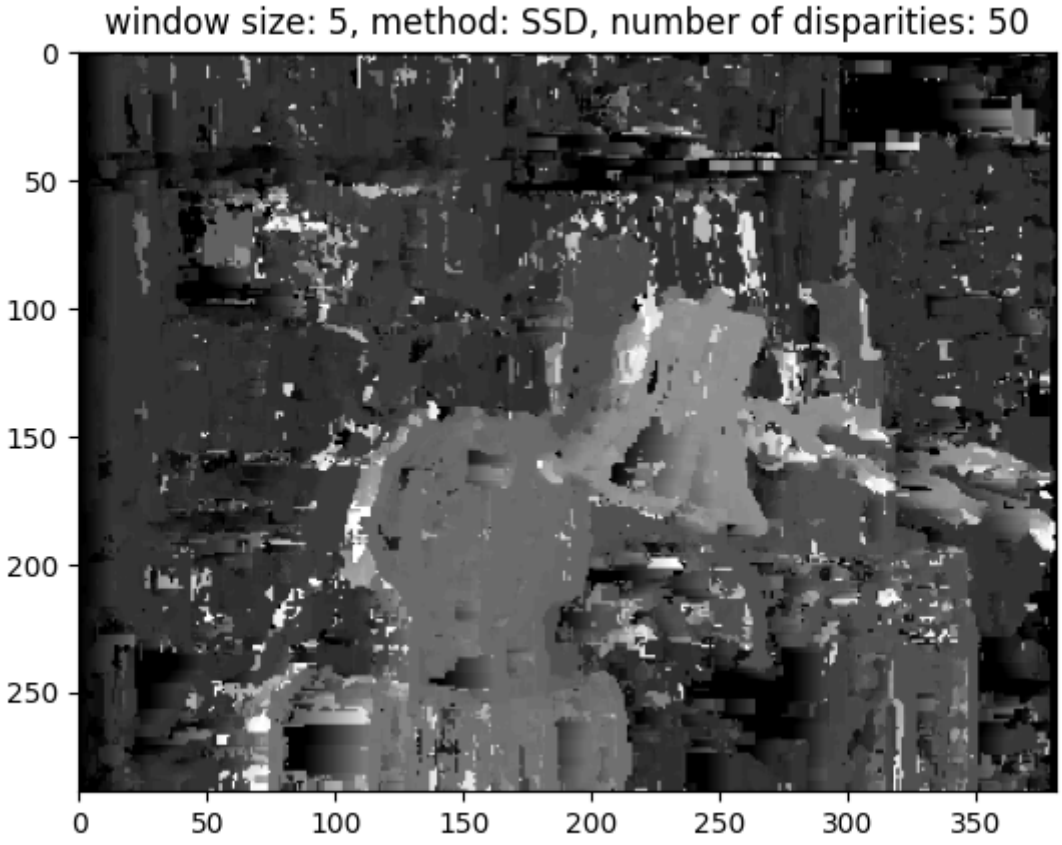
   For each pixel, the search range is the number of pixels to the left of the current pixel to search for the best match. A larger search range will give more accurate results, but will be computationally expensive. A smaller search range will give less accurate results, but will be less computationally expensive.

   We search the width of the image in this setup.

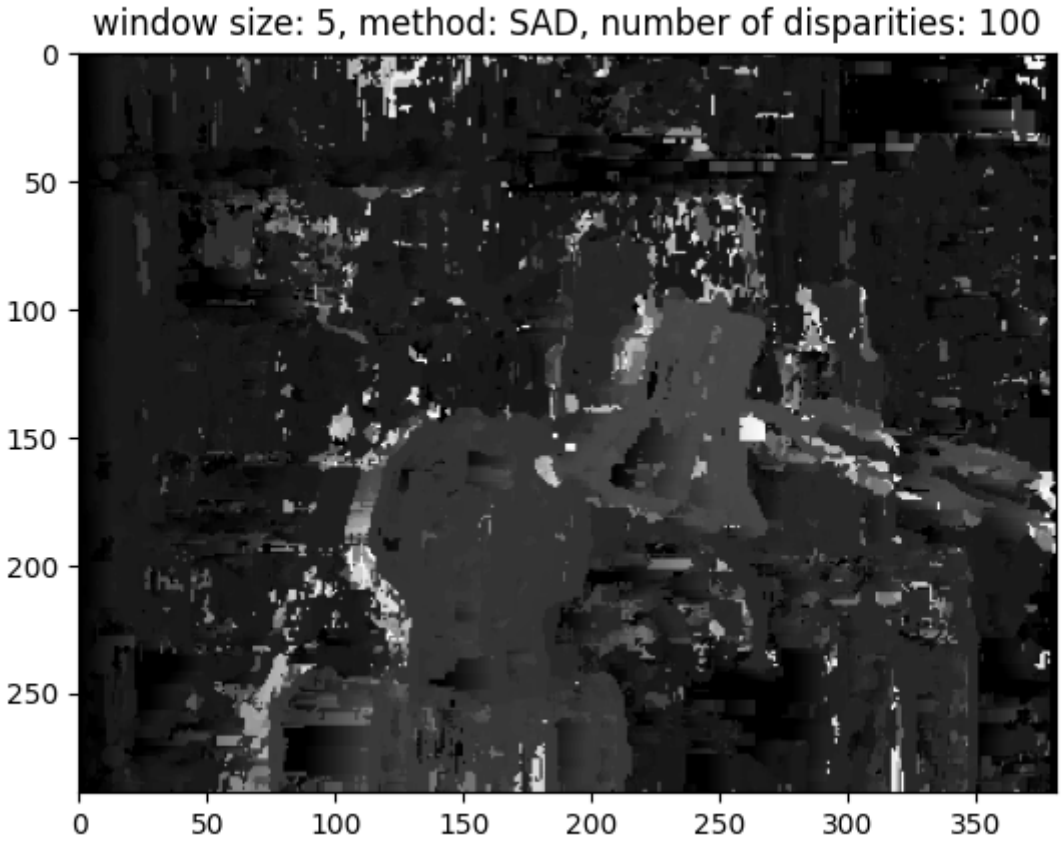
   The following results show the effect of the search range.


window size: 5, method: SAD, number of disparities: 50
0
50
100
150
200
250
0
50
100
150
200
250
300
350

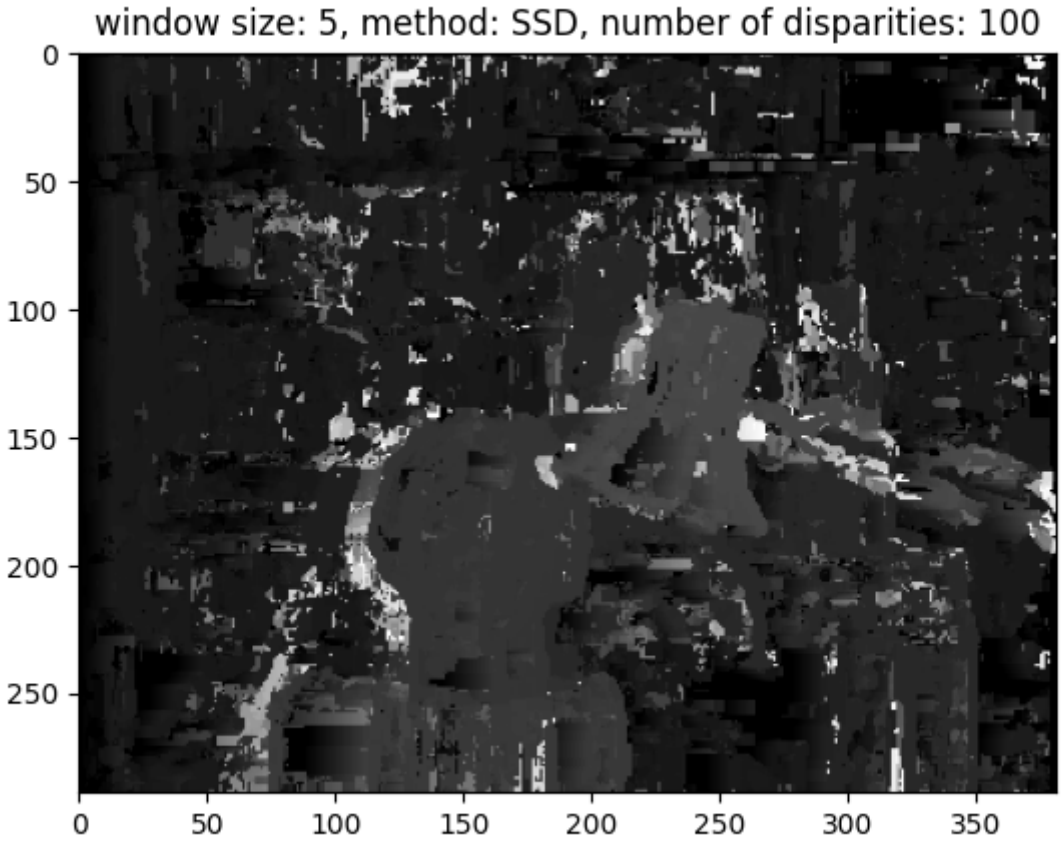
SAD, range = 50


window size: 5, method: SSD, number of disparities: 50
0
50
100
150
200
250
0
50
100
150
200
250
300
350

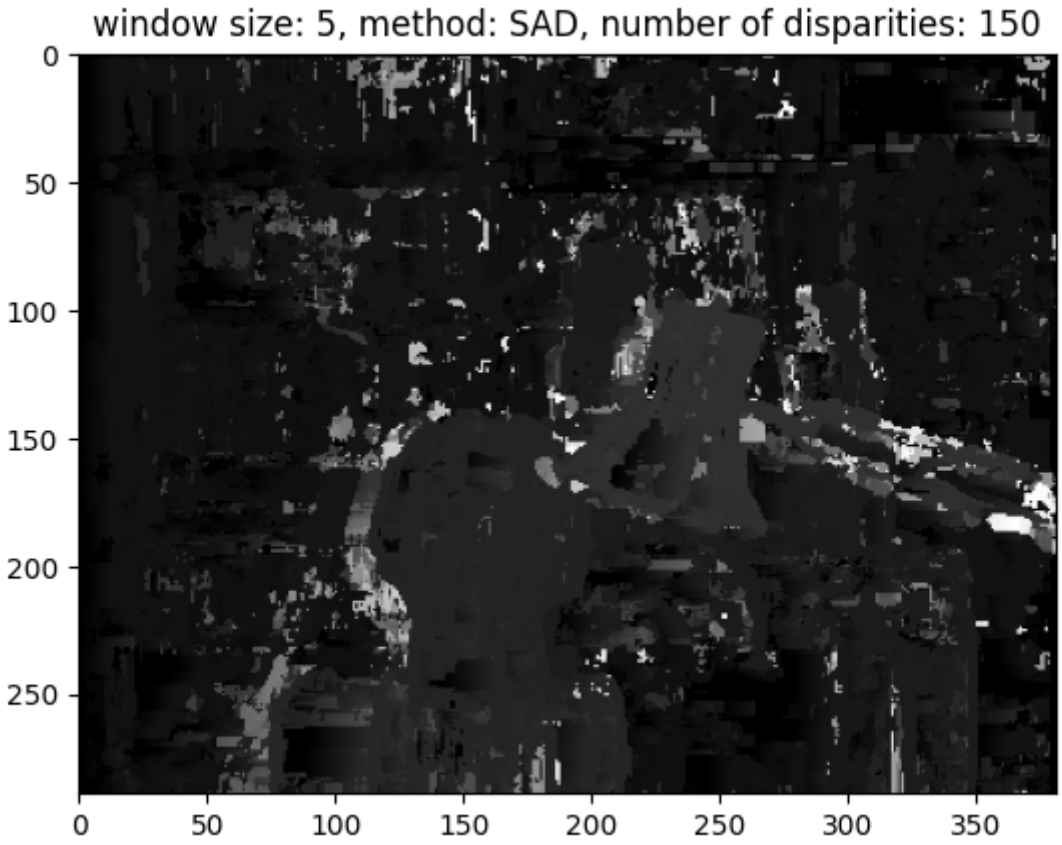
SSD, range = 50



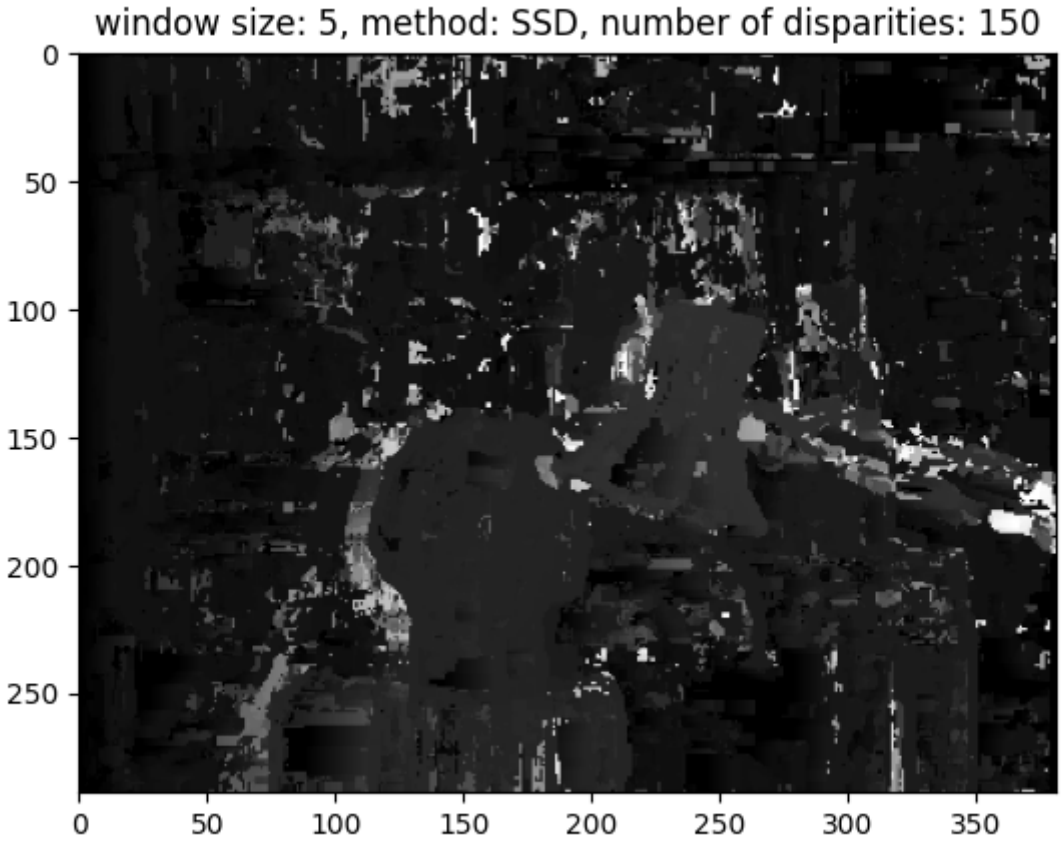
SAD, range = 100



SSD, range = 100



SAD, range = 150



SSD, range = 150

Window Size

The window size is the size of the window used to compare the blocks. The window size is a square, so the window size is the length of one side of the square.

Search Range

The search range is the number of disparities to search for. The search range is the number of pixels to the left of the current pixel to search for the best match.

Method

The method is the method used to compute the distance between two pixels. The method can be either SAD or SSD.

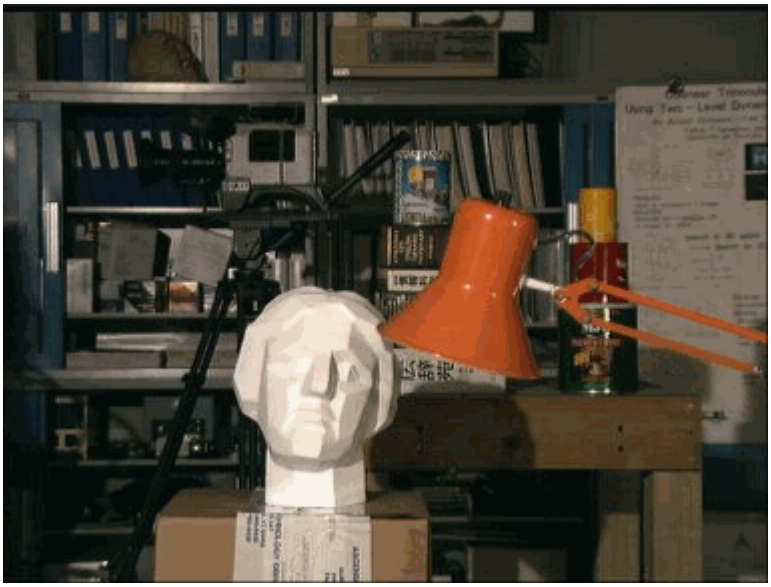
SSD is the sum of the squared differences between the two pixels. SSD is sensitive to intensity offsets. SAD is the sum of the absolute differences between the two pixels.

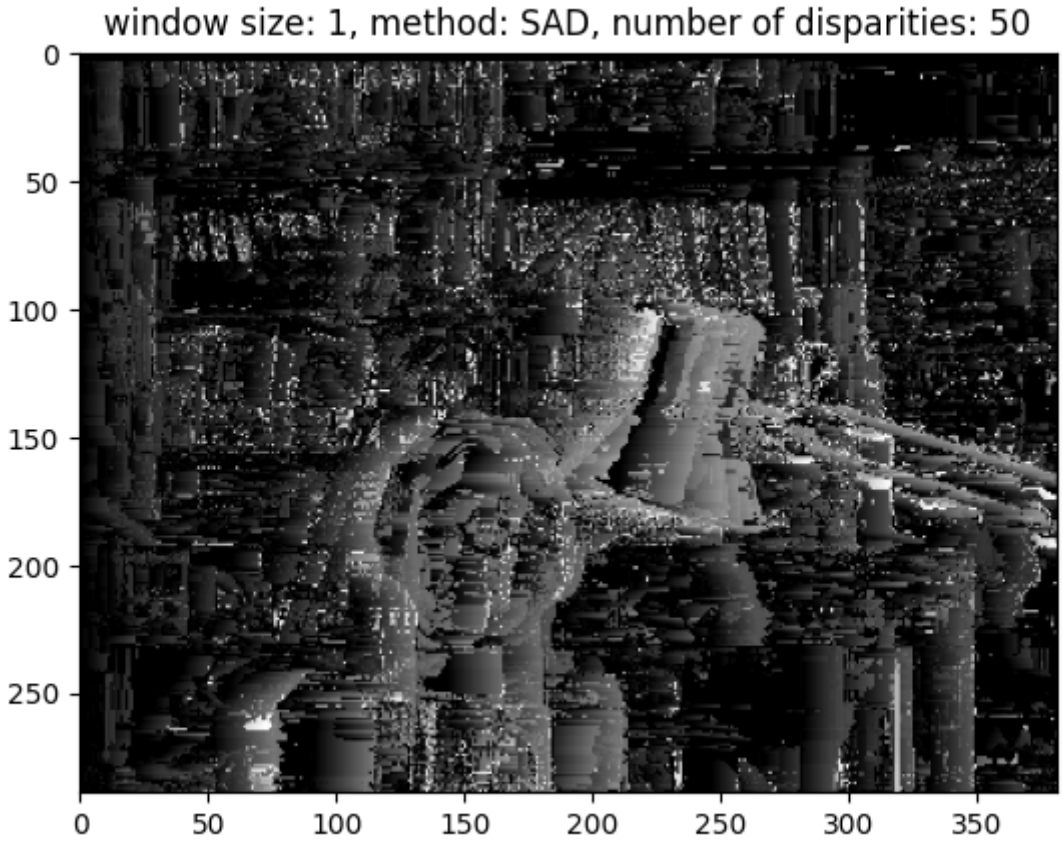
Results in the previous and next sections show the effect of the method.

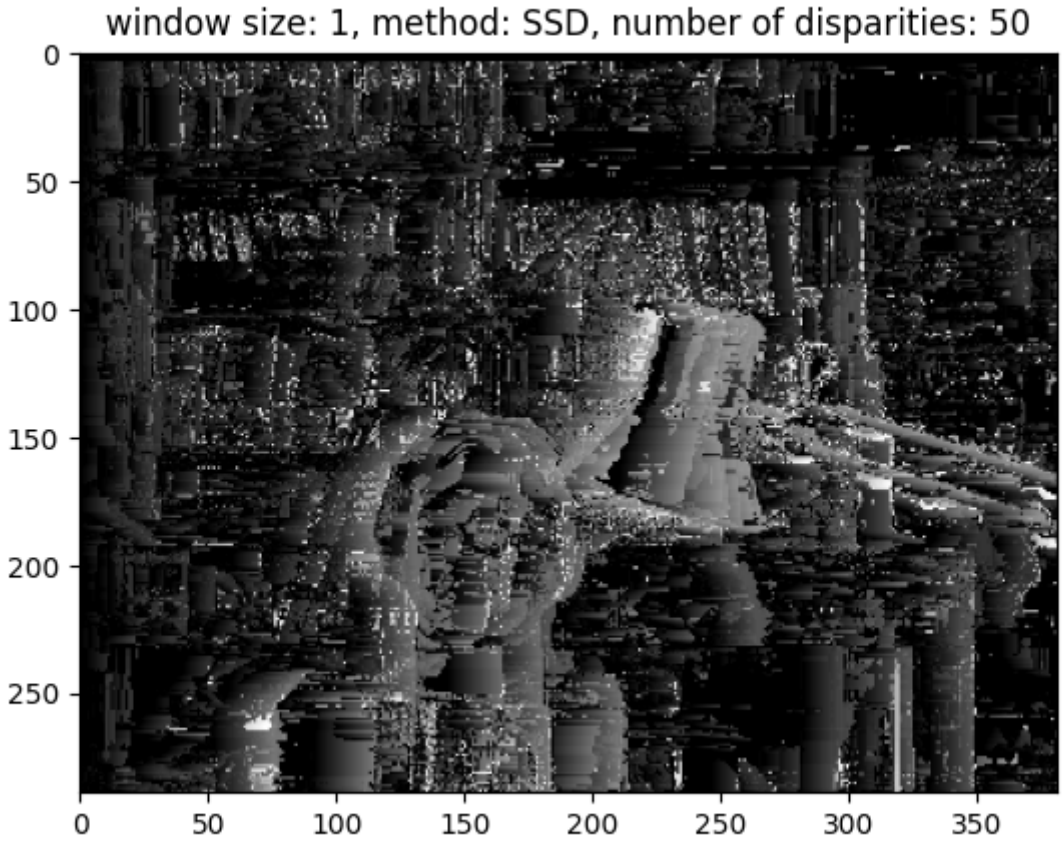
Results

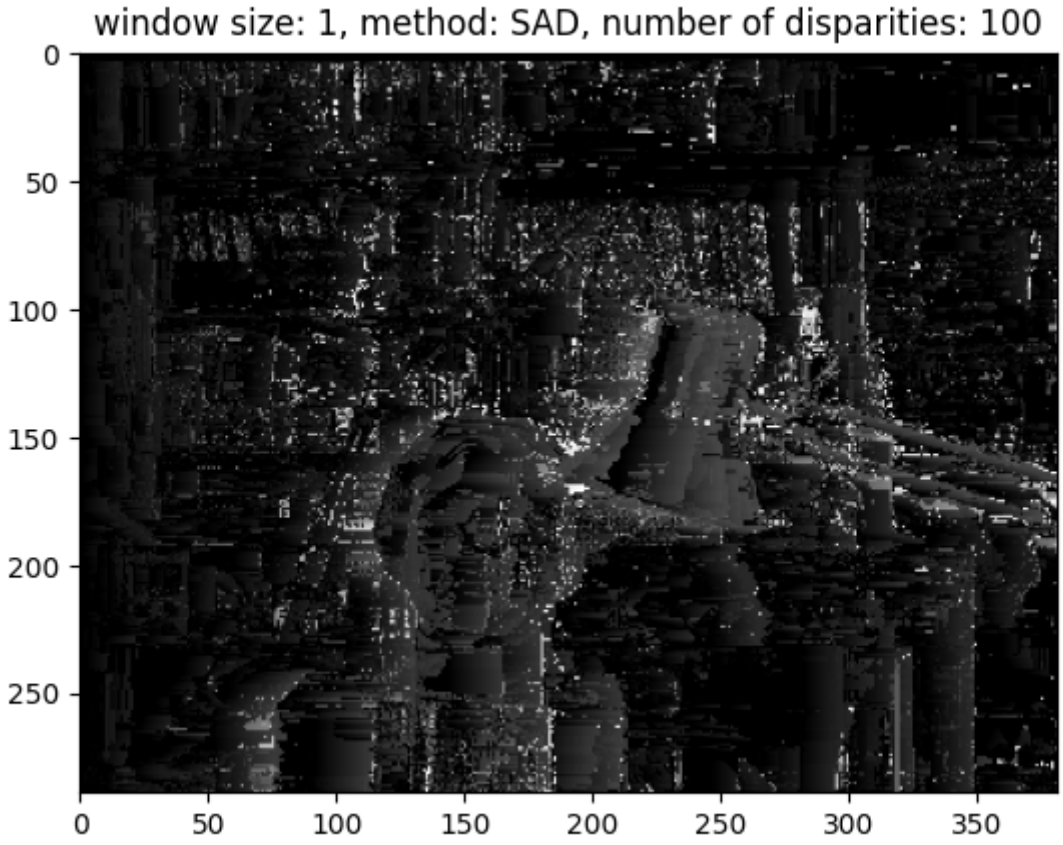
We show the dispaity maps for 3 images with different window sizes and methods.

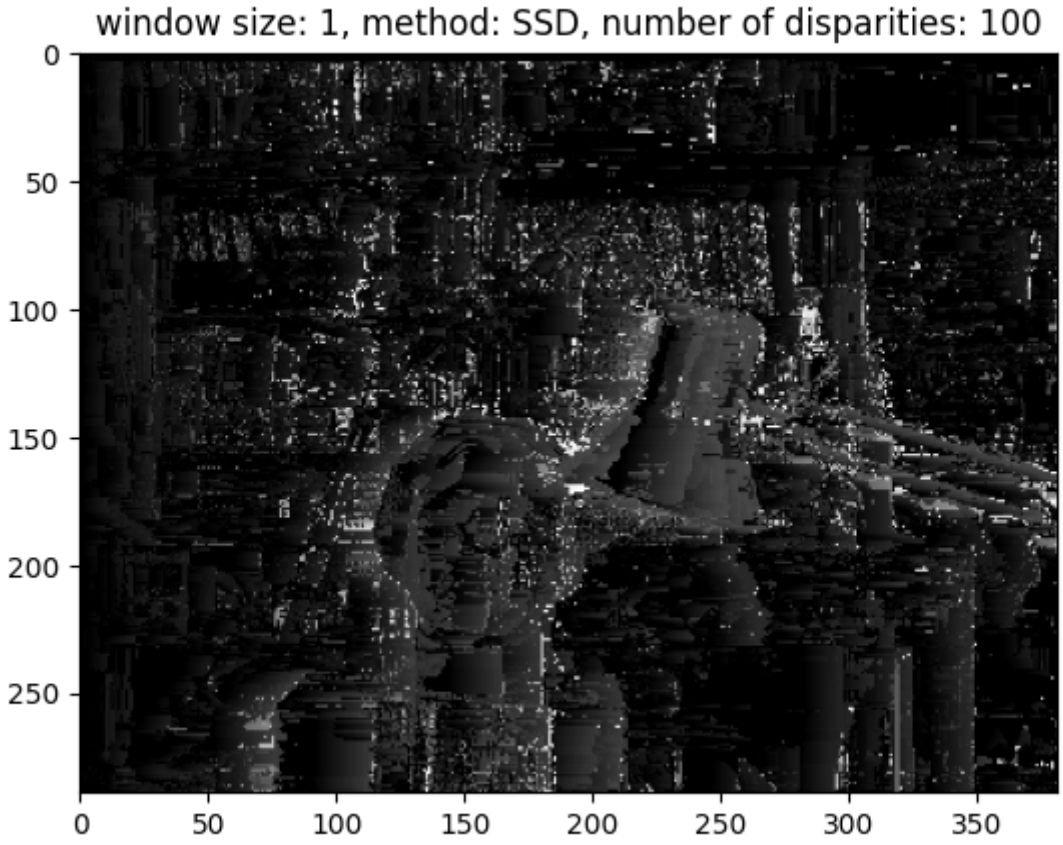
First Image

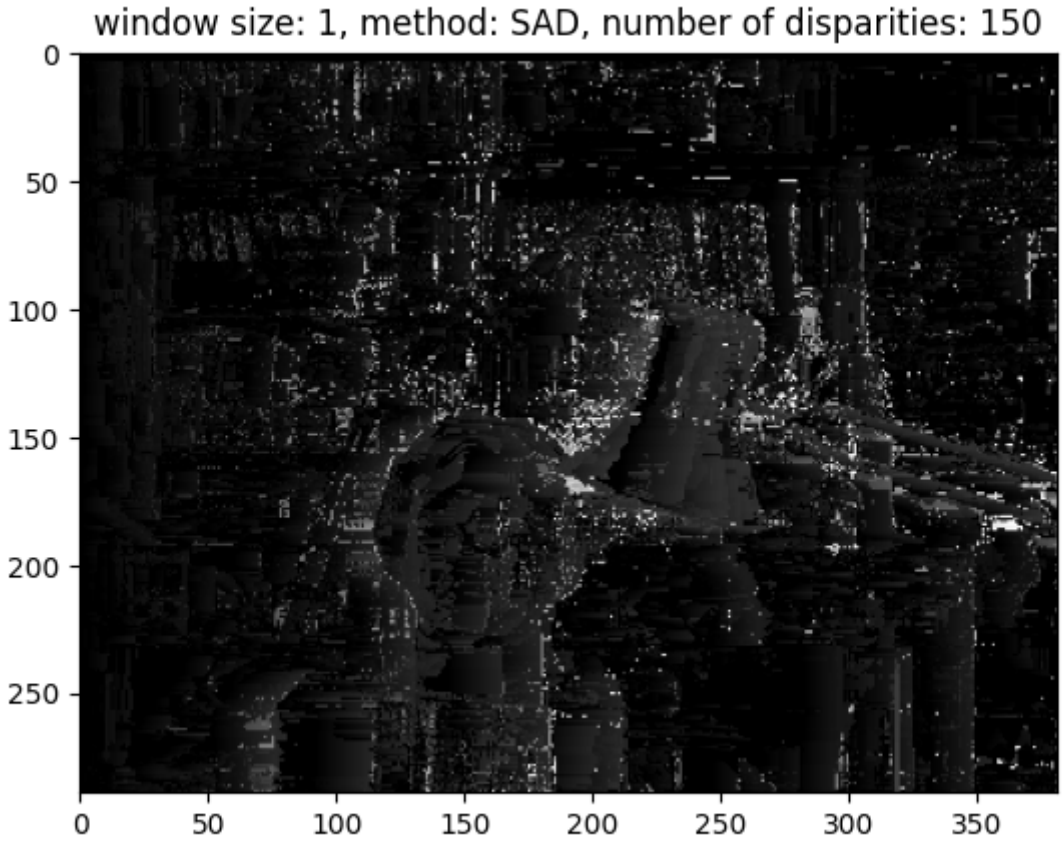


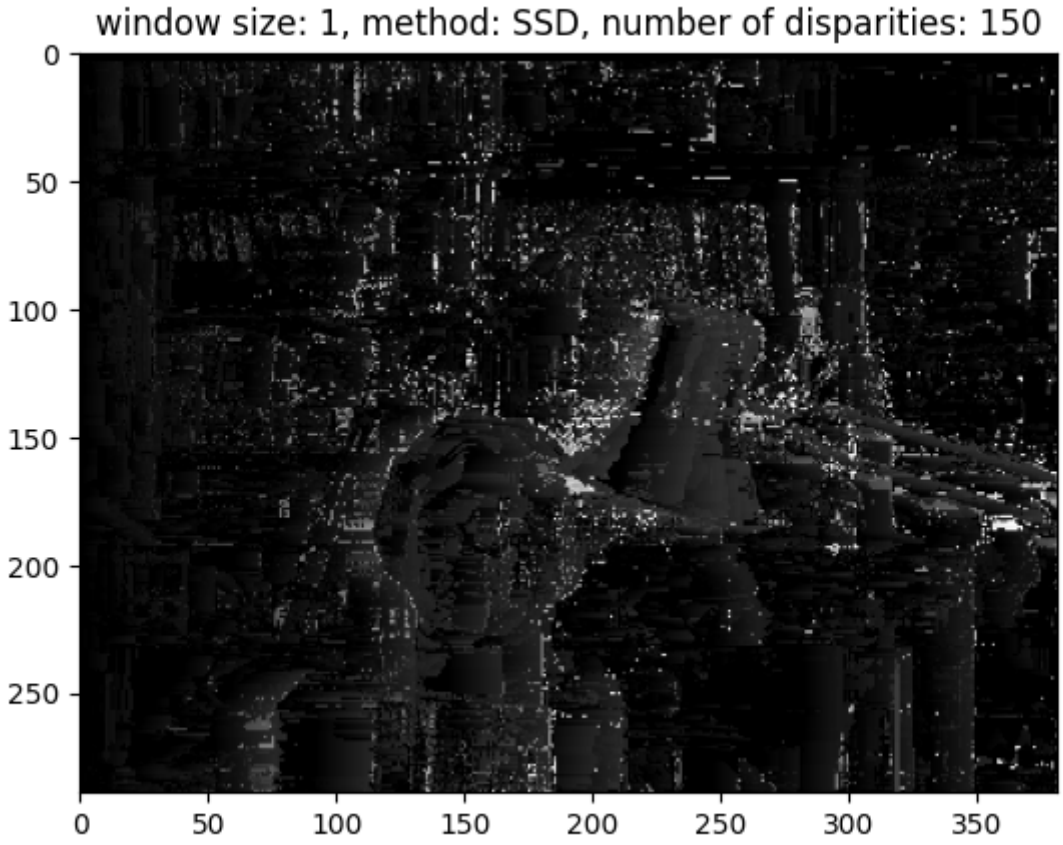

window size: 1, method: SAD, number of disparities: 50
0
50
100
150
200
250
0
50
100
150
200
250
300
350

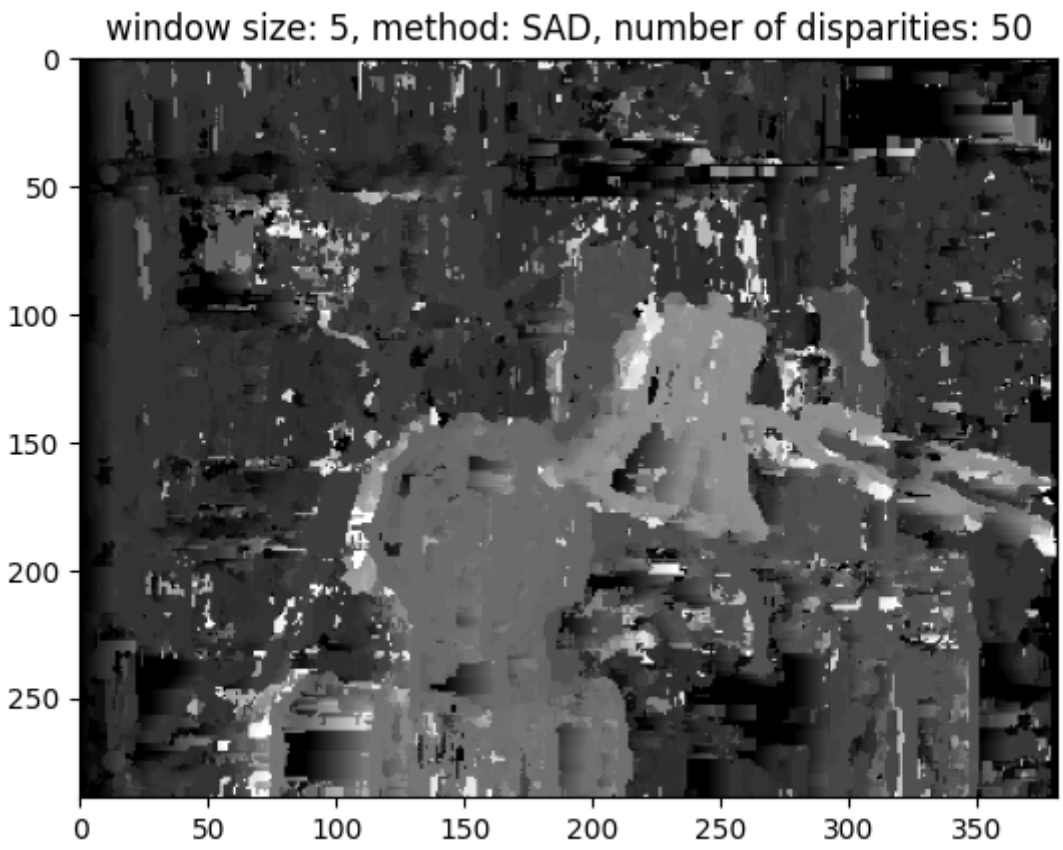

window size: 1, method: SSD, number of disparities: 50
0
50
100
150
200
250
0
50
100
150
200
250
300
350

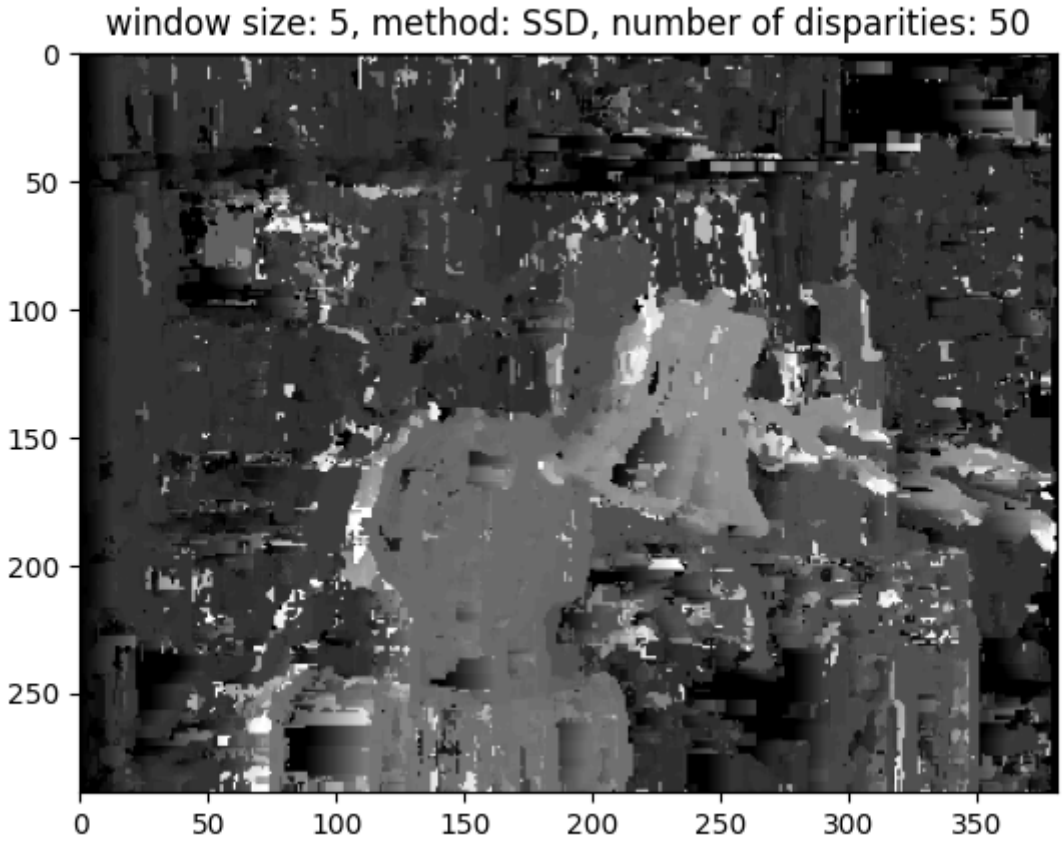

window size: 1, method: SAD, number of disparities: 100
0
50
100
150
200
250
0
50
100
150
200
250
300
350

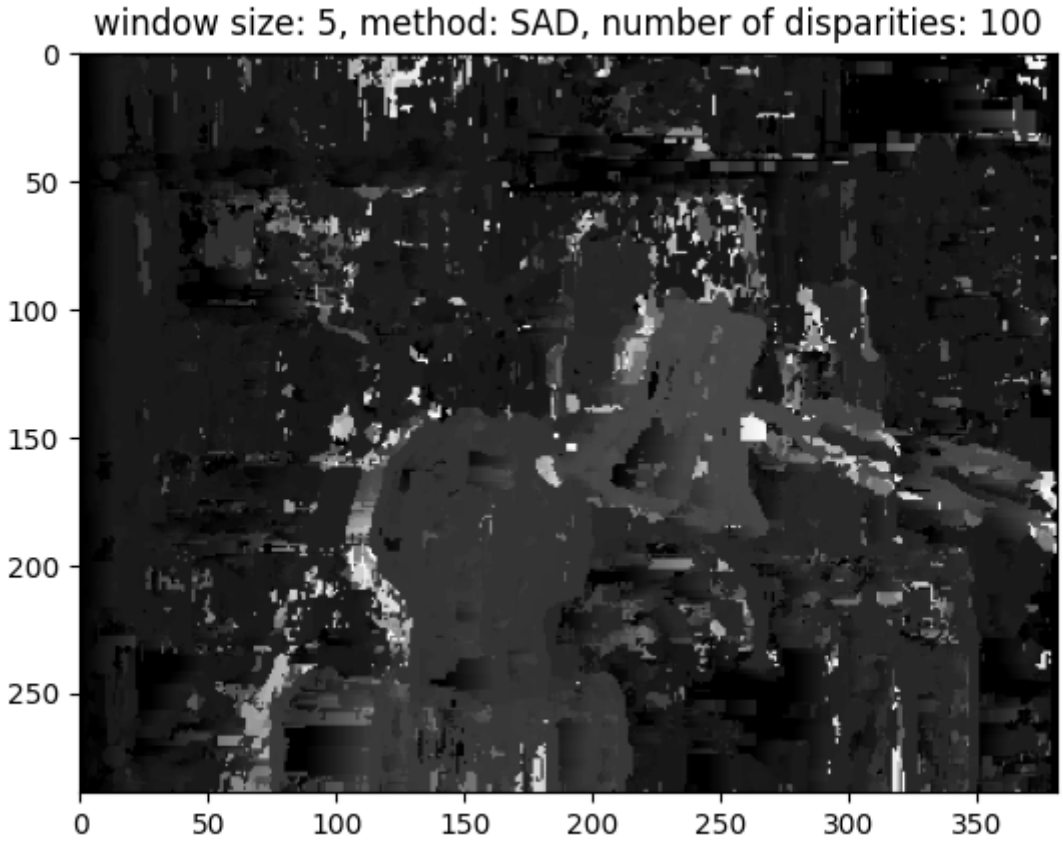

window size: 1, method: SSD, number of disparities: 100
0
50
100
150
200
250
0
50
100
150
200
250
300
350

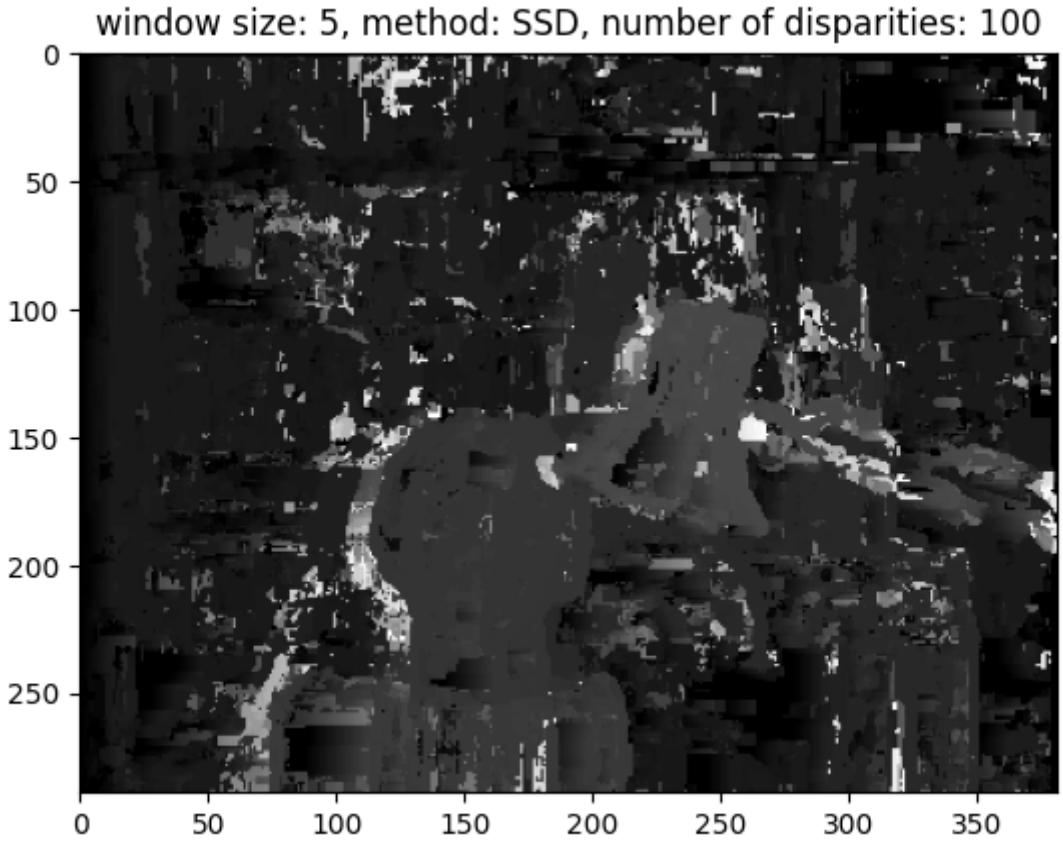

window size: 1, method: SAD, number of disparities: 150
0
50
100
150
200
250
0
50
100
150
200
250
300
350

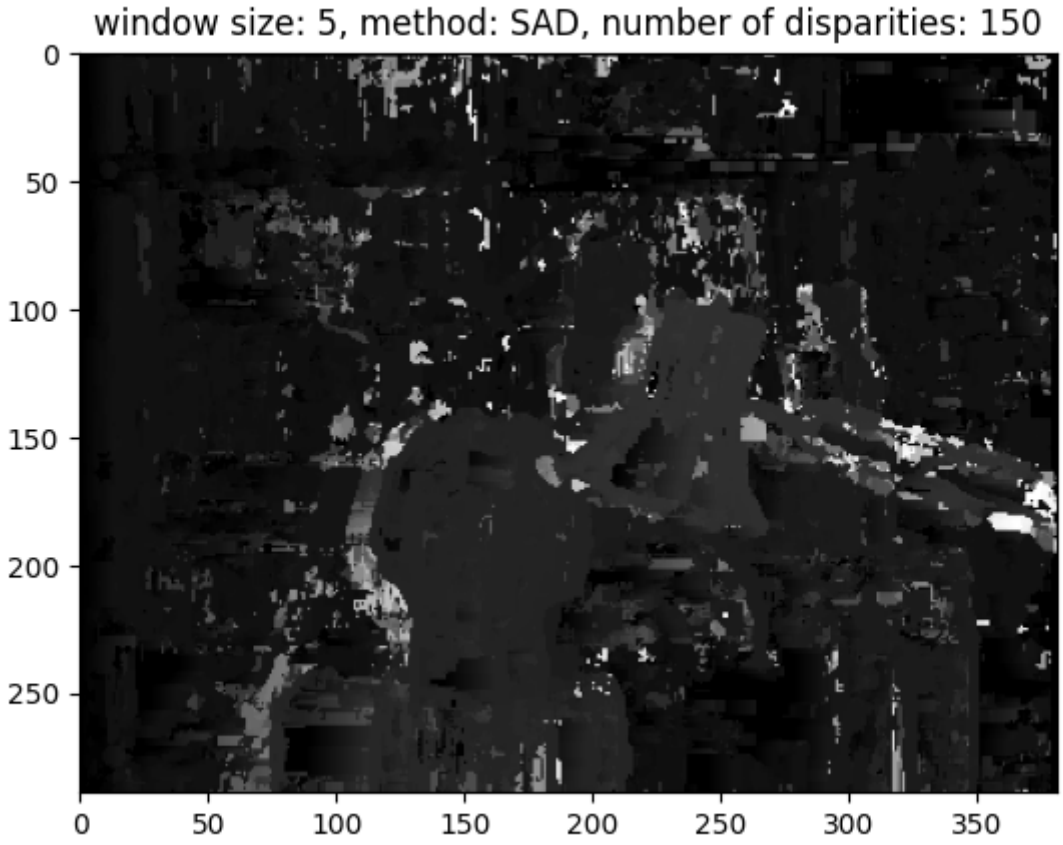

window size: 1, method: SSD, number of disparities: 150
0
50
100
150
200
250
0
50
100
150
200
250
300
350

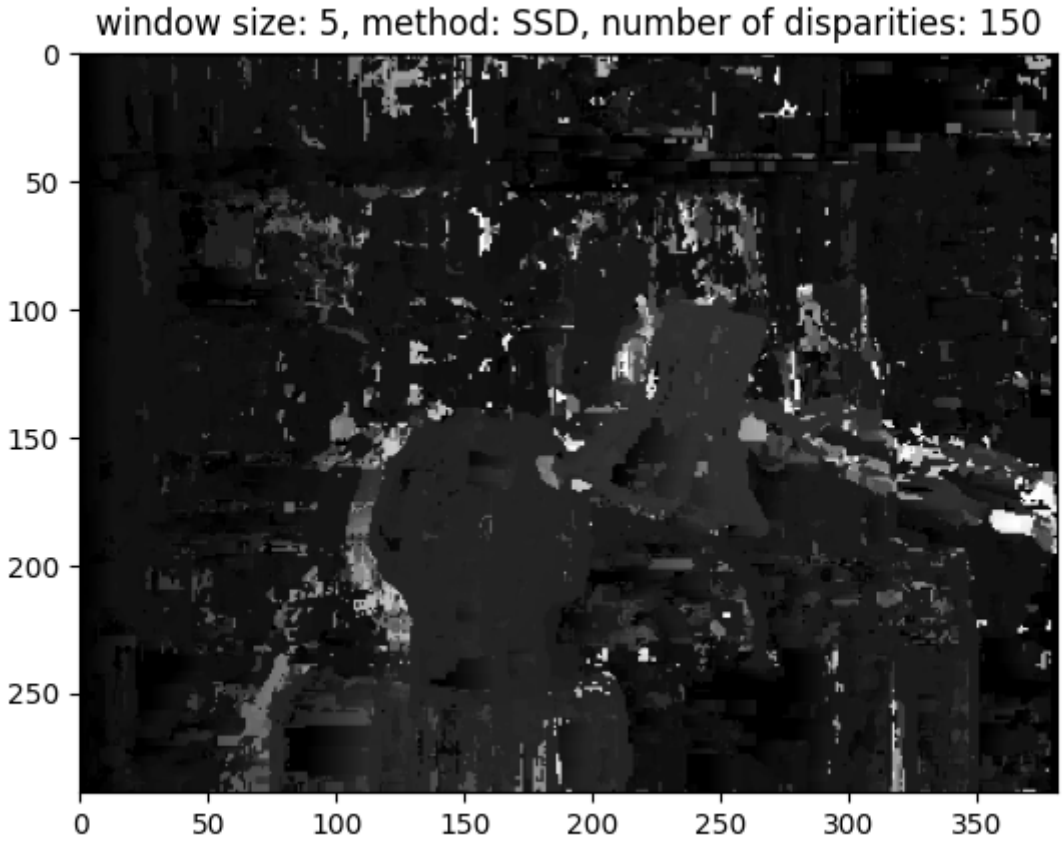

window size: 5, method: SAD, number of disparities: 50
0
50
100
150
200
250
0
50
100
150
200
250
300
350

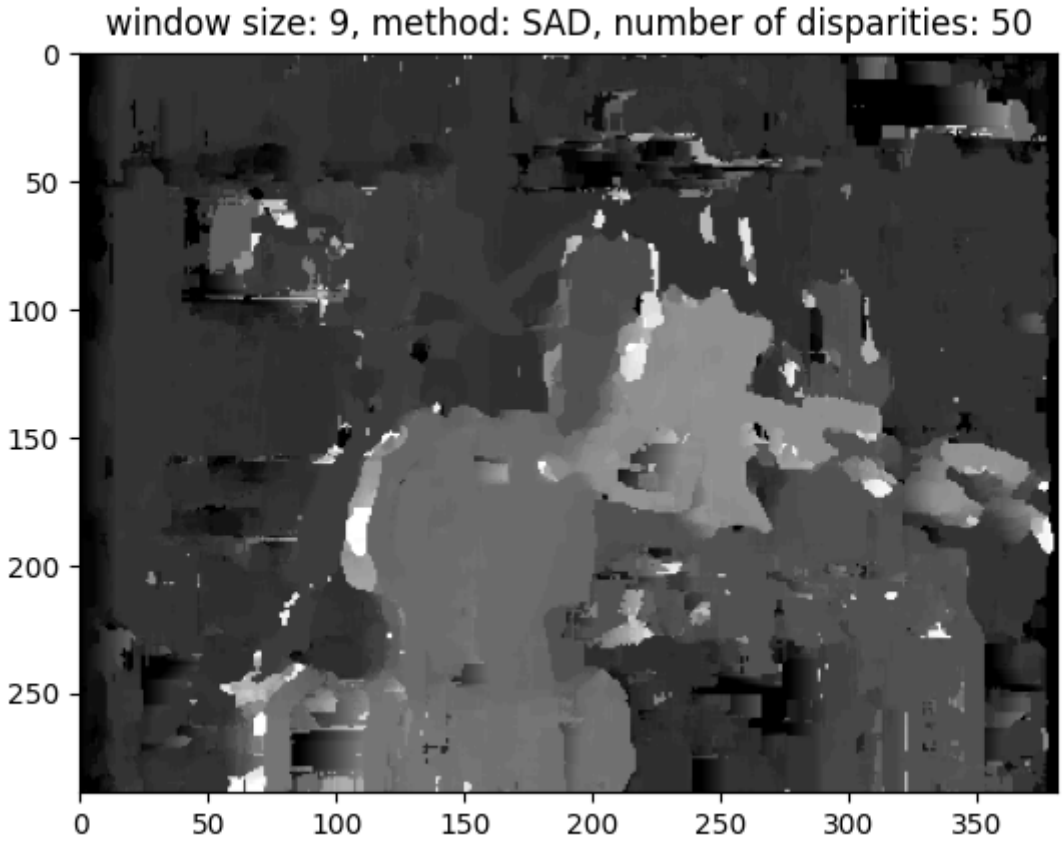

window size: 5, method: SSD, number of disparities: 50
0
50
100
150
200
250
0
50
100
150
200
250
300
350

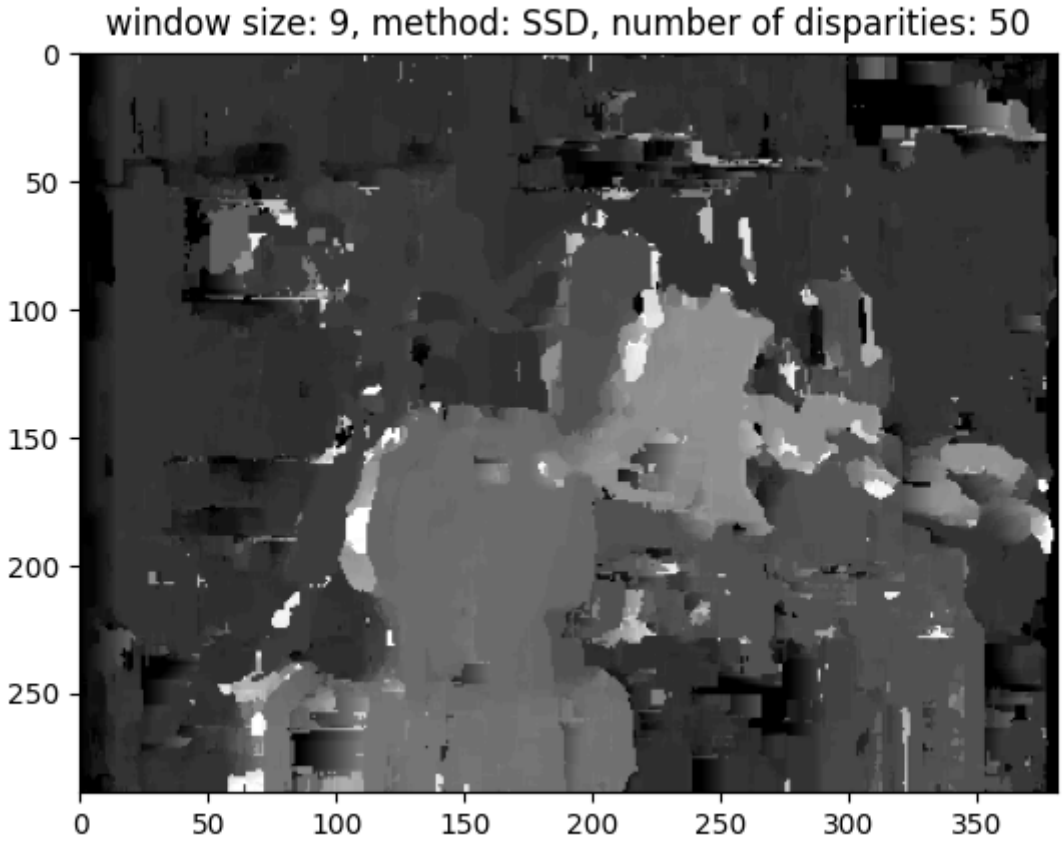

window size: 5, method: SAD, number of disparities: 100
0
50
100
150
200
250
0
50
100
150
200
250
300
350

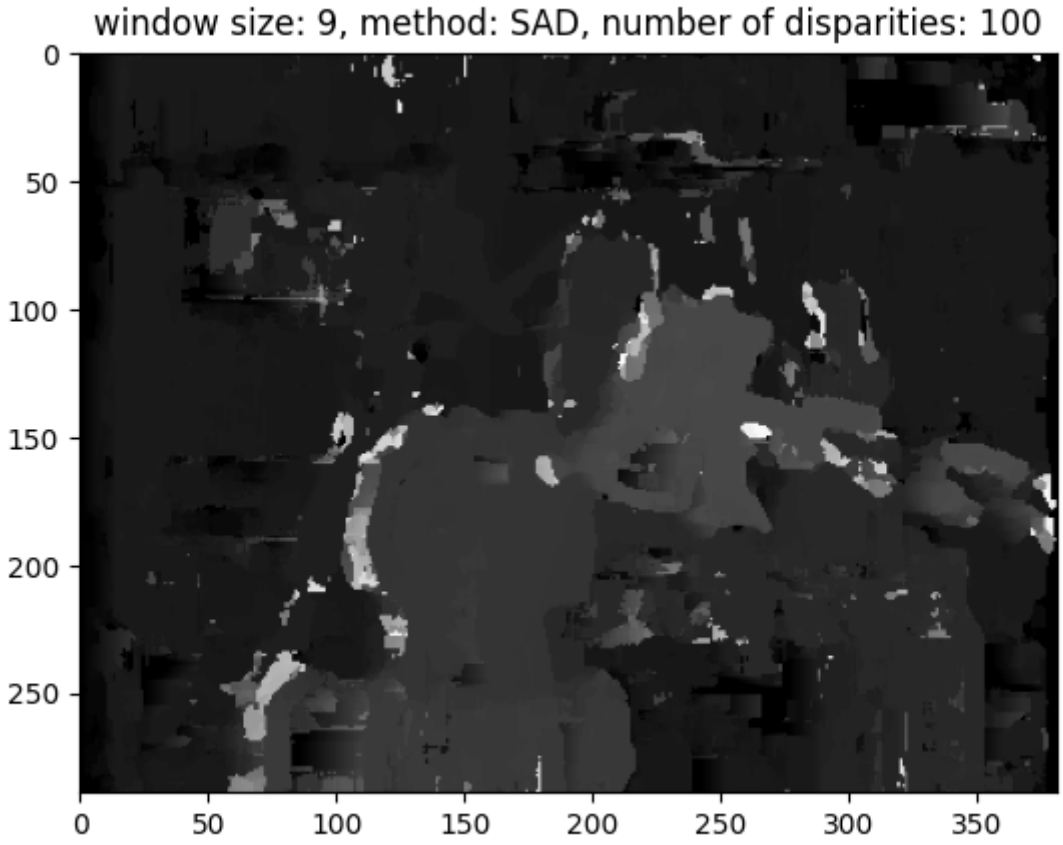

window size: 5, method: SSD, number of disparities: 100
0
50
100
150
200
250
0
50
100
150
200
250
300
350

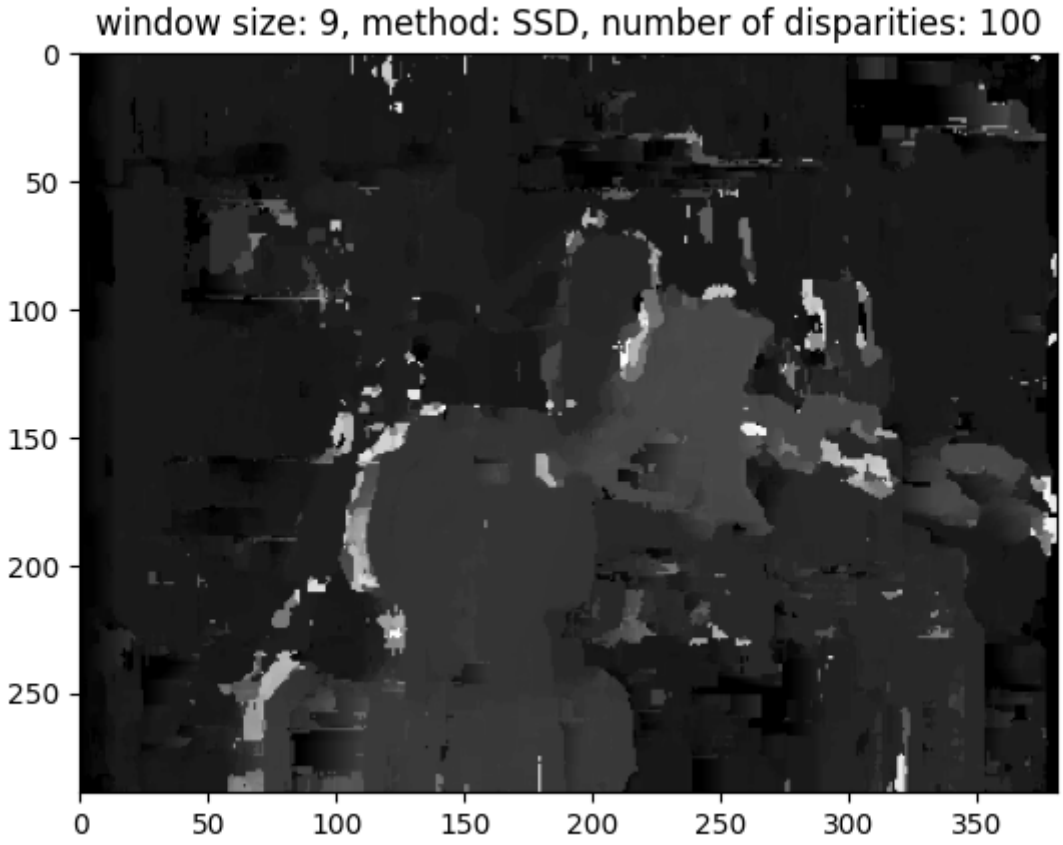

window size: 5, method: SAD, number of disparities: 150
0
50
100
150
200
250
0
50
100
150
200
250
300
350

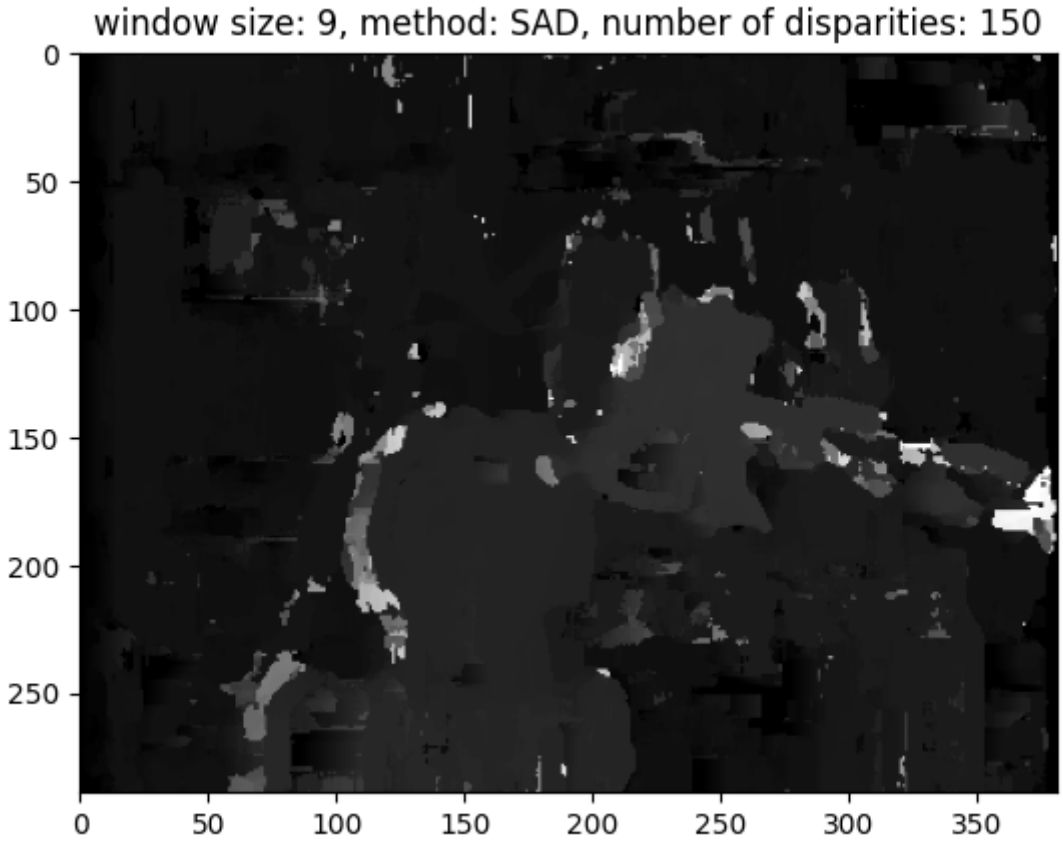

window size: 5, method: SSD, number of disparities: 150
0
50
100
150
200
250
0
50
100
150
200
250
300
350

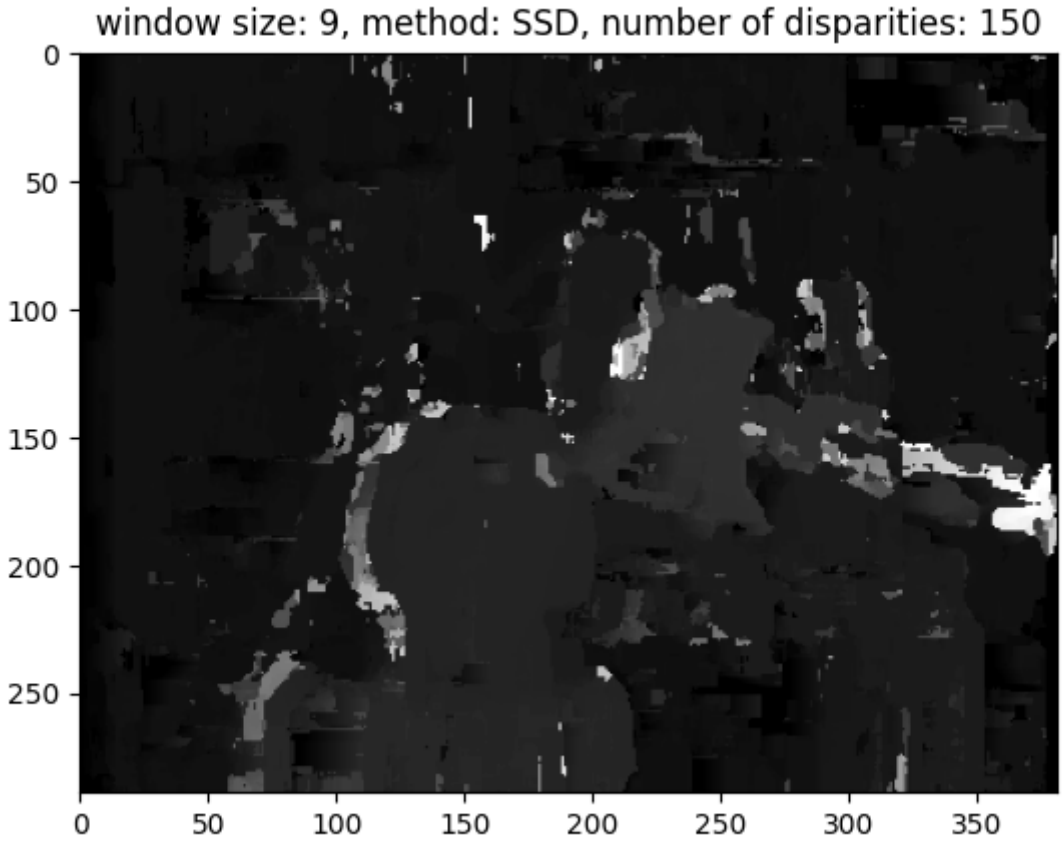

window size: 9, method: SAD, number of disparities: 50
0
50
100
150
200
250
0
50
100
150
200
250
300
350


window size: 9, method: SSD, number of disparities: 50
0
50
100
150
200
250
0
50
100
150
200
250
300
350


window size: 9, method: SAD, number of disparities: 100
0
50
100
150
200
250
0
50
100
150
200
250
300
350


window size: 9, method: SSD, number of disparities: 100
0
50
100
150
200
250
0
50
100
150
200
250
300
350

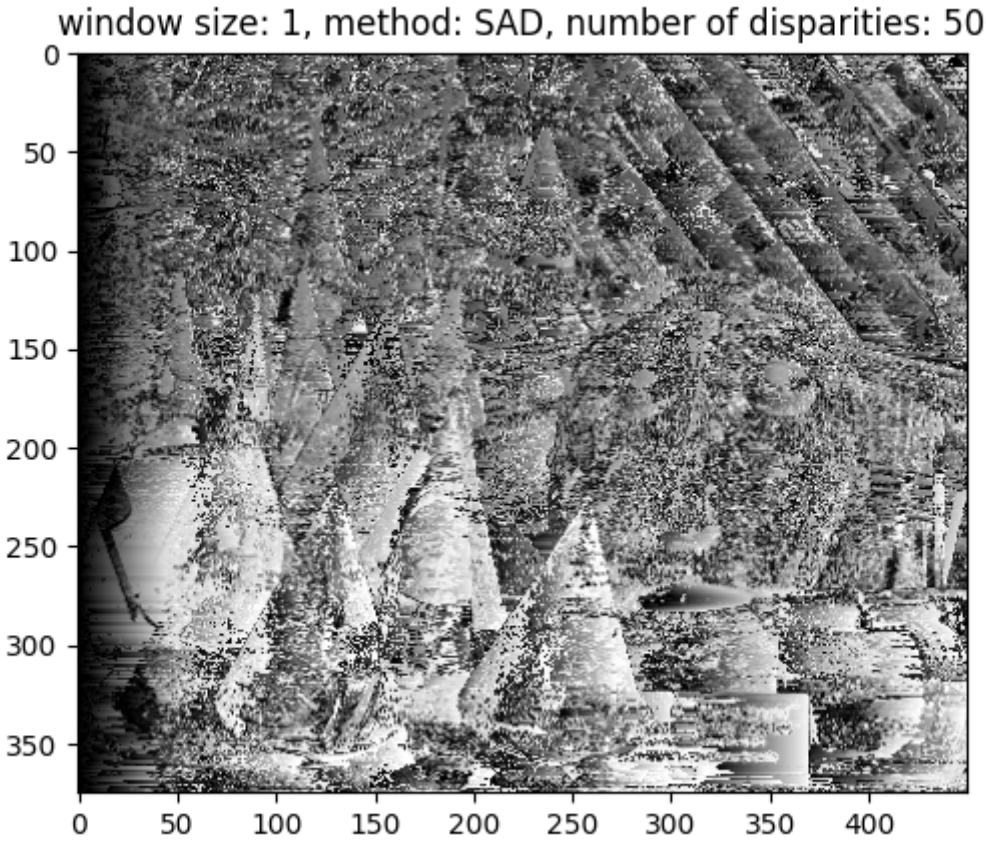


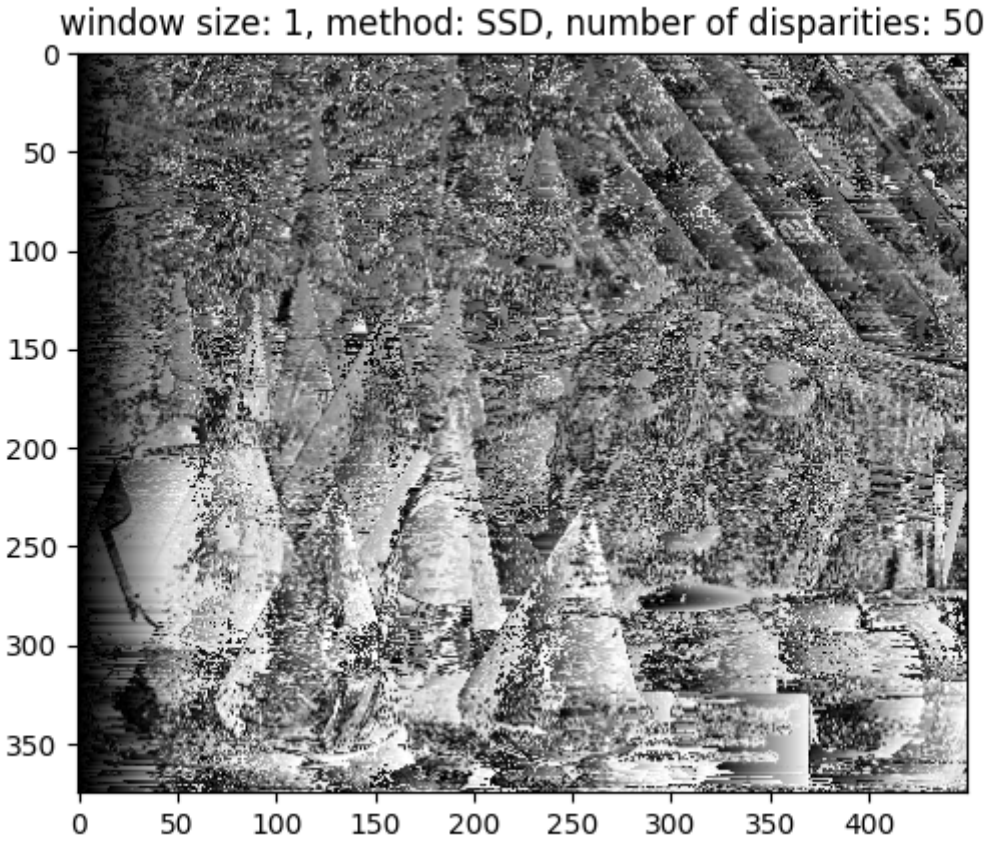


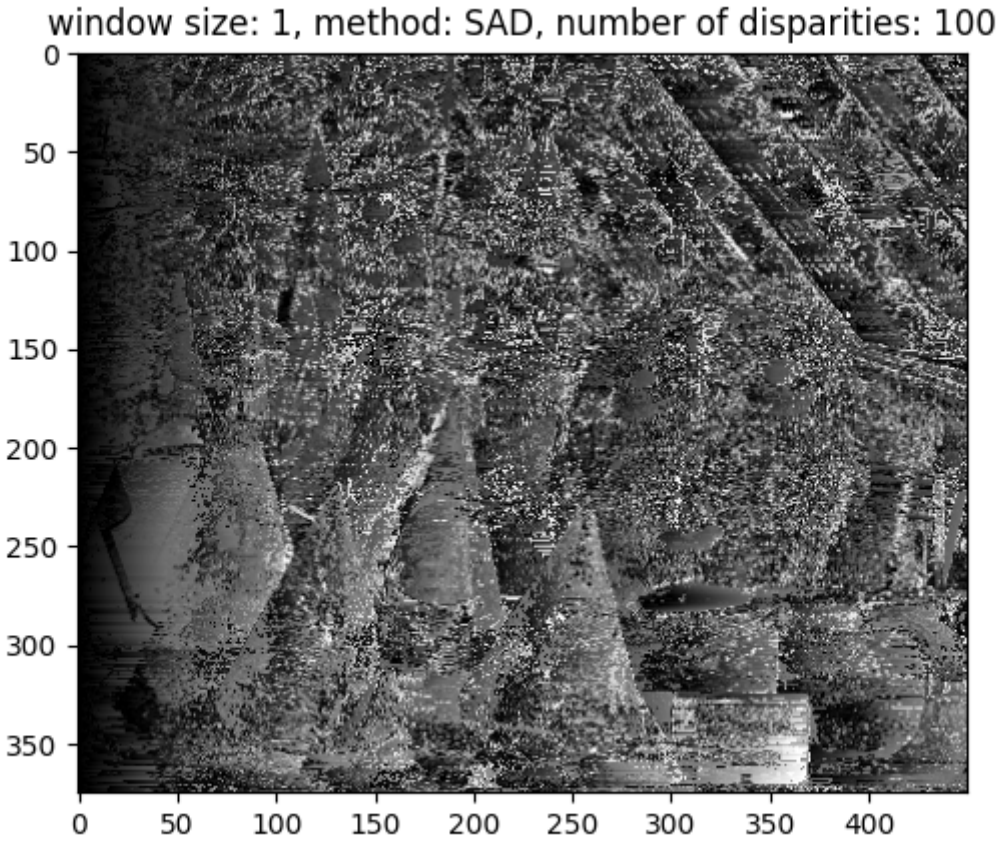
Second Image

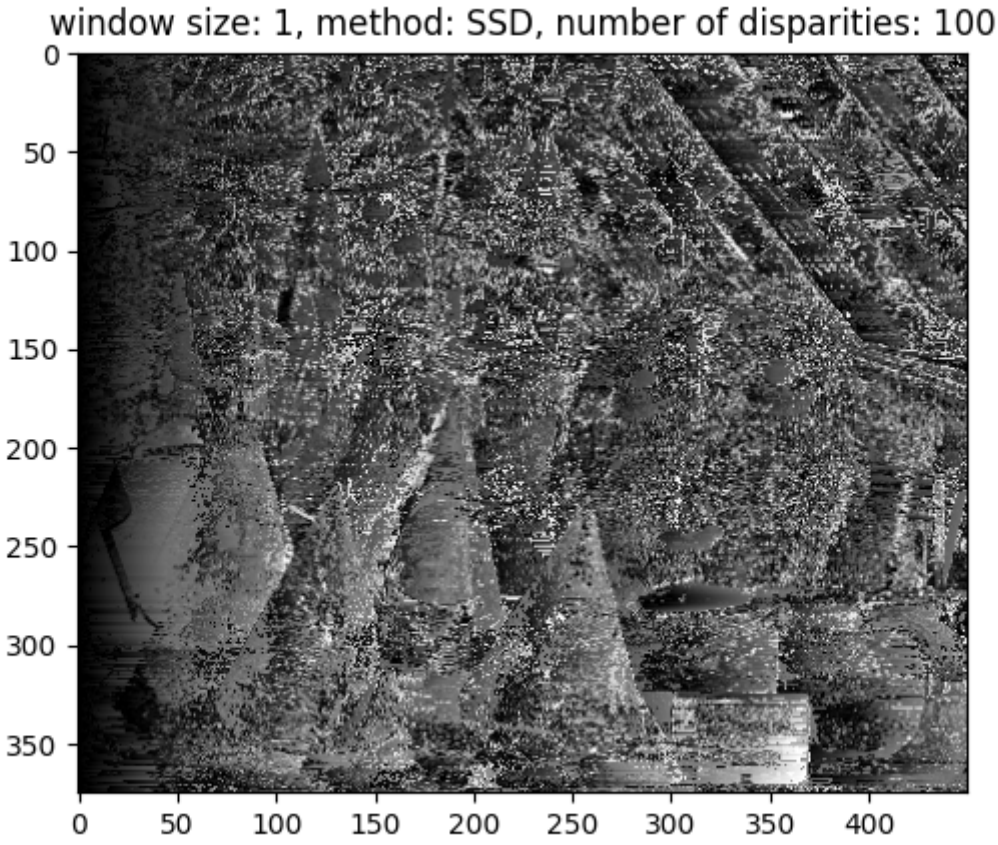

Fisherbrand
Safety Matches

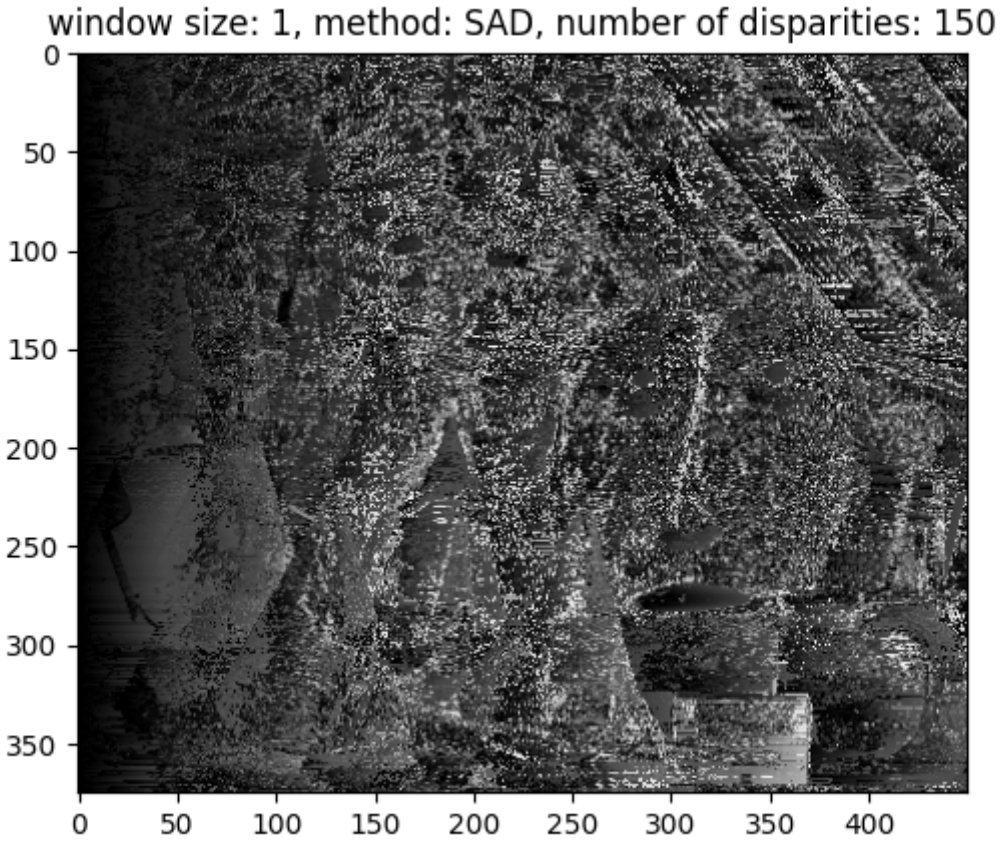

Fisherbrand
Safety Matches

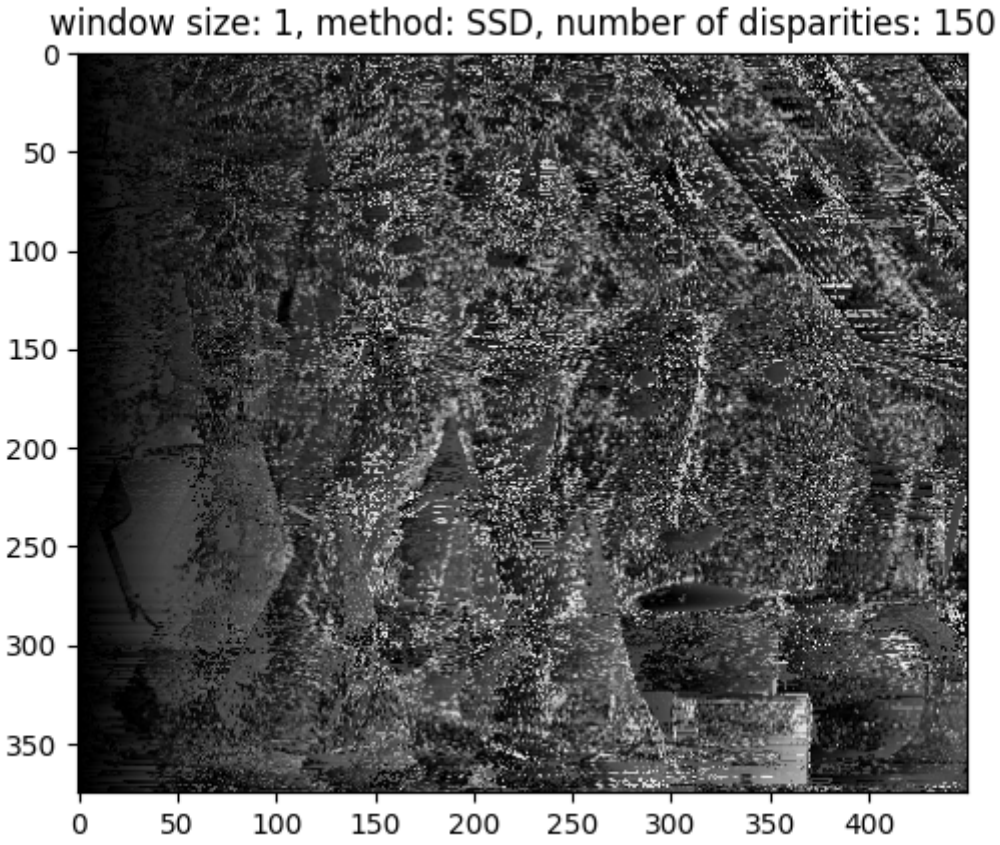

window size: 1, method: SAD, number of disparities: 50
0
50
100
150
200
250
300
350
0
50
100
150
200
250
300
350
400

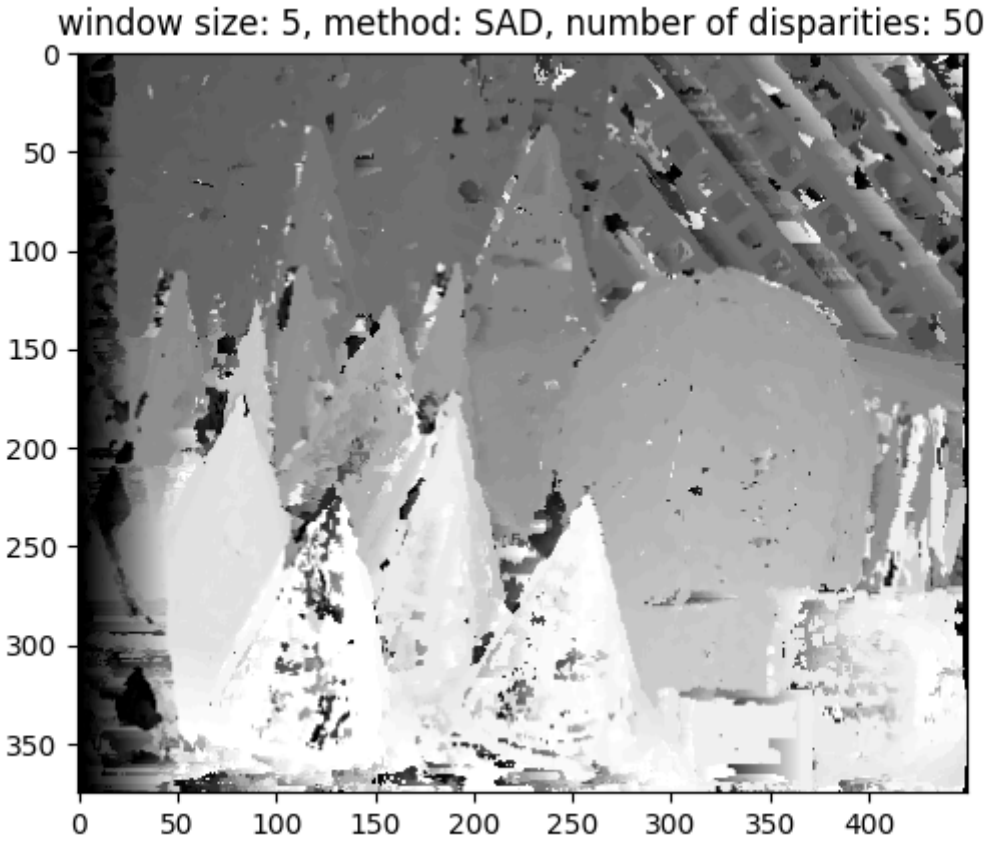

window size: 1, method: SSD, number of disparities: 50
0
50
100
150
200
250
300
350
0
50
100
150
200
250
300
350
400

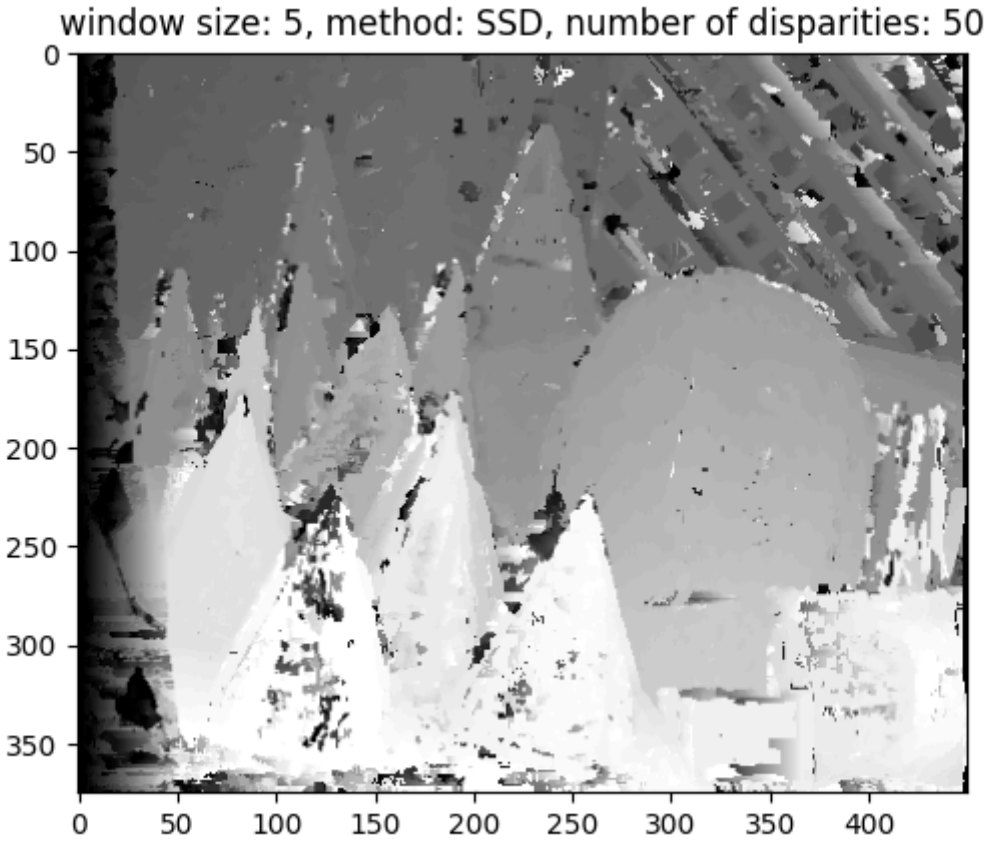

window size: 1, method: SAD, number of disparities: 100
0
50
100
150
200
250
300
350
0
50
100
150
200
250
300
350
400

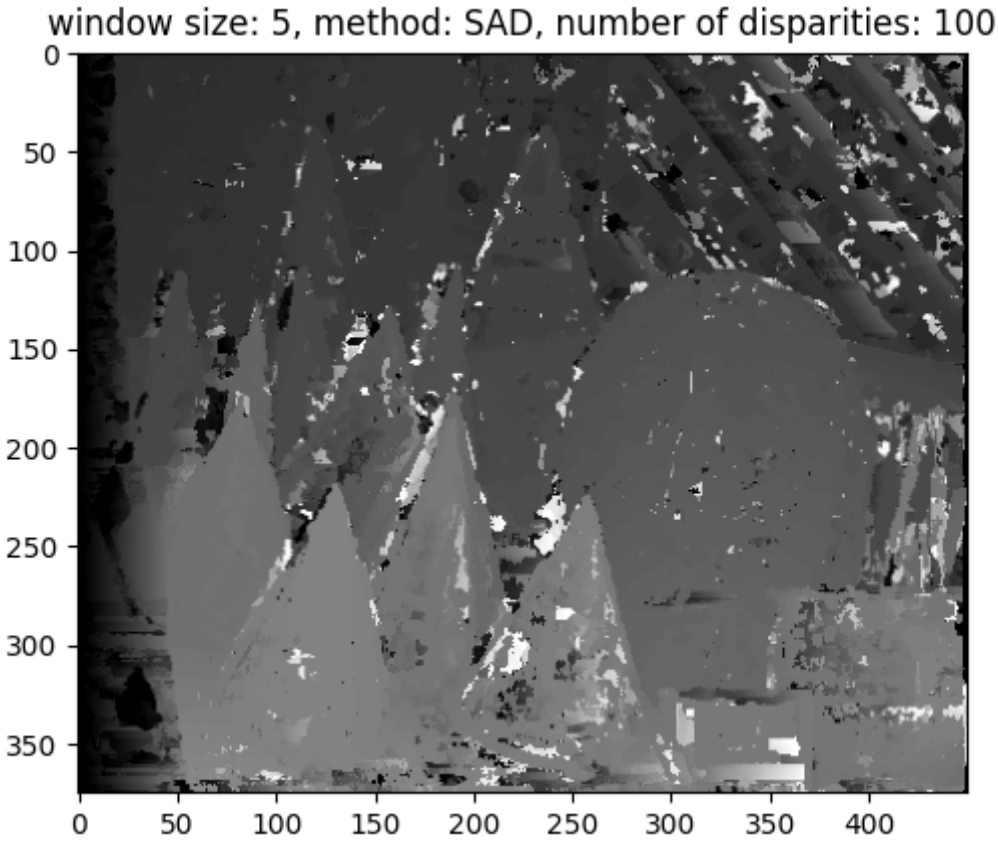

window size: 1, method: SSD, number of disparities: 100
0
50
100
150
200
250
300
350
0
50
100
150
200
250
300
350
400

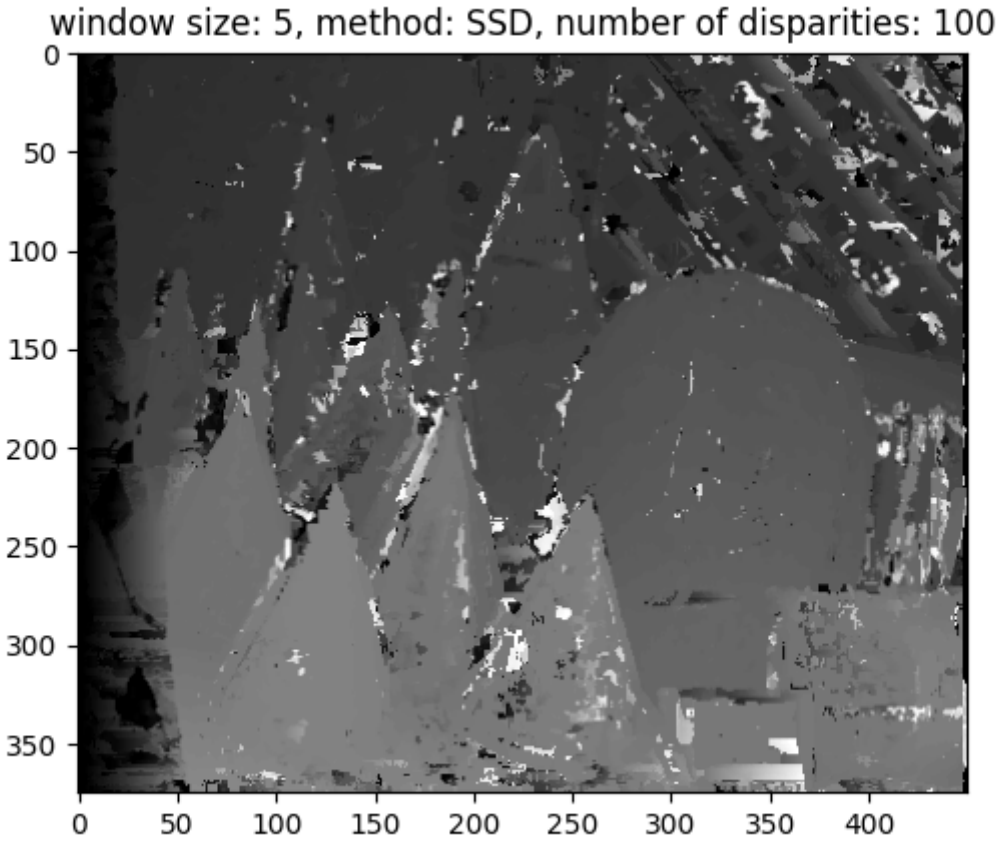

window size: 1, method: SAD, number of disparities: 150
0
50
100
150
200
250
300
350
0
50
100
150
200
250
300
350
400

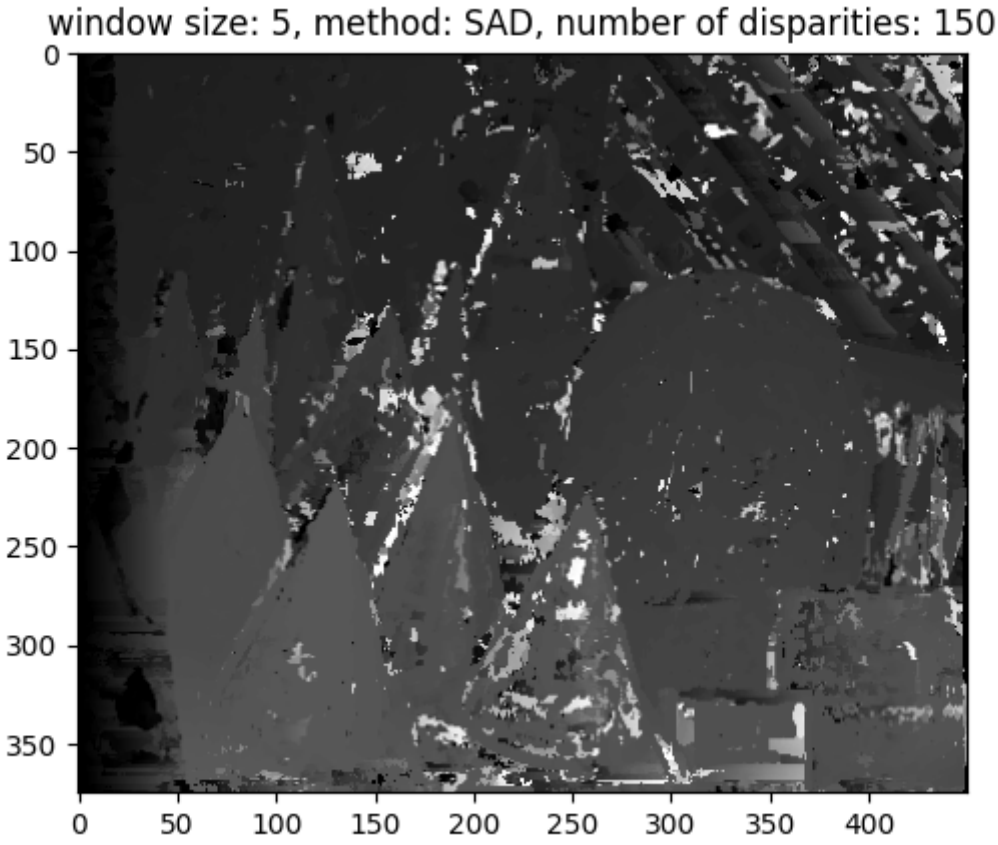

window size: 1, method: SSD, number of disparities: 150
0
50
100
150
200
250
300
350
0
50
100
150
200
250
300
350
400

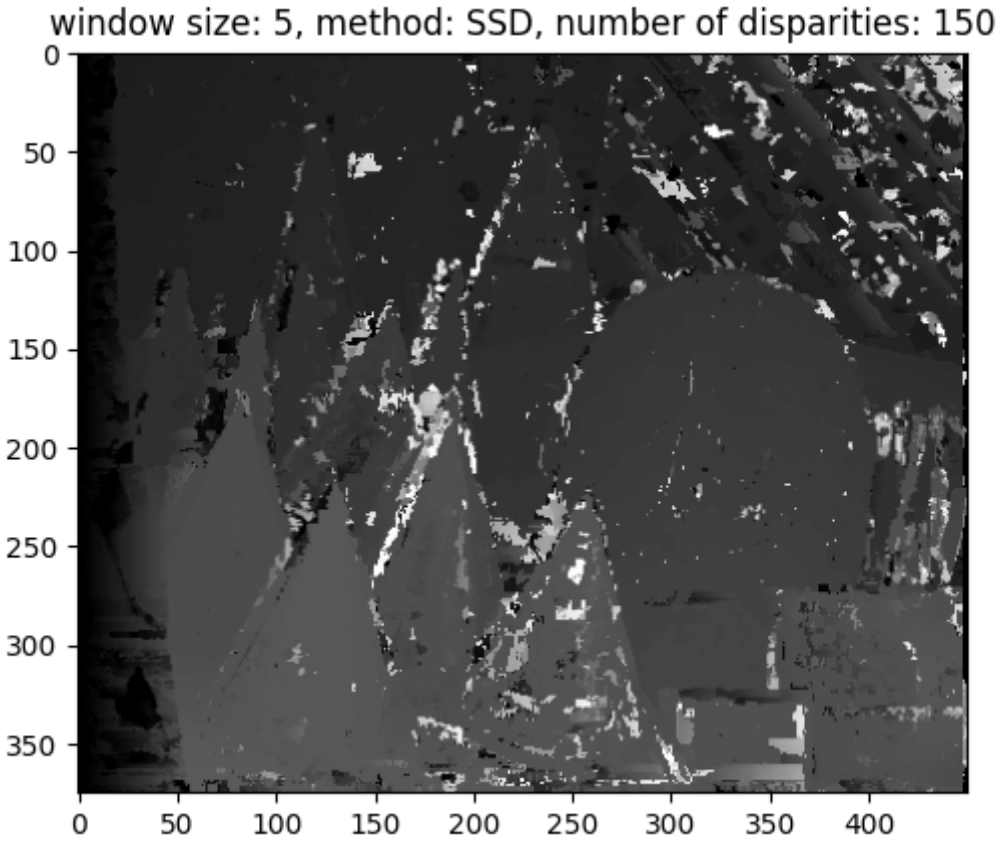

window size: 5, method: SAD, number of disparities: 50
0
50
100
150
200
250
300
350
0
50
100
150
200
250
300
350
400

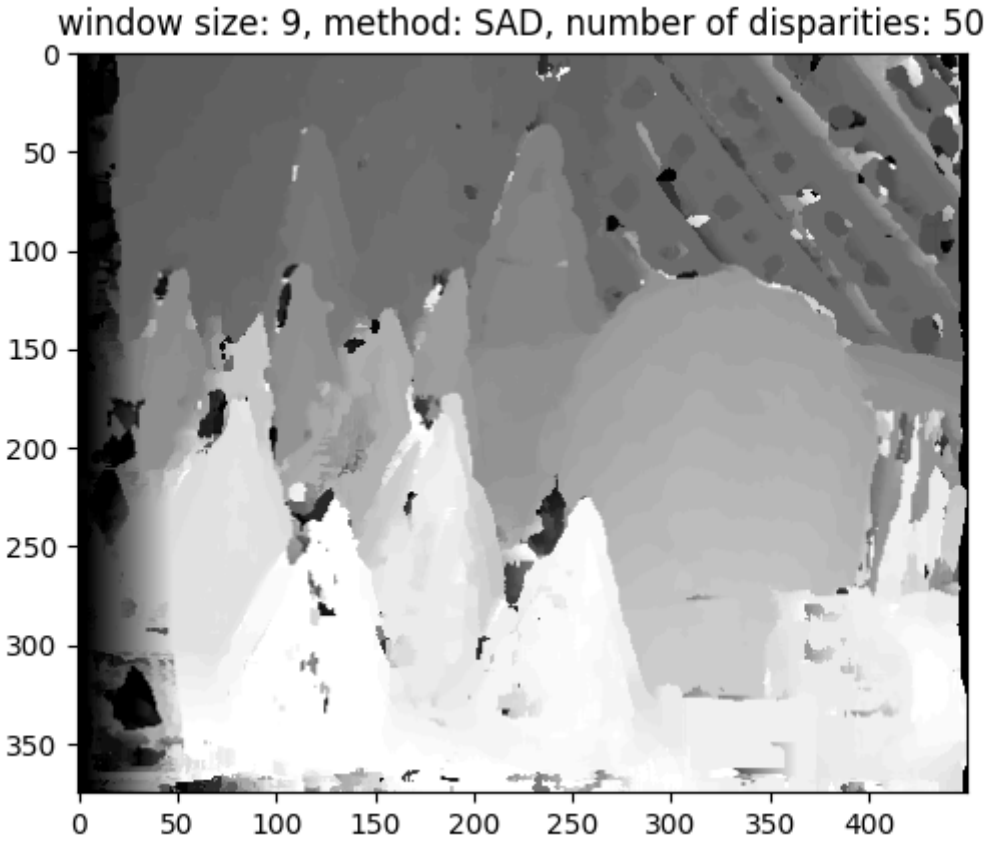

window size: 5, method: SSD, number of disparities: 50
0
50
100
150
200
250
300
350
0
50
100
150
200
250
300
350
400

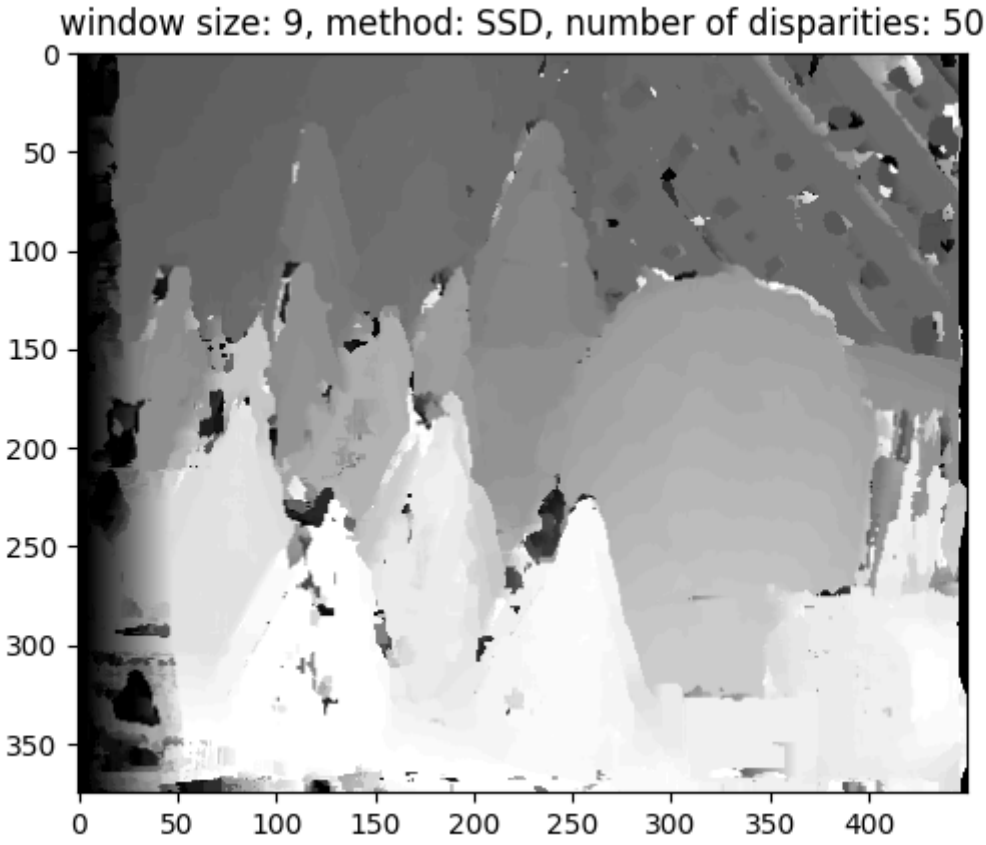

window size: 5, method: SAD, number of disparities: 100
0
50
100
150
200
250
300
350
0
50
100
150
200
250
300
350
400

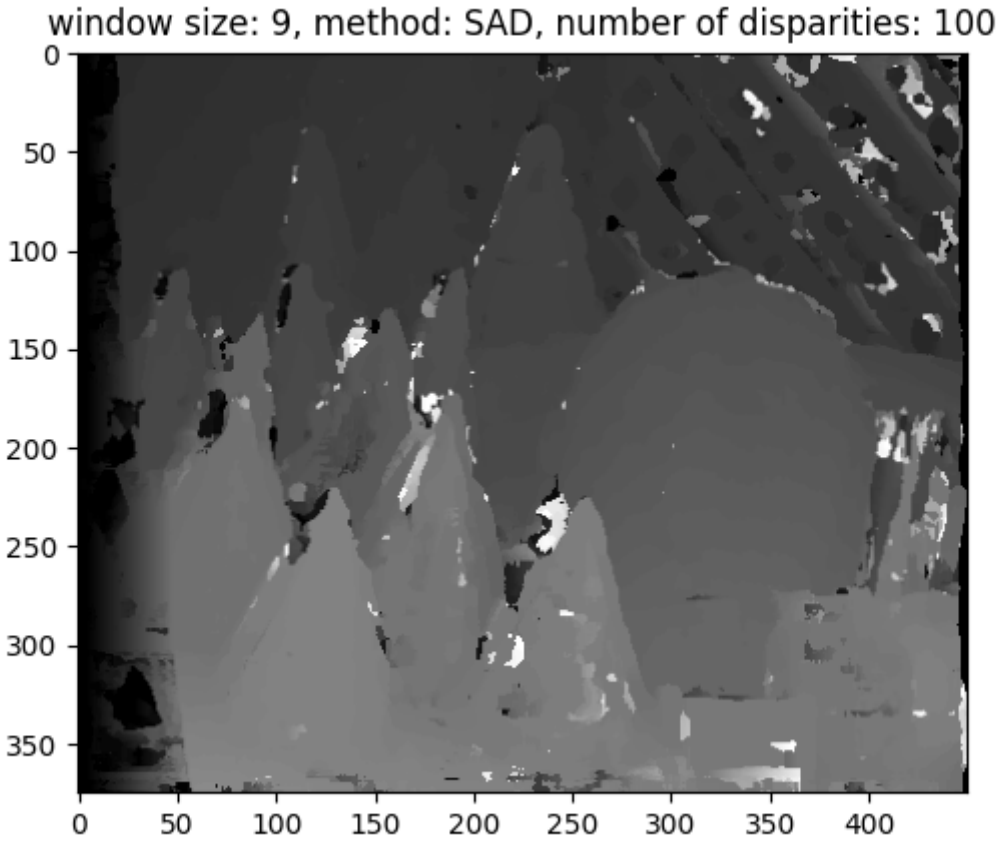

window size: 5, method: SSD, number of disparities: 100
0
50
100
150
200
250
300
350
0
50
100
150
200
250
300
350
400

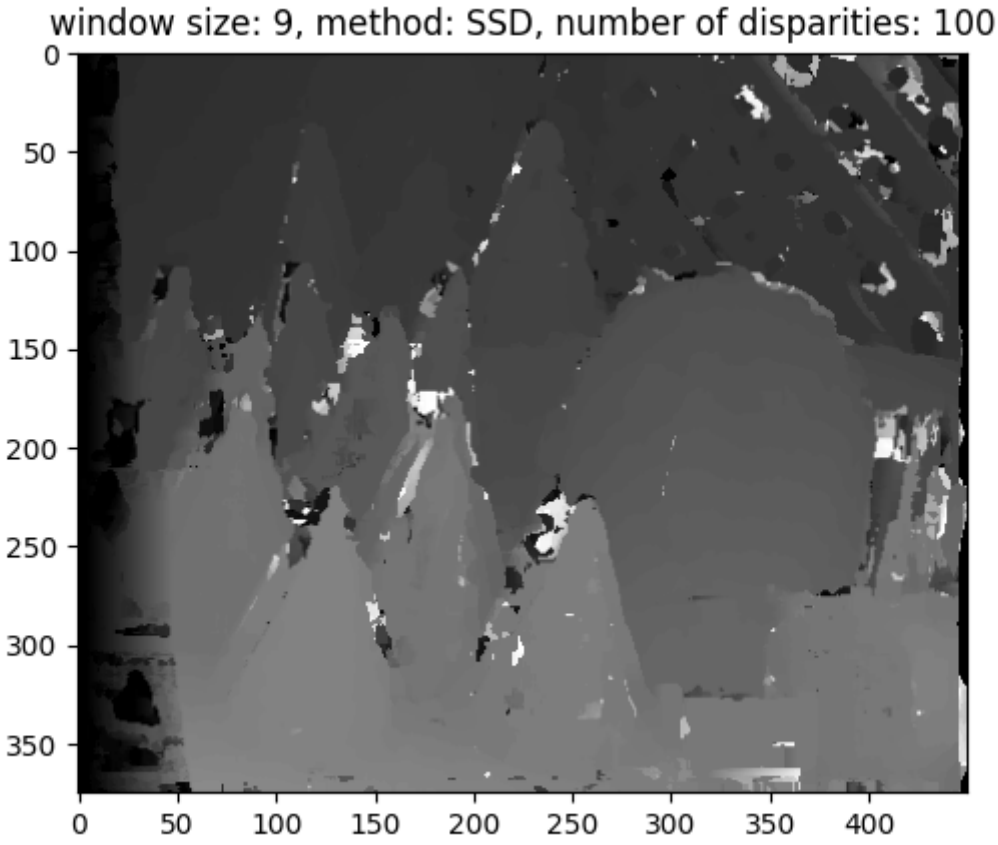

window size: 5, method: SAD, number of disparities: 150
0
50
100
150
200
250
300
350
0
50
100
150
200
250
300
350
400

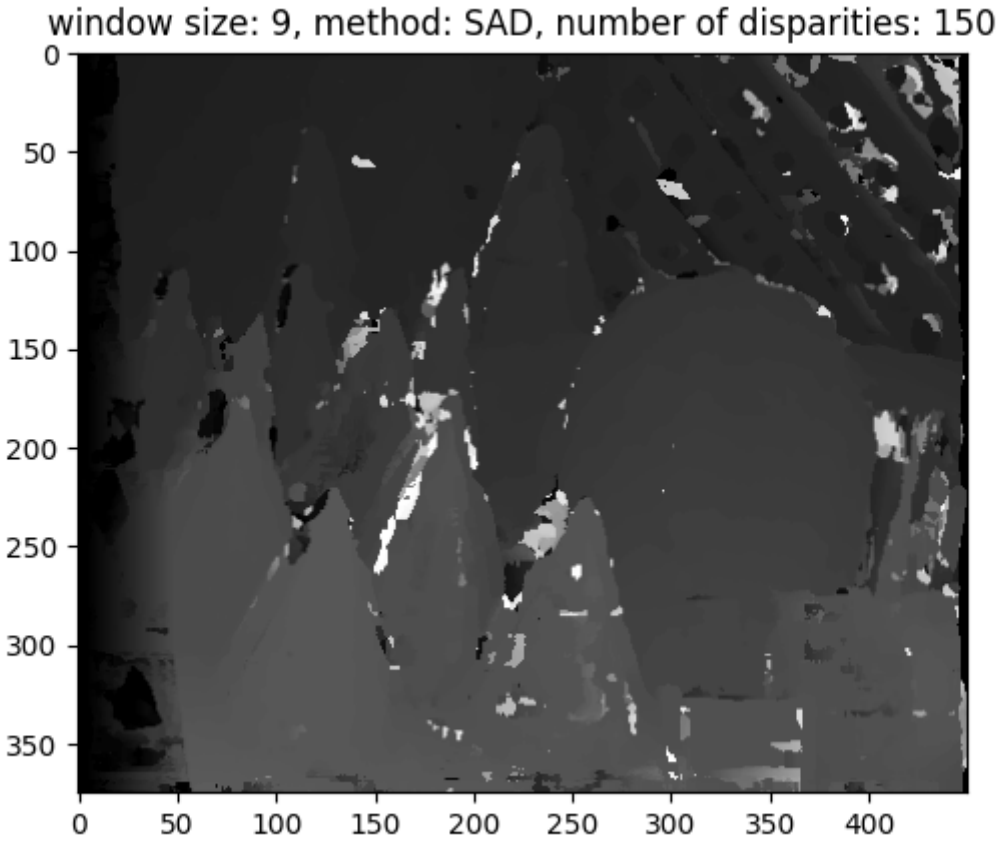

window size: 5, method: SSD, number of disparities: 150
0
50
100
150
200
250
300
350
0
50
100
150
200
250
300
350
400

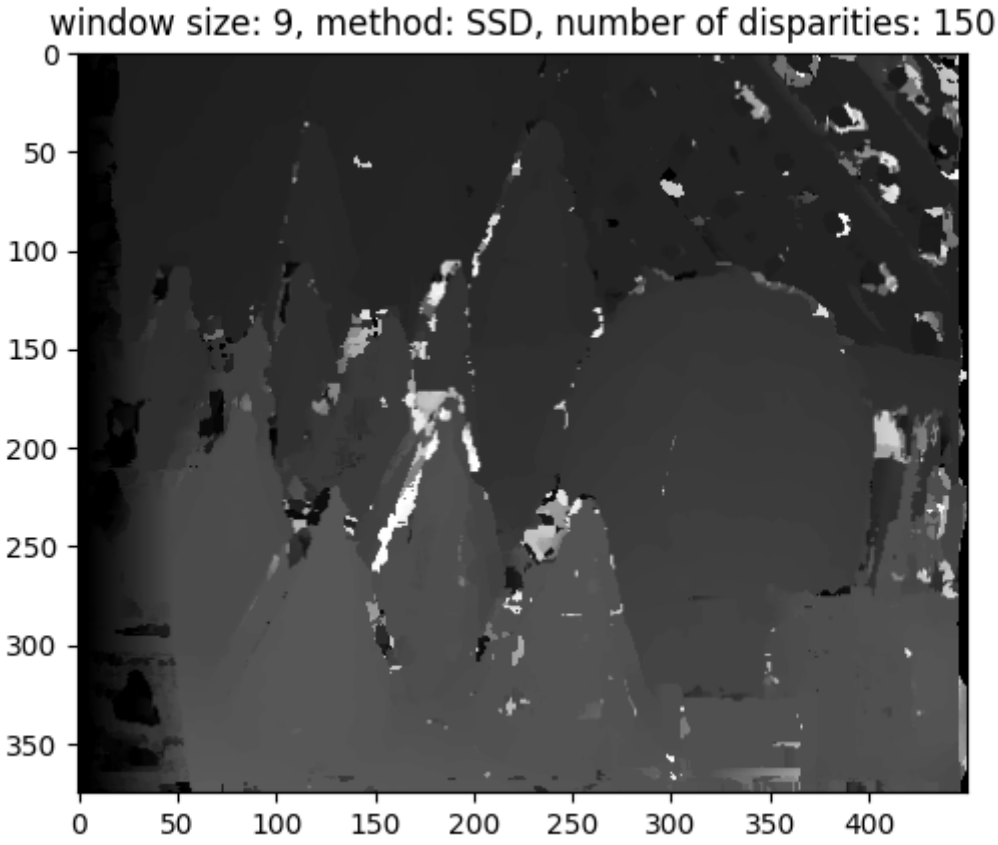

window size: 9, method: SAD, number of disparities: 50
0
50
100
150
200
250
300
350
0
50
100
150
200
250
300
350
400


window size: 9, method: SSD, number of disparities: 50
0
50
100
150
200
250
300
350
0
50
100
150
200
250
300
350
400

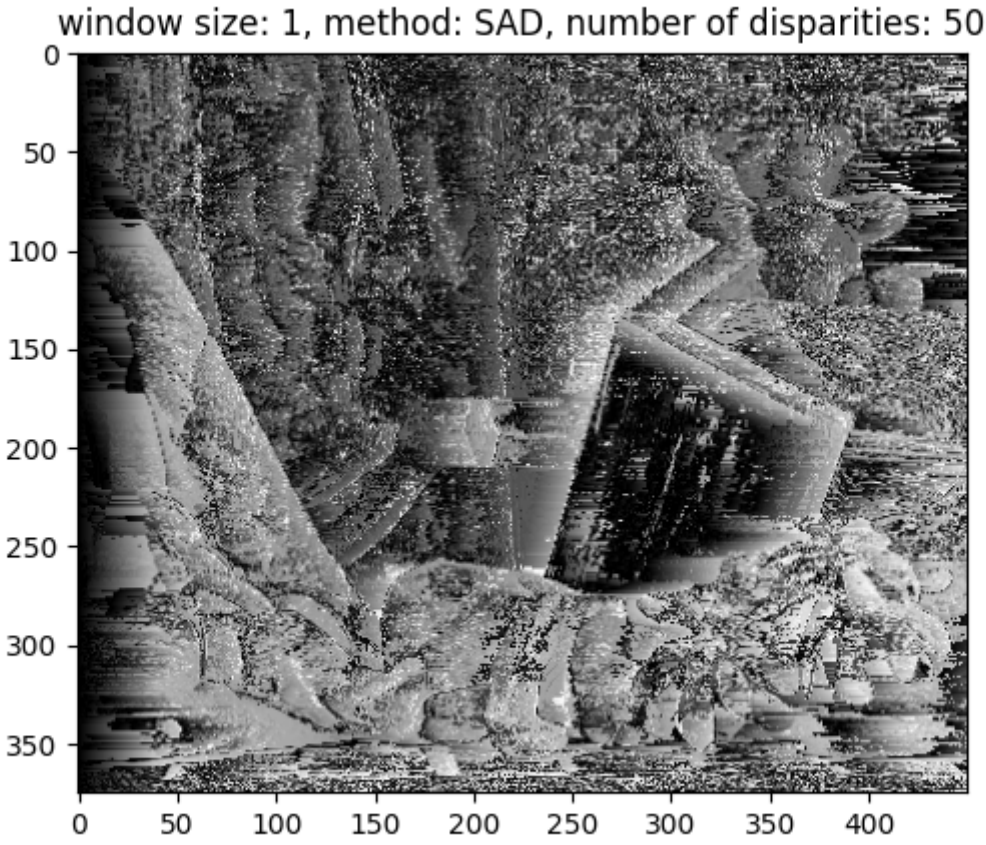

window size: 9, method: SAD, number of disparities: 100
0
50
100
150
200
250
300
350
0
50
100
150
200
250
300
350
400

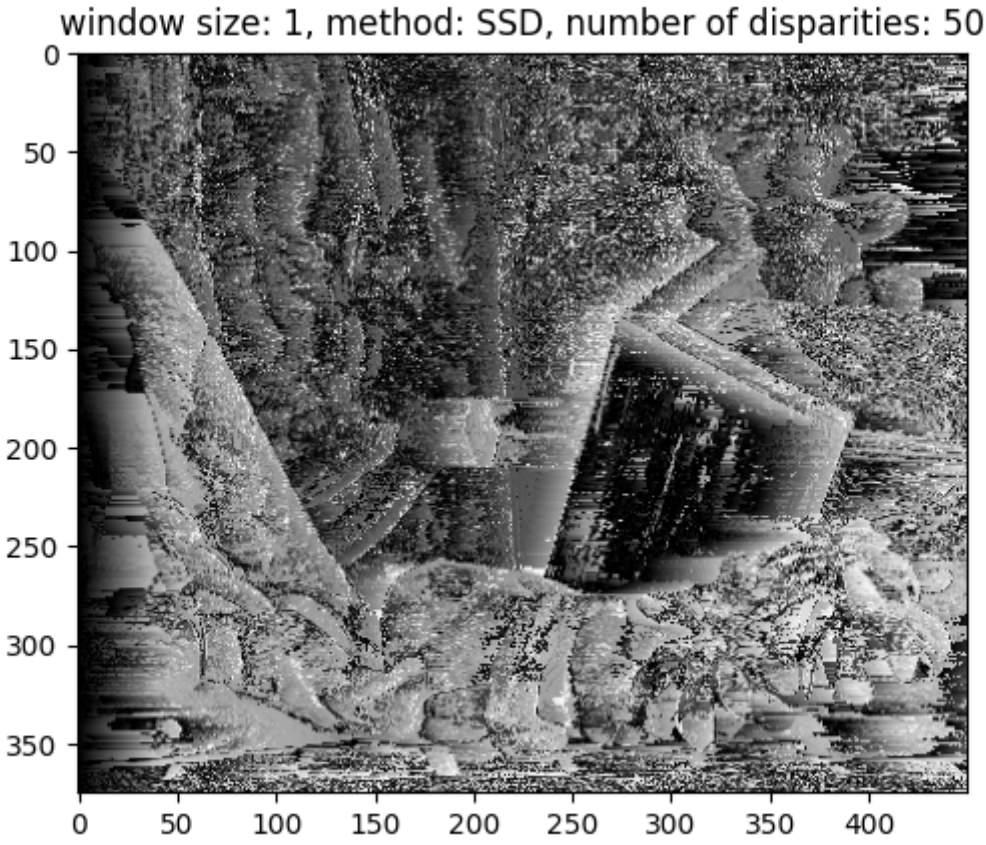

window size: 9, method: SSD, number of disparities: 100
0
50
100
150
200
250
300
350
0
50
100
150
200
250
300
350
400

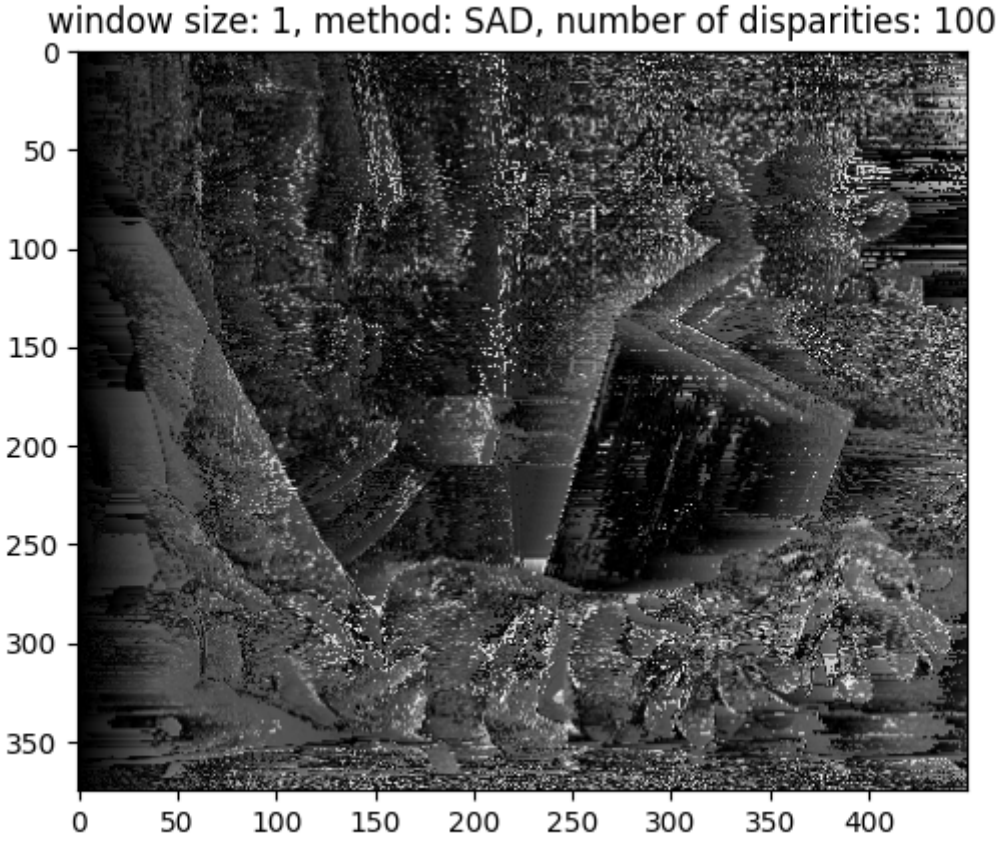

window size: 9, method: SAD, number of disparities: 150
0
50
100
150
200
250
300
350
0
50
100
150
200
250
300
350
400

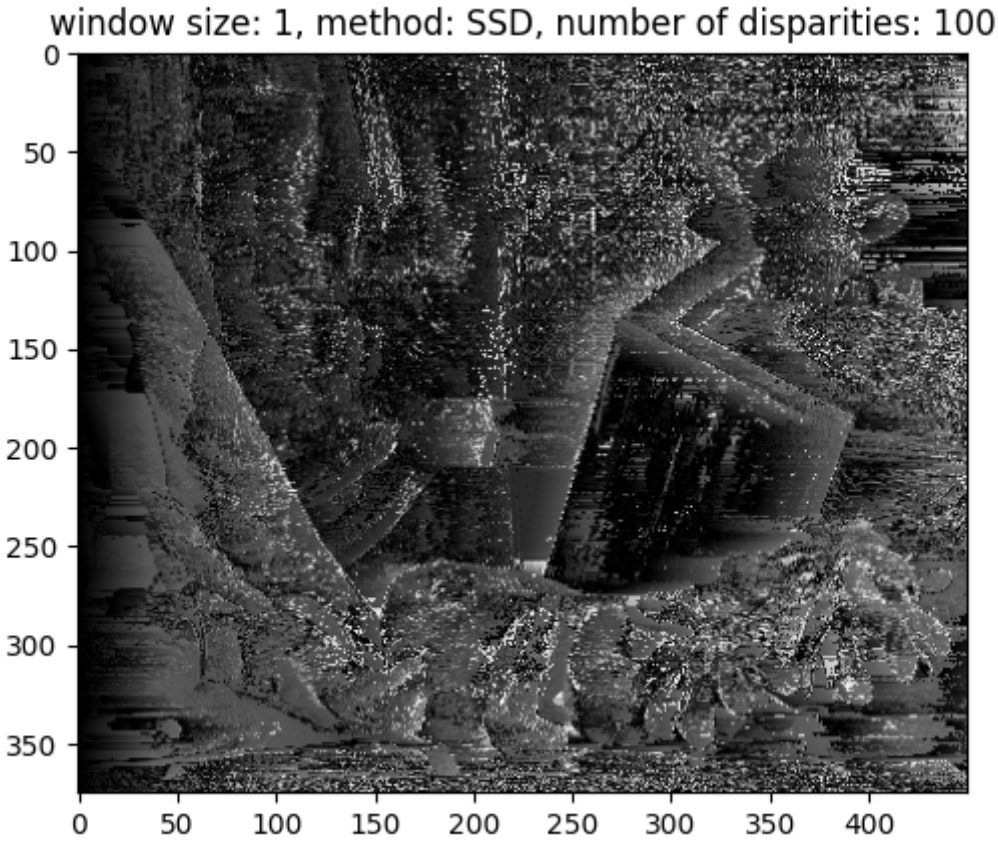

window size: 9, method: SSD, number of disparities: 150
0
50
100
150
200
250
300
350
0
50
100
150
200
250
300
350
400

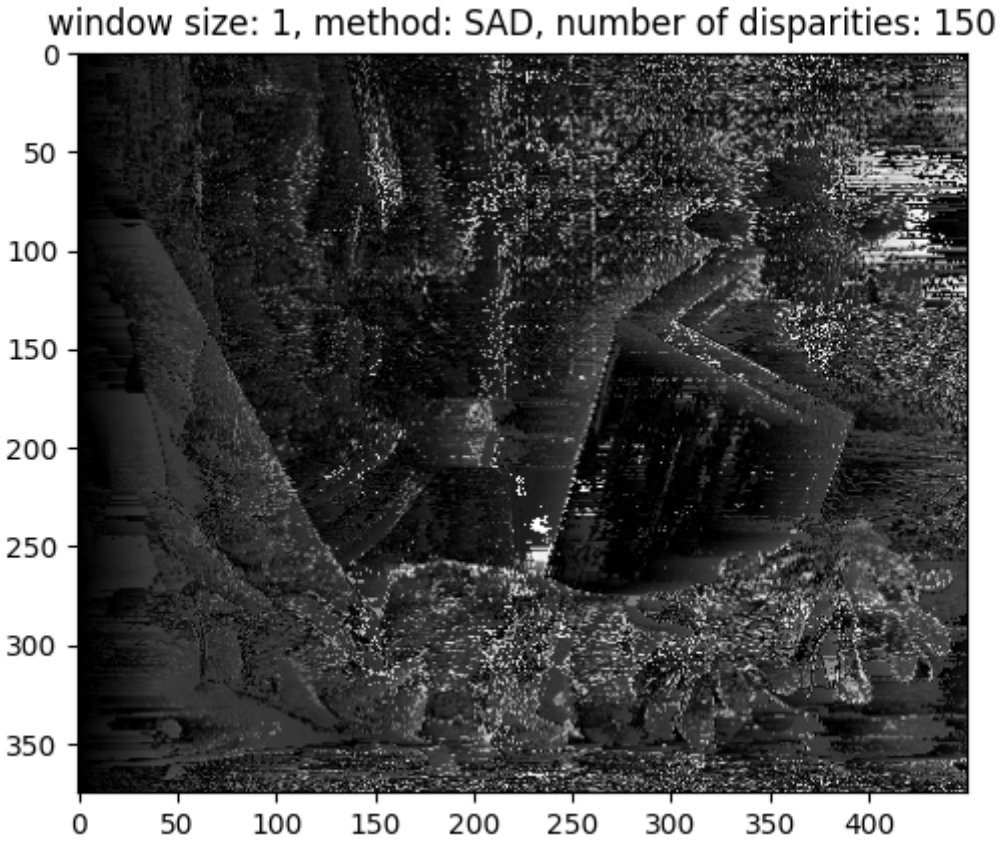
Third Image

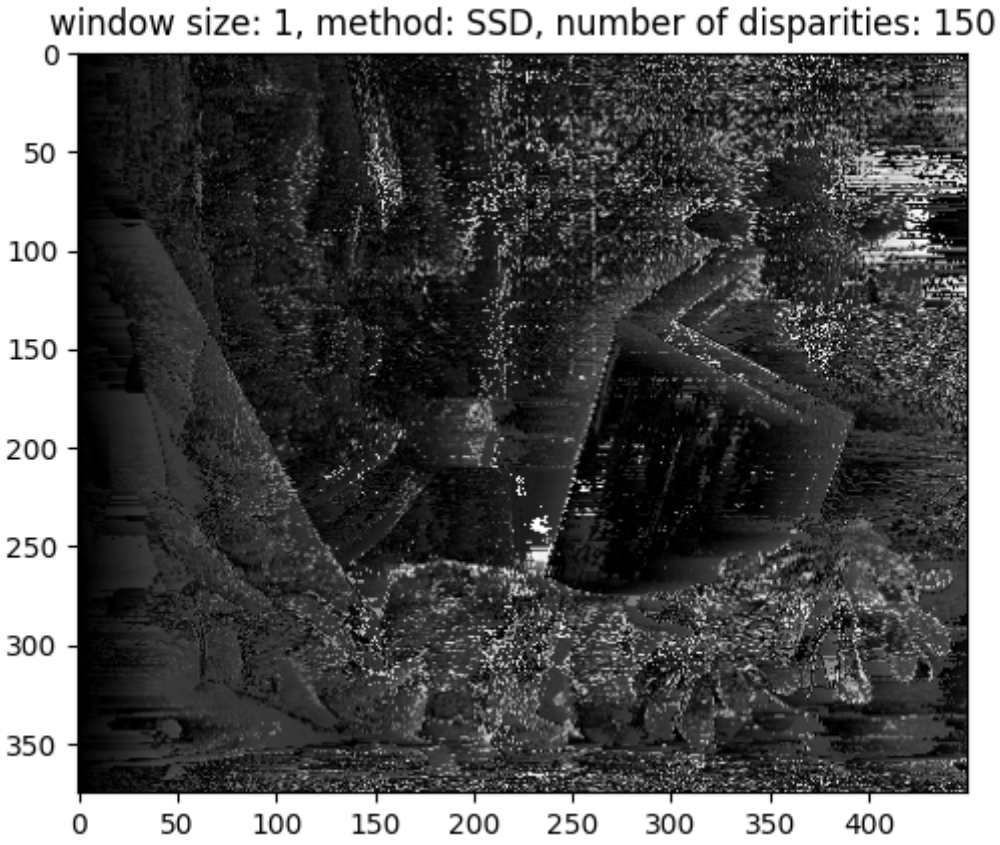

window size: 1, method: SAD, number of disparities: 50
0
50
100
150
200
250
300
350
0
50
100
150
200
250
300
350
400

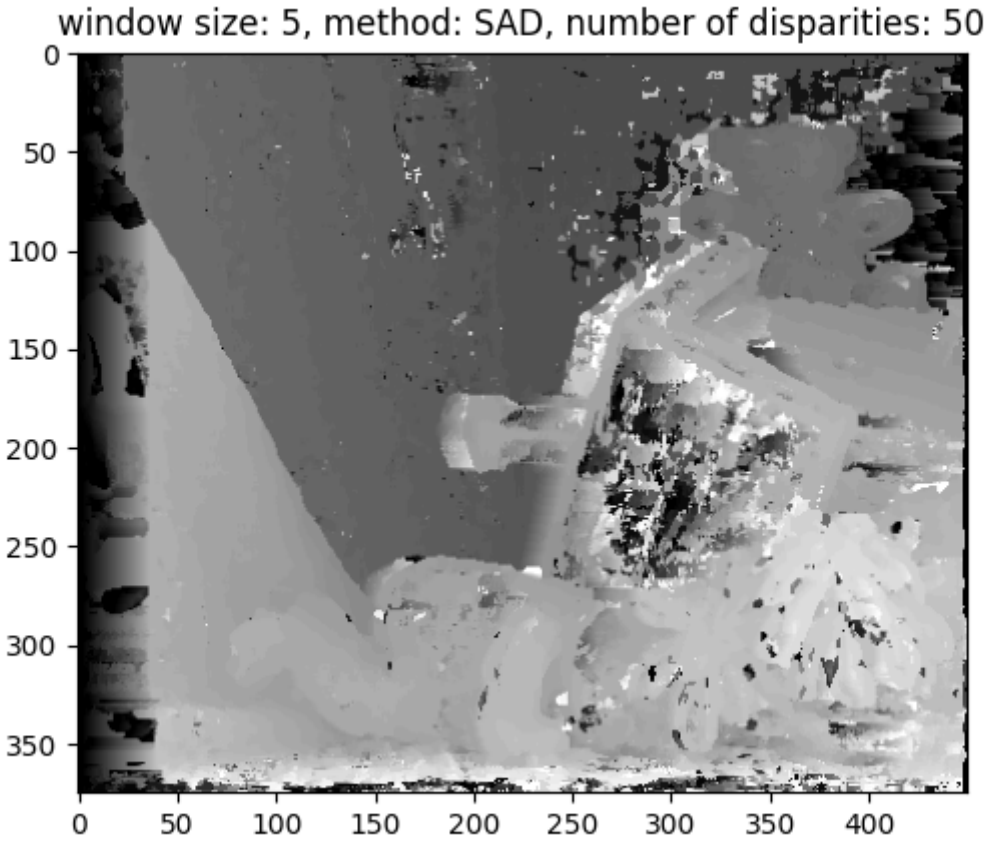

window size: 1, method: SSD, number of disparities: 50
0
50
100
150
200
250
300
350
0
50
100
150
200
250
300
350
400

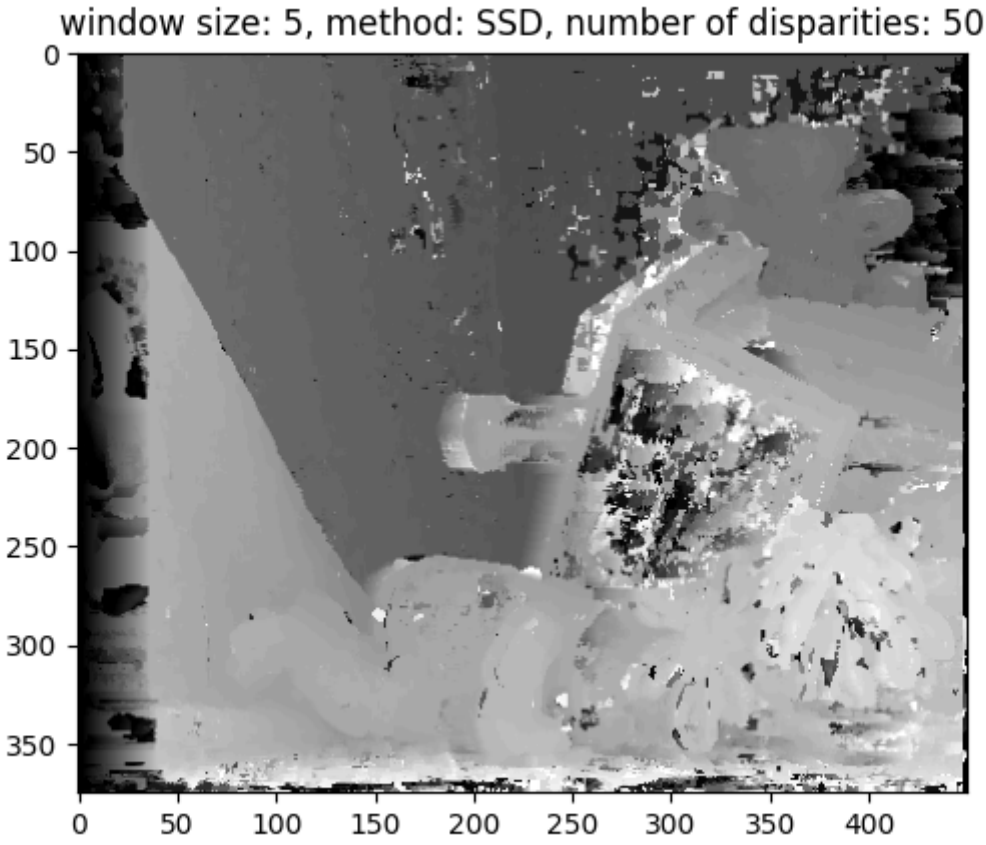

window size: 1, method: SAD, number of disparities: 100
0
50
100
150
200
250
300
350
0
50
100
150
200
250
300
350
400

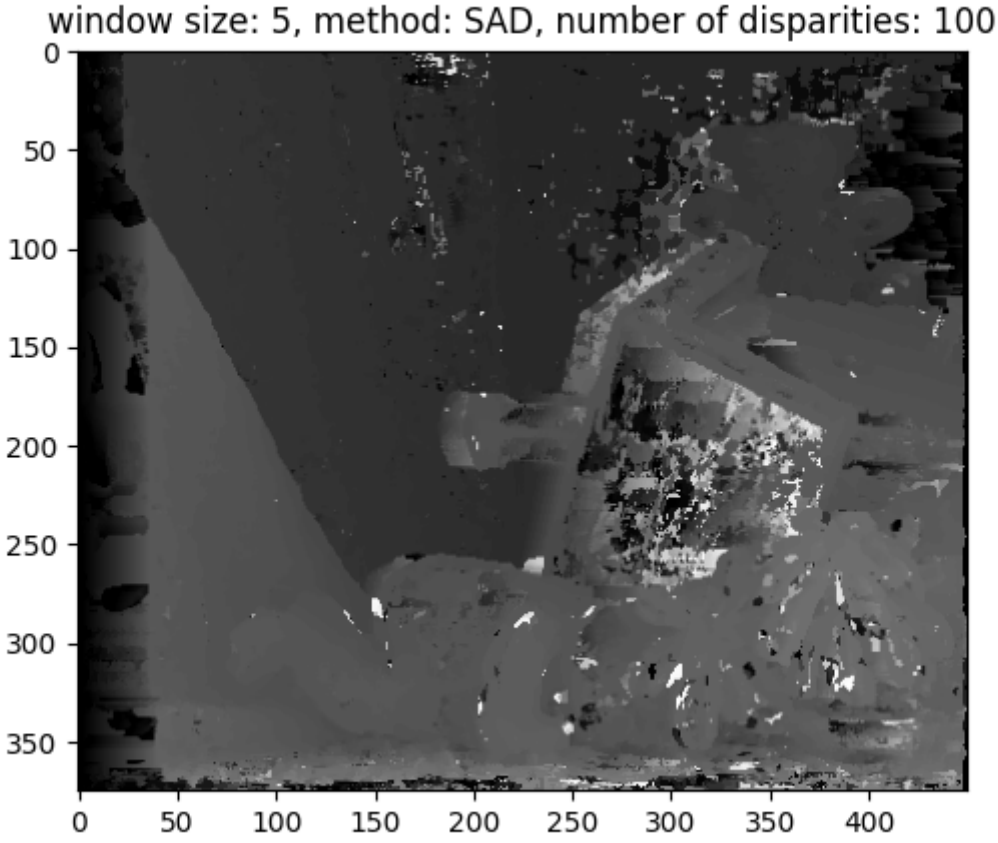

window size: 1, method: SSD, number of disparities: 100
0
50
100
150
200
250
300
350
0
50
100
150
200
250
300
350
400

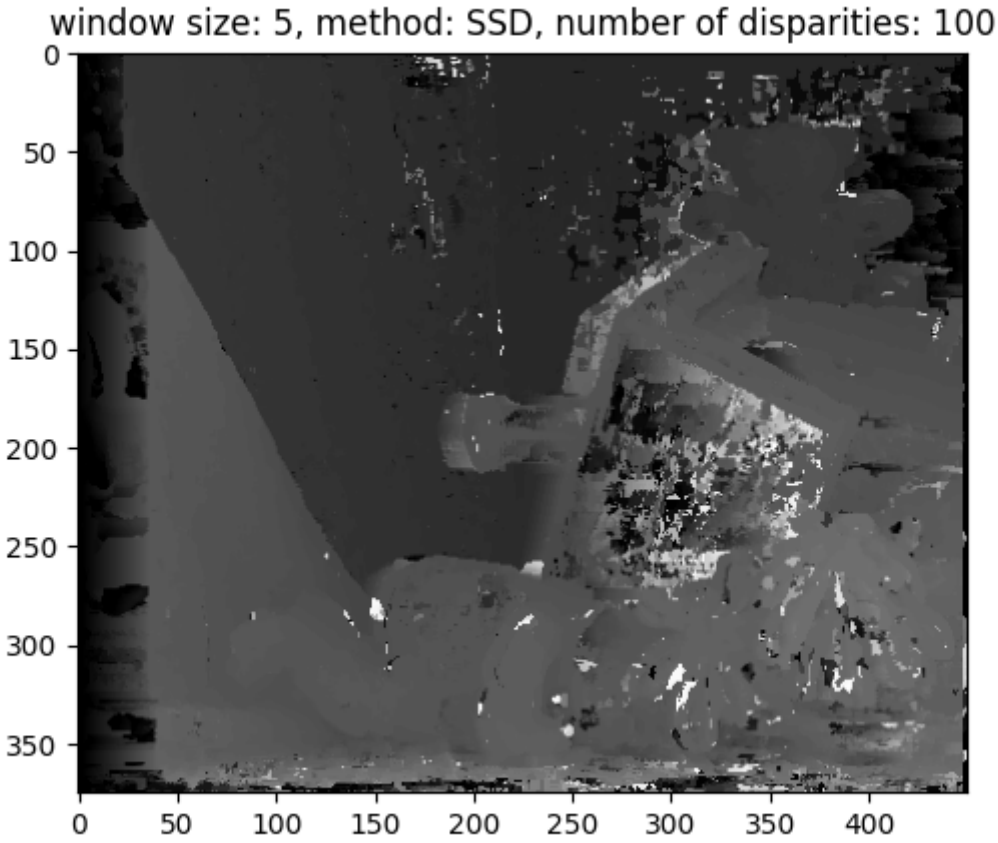

window size: 1, method: SAD, number of disparities: 150
0
50
100
150
200
250
300
350
0
50
100
150
200
250
300
350
400

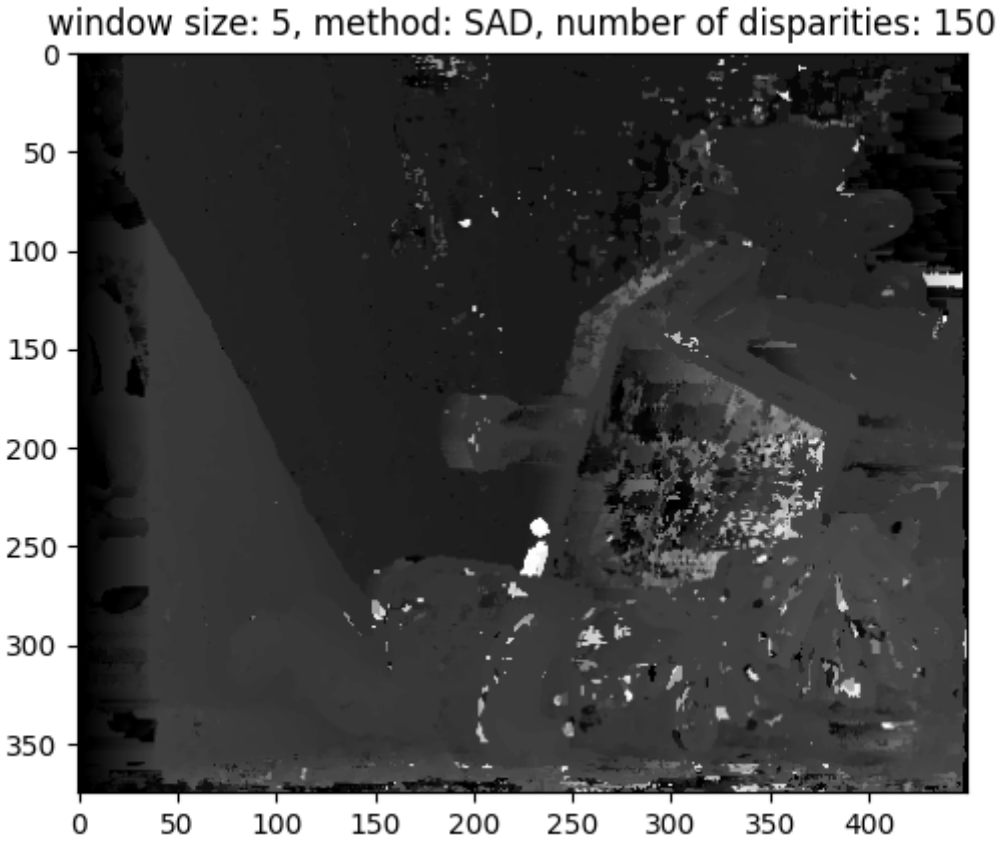

window size: 1, method: SSD, number of disparities: 150
0
50
100
150
200
250
300
350
0
50
100
150
200
250
300
350
400

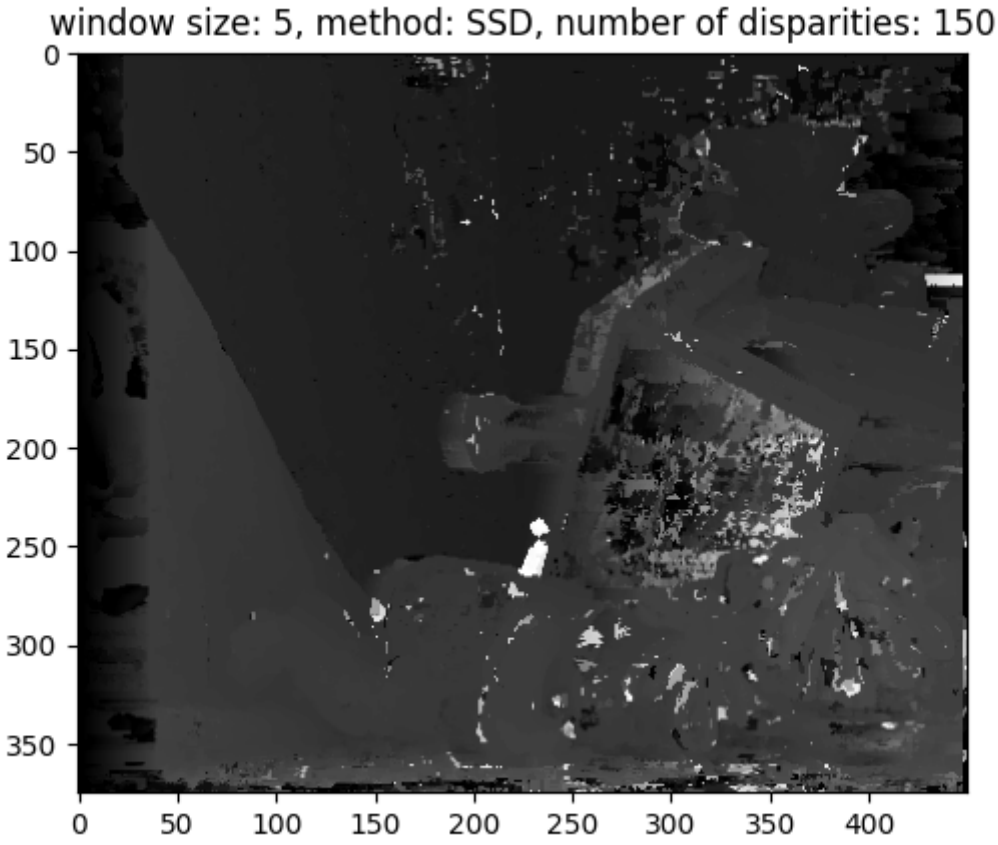

window size: 5, method: SAD, number of disparities: 50
0
50
100
150
200
250
300
350
0
50
100
150
200
250
300
350
400

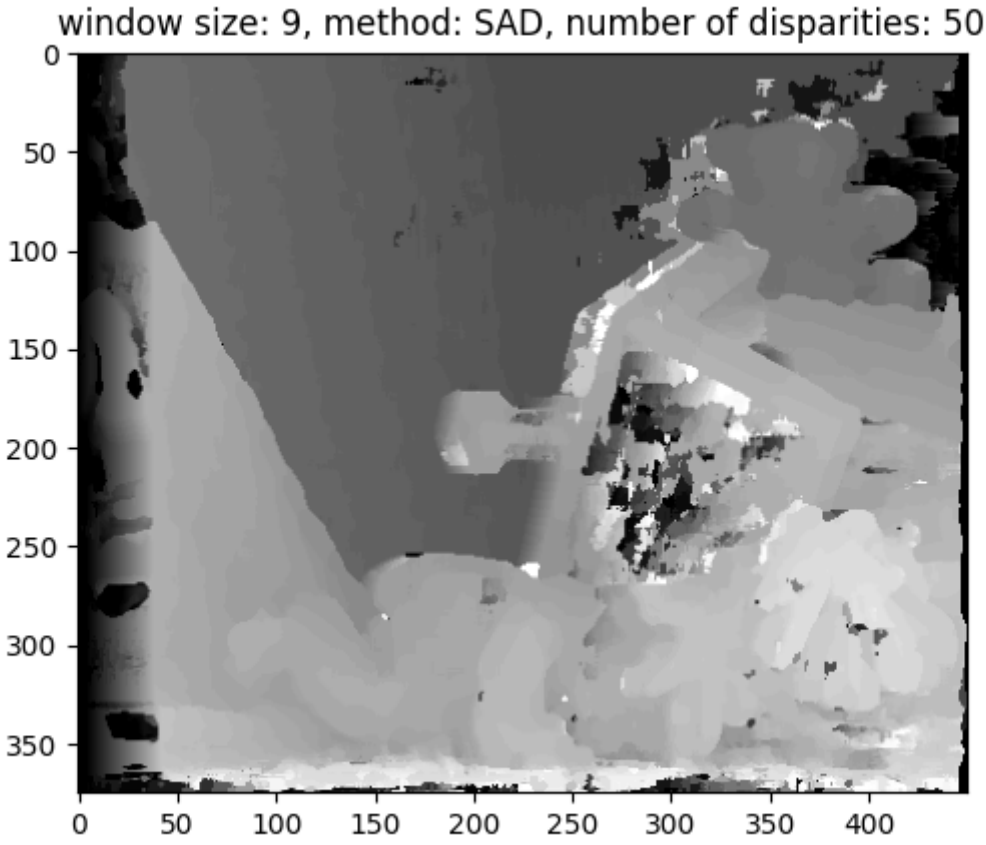

window size: 5, method: SSD, number of disparities: 50
0
50
100
150
200
250
300
350
0
50
100
150
200
250
300
350
400

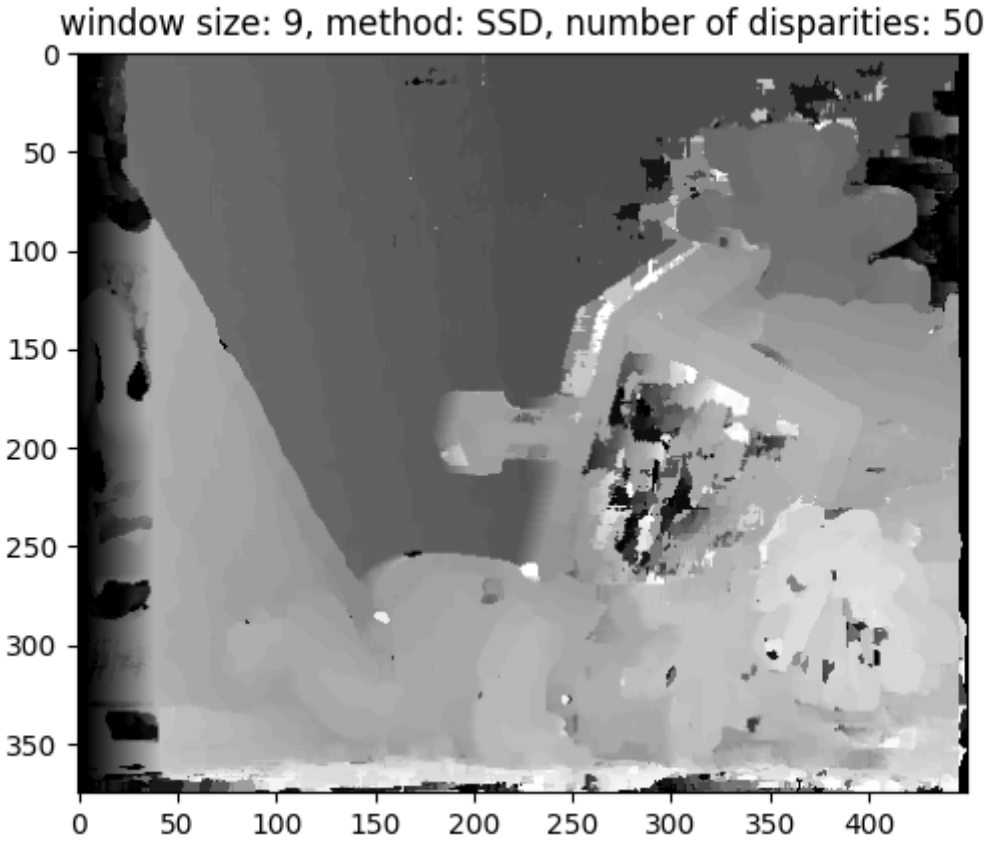

window size: 5, method: SAD, number of disparities: 100
0
50
100
150
200
250
300
350
0
50
100
150
200
250
300
350
400

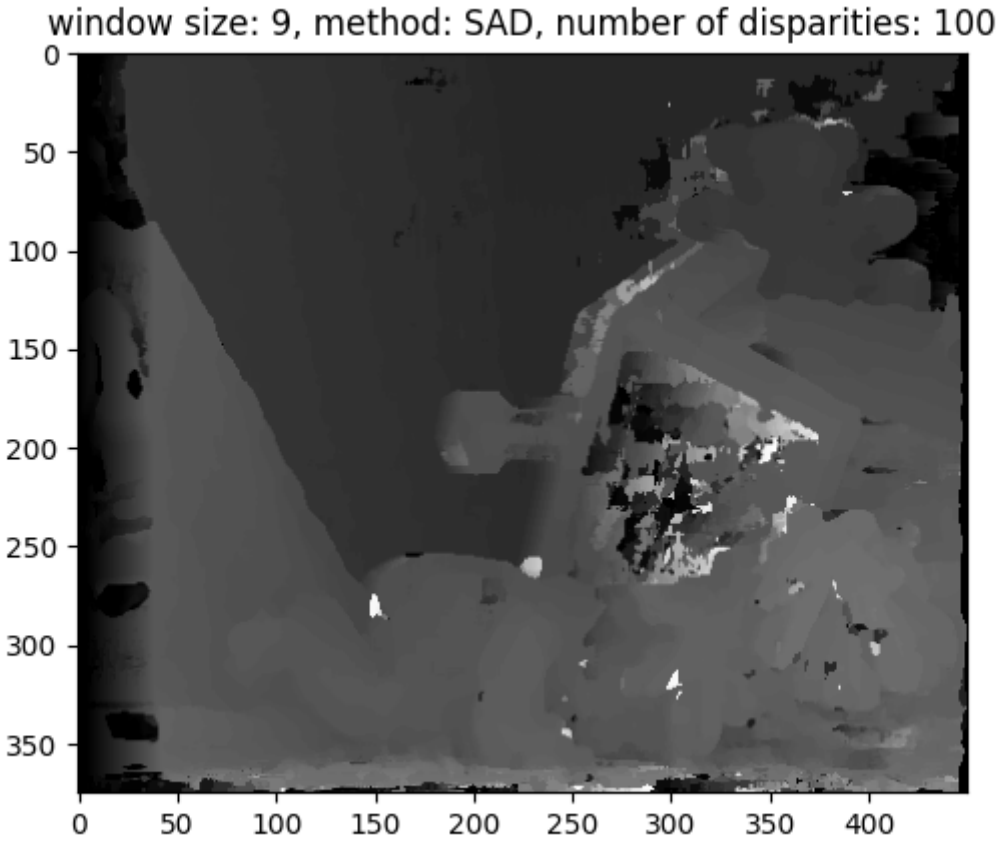

window size: 5, method: SSD, number of disparities: 100
0
50
100
150
200
250
300
350
0
50
100
150
200
250
300
350
400

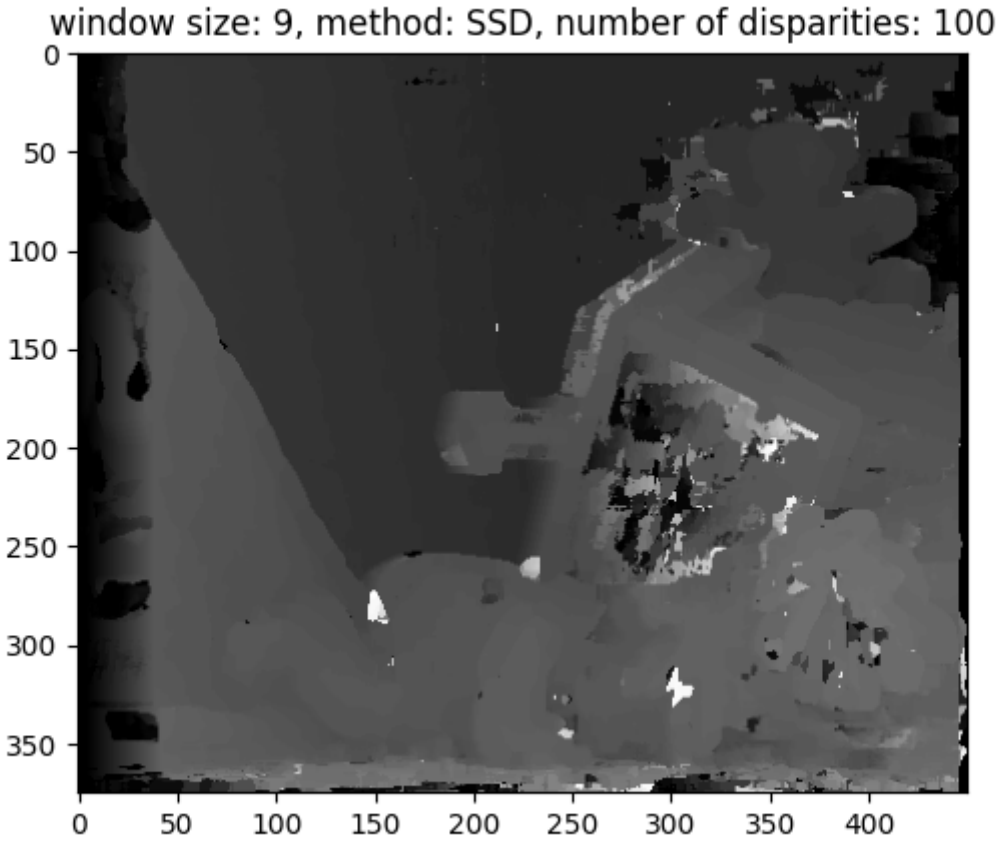

window size: 5, method: SAD, number of disparities: 150
0
50
100
150
200
250
300
350
0
50
100
150
200
250
300
350
400

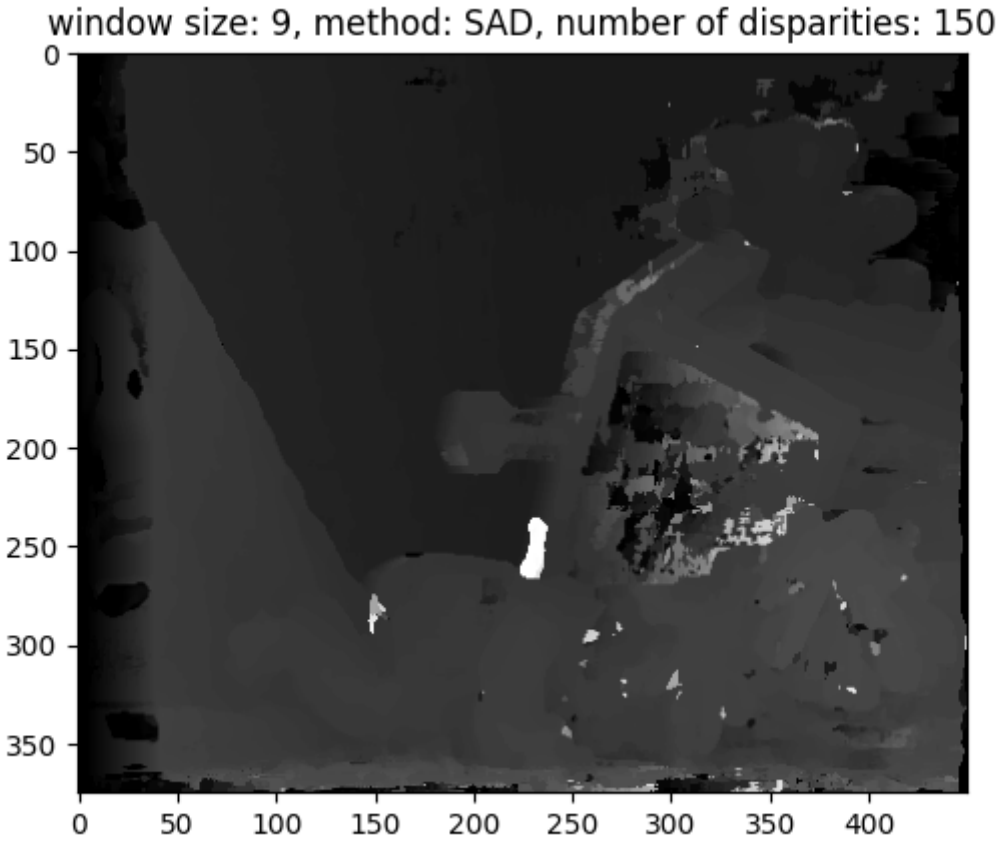

window size: 5, method: SSD, number of disparities: 150
0
50
100
150
200
250
300
350
0
50
100
150
200
250
300
350
400

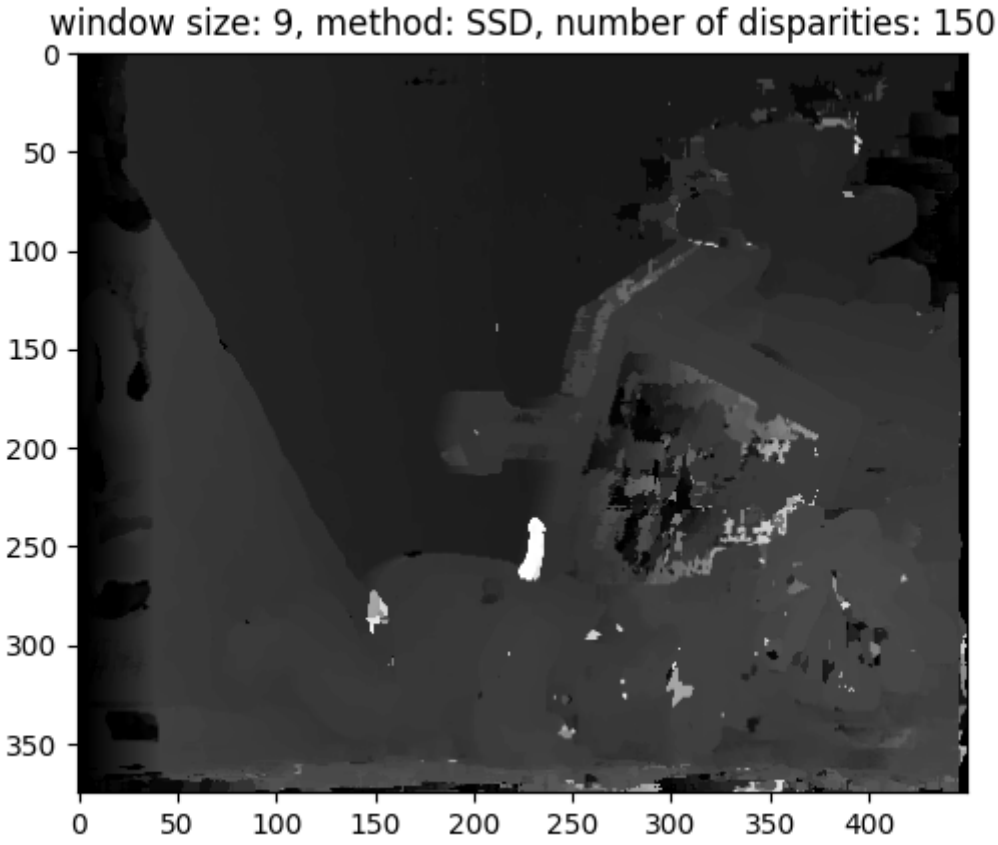

window size: 9, method: SAD, number of disparities: 50
0
50
100
150
200
250
300
350
0
50
100
150
200
250
300
350
400


window size: 9, method: SSD, number of disparities: 50
0
50
100
150
200
250
300
350
0
50
100
150
200
250
300
350
400


window size: 9, method: SAD, number of disparities: 100
0
50
100
150
200
250
300
350
0
50
100
150
200
250
300
350
400


window size: 9, method: SSD, number of disparities: 100
0
50
100
150
200
250
300
350
0
50
100
150
200
250
300
350
400


window size: 9, method: SAD, number of disparities: 150
0
50
100
150
200
250
300
350
0
50
100
150
200
250
300
350
400


window size: 9, method: SSD, number of disparities: 150
0
50
100
150
200
250
300
350
0
50
100
150
200
250
300
350
400