In: ASER 2004 2nd International Workshop on Advances in Service Robotics
Feldafing, Germany, 20-21 May 2004

SIMERO: Camera Supervised Workspace for Service Robots

Thorsten Gecks and Dominik Henrich

Embedded Systems and Robotics Lab (RESY)
University of Bayreuth, D-95445 Bayreuth, Germany
E-Mail: {thorsten.gecks|dominik.henrich}@uni-bayreuth.de
http: //ai3.inf.uni-bayreuth.de.de/

We present an industrial robot system whose workspace is supervised by several stationary cameras detecting obstacles using a difference image method. All robot transfer motions are checked for collision by an image-based method. If any collision is detected, the robot motion path is changed accordingly. The image processing is simple and robust and could be extended to mobile robots for example for services within a household. Experiments prove good performance.

Keywords: image processing, industrial robot, workspace supervision, difference image, path planning

1. Introduction

Humans need reliable and safe interaction with robots in order to accept and trust the robotic partner at work and even more at home. In order to achieve this, our project aims at a secure interaction between humans and robots to prevent severe injuries to human partners and to prevent (costly) collisions with environment obstacles.

The system therefore needs to detect dynamic obstacles like humans and determine appropriate measures (path planning) to avoid collisions with the detected obstacles. The system shall be safe for a set of tasks, like transportation of objects.

To achieve this goal, the current state of the environment needs to be acquired. Cameras are convenient sensors for this task as they are widely available and cost effective regarding to several criteria such as resolution and update rate.

In the following sections we give an overview of existing approaches in robot safety systems (Section 2), then give a short introduction to our safety system and its application in transport services (Section 3). Subsequently, we present experimental results of our safety system (Section 4) and draw a conclusion (Section 5).

2. State of the art

In the past, many approaches have been discussed for sensor-based collision avoidance. However, most of them use sensors that provide only local information. For example, in [Novak92] and [Feddema94] capacitance sensors were used as sensor skin. In [Lumelsky93], algorithms for whole-arm collision avoidance for robots with sensor skins were presented. In [Yu99], a wrist-mounted laser scanner was used. With only local sensor information available, only configurations close to the current robot configuration can be examined and thus only local planning is possible.

[Noborio01] presented an approach for image-based path planning in configuration space; however, the use of a wrist-mounted sensor is assumed and the approach required that the image of the scene in the target configuration to be known.

A general method for acquiring the physical extent of objects from multiple images is the back-projection. It is widely used in Computer Graphics, for example in [Eckert00] and [Eisert00]. However, the focus here was on the precise reconstruction of the object in 3D space, including the texture information. This is not necessary for our problem, as the object can be coarsely reconstructed and there is no need to process the object texture. Also, only single objects were reconstructed, while in our problem multiple obstacles exist in the scene.

[Meisel91], [Meisel94] and [Ameling96] presented a system for obstacle detection with stationary CCD-Cameras. These cameras were used to establish multiple passive light barriers, a concept related to back-projection. Evenly distributed points in the robot workspace were mapped to a monitored pixel in each camera. If the features of the pixel differ from the given background values beyond a certain threshold, the system assumed the corresponding beam of light to be interrupted. If an interrupted beam existed within the robots path, the current robot movement was stopped. A path planning based on the obstacle information was not realized.

The MEPHISTO system allowed for robot-human coexistence in the field of mobile robots [Steinhaus99]. It combined robot-based sensor systems (laser scanner) with a global monitoring system consisting of color cameras surveying the floor upon which robots and humans move. The images obtained by the cameras were compared to a reference image that was continuously updated. The difference image was mapped on the floor in the form of a polygonal region, which allowed for fast collision detection. The system provided the observed mobile robots with a path planning service in a 3-dimensional (2D-position and rotation around axis perpendicular to the ground plane) configuration space.

Our goal is to develop a system capable of global path planning, which is more suitable for service applications than local path planning based on short-range proximity sensors as the robot can cope with traps resulting from certain obstacle constellations. Therefore we employ cameras as global sensors. Comparable to [Meisel91] and the back projection method, we use a combination of several cameras to detect obstacles within the work space. The idea of an image-based path planning from [Steinhaus99] is extended to 3D workspace and 6D configuration space path planning for an industrial robot arm. This concept seems to be a promising technique for the realization of safe robot services.

3. System Concept

We now briefly present the realization of our prototype system of a camera supervised workspace. Beginning with a hardware description and software overview (Section 3.1), we describe necessary setup steps (Section 3.2) and develop the system data flow sequentially (Sections 3.3, 3.4). Additionally we describe how the system copes with gripped objects for carrying tasks in section 3.5.

3.1 System Overview

The workspace of the robot is monitored by several stationary cameras, each one monitoring the entire workspace shared by humans and the robot (Figure 1). Additionally, the current robot position is acquired from the robot controller via a network connection. The processing of the sensor data and the generation of robot movement commands is done on a single-processor standard workstation PC.

Figure 1: View of the realized prototype system comprising four cameras (black circles)

The software computes a path around obstacles such as humans. To detect the obstacles within the work space, the image processing subsystem applies a difference image method. It detects the humans and other obstacles via comparison of the current images to reference work cell images [Ebert01, Ebert02a, Ebert02b, Ebert03a]. Generally the *reference images* comprise the work cell without humans and robot containing static obstacles only.

3.2 Reference Image Generation

These reference images are generated in a preliminary setup step. The robot is driven into various positions and a picture is captured of each one. Then, a median value is calculated for every single pixel in all captured images of each camera. The robot arm disappears, because he is supposed to exist only once at all positions (except the robot base, which is immovable for the shown robot type) and the median operation selects the pixel value present in the majority of the images, wich represent the background.

3.3 Difference Image Calculation

For calculating a difference image of the work cell, the currently captured gray scale images are each subdivided into non-overlapping tiles on a grid, such that each tile contains several pixels. This subdivision is also applied to the corresponding reference image. For each tile, features are calculated based on the given pixel values (Figure 2 a,b). These features are then compared to the corresponding features of the reference images (Figure 2 c,d,e). The classification yields two classes: The tile is classified as *Background* if no significant changes to the reference image exist and as *Foreground* if significant changes exist. Foreground tiles are regarded as parts of obstacles. Several classification methods and automatic classification parameter optimization strategies can be applied [Heinzen03]. Linear classification or statistical Bayes classification can achieve circa 97% correctly detected foreground pixels.

3.4 Collision Test and Path Planning

The prerequisite for the collision test is to eliminate the robot from the current set of foreground pixels, because otherwise the robot would occur as obstacle and path planning in image space would be impossible. Therefore, the system computes a robot model given the current robot configuration and projects it into the camera views. These robot views are used to mask the robot in its current position [Ebert02b]. Concurrently, only foreground pixels belonging to obstacles remain for consideration in the path planning process (Figure 2 f,g,h,i,j). This image is called *cleaned workspace image*.

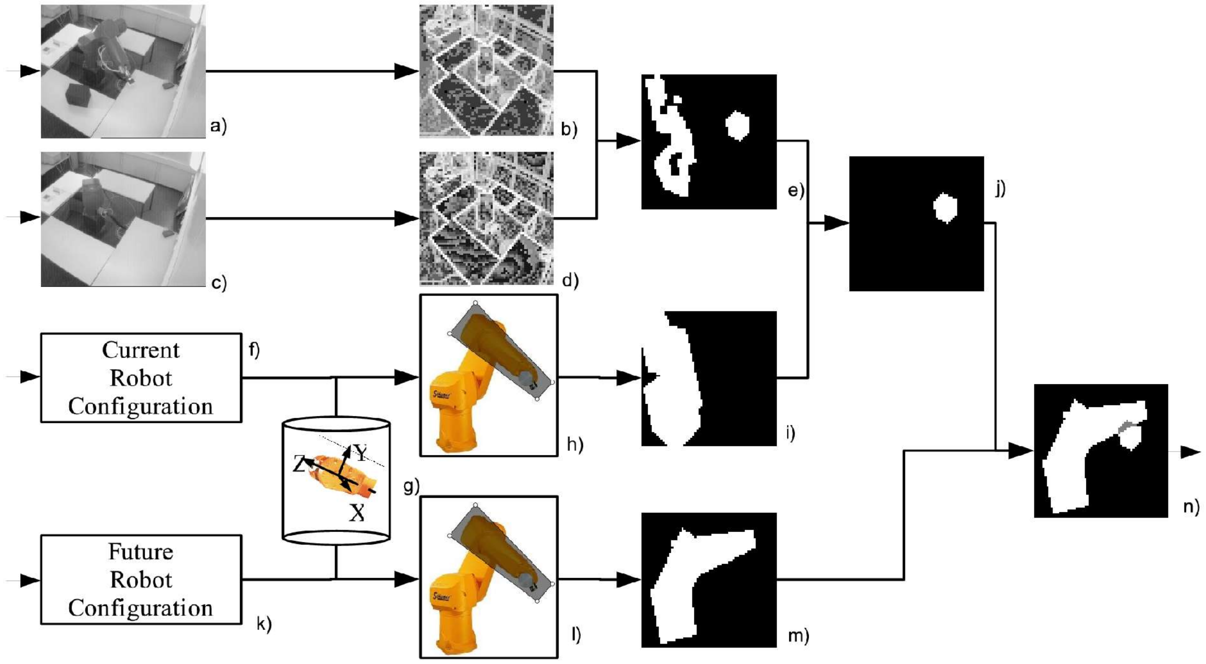


Figure 2: Data flow with simplified computation steps (no iterations shown)

The path planning takes place in a discrete subspace of the robot configuration space. Path planning algorithms operating on this discrete space check future robot configurations for collisions with obstacles. The future robot configurations are therefore projected into the camera views with the help of the robot model and checked for intersection with the set of foreground pixels from the cleaned workspace image (Figure 2 k,l,m,n). The cameras delivering no intersection are counted and if their number exceeds a certain threshold the robot configuration is considered collision-free.

3.5 Object transportation service

Regarding the transportation service task the system needs to be extended as the system described above only comprises safe movements of the robot body and transportation of objects needs special considerations as discussed below.

Objects within the gripper of the robot are a potential source of collisions with the environment. Therefore the collision test needs to reflect the existence of a carried object.

Carried objects on the other hand are a special type of objects because they are in contact with the robot, in fact they “collide” with the robot. To provide a transportation service, they must not to be regarded as obstacles, because in this case the system would try to avoid them. Therefore, these objects need to be distinguished from the rest of the obstacle scene.

The solution is to include the carried objects as part of the robot body. We are then able to perform the collision test in image space including the obstacle by rendering the carried object as part of an extended robot model into the images.

To include carried objects in the robot model, their geometry needs to be known, which generally applies to manipulated objects in an industrial environment. Depending on the object size we can subdivide the workpieces into two classes. In the first class, there are objects smaller than the robot gripper. These objects can easily be integrated because it is irrelevant for the robot model whether they are gripped or not, as their few extra pixels can be incorporated into the standard robot model without being inconvenient when the robot is not carrying objects.

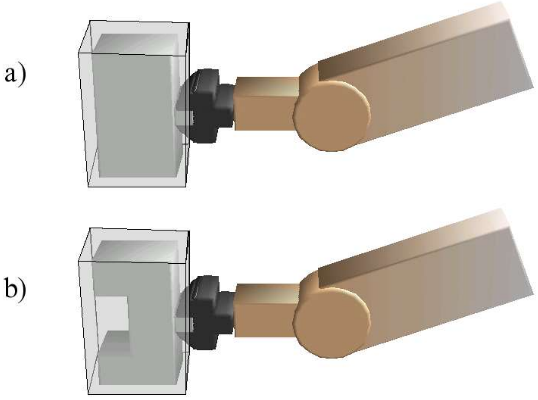


Figure 3: Generic Volume

Secondly there are objects too big to be neglected, which need to be modeled, preferably with a generic volume, such as a cube or any other compact geometric primitive (Figure 3 a). The generic model simplifies the image rendering considerably, therefore it is preferred whenever feasible, even for large objects. This technique is especially useful compared to exact modeling, if the robot is manipulating machined workpieces in an industrial environment that change their shape after being processed, for example, by milling machines. The generic volume remains unchanged after the processing step as it still comprises the object (Figure 3 b).

4. Experiments

The figures listed in Table 1 represent the specifications of a prototype system and the major results and parameters.

The setup with comparatively cheap PC-Hardware providing only low computing power nevertheless achieves considerable robot speeds. As the system architecture allows for extensive parallelization, a high scalability regarding workspace volume, camera resolution and update rates is realistic.

The accuracy provided by the system allows even close (30 – 50 cm) approaches to obstacles, such as humans.

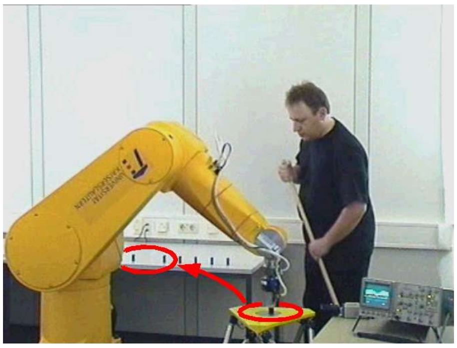
Figure	***Value***
No. of cameras	4 Grayscale cameras
Camera resolution	704x576
Difference image resolution	64x64
PC specifications	1 Processor AMD Athlon 2200+ 512 MB DDR-RAM Windows NT 4.0 SP 6, Visual C++ 6.0
Network connection	10 Mbit Ethernet
Robot work space and No. of configurations	work space was a cube of ca. 1.5x2x2 m 8192 configurations
Update rate	Ca 7 Hz
Robot speed	70 cm/s

Table 1: Prototype system measures

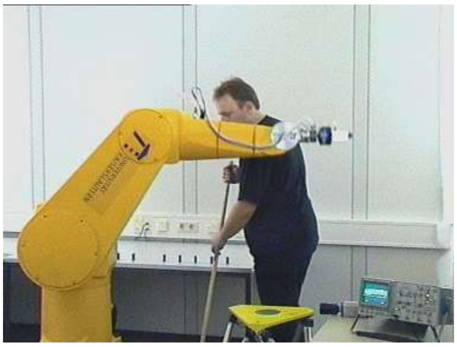
The following image sequence presents a prototypical application example. The systems task is to transport a small black wooden work piece from a point of processing to another (marked by the red ellipses in Figure 4a), in this case from a symbolic quality check to a placement area. The standard, uninterrupted robot path is depicted by the red arrow in figure Figure 4a.

The sequence shows the reaction of the safety system to the interruption of the standard robot path. The system dynamically plans a path around a moving obstacle, such as the human coworker, who is cleaning the robot work cell in this example (Figure 4a-f).

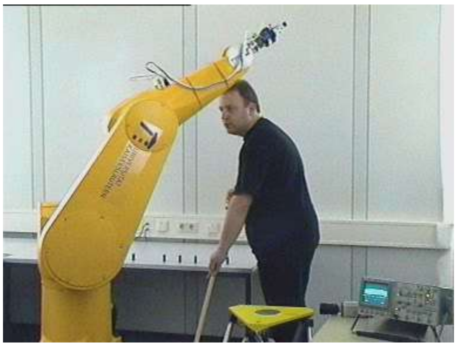
a)



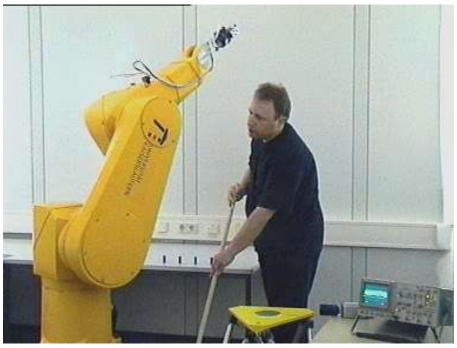
b)



c)



d)



e)

f)

Figure 4: Image sequence of a service application example

5. Conclusions

The presented supervision system shows the conceptual feasibility of our system for applications where humans and robots coexist in a common workspace. A basic service functionality is already available with the transportation task and will be extended in future development steps. The system currently requires a mostly static background of the supervised workspace, making it feasible for applications in industrial plants like safe pick-and-place operations or museums. The limitation to a static background is subject to further research.

References

[Ameling96] Ameling, W. (Ed.): „Flexible Handhabungsgeräte im Maschinenbau“; Ergebnisse aus dem Sonderforschungsbereich 208, VCH Publishing, 1996

[Ebert01] Ebert, D., Henrich, D.: „Safe Human-Robot-Cooperation:Problem Analysis,System Concept and Fast Sensor Fusion" In: IEEE Conference on Multisensor Fusion and Integration for Intelligent Systems, pp. 239-244, Baden-Baden, Germany, August 20 - 22, 2001

[Ebert02a] Ebert, D., Henrich, D.: „SIMERO - Sicherheitsstrategien für die Mensch-Roboter-Kooperation" In: „OTS-Systeme in der Robotik – Roboter Ohne Trennende Schutzeinrichtungen“, Reihe BKM Berichte, Herbert Utz Publishing, pp. 5.1-5.17, München, June 25 2002

[Ebert02b] Ebert, D., Henrich, D.: „Safe Human-Robot-Cooperation: Image-based collision detection for Industrial Robots" In: IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 1826-1831, Lausanne, October 2 - 4, 2002

[Ebert03a] Ebert, D., Henrich, D.: „SIMERO - Sichere Mensch-Roboter-Koexistenz“, In: „2. Workshop für OTS-Systeme in der Robotik – Mensch und Roboter ohne trennende Schutzsysteme“, Stuttgart, June 24 2003

[Ebert03b] Ebert,D.: “Bildbasierte Erzeugung kollisionsfreier Transferbewegungen für Industrieroboter“ PhD Thesis, Informatics Faculty, University of Kaiserslautern, Germany, 2003

[Eckert00] Gerald E.: “Automatic Shape Reconstruction of Rigid 3-D Objects from Multiple Calibrated Images”, In: Eusipco 2000 Proceedings, Tampere, Finland , 2000.

[Feddema94] Feddema J.T., Novak J.L.: “Whole Arm Obstacle Avoidance for Teleoperated Robots”. In: IEEE Robotics and Automation Proceedings, pp.3303 – 3309, 1994.

[Gecks03] Gecks, T.: „SIMERO - Erzeugung von flüssigen, schnellen Roboterbewegungen“, Diploma Thesis, Informatics Faculty, University of Kaiserslautern, Germany, 2003

[Heinzen03] Heinzen, F.: „SIMERO – Robuste und Schnelle Erzeugung von Silhouetten aus Grauwertbildern“, Diploma Thesis, Informatics Faculty, University of Kaiserslautern, Germany, 2003

[Lumelsky93] Lumelsky V., Cheung E.: “Real-Time Collision Avoidance in Teleoperated Whole-Sensitive Robot Arm Manipulators”. In: IEEE Transactions on Systems, Man and Cybernetics, Vol.23 No.1, pp.194-203,1993.

[Meisel91] Meisel, A.; Föhr, R.; Ameling, W.:“3D-Kollisionsschutzsensor auf der Basis von CCD-Kameras“, In SENSOR 91, pp. 157-170, Nürnberg, May 1991

[Meisel94] Meisel A: “3D-Bildverarbeitung für feste und bewegte Kameras”, Vieweg Publishing, Fortschritte der Robotik Nr. 21, 1994

[Noborio01] Noborio H., Nishino Y.: “Image-based Path-Planning Algorithm on the Joint Space”. In: IEEE International Conference on Robotics and Automation, pp. 1180-1187,Seoul, 2001.

[Novak92] Novak J.L., Feddema J.T.: ”A Capacitance-Based Proximity Sensor for Whole Arm Obstacle Avoidance”. In: IEEE Proceedings of the Intl. Conf. on Robotics and Automation, pp. 1307-1314, 1992.

[Yu99] Yu Y., Gupta K.: „Sensor-Based Roadmaps for Motion-Planning for Articulated Robots in Unknown Environment: Some Experiments with an Eye-in-hand System“. In: IEEE International Conference on Intelligent Robots and Systems, pp.1707-1714, 1999.

[Steinhaus99] Peter Steinhaus, Markus Ehrenmann, Rüdiger Dillmann: MEPHISTO: A Modular and Existensible Path Planning System Using Observation. ICVS 1999 361-375