UNIVERSITÉ Concordia UNIVERSITY
FACULTY OF ENGINEERING AND COMPUTER SCIENCE
Department of Computer Science and Software Engineering

COMP498G/691G COMPUTER VISION

TUTORIAL 5
Stereo Matching

Copyright © Xichen Zhou

Tutorial Overview

- Camera calibration (review)
- Stereo matching

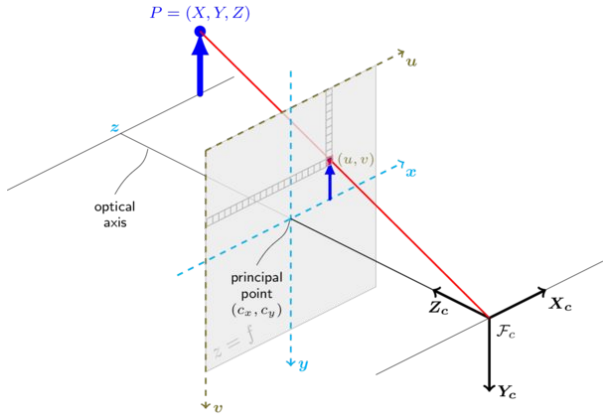
Copyright © Xichen Zhou

OpenCV Camera Model

OpenCV camera calibration is based on a pinhole camera model.

$$s\, m' = A[R|t]M'$$

$$s \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$



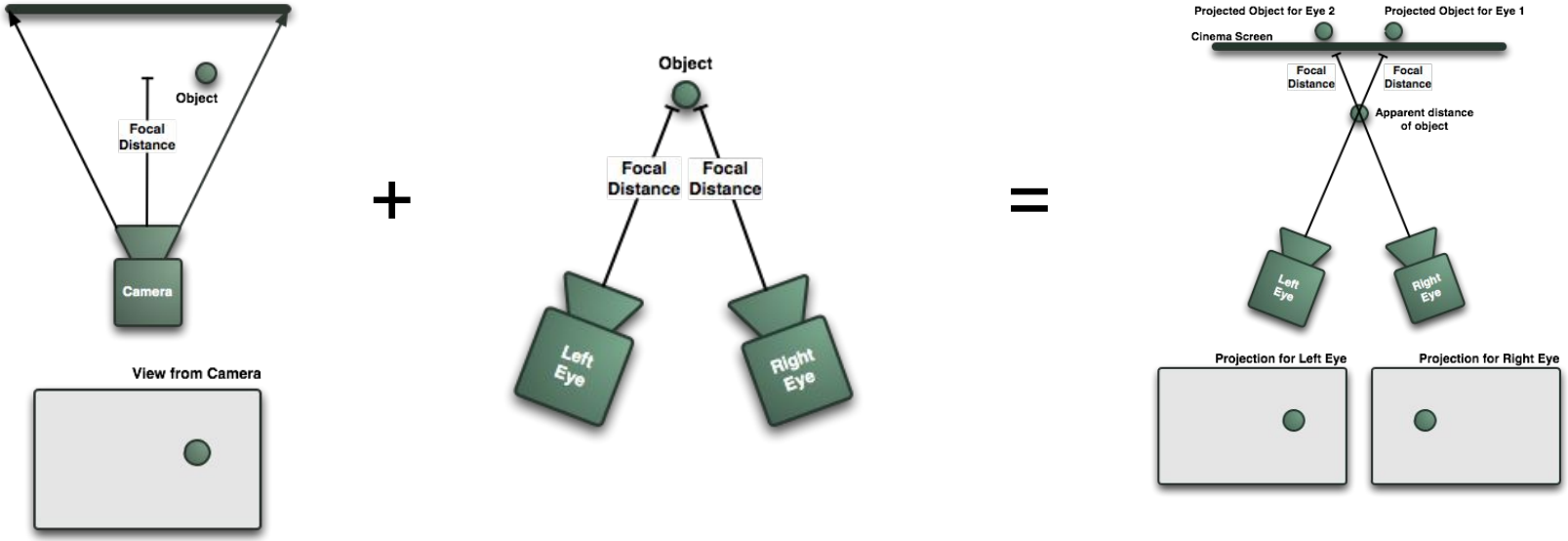
camera model

http://docs.opencv.org/3.1.0/dc/dbb/tutorial_py_calibration.html

Copyright © Xichen Zhou

OpenCV Camera Model

What if we have two camera



http://taishimizu.com/193/3D-Considered-Harmful

Copyright © Xichen Zhou

OpenCV Camera Model

Real world examples

zed 2000 stereo cam

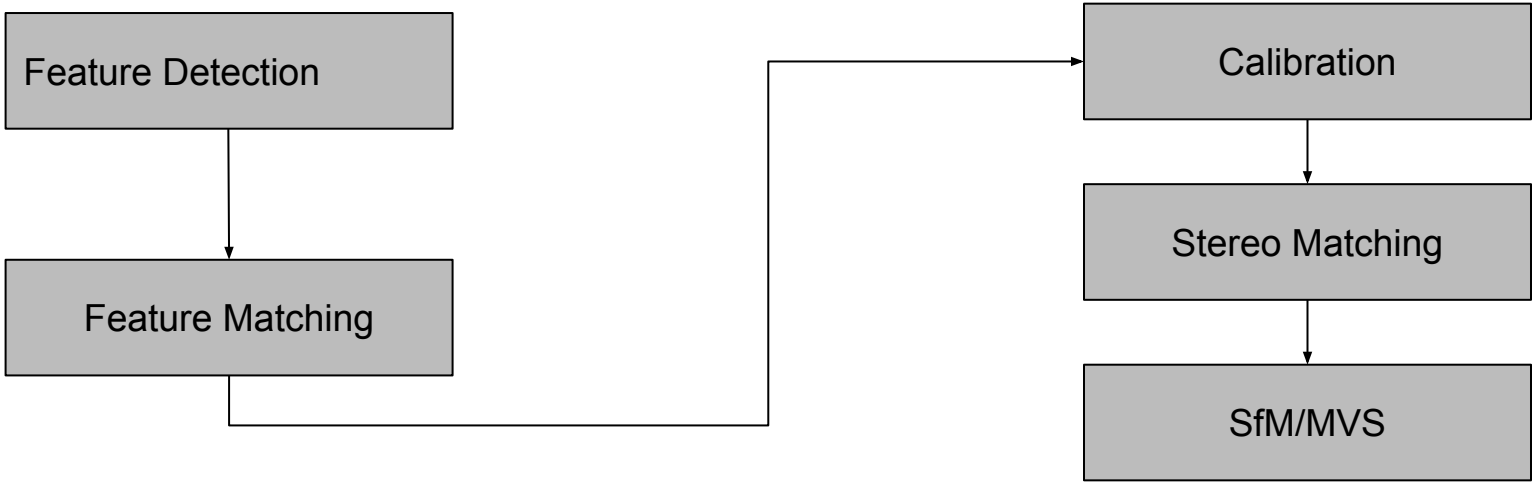
pointgrey Bumblebee2
1394a

Kinect

DUO3D MLX

Copyright © Xichen Zhou

Camera Calibration Procedure



http://docs.opencv.org/3.1.0/dc/dbb/tutorial_py_calibration.html

Copyright © Xichen Zhou

OpenCV Calib3d library

OpenCV StereoBM class

```
// Block matching stereo correspondence algorithm class StereoBM {
    enum { NORMALIZED_RESPONSE = CV_STEREO_BM_NORMALIZED_RESPONSE,
        BASIC_PRESET=CV_STEREO_BM_BASIC,
        FISH_EYE_PRESET=CV_STEREO_BM_FISH_EYE,
        NARROW_PRESET=CV_STEREO_BM_NARROW };
    StereoBM();
    // the preset is one of ..._PRESET above.
    // ndisparities is the size of disparity range,
    // in which the optimal disparity at each pixel is searched for.
    // SADWindowSize is the size of averaging window used to match pixel blocks
    //    (larger values mean better robustness to noise, but yield blurry disparity maps)
    StereoBM(int preset, int ndisparities=0, int SADWindowSize=21);
    // separate initialization function
    void init(int preset, int ndisparities=0, int SADWindowSize=21);
    // computes the disparity for the two rectified 8-bit single-channel images.
    // the disparity will be 16-bit signed (fixed-point) or 32-bit floating-point image of
the same size as left.
    void operator()( InputArray left, InputArray right, OutputArray disparity, int
disptype=CV_16S );
    Ptr<CvStereoBMState> state;
};
```

http://docs.opencv.org/3.1.0/dc/dbb/tutorial_py_calibration.html

Copyright © Xichen Zhou

OpenCV Calib3d library

StereoBM::StereoBM(int **preset**, int **ndisparities**=0, int **SADWindowSize**=21)

preset – specifies the whole set of algorithm parameters, one of:

- BASIC_PRESET - parameters suitable for general cameras
- FISH_EYE_PRESET - parameters suitable for wide-angle cameras
- NARROW_PRESET - parameters suitable for narrow-angle cameras

ndisparities – the disparity search range. For each pixel algorithm will find the best disparity from 0 (default minimum disparity) to ndisparities. The search range can then be shifted by changing the minimum disparity.

SADWindowSize – the linear size of the blocks compared by the algorithm. The size should be odd (as the block is centered at the current pixel). Larger block size implies smoother, though less accurate disparity map. Smaller block size gives more detailed disparity map, but there is higher chance for algorithm to find a wrong correspondence.

http://docs.opencv.org/3.1.0/dc/dbb/tutorial_py_calibration.html

Copyright © Xichen Zhou

OpenCV Calib3d library

void **StereoBM::operator()**(InputArray **left**, InputArray **right**, OutputArray **disparity**, int **disptype**=CV_16S)

left and **right** are single channel images
disparity – Output disparity map

http://docs.opencv.org/3.1.0/dc/dbb/tutorial_py_calibration.html

Copyright © Xichen Zhou

Concordia UNIVERSITY

OpenCV Calib3d library

Calibrate Camera with stereo image points

double **stereoCalibrate**(InputArrayOfArrays **objectPoints**, InputArrayOfArrays **imagePoints1**, InputArrayOfArrays **imagePoints2**, InputOutputArray **cameraMatrix1**, InputOutputArray **distCoeffs1**, InputOutputArray **cameraMatrix2**, InputOutputArray **distCoeffs2**, Size **imageSize**, OutputArray **R**, OutputArray **T**, OutputArray **E**, OutputArray **F**, TermCriteria **criteria**=TermCriteria(TermCriteria::COUNT+TermCriteria::EPS, 30, 1e-6), int **flags**=CALIB_FIX_INTRINSIC)

- Object points set
- image points set 1,
- image points set 2,

http://docs.opencv.org/3.1.0/dc/dbb/tutorial_py_calibration.html

Copyright © Xichen Zhou

OpenCV Calib3d library

Image Rectification Transformation

void stereoRectify(InputArray **cameraMatrix1**, InputArray **distCoeffs1**, InputArray **cameraMatrix2**, InputArray **distCoeffs2**, Size **imageSize**, InputArray **R**, InputArray **T**, OutputArray **R1**, OutputArray **R2**, OutputArray **P1**, OutputArray **P2**, OutputArray **Q**, int **flags**=CALIB_ZERO_DISPARITY, double **alpha**=-1, Size **newImageSize**=Size(), Rect* **validPixROI1**=0, Rect* **validPixROI2**=0)

bool stereoRectifyUncalibrated(InputArray **points1**, InputArray **points2**, InputArray **F**, Size **imgSize**, OutputArray **H1**, OutputArray **H2**, double **threshold**=5)

http://docs.opencv.org/3.1.0/dc/dbb/tutorial_py_calibration.html

Copyright © Xichen Zhou

Concordia UNIVERSITY

OpenCV Calib3d library

Other important functions in calib3d library

void triangulatePoints(InputArray **projMatr1**, InputArray **projMatr2**, InputArray **projPoints1**, InputArray **projPoints2**, OutputArray **points4D**)

Mat findFundamentalMat(InputArray **points1**, InputArray **points2**, int **method**=FM_RANSAC, double **param1**=3., double **param2**=0.99, OutputArray **mask**=noArray())

int estimateAffine3D(InputArray **src**, InputArray **dst**, OutputArray **out**, OutputArray **inliers**, double **ransacThreshold**=3, double **confidence**=0.99)

void reprojectImageTo3D(InputArray **disparity**, OutputArray **_3dImage**, InputArray **Q**, bool **handleMissingValues**=false, int **ddepth**=-1)

http://docs.opencv.org/3.1.0/dc/dbb/tutorial_py_calibration.html

Copyright © Xichen Zhou

CONCORDIA.CA

Copyright © Xichen Zhou