1 Introduction

The video-based exploration of interiors with autonomous and mobile service robots is a task that requires much programming effort. Additionally, the programming tasks differ in the necessary modules. Commands, which control the technical basis equipment, must consider the reality of the robot. These commands activate the breaks and the actuation. The parts are basically included in the delivery. Often a mobile robot additionally possesses sonar, ultrasonic, and cameras, which constitute the perception function of the robot. The programming of such a mobile robot is a very difficult task if no control software comes with the robot. First, the programmer must develop the necessary drivers. As a rule the manufacturer includes a software library into the scope of the supply. This enables programs in a high-level language like C++ to be created very comfortably to control most or all parts of the robot’s basic equipment. Typically operators are available. The user can transfer values to the arguments whose domain depends on the device that is to be controlled, and the admitted measurement. Operators, which enable rotary motions, may take values in degrees or radians. The velocity can be adjusted with values, which are specified in meters per second or yards per second. Video cameras are sometimes also part of a mobile robot’s basic equipment, but further software and/or hardware must be acquired generally. A frame grabber is required. This board digitizes the analog signal of the camera. The gained digital image can then be processed with an image-processing library. Such a library provides operators for the image processing that can also be included into a high-level program. If the camera comes with the robot, the manufacturer provides two radio sets if the computer that controls the robot is not physically compounded with the robot. One radio set is necessary to control the robot’s basic equipment from a static computer. The second radio set transmits the analog camera signals to the frame grabber. Nowadays, robots are often equipped with a computer. In this case radio sets are not necessary, because data transfer between a robot’s equipment and a computer can be directly conducted by the use of cables. Additionally, a camera head can be used that connects a camera with a robot and enables software-steered panning and tilting of the camera. Mobile service robots use often a laser that is, as a rule, not part of a robot. They are relatively expensive, but sometimes the robot-control software provided involves drivers for commercial lasers.
The application areas for mobile service robots are manifold. For example, a real-
ized application that guided people through a museum has been reported. The
robot, named RHINO [1], was mainly based on a laser device, but many imaginable
areas require the robot to be producible cheaply. Therefore, RHINO will not be
further considered in this book. This book proposes robot-navigation software that
uses only cheap off-the-shelf cameras to fulfill its tasks. Other imaginable applica-
tions could be postal delivery or a watchman in an office environment, service
actions in a hospital or nursing home, and so forth. Several researchers are currently
working on such applications, but a reliable application does not currently exist.
Therefore, at this point a possible scenario for a mobile service robot that works
autonomously will be illustrated.
Postal delivery is considered, as mentioned before. First, it should be said that
such a task can not be realized with a robot that works exclusively with lasers,
because the robot must be able to read. Otherwise it can not allocate letters to the
particular addresses. Therefore, a camera is an essential device. If the mobile robot
is to work autonomously, it is necessary that it knows the working environment. If
the robot needs to be able to work in arbitrary environments, it is a problem if a
human generates a necessary navigation map, which can be considered as a city
map, offline. If the robot’s environment changes, a new map must be created manu-
ally, which increases the operating costs. To avoid this, the robot must acquire the
map autonomously. It must explore the environment before the operating phase
starts. During the operating phase, the robot uses the created map to fulfill its tasks.
Of course, its environment changes, and therefore it is necessary to update the map
during operation. Some objects often change their positions like creatures; others
remain rather permanently in the map. Desks are an example. The robot must also
be able to detect unexpected obstacles, because collision avoidance must be exe-
cuted. If letters are to be distributed in an office environment, and the robot was just
switched on to do this task, it must know its actual position. Because the robot was
just activated, it has no idea where it is. Therefore, it tries a self-localization that
uses statistical methods like Monte Carlo. If the localization is successful, the robot
has to drive to the post-office boxes. It knows the location by the use of the naviga-
tion map. As a rule the boxes are equipped with names. The robot must therefore be
able to read the names, which assures a correct delivery. An OCR (optical character
recognition) module must therefore be part of the navigation map. It shall be
assumed that only one post-office box contains a letter. The robot then has to take
the letter. The robot must also read the doorplates during the map acquisition, so it
is able to determine to which address the letter must be brought. A path scheduler
can then examine a beneficial run. If the robot has reached the desired office, it can
place the letter on a desk, which should also be contained in the navigation map.
In Figure 1 is shown a possible architecture of a video-based robot-navigation pro-
gram on a rather abstract level.
A mobile robot, which is equipped with a camera, is sketched in the lower right area of the figure. The image data are read from an image-processing module that must try to detect an object in the image using image-processing operators. The object is analyzed and reconstructed after a successful detection. It is necessary to determine its world coordinates. The three-dimensional world coordinate system is independent of the robot’s actual position. Its origin can be arbitrarily chosen. For example, the origin could be that point from which a robot starts its interior exploration. The origin of the three-dimensional camera coordinate system is determined by the focal point of the camera. If object coordinates are actually known in the camera coordinate system, it is possible to derive the world coordinates. The determination of the coordinates can use a stereo technique. At least two images from different positions are necessary for these purposes. Corresponding pixels belonging to that image region, which represents the desired object, must be detected in both images. Stereo triangulation exploits geometrical realities to determine the distance of the object point from the focal point. Additionally, the technical data of the camera must be considered for the depth estimation. Calibration techniques are available for these purposes. If the object coordinates are known, the object and its parts can be measured. In many cases the robot will be forced to take many more than two images for a reliable three-dimensional reconstruction, because three-dimensional objects often look different when viewed from different positions. The acquired data should then enable a CAD (computer-aided design) model to be produced. This can be a wire frame represented with a graph. For example, if the CAD model of an office table is to be obtained, the table legs and the desktop can be represented with
edges. The program must determine for every edge the length and its start and endpoints, which are represented by nodes. Coordinates are then attached to every node. The CAD module can additionally use a knowledge base for the proper reconstruction of the object. For example, the knowledge base can supply an image-processing program with important information about the configuration and quantity of object parts like the desktop and the table legs. After the three-dimensional object reconstruction is completed, the examined data can be collected in the navigation map. All these tasks must take place before the operating phase can start. The autonomous navigation uses the calculated map and transmits control commands to the mobile robot to fulfill the work necessary that depends on the particular scenario.

As noted before, such a service robot must be producible at a very low price if it is to fulfill its tasks cost effectively. Beside the use of very cheap equipment, the aim can be realized with the creation of favorable software. In particular, the software should be portable, work on different operating systems, and be easily maintainable.

Chapter two discusses some image-processing operators after these introductory words. The purpose of the chapter is not to give a complete overview about existing operators. Several textbooks are available. Image-processing operators are discussed that seem appropriate for machine-vision tasks. Most of the operators explained are used in experiments to test their presumed eligibility. Although cheap color cameras are available, the exploitation of color features in machine-vision applications is not often observed. Different color models are explained with regard to their possible field of application in machine-vision applications following elementary elucidations.

There then follows a section that relates to Kalman filter that is not a pure image-processing operator. In fact the Kalman filter is a stochastic method that can basically be used in many application areas. The Kalman filter supports the estimation of a model’s state by the use of appropriate model parameters. Machine-vision applications can use the Kalman filter for the three-dimensional reconstruction by analyzing an image sequence that shows the object at different points in time. The image sequence can be acquired with a moving camera that effects the state transitions.

Video-based exploration with autonomous robots can be damaged by illumination fluctuations. The alterations can be effected by changes in the daylight, which can determine the robot’s working conditions. For example, lighting alterations may be observable if the robot’s working time comprises the entire day. Experiments showed that a Gabor filter can mitigate the effects of inhomogeneous illumination. The chapter discusses an application that uses the Gabor filter in a machine-vision application and reports the results.

Subsequent paragraphs describe fundamental morphological operators that are not typical for video-based machine-vision applications, but as a rule they are used in almost every image-processing program and thus also in experiments that are explained in the later chapters of this book. They are therefore explained for clarity. Further basis operators are edge detection, skeleton procedure, region building, and threshold operator. The skeleton procedure is not so frequently observed in
machine-vision applications as the other listed operators, but it seems to be principally an appropriate technique if the three-dimensional reconstruction with wire frames is required. The skeleton procedure is therefore discussed for the sake of completeness.

Chapter three is devoted to navigation. Applications that control mobile service robots are often forced to use several coordinate systems. The camera’s view can be realized with a three-dimensional coordinate system. Similar ideas can hold for a robot gripper when it belongs to the equipment of a mobile robot. Further coordinate systems are often necessary to represent the real world and the robot’s view that is called the egocentric perception of the robot. Transformations between different coordinate systems are sometimes required. An example of this was mentioned before.

Map appearances can be coarsely divided into grid-based maps and graph-based maps. Graph-based maps are appropriate if quite an abstract modeling of the environment is to be realized. They offer the possibility that known algorithms for graphs can be used to execute a path plan between a starting point and an arrival point. For example, the path planning can be calculated on the condition that the shortest path should be found. Grid-based maps offer the possibility that the environment can be modeled as detailed as is wished. The grid technique was originally developed for maps used by human beings like city maps, atlases, and so forth.

After the discussion of several forms of grid-based maps, path planning is explained. The path length, the actual necessary behavior, and the abstraction level of the planning influence the path planning. One application is then exemplified that combines two abstraction levels of path planning.

The next section shows an example of an architecture that involves different map types. The chapter finishes with an explanation of the robot’s self-localization.

Chapter four deals with vision systems. Machine vision is doubtless oriented to the human visual apparatus that is first illustrated. The similarity between the human visual apparatus and the technical vision system is then elaborated. To this belongs also behavior-based considerations like the attention control that determines how the next view is selected. Further sections consider interactions between observer and environment.

The remainder of chapter four explains current technical vision systems, which can be low priced. CMOS cameras are more expensive cameras. They are not considered because affordable development of mobile service robots is not possible with such cameras.

The content of chapter five is the three-dimensional reconstruction of objects. CAD techniques are especially considered, but other methods are also described. The application area for CAD techniques was originally industrial product development. The strategy for object reconstruction from image data differs therefore from the original application that uses CAD to model a new product, but object reconstruction uses image data to gain a CAD model from an existing object. Nevertheless, CAD techniques are appropriate for machine-vision applications. This is shown in the chapter. First, widespread CAD techniques are regarded and then followed by approximate modeling methods. Some models are a composite of different ap-
proaches. These are the hybrid models. Automated conversions between different models are proposed. One approach is then discussed that creates a CAD model from image data. The drawback of this is an elaborate calculation procedure. This is often observed if CAD models in machine-vision applications are used. But alternative approaches, whose purpose is not the explicit generation of a CAD model and sometimes not even a complete object reconstruction, also frequently suffer from this problem.

Knowledge-based approaches seem to be appropriate to diminish the calculation effort. The last application proposes a direct manipulation of the object, which is to be reconstructed, with marks. This strategy offers possibilities for simplification, but in some applications such marks may be felt to be disturbing. This may hold especially for applications with service robots, because human beings also use the robot’s working environment. Mark-based procedures also require additional work or are impracticable. An application for a service robot scenario can not probably use the strategy, because too many objects have to be furnished with such marks.

Chapter six covers stereo vision that tries to gain depth information of the environment. The configuration of the used cameras provides geometrical facts, which can be used for the depth estimation. The task is the three-dimensional reconstruction of a scene point if only corresponding points in two or more images are known. The examination of corresponding points is sometimes very difficult, but this depends on the particular application. Three-dimensional reconstruction can also be gained from image sequences that were taken from a moving camera. In this case the Kalman filter can be used.

Chapter seven discusses the camera calibration that is a prerequisite for a successful reconstruction, because the camera parameters are determined with this strategy. The simplest calibration strategy is the pinhole camera calibration that determines only the camera’s basis parameters like the focal length. But approaches also exist that consider further parameters. Lens distortion is an example of such parameters. Special calibration approaches exist for robot-vision systems. In this case the robot can be used to perform a self-calibration.

Several computer-vision applications use self-learning algorithms (Chapter 8), which can be realized with neural networks. OCR (Chapter 9) in computer vision is an example. Self-learning algorithms are useful here, because the appearance of the characters varies depending on the environment conditions. But changing fonts can also be a problem.

Until now the work can be considered as tutorial and shows that known methods are insufficient to develop a reliable video-based application for a mobile and autonomous service robot. In the next chapters methods are explained that will close the gap.

Chapter 10 proposes the use of redundant programs in robot-vision applications. Although redundant programming is, in general, a well-known technique and was originally developed to enhance the robustness of operating systems [2], it is not common to consider the use in computer-vision applications. First, the chapter describes the basics and elaborates general design guidelines for computer-vision applications that use the redundancy technique. The capability was tested with a
robot-vision program that reads numbers on a doorplate. A high recognition rate was obtained.

A further drawback for a potential developer is the fact that no evaluation attempts can be found in the literature to compare different algorithms for service-robot applications. Chapter 11 reports on executed comparisons among algorithms. The algorithms are explained and then compared to experiment.

Chapter 12 explains a cost-effective calibration program that is based on pinhole-camera calibration. Most existing implementations use commercial software packages. This restricts the portability and increases the costs for licenses. In contrast the proposed implementation does not show these drawbacks. Additionally it is shown how a precise calibration object can be simply and cheaply developed.

Chapter 13 shows the superiority of the redundant programming technique in the three-dimensional reconstruction by the use of the CAD technique. A new CAD modeling method was developed for robot-vision applications that enables the distance-independent recognition of objects, but known drawbacks like mathematically elaborate procedures can not be observed. The CAD model extraction from image data with the new method is tested with a program. The results gained are reported. The sample images used were of extremely poor quality and taken with an off-the-shelf video camera with a low resolution. Nevertheless, the recognition results were impressive. Even a sophisticated but conventional computer-vision program will not readily achieve the reported recognition rate.