create()

```
static Ptr<StereoSGBM> cv::StereoSGBM::create ( int minDisparity = 0 ,
                                                int numDisparities = 16 ,
                                                int blockSize = 3 ,
                                                int P1 = 0 ,
                                                int P2 = 0 ,
                                                int disp12MaxDiff = 0 ,
                                                int preFilterCap = 0 ,
                                                int uniquenessRatio = 0 ,
                                                int speckleWindowSize = 0 ,
                                                int speckleRange = 0 ,
                                                int mode = StereoSGBM::MODE_SGBM
                                              )
```

Python:

```
cv.StereoSGBM.create(  [, minDisparity[, numDisparities[, blockSize[, P1[, P2[, disp12MaxDiff[, preFilterCap[, uniquenessRatio[, speckleWindowSize[, speckleRange[, mode]]]]]]]]]]]]
```

```
cv.StereoSGBM_create( [, minDisparity[, numDisparities[, blockSize[, P1[, P2[, disp12MaxDiff[, preFilterCap[, uniquenessRatio[, speckleWindowSize[, speckleRange[, mode]]]]]]]]]]]]
```

Creates **StereoSGBM** object.

Parameters

minDisparity	Minimum possible disparity value. Normally, it is zero but sometimes rectification algorithms can shift images, so this parameter needs to be adjus accordingly.
numDisparities	Maximum disparity minus minimum disparity. The value is always greater than zero. In the current implementation, this parameter must be divisib
blockSize	Matched block size. It must be an odd number >=1 . Normally, it should be somewhere in the 3..11 range.
P1	The first parameter controlling the disparity smoothness. See below.
P2	The second parameter controlling the disparity smoothness. The larger the values are, the smoother the disparity is. P1 is the penalty on the disp change by plus or minus 1 between neighbor pixels. P2 is the penalty on the disparity change by more than 1 between neighbor pixels. The algor requires P2 > P1 . See stereo_match.cpp sample where some reasonably good P1 and P2 values are shown (like 8*number_of_image_channels*blockSize*blockSize and 32*number_of_image_channels*blockSize*blockSize , respectively).
disp12MaxDiff	Maximum allowed difference (in integer pixel units) in the left-right disparity check. Set it to a non-positive value to disable the check.
preFilterCap	Truncation value for the prefiltered image pixels. The algorithm first computes x-derivative at each pixel and clips its value by [-preFilterCap, preF interval. The result values are passed to the Birchfield-Tomasi pixel cost function.
uniquenessRatio	Margin in percentage by which the best (minimum) computed cost function value should "win" the second best value to consider the found match Normally, a value within the 5-15 range is good enough.
speckleWindowSize	Maximum size of smooth disparity regions to consider their noise speckles and invalidate. Set it to 0 to disable speckle filtering. Otherwise, set it somewhere in the 50-200 range.
speckleRange	Maximum disparity variation within each connected component. If you do speckle filtering, set the parameter to a positive value, it will be implicitly by 16. Normally, 1 or 2 is good enough.
mode	Set it to **StereoSGBM::MODE_HH** to run the full-scale two-pass dynamic programming algorithm. It will consume O(W*H*numDisparities) bytes, large for 640x480 stereo and huge for HD-size pictures. By default, it is set to false .

The first constructor initializes **StereoSGBM** with all the default parameters. So, you only have to set StereoSGBM::numDisparities at minimum. The second constructor enabl set each parameter to a custom value.