



В данной работе познакомимся с функциями OpenCV, предназначенными для работы со стереозрением.

Необходимый bag-файл (stereo.bag) находится в каталоге ./imageStereoWork.

По аналогии с предыдущим заданием будем использовать зависимости для пакета:

- 1) sensor_msgs
- 2) cv_bridge
- 3) roscpp
- 4) std_msgs
- 5) image_transport

Также в каталоге находится исходный файл stereosubscriber.cpp.

В качестве исследуемого алгоритма будем использовать алгоритм BM (block matching algorithm).

- 1) Создадим глобальные переменные, которые будут использоваться для динамической настройки параметров стереоизображения:

```
static int StereoNDisp = 35;  
static int StereoWinSize = 10;  
static int StereoPresent = 2;  
static int StereoTexTh = 5;
```

- 2) Для работы нам потребуются следующие объекты:

```
cv::Mat left_img_rgb;  
cv::Mat right_img_rgb;  
cv::Mat left;  
cv::Mat right;  
cv::Mat disp;  
  
cv::gpu::GpuMat d_left, d_right;  
cv::gpu::GpuMat disp_dst;  
  
cv::gpu::StereoBM_GPU bm;
```

- 3) Интерфейс настройки параметров:

```
bm.avergeTexThreshold = 35;  
bm.ndisp = 128;  
bm.preset = 2;
```

```
bm.winSize = 32;
```

```
cv::createTrackbar("NDisp ", "Stereo Control", &StereoNDisp, 2048);  
cv::createTrackbar("WinSize", "Stereo Control", &StereoWinSize, 50);  
cv::createTrackbar("Present", "Stereo Control", &StereoPresent, 1000);  
cv::createTrackbar("TexCH ", "Stereo Control", &StereoTexTh, 1000);
```

```
bm.ndisp = StereoNDisp;  
bm.winSize = StereoWinSize;  
bm.preset = StereoPresent;  
bm.avergeTexThreshold = StereoTexTh;
```

4) Подготовка изображений:

```
left_img_ptr = cv_bridge::toCvCopy(msg1,sensor_msgs::image_encodings::BGR8);  
right_img_ptr = cv_bridge::toCvCopy(msg2,sensor_msgs::image_encodings::BGR8);
```

```
left_img_rgb = left_img_ptr->image;  
right_img_rgb = right_img_ptr->image;
```

```
cv::Mat disp(left_img_rgb.size(), CV_8U);  
cv::gpu::GpuMat d_disp(left_img_rgb.size(), CV_8U);
```

```
cv::cvtColor(left_img_rgb, left, CV_BGR2GRAY);  
cv::cvtColor(right_img_rgb, right, CV_BGR2GRAY);
```

```
d_left.upload(left);  
d_right.upload(right);
```

5) Работа с BM

```
bm(d_left, d_right, d_disp);
```

6) Визуализация

Задание:

По аналогии с BM реализовать модули, использующие алгоритмы BP или CSBP.

Описание данных объектов:

http://docs.opencv.org/2.4/modules/gpu/doc/camera_calibration_and_3d_reconstruction.html