- Obtaining Depth Information from Stereo Images -

Whitepaper

ENSENSO

IDS Imaging Development Systems GmbH
Dimbacher Str. 6-8
74182 Obersulm
Tel. +49-(0)7134-96196-0
Fax +49-(0)7134-96196-99

sales@ids-imaging.de
www.ids-imaging.de

Inhalt

1 Abstract

This paper gives an overview of the main processing steps for depth perception with a stereo camera. After a description of the general techniques, we discuss the specifics of the Ensenso N10 stereo camera in combination with the NxLib stereo processing library.

2 A General Stereo Vision Configuration

Depth perception from stereo vision is based on the triangulation principle. We use two cameras with projective optics and arrange them side by side, such that their view fields overlap at the desired object distance. By taking a picture with each camera we capture the scene from two different viewpoints. This setup is illustrated in Figure 1.

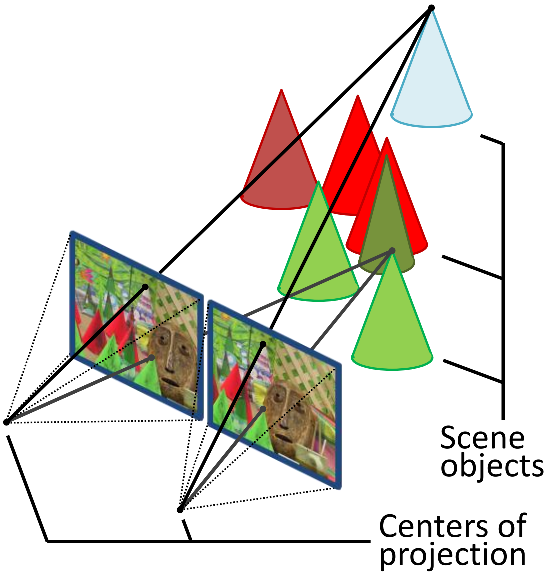

Figure 1: An example from the Middlebury Stereo Dataset ‘Cones’. A scene with paper cones is imaged with a stereo camera. The projection rays of two cone tips into both camera images are exemplarily marked.

For each surface point visible in both images, there are two rays in 3D space connecting the surface point with each camera’s centre of projection. In order to obtain the 3D position of the captured scene we mainly need to accomplish two tasks: First, we need to identify where each surface point that is visible in the left image is located in the right image. And second, the exact camera geometry must be known to compute the ray intersection point for associated pixels of the left and right camera. As we assume the cameras are firmly attached to each other, the geometry is only computed once during the calibration process.

3 Calibration

The geometry of the two-camera system is computed a priori in the stereo calibration process. First, we need a calibration object. Usually this is a planar calibration plate with a checkerboard or dot pattern of known size. Then we capture synchronous image pairs, showing the pattern different positions, orientations and distances in both cameras. One can then use the pixel locations of the pattern's dots in each image pair and their known positions on the calibration plate to compute both, the 3D poses of all observed patterns , and an accurate model of the stereo camera. The model consists of the so-called intrinsic parameters of each camera like the camera's focal length and distortion and the extrinsic parameters, i.e. the rotation and shift in three dimensions between the left and right camera. We can use this calibration data to triangulate corresponding points that have been identified in both images and recover their metric 3D coordinates with respect to the camera.

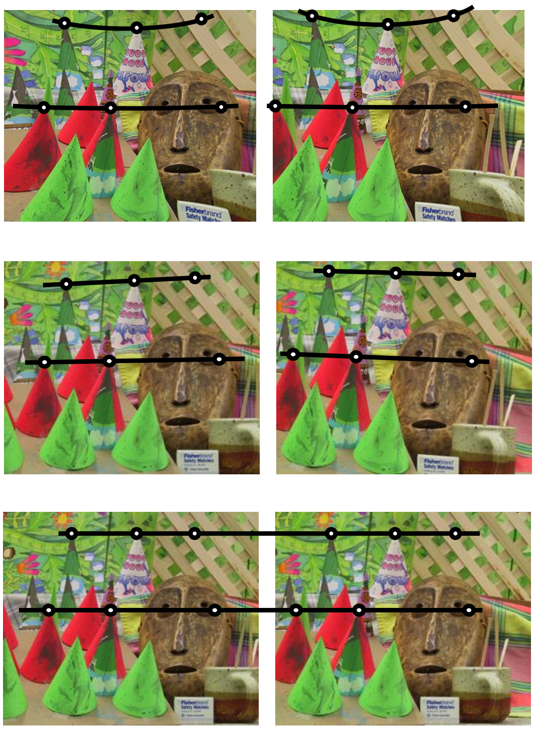

Figure 2: Search space to match image locations is only one dimensional. Top: The epipolar lines are curved in the distorted raw images. Middle: Removing image distortions results in straight epipolar lines. Bottom: Rectification makes epipolar lines aligned with the image axes. Correspondence search can be carried out along image scanlines.

3.1 Processing Steps for Depth Computation

The following three sections describe the processing steps, necessary for computing the 3D location for each pixel of an image pair. These steps have to be performed in real time for each captured stereo image to obtain a 3D point cloud or surface of the scene.

3.2 Rectification

In order to triangulate the imaged points we need to identify corresponding image parts in the left and right image. Considering a small image patch from the left image, we could simply search the entire right image for a sufficiently good match. This would be too time consuming to be done in real time. Consider the example image pair in Figure 3 with the cone tip visible in the top of the left image. Intuitively it does not seem necessary to search for the tip of the cone in the bottom half of the right image, when the cameras are mounted side by side. In fact the geometry of the two projective cameras allows to restrict the search to a one dimensional line in the right image, the so called epipolar line.

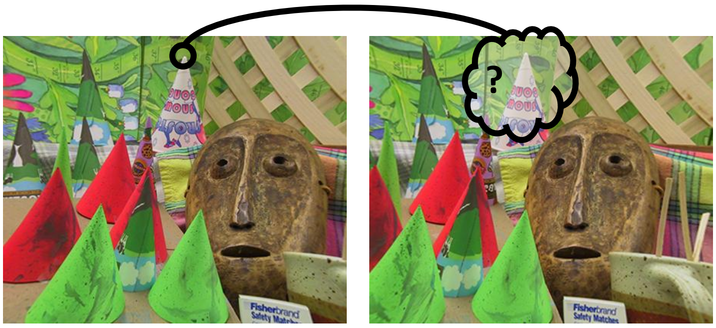

Figure 3: A stereo image pair. Where do we have to search for the cone tip in the right image?

Figure 2 (top) shows a few hand marked point correspondences and their epipolar lines. In the raw camera images the epipolar lines will be curved due to distortions caused by the camera optics. Searching correspondences along these curved lines will be quite slow and complicated, but we can remove the image distortions by reversely applying the distortion learnt during the calibration process. The resulting undistorted images have straight epipolar lines, depicted in Figure 2 (middle).

Although being straight, the epipolar lines will have different orientations in different parts of each image. This is caused by the image planes (i.e. the camera sensors) neither being perfectly coplanar nor identically oriented. To further accelerate the correspondence search we can use the camera geometry from the calibration and apply an additional perspective transformation to the images, such that the epipolar lines are aligned with the image scanlines[2]. This step is called rectification. The search for the tip of the white cone can now be carried out by simply looking at the same scanline in the right image and finding the best matching position. All further processing will take place in the rectified images only, the resulting images are shown in Figure 2 (bottom).

3.3 Stereo Matching

For each pixel in the left image, we can now search for the pixel on the same scanline in the right image, which captured the same object point. Because a single pixel value is typically not discriminative enough to reliably find the corresponding pixel, one usually tries to match small windows (e.g. 7x7 pixels) around each pixel against all possible windows in the right image on the same row. As further restriction, we don't need to search the entire row but only a limited number of pixels to the left of the left image pixel's x-coordinate, corresponding to the slightly cross-eyes gaze necessary to focus near objects. This accelerates the matching and restricts the depth range where points can be triangulated. If a sufficiently good and unique match was found, we associate the left image pixel with the corresponding right image pixel. The association is stored in the disparity map in form of an offset between the pixels x-positions (see Figure 4).

This matching technique is called local stereo matching, as it only uses local information around each pixel. Obviously, we can only match a region between left and right image when it is sufficiently distinct from other image parts on the same scanline. Thus, local stereo matching will fail in regions with poor or repetitive texture. Other methods, known as global stereo matching, can also exploit neighboring information. They don't just consider each pixel (or image patch) individually to search for a matching partner, instead they try to find an assignment for all left and right image pixels at once. This global assigment also takes into account that surfaces are mostly smooth and thus neighboring pixels will often have similar depths. Global methods are more complex and need more processing power than the local approach, but they require less texture on the surfaces and deliver more accurate results, especially at object boundaries.

[2] Mathematically speaking the rays captured by the same scanline of the rectified left and right image are all contained within a single plane.

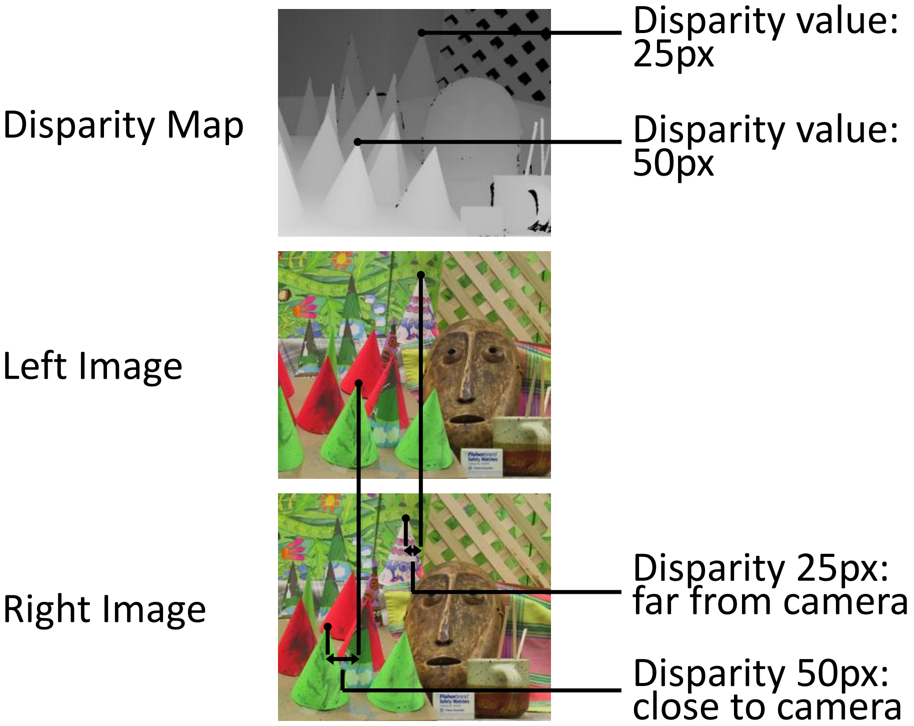

Figure 4: Result of image matching. The disparity map represents depth information in form of pixel shifts between left and right image. A special value (here black) is used to indicate, that a pixel could not be identified in the right image. This will happen for occluded areas or reflections on the object, which appears differently in both cameras.

3.4 Reprojection

Regardless of what matching technique is used, the result is always an association between pixels of the left and right image, stored in the disparity map. The values in the disparity map encode the offset in pixels, where the corresponding location was found in the right image. Figure 4 illustrates the disparity notion. We can then again use the camera geometry obtained during calibration to convert the pixel based disparity values into actual metric X, Y and Z coordinates for every pixel. This conversion is called reprojection. We can simply intersect the two rays of each associated left and right image pixel, as illustrated earlier in Figure 1. The resulting XYZ data is called a point cloud. It is often stored as a three channel image to also keep the point's neighboring information from the image's pixel grid. A visualization of the point cloud is shown in Figure 5.

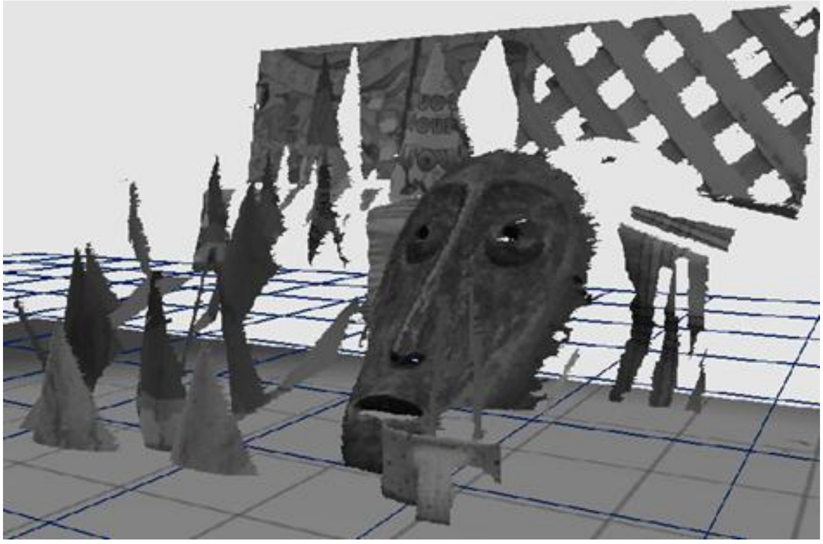

Figure 5: View of the 3D Surface generated from the disparity map and the camera calibration data. The surface is also textured with the left camera image (here converted to gray scale).

3.5 Application specific processing

The three described processing steps have to be carried out on the stereo image pair in order to obtain the full 3D point cloud of the scene. The point cloud then needs to be processed further to realize a specific application. It can be used to match the surface of the scene against a known object, either learned from a previous point cloud or a CAD model. If the part can be located uniquely in the captured scene surface, the complete position and rotation of the object can be computed and it could, for example, be picked up by a robot.

3.6 N10 Stereo Camera

As mentioned earlier, all stereo matching techniques require textured objects to reliably determine the correspondences between the left and right image. The Ensenso N10 stereo camera therefore integrates an additional texture projection unit. During the image capture the texture projection unit augments the object's own texture with a highly structured pattern to eliminate ambiguities in the stereo matching step. This ensures a dense 3D point cloud even on unicolored or ambiguously textured surfaces. The projector and the cameras are also synchronized by a hardware trigger signal to ensure consistent image pairs when capturing moving objects.

3.7 NxLib Stereo Processing Library

The NxLib library interfaces the N10 camera and implements the entire stereo processing pipeline including calibration. It combines texture projection with a global matching technique and provides dense, high-quality point clouds. The strictly parallelized global matching algorithm can use all processor cores to achieve real time performance.