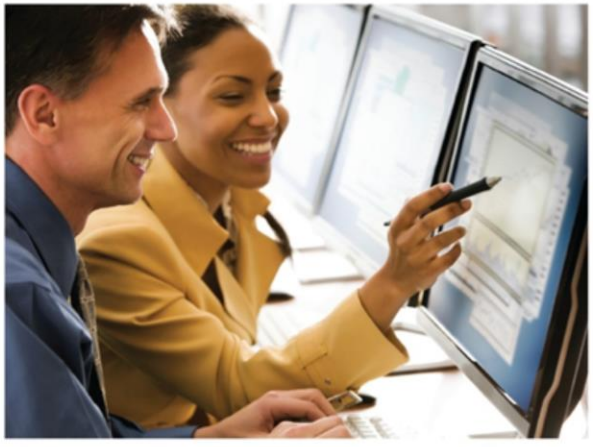
MathWorks | Training Services

Exercises: Stereo Vision

AUVSI Foundation: Computer Vision Training



MathWorks | Training Services

Vehicular Stereo Vision

Extract depth-based images using the stereo images from the perspective of a vehicle.

1. Open the starter script `vehicleDepthExtraction_start`. This already has starter code that loads a pair of stereo images. Fill in the rest of the code according to the comments in the starter code. The following points help in implementing the rest of the code.
2. Load the provided stereo parameters from the file `stereoParams.mat`.
3. Rectify the images using `rectifyStereoImages`.
4. Generate a disparity map using `disparity`. Use a `DisparityRange` of `[0 64]`.
5. Create a point cloud using `reconstructScene`.
    a. Convert from "millimeters" to "meters".
    b. Limit the range of point clouds using the provided utility function `thresholdPC` with the thresholds `[-5 5;-5 10;0 30]`.
    c. View the point clouds.
6. Try to extract the parts of the image which correspond to the range between 20 and 30 meters.
    a. Extract the z layer of the point cloud.
    b. Create a mask where pixels outside of the range are 'true'.
    c. Use `repmat` to replicate the mask in the third dimension.
    d. Use this mask on an image to set pixels which are outside of the range to 0.

Solution

Solution for vehicular stereo vision with a single frame

```
>> vehicleDepthExtraction
```

Solution for Extra 2

```
>> vehicleDepthEstimation
```

Solution for Extra 2 with a stream of frames

```
>> vehicleDepthEstimation_vid
```

Extra 1:

Use images from `CalibrationImages` folder with the Stereo Calibration app to determine stereo parameters instead of using the provided parameters.

Extra 2:

1. Use a trained cascade object detector to identify the centroid of the vehicle in front. Determine the distance to the vehicle.
2. Load the trained cascade object detector `CarDetector.xml` which has already been trained to identify vehicles.
3. Use the trained cascade object detector to identify the centroid of the vehicle directly ahead.
4. Use this centroid to determine the distance to the vehicle.
5. Try this with all images in the `RecordedImages` folder in sequence.