**Third party libraries information**

**CV2**

**Object detection using sift/surf:**

**\* General**

**1.** Introduction to SURF (Speeded-Up Robust Features) <http://docs.opencv.org/trunk/df/dd2/tutorial_py_surf_intro.html>

**2.** Introduction to SIFT (Scale-Invariant Feature Transform)

<http://docs.opencv.org/trunk/da/df5/tutorial_py_sift_intro.html>

**3.** Various Feature Matching techniques

[**http://docs.opencv.org/3.0-**](http://docs.opencv.org/3.0-) [**beta/doc/py\_tutorials/py\_feature2d/py\_matcher/py\_matcher.html**](http://docs.opencv.org/3.0-beta/doc/py_tutorials/py_feature2d/py_matcher/py_matcher.html)

**\* Specific**

**4.** cv2.getPerspectiveTransform(src, dst) → retval

Calculates a perspective transform from four pairs of the corresponding points.

|  |  |
| --- | --- |
| Parameters: | * Src – Coordinates of quadrangle vertices in the source image. * Dst – Coordinates of the corresponding quadrangle vertices in the destination image. |

Link: <http://docs.opencv.org/2.4/modules/imgproc/doc/geometric_transformations.html>

**5. cv2.warpPerspective**

  cv2.**warpPerspective**(src, M, dsize[, dst[, flags[, borderMode[, borderValue]]]]) → dst

|  |  |
| --- | --- |
| Parameters: | * Src – input image. * Dst – output image that has the size dsize and the same type as src. * M –  transformation matrix. * Dsize – size of the output image. * Flags – combination of interpolation methods (INTER\_LINEAR or INTER\_NEAREST) and the optional flag WARP\_INVERSE\_MAP, that sets M as the inverse transformation (  ). * borderMode–pixel extrapolation method (BORDER\_CONSTANT or BORDER\_REPLICATE). * BorderValue – value used in case of a constant border; by default, it equals 0. |

Link:

http://docs.opencv.org/2.4/modules/imgproc/doc/geometric\_transformations.html

**-----------------------------------------------------------------------------------------------**

**four\_point\_transform** and **order points:**

[**http://www.pyimagesearch.com/2014/08/25/4-point-opencv-**](http://www.pyimagesearch.com/2014/08/25/4-point-opencv-) [**getperspective-transform-example/**](http://www.pyimagesearch.com/2014/08/25/4-point-opencv-getperspective-transform-example/)

**-----------------------------------------------------------------------------------------------**

**KeyPoint object class (**Kp1):

<http://docs.opencv.org/trunk/d2/d29/classcv_1_1KeyPoint.html>

**gaussian\_gradient\_magnitude (Scipy)**

scipy.ndimage.gaussian\_gradient\_magnitude(*input*, *sigma*, *output=None*, *mode='reflect'*, *cval=0.0*, *\*\*kwargs*)

|  |  |
| --- | --- |
| Parameters: | **Input** *: array\_like*  Input array to filter.  **Sigma** *: scalar or sequence of scalars*  The standard deviations of the Gaussian filter are given for each axis as a sequence, or as a single number, in which case it is equal for all axes..  **output** *: array, optional*  The *output* parameter passes an array in which to store the filter output. Output array should have different name as compared to input array to avoid aliasing errors.  **Mode** *: str or sequence, optional*  The *mode* parameter determines how the array borders are handled. Valid modes are {‘reflect’, ‘constant’, ‘nearest’, ‘mirror’, ‘wrap’}. *Cval* is the value used when mode is equal to ‘constant’. A list of modes with length equal to the number of axes can be provided to specify different modes for different axes. Default is ‘reflect’  **cval** *: scalar, optional*  Value to fill past edges of input if *mode* is ‘constant’. Default is 0.0  **Extra keyword arguments will be passed to gaussian\_filter().** |
| Returns: | **gaussian\_gradient\_magnitude** *: ndarray*  Filtered array. Has the same shape as *input*. |

Link : https://docs.scipy.org/doc/scipy/reference/generated/scipy.ndimage.gaussian\_gradient\_magnitude.html

# **scipy.ndimage.label**

scipy.ndimage.label(*input*,*structure=None*,*output=None*)

Label features in an array.

|  |  |
| --- | --- |
| Parameters: | **Input** *: array\_like*  An array-like object to be labeled. Any non-zero values in *input* are counted as features and zero values are considered the background.  **Structure** *: array\_like, optional*  A structuring element that defines feature connections. *Structure* must be symmetric. If no structuring element is provided, one is automatically generated with a squared connectivity equal to one. That is, for a 2-D *input* array, the default structuring element is:  [[0,1,0],  [1,1,1],  [0,1,0]]  **output** *: (None, data-type, array\_like), optional*  If *output* is a data type, it specifies the type of the resulting labeled feature array If *output* is an array-like object, then *output* will be updated with the labeled features from this function. This function can operate in-place, by passing output=input. Note that the output must be able to store the largest label, or this function will raise an Exception. |
| Returns: | **Label** *: ndarray or int*  An integer ndarray where each unique feature in *input* has a unique label in the returned array.  **num\_features** *: int*  How many objects were found.  If *output* is None, this function returns a tuple of (*labeled\_array*, *num\_features*).  If *output* is a ndarray, then it will be updated with values in *labeled\_array* and only *num\_features* will be returned by this function. |

Link :

https://docs.scipy.org/doc/scipy-0.19.0/reference/generated/scipy.ndimage.label.html

# **scipy.ndimage.measurements.find\_objects**

scipy.ndimage.measurements.find\_objects(*input*, *max\_label=0*)

Find objects in a labeled array.

|  |  |
| --- | --- |
| Parameters: | **Input** *: ndarray of ints*  Array containing objects defined by different labels. Labels with value 0 are ignored.  **max\_label** *: int, optional*  Maximum label to be searched for in *input*. If max\_label is not given, the positions of all objects are returned. |
| Returns: | **object\_slices** *: list of tuples*  A list of tuples, with each tuple containing N slices (with N the dimension of the input array). Slices correspond to the minimal parallelepiped that contains the object. If a number is missing, None is returned instead of a slice. |

Link:

<https://docs.scipy.org/doc/scipy-0.16.1/reference/generated/scipy.ndimage.measurements.find_objects.html>