GEODEZIJA IR KARTOGRAFIJA / GEODESY AND CARTOGRAPHY
2008
34(3): 92–96

UDK 528.72
doi: 10.3846/1392-1541.2008.34.92-96

ANALYSIS AND PROCESSING THE 3D-RANGE-IMAGE-DATA FOR ROBOT MONITORING[1]

Tobias Kohoutek

Swiss Federal Institute of Technology, ETH Zurich, Switzerland
E-mail: kohoutek@geod.baug.ethz.ch

Received 10 11 2007; accepted 19 06 2008

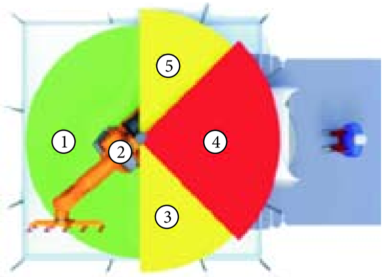
Abstract. Industrial robots are commonly used for physically stressful jobs in complex environments. In any case collisions with heavy and high dynamic machines need to be prevented. For this reason the operational range has to be monitored precisely, reliably and meticulously. The advantage of the SwissRanger® SR-3000 is that it delivers intensity images and 3D-information simultaneously of the same scene that conveniently allows 3D-monitoring. Due to that fact automatic real time collision prevention within the robots working space is possible by working with 3D-coordinates.

Keywords: range-image, motion analysis, object tracking, real-time, robot monitoring, security zone, optical flow.

1. Introduction

To record a moving object in 3D is possible using different methods. This paper shows the analysis and processing of images containing the local brightness and the distance for 25344 pixels. Those 3D-images are taken by the SwissRanger® SR-3000 by the time-of-flight (TOF) principle for measuring ranges. The miniaturized size (including the modulated infra-red light source of 54 LEDs), no movable components and the recording of kinematic processes (30 frames/seconds) are important advantages in contrast to laser scanners.

Camera systems are rarely used for monitoring working processes of machine tools and industrial robots. Today it is usual to apply shut-off mats or light barriers to detect objects entering the security zone, shown in Fig. 1.



1 – pivoting range
2 – robot
3 – stopping distance
4 – security zone
5 – stopping distance

Fig. 1. Robot working space (KUKA... 2006)

2. Components

2.1. SwissRanger® SR-3000

This distance measuring camera, based on combining CMOS/CCD-technology, is developed by Centre Suisse d'Electronique et de Microtechnique SA (CSEM), Zurich Switzerland (Lange, Seitz 2001). By this camera it is possible to acquire an amplitude image, that shows the local brightness in gray values (64 Bit), and a range image for the distances in every pixel. The distances are coded into gray values (64 Bit) in the range image. The distance measurements are realized for each individual pixel by exploiting the TOF-principle, working with a modulated infrared light source. Objects in a scene reflect the emitted light pulses back to the camera, where their precise time of arrival is measured at 4 points (CSEM... 2007).

In (Oggier *et al.* 2004) the phase map and finally a complete distance map can be acquired by detecting the phase delay between the emitted and the reflected signal in Fig. 2a. By sampling this signal the three unknown parameters of the modulated signal in Fig. 2b, the amplitude A, the offset I and the phase φ can be determined by the equations (1) to (3).

$$\varphi = \arctan\left(\frac{m4 - m2}{m1 - m3}\right), \quad (1)$$

$$A = \frac{\sqrt{(m3-m1)^2 + (m4-m2)^2}}{2}, \quad (2)$$

[1] Diploma thesis at Dresden University of Technology (TU Dresden), Institute of Photogrammetry and Remote Sensing, Germany, 2006.

ISSN 1392-1541 print / ISSN 1648-3502 online
http://www.gc.vgtu.lt

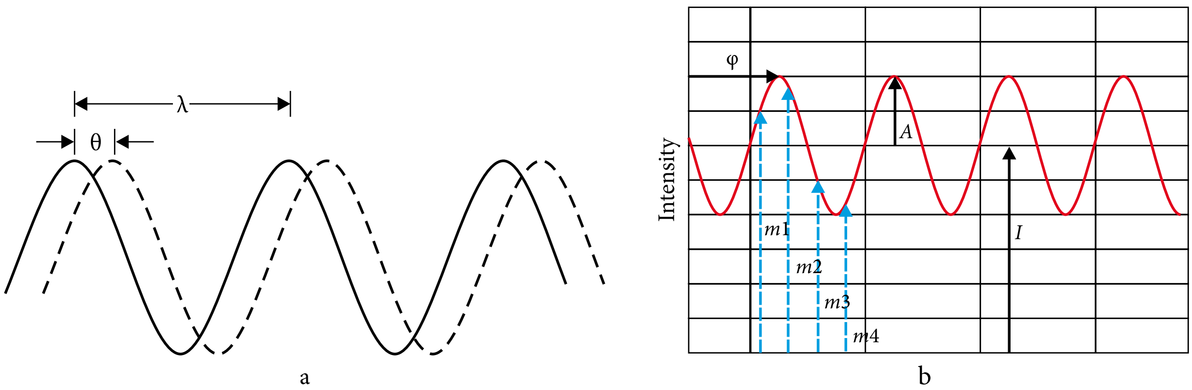


Fig. 2. Phase delay between the emitted and the reflected signal:
a) phase delay (Zhang 2005); b) modulated signal (Weingarten 2004)

$$B = \frac{m1 + m2 + m3 + m4}{4} . \qquad (3)$$

With (4) and (5) the distance and the accuracy of the depth measurements ΔD can be calculated. The maximum unambiguous distance range of 7,50 m is represented by $D_{\max}$.

$$D = D_{\max} * \frac{\varphi}{2\pi} , \qquad (4)$$

$$\Delta D = \frac{D_{\max}}{\sqrt{8}} \frac{\sqrt{B}}{2A} . \qquad (5)$$

2.2. LEGO® MINDSTORMS™ RIS 2.0

This robot was used to imitate the movements of an industrial robot. The Robotic Invention System includes a RCX-Microcomputer, a USB-Infra-red-Transmitter and the RIS Software (Windows™) as well as conventional LEGO® bricks and different sensors and motor types.

For interactive controlling it is not possible to use the RIS Software. Here the V2.1-Interface by Daniel Berger (2007) programmed in Microsoft® Visual C++ was adopted. Due to that fact the RCX-Microcomputer works only as an interface between the motors/sensors of the robot model and a desktop computer.

3. Analysis and processing of 3D-image-information

3.1. Robot detection

With the experimental setup, shown in Fig. 3, two different kinds of segmentation could be realized by using Microsoft® Visual C++ and the Open Source Computer Vision Library OpenCV.

The first method of segmentation is based on detection by a background image. Here the difference image of the background image and the first image with the robot displays as a result only the robot itself. Is it not possible to work with a background image for the second method, where segmentation is used.

Thresholding, morphological operations and edge detection are utilized algorithms for the segmentation. In every case the initial image presents the mean over 25 images to reduce the noise level. For N images the SNR in a mean image is $1/\sqrt{N}$ (Jähne 2005).

The range image was used for the edge detection, due to the fact that scattering the pixel values is bigger compared to the amplitude image. In Fig. 4 different edge detection algorithms were used like Sobel-, LaPlace- and Canny-algorithm.

The best result, a binary image, was given by the Canny-algorithm. Because more edges can be detected than are actually on the robot itself, a threshold was set up to remove those outlines with a fewer number of pi-

Fig. 3. Experimental setup

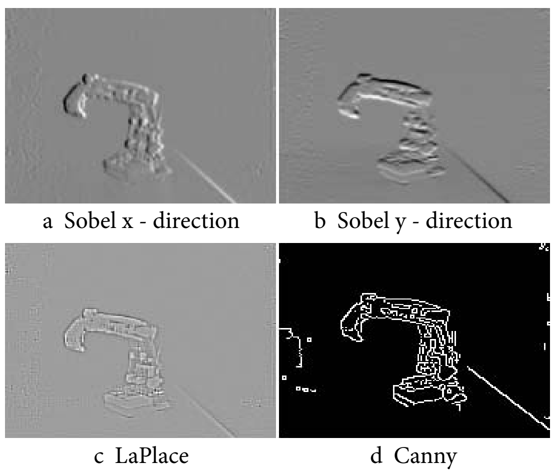


Fig. 4. Edge detection algorithms

xels as those at the robot contour. This step is followed by a dilation operation to fill the gaps in the contour. In the last step the robot structure is filled out white and features become marked is the binary image using a fixed raster over the white area shown in Fig. 5.

In the following steps only features at the robot's contour are required. A simple neighbourhood operation is used to minimize the features by proofing every feature and its four neighbours (up, down, left and right), whether they belong to the robot (white) or to the background (black). A feature is an edge feature, when three neighbours belong to the robot and the fourth to the background. It also counts as an edge feature when only two neighbours belong to the robot, but both should not be on opposite sides. In this case, the feature is a corner feature (Fig. 6). The result is a new image that contains only features at the robot contour.

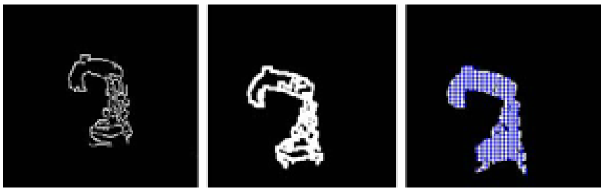


Fig. 5. Robot edges and features

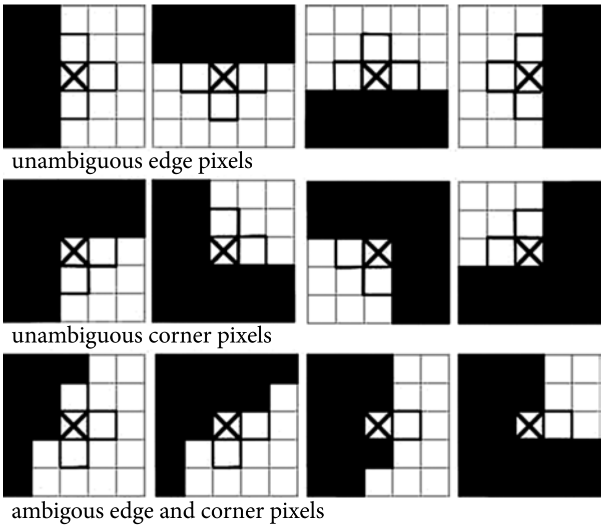


Fig. 6. Template for edge features

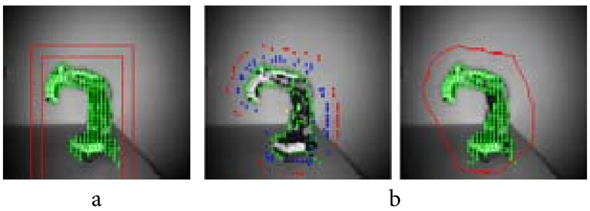


Fig. 7. Security zones: a – static; b – dynamic

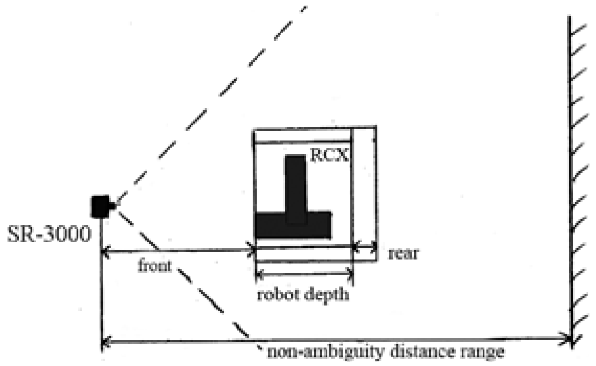


Fig. 8. Monitored areas

3.2. Virtual Security Zones

For monitoring and collision prevention a virtual security zone around the robot is created. There are two different shapes, shown in Fig. 7, where the first one is similar to a cube around the robot and called "static security zone". The second shape fits much better to the robot's shape and is called "dynamic security zone", because it will do the same movements like the robot.

In order to create the static zone, three special points at the robot's contour have to be known. These points are the furthest features to the left, right and top of the robot's edge. Knowing the x- and y-values in the image, it is possible to append the virtual cube to these features around the robot. The wall thickness of the cube is correlated to the robot's velocity and describes the security area. The stopping distance increases with higher speeds; consequently, the faster the robot moves, the thicker the wall is to be.

It is obvious that much more than necessary image space is controlled or not usable for other image operations by using such a "static" box. Due to that fact the dynamic zone was created. It fits much better to the robot's contour, because the shape is created by all edge detected features. Now it is important to find out, whether a feature is on the left or the right side of the robot, or at the top or the ground. For that a more sophisticated neighbourhood operation was used.

For all features (shown in green) it is known, on which side at the robot's contour they are. Using that information, they are shifted in radial direction for 5 pixels on a line from the focal point (yellow) to the image edge. These virtual inner boarder points (blue) were shifted a second time in the same direction for eight more pixels in order to create the outer barrier (red).

After definition and preparation of the security zones in the 2-dimensional image space, a concept for the 3-dimensional robot working space was generated. As seen in the top view of Fig. 8, the monitored working space was split into 3 areas (front, robot depth, rear).

The idea is to detect other objects entering the chosen security zone in 3 different distances. If there is an object entering the security zone between the camera and the robot (front), the robot has to stop immediately. In that case the robot would be in the so-called phantom space (more information in (Franz 2001)) of the other object and it is not possible anymore to guarantee a risk free working of the robot. The following controlled zone is called "robot depth" and is as depth as the robot itself. Directly behind the robot there is again a phantom space which is not controllable. It seems that objects entering the security zone from behind cannot be detected. But if they come nearer to the robot, they "grow" in the image space and the grey values in amplitude image and range image will change.

The conditions to have to be fulfilled to stop the robot's work are simple. An object is entering the security zone, if there are changes in the grey values of the amplitude image as well as changes in the range image.

There are 3 different distance thresholds for controlling the range image in the distances "front", "robot depth" and "rear".

4. Robot monitoring/ tracking and motion analysis

Motion in image sequences is always associated with changes between 2 images. By subtracting these 2 images, all differences become visible. It is important to know that grey value changes are not always related to object motions. Also changes of the light source or camera position generates such differences in the grey values, as can be seen in Fig. 9. The results are the motion field and the optical flow, where the motion field describes the real motion of an object in the 3D-scene projected onto the image plane (Jähne 2005). The optical flow is defined by the flow of gray values in the image. The existence of a constant light source and a pixel neighbourhood, that moves similar to the centre pixel (Fig. 10), during the exposures are pre-conditions for using the optical flow.

This displacement vector field is the projected 3D physical motion filed to the image plane and provides information about the arrangement and the changes of objects. The gray values "flow" over the image plane is equivalent to the flow of volume elements in liquids or gases. Motion is closely related to spatial and temporal grey value changes. That is why only the component of the displacement vector which is normal to an edge can be determined, while the component parallel remains unknown. That problem is called aperture problem and applies only to local operators. In Fig. 11a an unambiguous determination is only possible for an object corner that lies within the operator mask. The aperture problem is a special case of the correspondence problem, because distinguishing different points of an edge is not possible. A solution for that problem is shown in Fig. 11b by an image pyramid that reduces the resolution of an image gradually. While the original image I is the 0^{th} pyramid level, the resolution and size of the following I^L decreases by a factor of two. Large pictures get scaled down by using pyramids into a magnitude of local neighbourhood operations. Smaller image size neighbourhood operations made in the upper level of a pyramid can be performed more efficiently than for finer scales. The basis for the hierarchical image classification is built by important image features. A high robustness and a good local accuracy are expressed by image pyramids.

During the working process of the robot the motion of all object features will be calculated by the optical flow with sub-millimeter accuracy. Their displacements will be added to their corresponding points of the dynamic security zone, and so this zone follows the same movements as the robot.

During the real time robot monitoring (30 frames/ second) the security zone will be controlled in amplitude image and range image. The robot will stop when there is a detected grey value changing in both images at the same feature and his direct neighbourhood. If an object is detec-

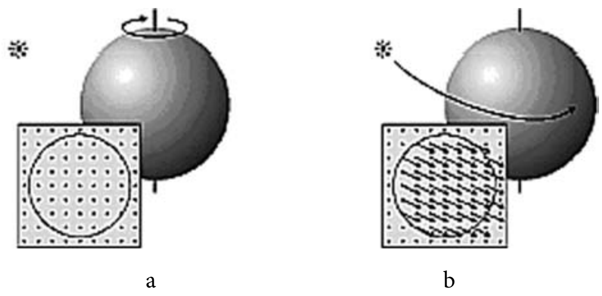


Fig. 9. Motion field vs. optical flow (Franz 2001):
a – motion of the sphere without grey value changing;
b – motion of the light source with grey value changing

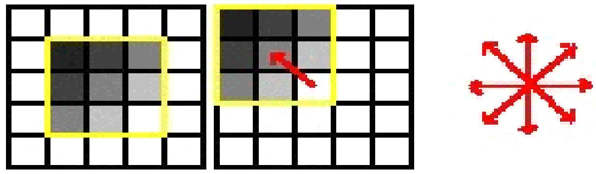


Fig. 10. Displacement vector field at optical flow

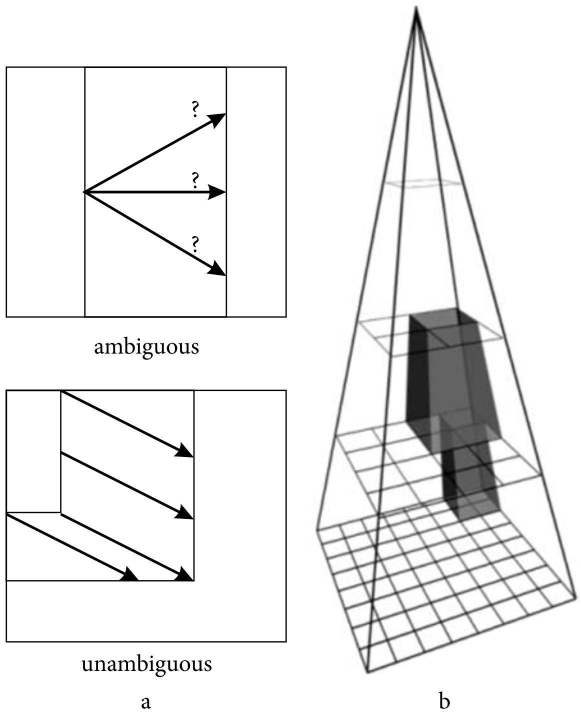


Fig. 11. Aperture problem (a), image pyramide (b)

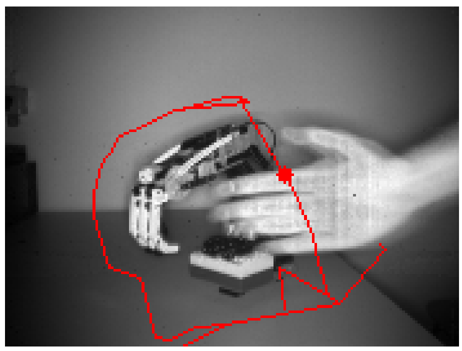


Fig. 12. Detected object

ted, the robot will stop in time to avoid a collision and the object will also be marked and displayed in Fig. 12.

5. Conclusion

The determination of a robot working space and its automatic real-time monitoring was carried out successfully. The results for the analysis and processing of the 3D-images are reliable. The distance measurement is influenced by multipath effects, temperature, distortion and effects of the objective.

The data for absolute and relative accuracy of the recorded objects will become reliable after a camera calibration of the SwissRanger® SR-3000.

References

Berger, D. 2007. *Advanced Lego Mindstorms Programming in Visual C++, 2005* [cited March 2007]. Available from Internet: <http://www.kyb.tuebingen.mpg.de/bu/people/berger/mindstorms.html>.

CSEM; SwissRanger SR-3000. Available from Internet: <http://swissranger.ch> [cited march 2007].

Franz, M. O. 2001. *Computer Vision in autonomous Systems*. Max Planck Institute for Biological Cybernetic, Tübingen.

Jähne, B. 2005. *Digital Image Processing*, 6th revised and extended edition, Springer-Verlag, Berlin Heidelberg New York.

KUKA Roboter GmbH Augsburg; KUKA SafeRobot V1.1, 2006.

Lange, R.; Seitz, P. 2001. Solid-State, Time-of-Flight Range Camera, *IEEE Journal of Quantum Electronics*.

Oggier, T.; Lehmann, M., *et al*. 2004. An all-solid-state optical range camera for 3D real-time imaging with sub-centimeter depth resolution (SwissRanger™), St. Etienne, *SPIE Optical Design and Engineering*.

Siebold, M. 2001. *Entwurf und Realisierung eines Konzeptes zur pyramidengestützten Bilddarstellung*. Master Thesis. FH Mainz.

Weingarten, J.W., Grüner, G., Siegwart, R. 2004. *A State-of-the-Art 3D Sensor for Robot navigation*, IROS, Sendai.

Zhang, J. 2005. *Lesson: Angewandte Sensorik – Abstandssensoren*.

Tobias KOHOUTEK. Dipl.-Ing., research and teaching Assistant. ETH Zurich, Institute of Geodesy and Photogrammetry
Ph +41 44 633 30 41, email: kohoutek@geod.baug.ethz.ch
Research interests: range imaging, engineering geodesy.