

Chapter 9

Robotics and Autonomous Robots

Henry I. Ibekwe¹ and Ali K. Kamrani²

Abstract Autonomous robots are rapidly changing manufacturing processes and industrial production systems. It is now an integral component essential for improved productivity and efficiency in industrial mechanized plants. In most modern large scale production facilities, tasks such as welding, forming, drilling, milling, and locating are completely performed by robots and the degree of autonomy will vary by industry. The emerging trend is to improve the robots' perceptual ability and cognition, thereby making them less reliant on an external controller and human intervention. This process is called autonomy, hence the name autonomous robots.

In this chapter, we shall discuss robot kinematics, basic task planning, and robot vision. Emphasis will be on the robot components that inherently improve robot autonomy. Arguably, the most important yet relatively underdeveloped robotic components are robot vision systems. The ability for a robot to “see” is an extremely difficult task because of the complex nature of visual perception. Comparable to human vision, an enormous amount of information is required to be processed and translated to the robot controllers. Humans perform visual tasks effortlessly without really understanding the tremendous processing ability of the brain. Till date, there are still major breakthroughs to be made in understanding vision and vision system as its application in robotics will significantly improve the autonomy of robots.

9.1 Introduction To Robotics

Robotics as defined by the *American Heritage Dictionary* is the science or study of the technology associated with the design, fabrication, theory, and application of robots. Robot, on the contrary, is a mechanical device that sometimes resembles a

¹Department of Industrial Engineering, University of Houston, Houston, Texas 77004 USA

²Associate Professor, Department of Industrial Engineering, University of Huston, Houston, Texas 77004, USA

human and is capable of performing a variety of often complex human tasks on command or by being programmed in advance. According to Fu et al. [13], the Robot Institute of America defines industrial robots as "... a reprogrammable multifunctional manipulator designed to move materials, parts, tools, or specialized devices, through variable programmed motions for the performance of a variety of tasks."

The study of robotics over the years has seen an immense growth in both technological application and otherwise. The recent advances in the manufacturing processes have necessitated the need to enable robots to be more autonomous. Autonomy simply means the ability of the robot to be independent, that is, intelligent. It should be understood that the mimicking of human intelligence and neural function is a relatively nascent research area and has significant strides to overcome in order to achieve this.

9.1.1 Application of Robots

There are numerous applications of robots today. They are applied anywhere from a highly sophisticated space robotic arm used by NASA's space exploration program to Lego® gadgets used for entertainment. Robots can also be found in various manufacturing industries, military applications, space exploration, and remote/hazardous environment. The application in industries has significantly increased in recent years because of improved productivity, reliability, and quality of end-products. This led to many manufacturing industries reformatting and remodeling plant layout to accommodate these changes.

9.1.1.1 Manufacturing Applications

The application of robots in the manufacturing industry includes operations such as drilling, cutting, deburring, parts installation, welding, painting, and inspecting. These tasks are just a few that can be performed at a high level of accuracy by the utilization of robots. Fewer system failures will occur because of improper machining and manufacturing than would otherwise have been performed by humans. An example of this occurs when uneven welds are created by machinists during the welding process. This can lead to failure of high-stress component within critical systems such as airplanes, bridges, and space shuttles. An example of a welding robot is shown in [Fig. 9.1](#)

Robots are vastly utilized in the electronics manufacturing because of the need for extreme accuracy, usually in the range of nanometers. Such levels of accuracy are nearly, if not, impossible to be performed by the average personnel. A pristine environment is also an added advantage. Tasks performed by the robots will minimize the possibility of debris and dust accumulation within the facility.

Painting and finishing are other common applications in the manufacturing environment. Automobile production lines utilize large number of robot manipulators for these operations. Uniformity and high accuracy is necessary during the paint and finishing phases.



Fig. 9.1 Fanuc Arc-Mate 120-i Robot Welding Cell (*Courtesy-Burns Machines*)

9.1.1.2 Assembly and Packaging Applications

Assembly operations have been quite cumbersome for application of robots; however, evolving techniques are reducing the level of difficulty required while maintaining quality standards. An example of this is a situation whereby difficulty arises when parts must be located and identified, carried in a particular order with many obstacles around the setup, fitted together, and then assembled [20]. Highly automated packaging companies solely apply robots for these purposes. The end result is often significant direct and indirect savings from reduced personnel cost to minimized product damage.

9.1.1.3 Remote and Hazardous Operations

Some tasks may involve high risk of fatality for humans. Robots that execute these tasks are often uniquely designed for very specific “workspace” conditions. Citing



Fig. 9.2 Mars rover (Courtesy-NASA)

the NASA Mars rover as an example, it was designed to collect vital research data about the inhospitable and rugged terrain of the planet Mars. Although there were minor glitches during the mission, the design of the autonomous rover (Fig. 9.2) has proven highly successful.

Remote-operated/autonomous robots have also found extensive use in underwater operations. Extremely high pressure at deep sea levels is an attribute to the dangers of deep sea exploration. Only recently could sunken naval, merchant, or cruise ships be explored. These robots can also be used for trans-Atlantic cable laying, deep sea dredging, and underwater pipeline inspection with the aid of vision systems.

9.1.1.4 Healthcare Applications

Imagine robots helping doctors perform complex operations like heart surgery or surgery for cancer using only a few tiny incisions. At first thought, it may come across as science fiction, but such tasks have been reliably performed tens of thousands of times already. This field of medicine is called *robotic-assisted minimally invasive surgery*. The design and production of a surgical robot is a formidable task since humans (a complex system) are now the “workpiece.” Numerous variables have to be considered and a high level of redundancy must be implemented in the design of the robotic surgical system.

Minimal invasive surgery (MIS) is performed using narrow, long-shafted surgical instruments and cameras, which are inserted through small incisions (Figs. 9.3 and 9.4) that serve as ports of entry to the body (e.g., abdominal wall) to reach the surgical

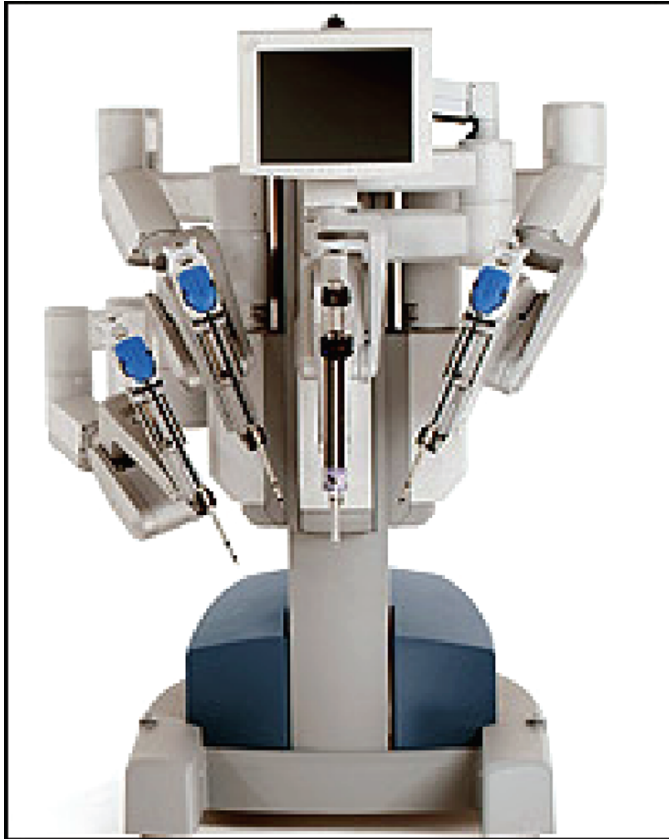


Fig. 9.3 The *da Vinci*® surgical system patient cart (*Courtesy of Intuitive Surgical, Inc.*)

site [12]. For many patients, the potential advantages of having an operation with a robotic surgical system include significantly less pain, less blood loss and need for transfusions, less risk of infection, a shorter hospital stay, quicker recovery time, and better outcomes, in many cases. Shorter hospital stay and simplified postoperative care may also lower hospital overhead and improve overall operational efficiencies. Potential surgeon benefits may include improved visualization, which is provided by a 3-D high-definition camera; improved precision, dexterity, and control provided by miniaturized, wristed instrumentation, tremor filtration, and motion scaling; and an intuitive interface that provides optimized hand-eye alignment.

The most complex subsystem of a surgical robot is the end effector (the instrument tip). The end-effector design must mimic the dexterity (skill in using the hands) of a human hand and wrist. The desire to perform surgery through the smallest incisions possible limits the available surgical maneuvers possible, as well as the “degrees of freedom”-or the angles and direction in which the “wrist” can move [12].

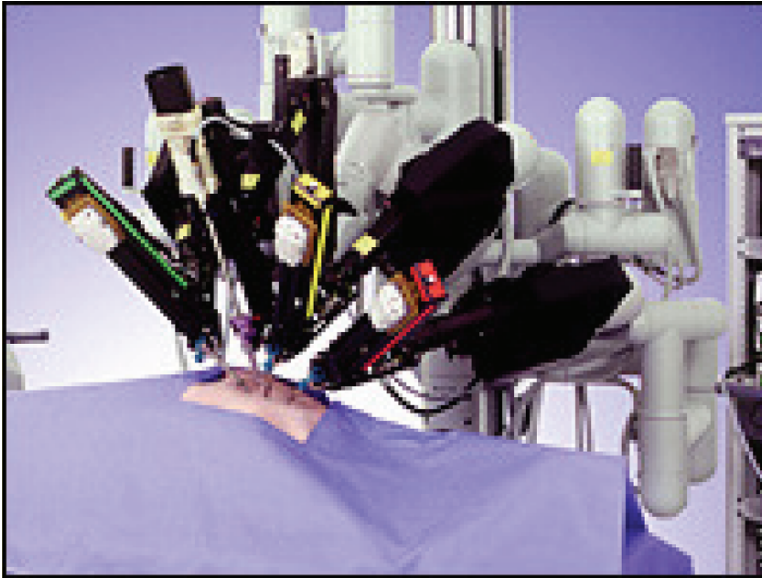


Fig. 9.4 Robotic-assisted surgery setup (Courtesy of Intuitive Surgical, Inc.)

9.1.2 Classes of Robots

Robots are classified according to their coordinate frames. These coordinate frames define the motion capabilities of the robot and are important in the robot kinematics analysis. They are Cartesian coordinate, cylindrical coordinate, spherical coordinate, articulated coordinate, and the selective compliance assembly robot arm (SCARA) robots. A brief discussion of these coordinate systems will follow. [Figure 9.5](#) identifies these robot classes.

9.1.2.1 Cartesian Coordinate Robots

Robots that fall under this class are often called rectangular or gantry robots and consist of three linear joints or axes. An example of a Cartesian coordinate robot is the IBM manufactured RS-1 robot. It has three prismatic joints and is modeled using the Cartesian coordinate system $(X, Y, Z) \rightarrow (X = a, Y = b, Z = c)$. Within a workspace, it travels linearly along any of the axis. During initialization it is important to note that starting axis and configuration settings are either pre-determined or dependent on various variables such as job type, work piece, and total repositioning time. Another similar robot is the *gantry robot* which operates on the same principle as the Cartesian coordinate robot. In this case, the robot is mounted on an overhead gantry.

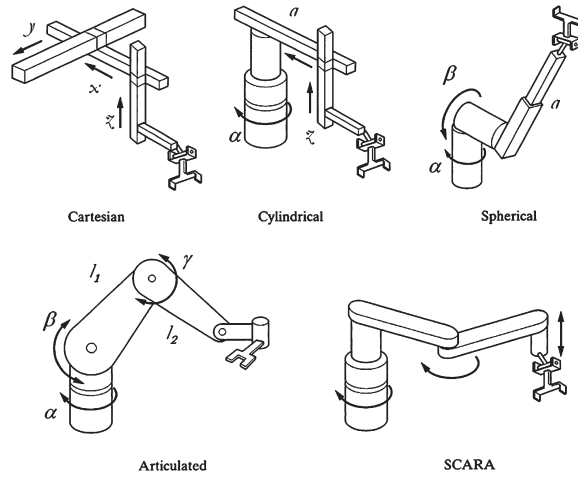


Fig. 9.5 Classes of robots (Courtesy-Niku [20])

9.1.2.2 Cylindrical Coordinate Robots

The cylindrical coordinate robots have two prismatic joints or linear axes and one revolute joint or rotary axis. The workspace for this class of robot resembles a cylinder as the name implies. The equivalent Cartesian coordinates can be found by the following equations

$$X = a \cos \alpha \quad (9.1)$$

$$Y = a \sin \alpha \quad (9.2)$$

$$Z = c \quad (9.3)$$

9.1.2.3 Spherical Coordinate Robots

The spherical coordinate robots have one prismatic or linear axis and two revolute joints or rotary axes. They have significant application in the manufacturing industry such as welding, cutting, and material handling. The corresponding X , Y , Z coordinates are

$$X = a \cos \alpha \cos \beta \quad (9.4)$$

$$Y = a \sin \alpha \cos \beta \quad (9.5)$$

$$Z = c \sin \beta \quad (9.6)$$

9.1.2.4 Articulated Coordinate Robots

The articulated coordinate robots are sometimes referred to as revolute or anthropomorphic robots. In this robot class, all joints are revolute or rotary and are comparable to the human arm. They are also highly utilized in industries and provide improved flexibility from the rectangular, cylindrical, or spherical coordinate robots. These axes connect three rigid links and the base. The corresponding X , Y , Z locations are

$$X = [l_1 \cos \beta + l_2 \cos (\beta + \gamma)] \cos \alpha \quad (9.7)$$

$$Y = [l_1 \cos \beta + l_2 \cos (\beta + \gamma)] \sin \alpha \quad (9.8)$$

$$Z = l_1 \cos \beta + l_2 \cos (\beta + \gamma) \quad (9.9)$$

9.1.2.5 SCARA Robots

This is the fifth class of robots. SCARA is the acronym for Selective Compliance Assembly Robot Arm. SCARA robots have two parallel rotary or revolute joints and a linear or prismatic joint. The two rotary joints enable the robot to relocate with the horizontal plane, while the linear joint enables it to relocate or move within the vertical plane. They have relatively better flexibility along the x -axis and the y -axis in comparison to that along the z -axis and hence are employed in assembly tasks.

9.1.3 Components of Robots

The robotic manipulator is composed of several main subsystems. They are the main frame or manipulator, actuators, sensors, controllers, end effector, and computer processor/operating systems. These subsystems interact in unison to produce the desired task to be performed by the robot.

9.1.3.1 Body/Main Frame

The main frame of a robotic manipulator is the rigid structure of the robot. They are serially connected to various joint configurations as discussed in Sect. 9.3. The frame can sometimes be called the manipulator. Care must be taken in using the term manipulator as the manipulator alone does not constitute a robot.

9.1.3.2 Actuators

Actuators are essentially the “driver” of a robot. They provide the necessary forces and moment to translate the robot from one spatial coordinate to another. There are

numerous actuators that can be used in the design of a robotic system. The more commonly used are pneumatic cylinder systems, hydraulic cylinder systems, servomotors, etc. Obviously, the actuators will require some level of control. This process is performed via the use of a controller.

9.1.3.3 Sensors

Robot sensors enable the robots to have some level of intelligence. The sophistication and complexity of the sensors provide increased autonomy for the robot. They are linked with robot controllers, discussed in subsequent sections, that provide a feedback system where information gathered can be analyzed and processed. Real-time information of the robot's internal state is collected through these sensors and allows the robot to interact with its environment with relative flexibility. The types of sensors commonly used are tactile sensors, vision sensors, speech sensors, etc.

9.1.3.4 Controllers

The controller moderates the movement and motions of the links of the robot. It verifies each link and the proper spatial coordinates in order to perform a task; therefore, it acquires data from the main computer process and transfers the information to the actuators.

9.1.3.5 End Effector

The end effector is the last element in the linkage of the robot arm. It may be referred to as the most important part of the robot. Specific design tasks will dictate the type of end effector on a robot manipulator. While design tasks of a robot vary, the main body of the robot can be consistent and require little alteration. For example, if a robot that primarily performs a pick-and-place task need to be redesigned to perform welding operations, the end effector will simply have to be modified and very little change made to the main body.

9.1.3.6 Computer Processor/Operating Systems

The computer processor is the "brain" of the robot manipulator. It controls the decision making and receives input signals from the sensors. The computer processor transmits the output signals to the robot controller. These signals are fed to actuators, which, in turn, provide the necessary forces to drive the robot.

Operating system is the software component of the computer processing system that enables the programmer to write computer algorithms and codes meant to run the robot manipulator.

implemented in industries. An industrial robot is a general-purpose, computer-controlled manipulator consisting of several rigid links connected in series by revolute or prismatic joints [13]. It can be modeled as an open-loop articulated chain consisting of the aforementioned rigid links typically connected in series prismatic or revolute joints. These joints are powered by actuators which are devices that produce forces to drive the systems to specified positions and orientations.

Robot manipulator kinematics involves the mathematical analysis of the geometry of motion of a robot manipulator taking into consideration a fixed reference coordinate system. The forces, moments that generate these motions, will not be considered during analysis but rather reflects on the special displacement of the robot as a function of time. A strong background in elementary vector analysis will be a definite requisite force for the study and understanding of robot kinematics.

Kinematics pertaining to robots is divided into two groups, forward kinematics and reverse kinematics. If all positional/joint variables of all joints and links of the robot are known, then the robot end-effector (hand) position can be determined by forward kinematics. Conversely, if the positional/joint variables of the end effector are known, then all the positional and joint variables of the joints and links can be determined by inverse or reverse kinematics.

Arguably the most important component of a robot manipulator is the end effector since it will primarily perform the required task. Therefore, it is a necessity to identify the position of each joint variable space with respect to the position of the end effector. The flexibility of the end effector is also important in analysis as this will determine its possible orientation.

9.2.1 Representation of Translating Bodies

In order to adequately derive and describe robot kinematics models, we shall revise the matrix representation of a point and a vector in 3-D space. This will give better insight in understanding the translation of body-attached coordinate. The basic knowledge of matrix algebra and vector analysis is also a necessity.

9.2.1.1 Representation of a Point and a Vector in 3-D Space

Consider a point \mathbf{p}_{xyz} in space (Fig. 9.7). We can represent this point by its three coordinates with respect to the reference coordinate frame.¹ That is,

$$\mathbf{p}_{xyz} = p_x \mathbf{i} + p_y \mathbf{j} + p_z \mathbf{k} \quad (9.10)$$

¹The reference coordinate frame is based on the *right-handed* Cartesian coordinate system and is called *orthonormal*.

where p_x , p_y , and p_z are the components of \mathbf{p}_{xyz} along the x -axis, y -axis, and z -axis, respectively, and \mathbf{i} , \mathbf{j} , \mathbf{k} are unit vectors along the x -axis, y -axis, z -axis, respectively. If there is a vector that originates from the origin $(0, 0, 0)$ of the reference coordinate axis, then the vector from origin O to point \mathbf{p}_{xyz} can be represented as \mathbf{p} such that

$$\mathbf{p}_{xyz} = p_x \mathbf{i} + p_y \mathbf{j} + p_z \mathbf{k} \quad (9.11)$$

Equation 9.11 shall subsequently be used to develop matrix transformation equations.

9.2.1.2 Rotation Matrices

Rotations matrices of the robot links having dimensions of a square matrix, $m \times m$ where $m = 3$, can be defined as a transformation matrix which functions on a position vector in a 3-D space. There are two coordinate systems, the reference coordinate system $OXYZ$ and the body-attached coordinate system $OUVW$ (Fig. 9.7). The components of the reference coordinate system, $OXYZ$, are OX , OY , and OZ , while the components of the body-attached coordinate system, $OUVW$, are OU , OV , and OW . The body-attached coordinate systems are fixed onto links of a robot and can translate or rotate.

Let \mathbf{i}_x , \mathbf{j}_y , and \mathbf{k}_z be the unit vectors along the coordinate axes of $OXYZ$ and let \mathbf{i}_u , \mathbf{j}_v , and \mathbf{k}_w be the unit vectors along the coordinate axes of $OUVW$. Then a point \mathbf{p} at rest and fixed with respect to the $OUVW$ coordinate frame in space can be represented by its coordinates with respect to $OXYZ$ and $OUVW$. Therefore

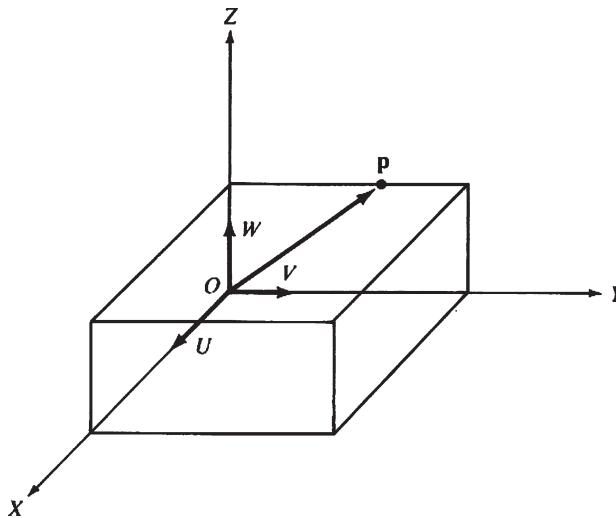


Fig. 9.7 Reference and the body-attached coordinate system

$$\mathbf{p}_{xyz} = (p_x, p_y, p_z)^T \quad (9.12)$$

$$\mathbf{p}_{uvw} = p_u, p_v, p_w \quad (9.13)$$

where $\mathbf{p}_{xyz} = \mathbf{p}_{uvw}$ meaning that they are at the same point in space but are based on different coordinate systems and are represented by a 3×1 matrix. If the coordinate reference frame $OUVW$ is rotated, the problem is to develop a 3×3 transformation matrix \mathbf{R} that will transform the coordinates of \mathbf{p}_{uvw} to the coordinates expressed with respect to the reference coordinate frame $OXYZ$. In other words

$$\mathbf{P}_{xyz} = \mathbf{R} \mathbf{p}_{uvw} \quad (9.14)$$

This transformation will result in an *orthogonal* transformation. Rewriting \mathbf{p}_{uvw} and \mathbf{p}_{xyz} in terms of their components we have

$$\mathbf{p}_{uvw} = p_u \mathbf{i}_u + p_v \mathbf{j}_v + p_w \mathbf{k}_w \quad (9.15)$$

$$\mathbf{p}_{xyz} = p_x \mathbf{i}_x + p_y \mathbf{j}_y + p_z \mathbf{k}_z \quad (9.16)$$

where $p_u, p_v,$ and p_w represent the components of \mathbf{p} along $OU, OV,$ and OW and $p_x, p_y,$ and p_z represent the components of \mathbf{p} along $OX, OY,$ and OZ . Expressing \mathbf{p} in terms of the components $p_x, p_y,$ and p_z we have

$$p_x = \mathbf{i}_x \cdot \mathbf{p} = \mathbf{i}_x \cdot \mathbf{i}_u p_u + \mathbf{i}_x \cdot \mathbf{j}_v p_v + \mathbf{i}_x \cdot \mathbf{k}_w p_w \quad (9.17)$$

$$p_y = \mathbf{j}_y \cdot \mathbf{p} = \mathbf{j}_y \cdot \mathbf{i}_u p_u + \mathbf{j}_y \cdot \mathbf{j}_v p_v + \mathbf{j}_y \cdot \mathbf{k}_w p_w \quad (9.18)$$

$$p_z = \mathbf{k}_z \cdot \mathbf{p} = \mathbf{k}_z \cdot \mathbf{i}_u p_u + \mathbf{k}_z \cdot \mathbf{j}_v p_v + \mathbf{k}_z \cdot \mathbf{k}_w p_w \quad (9.19)$$

Rewriting in matrix form, we have

$$\begin{bmatrix} p_x \\ p_y \\ p_z \end{bmatrix} = \begin{bmatrix} \mathbf{i}_x \cdot \mathbf{i}_u & \mathbf{i}_x \cdot \mathbf{j}_v & \mathbf{i}_x \cdot \mathbf{k}_w \\ \mathbf{j}_y \cdot \mathbf{i}_u & \mathbf{j}_y \cdot \mathbf{j}_v & \mathbf{j}_y \cdot \mathbf{k}_w \\ \mathbf{k}_z \cdot \mathbf{i}_u & \mathbf{k}_z \cdot \mathbf{j}_v & \mathbf{k}_z \cdot \mathbf{k}_w \end{bmatrix} \begin{bmatrix} p_u \\ p_v \\ p_w \end{bmatrix} \quad (9.20)$$

Let $\mathbf{R} = \begin{bmatrix} p_x \\ p_y \\ p_z \end{bmatrix}$, then:

$$\mathbf{R} = \begin{bmatrix} \mathbf{i}_x \cdot \mathbf{i}_u & \mathbf{i}_x \cdot \mathbf{j}_v & \mathbf{i}_x \cdot \mathbf{k}_w \\ \mathbf{j}_y \cdot \mathbf{i}_u & \mathbf{j}_y \cdot \mathbf{j}_v & \mathbf{j}_y \cdot \mathbf{k}_w \\ \mathbf{k}_z \cdot \mathbf{i}_u & \mathbf{k}_z \cdot \mathbf{j}_v & \mathbf{k}_z \cdot \mathbf{k}_w \end{bmatrix} \quad (9.21)$$

We can also develop a matrix \mathbf{Q} that will transform the coordinates of \mathbf{p}_{xyz} to the coordinates expressed with respect to the reference coordinate frame $OUVW$. That is

$$\mathbf{p}_{UVW} = \mathbf{Q} \mathbf{p}_{xyz} \quad (9.22)$$

Expressing \mathbf{p} in terms of the components p_u , p_v , and p_w , we have

$$p_u = \mathbf{i}_u \cdot \mathbf{p} = \mathbf{i}_u \cdot \mathbf{i}_x p_x + \mathbf{i}_u \cdot \mathbf{j}_y p_y + \mathbf{i}_u \cdot \mathbf{k}_z p_z \quad (9.23)$$

$$p_v = \mathbf{j}_v \cdot \mathbf{p} = \mathbf{j}_v \cdot \mathbf{i}_x p_x + \mathbf{j}_v \cdot \mathbf{j}_y p_y + \mathbf{j}_v \cdot \mathbf{k}_z p_z \quad (9.24)$$

$$p_w = \mathbf{k}_w \cdot \mathbf{p} = \mathbf{k}_w \cdot \mathbf{i}_x p_x + \mathbf{k}_w \cdot \mathbf{j}_y p_y + \mathbf{k}_w \cdot \mathbf{k}_z p_z \quad (9.25)$$

Rewriting in matrix form, we have,:

$$\begin{bmatrix} p_u \\ p_v \\ p_w \end{bmatrix} = \begin{bmatrix} \mathbf{i}_u \cdot \mathbf{i}_x & \mathbf{i}_u \cdot \mathbf{j}_y & \mathbf{i}_u \cdot \mathbf{k}_z \\ \mathbf{j}_v \cdot \mathbf{i}_x & \mathbf{j}_v \cdot \mathbf{j}_y & \mathbf{j}_v \cdot \mathbf{k}_z \\ \mathbf{k}_w \cdot \mathbf{i}_x & \mathbf{k}_w \cdot \mathbf{j}_y & \mathbf{k}_w \cdot \mathbf{k}_z \end{bmatrix} \begin{bmatrix} p_x \\ p_y \\ p_z \end{bmatrix} \quad (9.26)$$

Let $\mathbf{Q} = \begin{bmatrix} p_u \\ p_v \\ p_w \end{bmatrix}$, then

$$\mathbf{Q} = \begin{bmatrix} \mathbf{i}_u \cdot \mathbf{i}_x & \mathbf{i}_u \cdot \mathbf{j}_y & \mathbf{i}_u \cdot \mathbf{k}_z \\ \mathbf{j}_v \cdot \mathbf{i}_x & \mathbf{j}_v \cdot \mathbf{j}_y & \mathbf{j}_v \cdot \mathbf{k}_z \\ \mathbf{k}_w \cdot \mathbf{i}_x & \mathbf{k}_w \cdot \mathbf{j}_y & \mathbf{k}_w \cdot \mathbf{k}_z \end{bmatrix} \begin{bmatrix} p_x \\ p_y \\ p_z \end{bmatrix} \quad (9.27)$$

Since $\mathbf{p}_{xyz} = \mathbf{R} \mathbf{p}_{uvw}$ and $\mathbf{p}_{uvw} = \mathbf{Q} \mathbf{p}_{xyz}$, we have

$$\mathbf{Q} = \mathbf{R}^T = \mathbf{R}^{-1} \quad (9.28)$$

$$\therefore \mathbf{Q} \mathbf{R} = \mathbf{R}^T \mathbf{R} = \mathbf{R}^{-1} \mathbf{R} = \mathbf{I}_3 \quad (9.29)$$

where \mathbf{I}_3 is a 3×3 identity matrix, that is, $\mathbf{I}_3 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$ and since the vectors

in the dot product are all unit vectors, it is also called *orthonormal* transformation. The problem now is to find the rotation matrix \mathbf{R} such that if the *OUVW* coordinate system is rotated about any axis, it can be represented with respect to the *OXYZ*. We shall consider the three cases that will arise.

- Rotation of the *OUVW* system about the *OX*-axis by an angle α
- Rotation of the *OUVW* system about the *OY*-axis by an angle ϕ
- Rotation of the *OUVW* system about the *OZ*-axis by an angle θ

9.2.1.3 Rotation About the *OX*-Axis

For *case 1* (Rotation of the *OUVW* system about the *OX*-axis by an angle α) (Fig. 9.8), the rotation matrix $\mathbf{R}_{x,\alpha}$ can be derived as follows

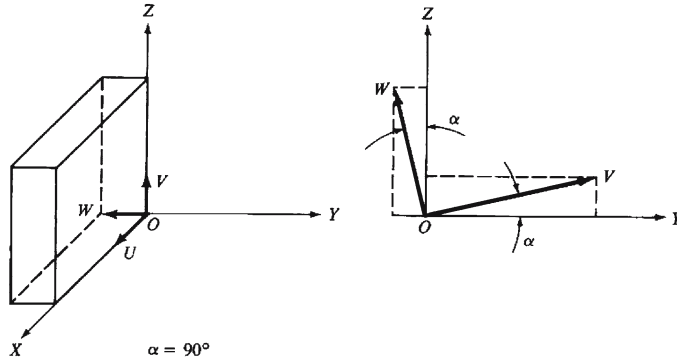


Fig. 9.8 Rotation of the $OUVW$ system about the OX -axis

$$\mathbf{p}_{xyz} = \mathbf{R}_{x,\alpha} \mathbf{p}_{uvw} \quad (9.30)$$

$$\mathbf{R}_{x,\alpha} = \begin{bmatrix} \mathbf{i}_x \cdot \mathbf{i}_u & \mathbf{i}_x \cdot \mathbf{j}_v & \mathbf{i}_x \cdot \mathbf{k}_w \\ \mathbf{j}_y \cdot \mathbf{i}_u & \mathbf{j}_y \cdot \mathbf{j}_v & \mathbf{j}_y \cdot \mathbf{k}_w \\ \mathbf{k}_z \cdot \mathbf{i}_u & \mathbf{k}_z \cdot \mathbf{j}_v & \mathbf{k}_z \cdot \mathbf{k}_w \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \alpha & -\sin \alpha \\ 0 & \sin \alpha & \cos \alpha \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & C\alpha & -S\alpha \\ 0 & S\alpha & C\alpha \end{bmatrix}$$

where $\mathbf{i}_x \cdot \mathbf{i}_u$ and by the definition, the dot product of two vectors \mathbf{a} and \mathbf{b} will result in a scalar $\mathbf{a} \cdot \mathbf{b} = |\mathbf{a}||\mathbf{b}|\cos\theta$. Also to simplify matrix element notation we can write $\cos\alpha$ as $C\alpha$ and similarly $\sin\alpha$ as $S\alpha$. This notation will be frequently used in further discussion.

9.2.1.4 Rotation About the OY -Axis

For *case 2* (Rotation of the $OUVW$ system about the OY -axis by an angle ϕ) (Fig. 9.9), the rotation matrix $\mathbf{R}_{y,\phi}$ can be derived as follows

$$\mathbf{p}_{xyz} = \mathbf{R}_{y,\phi} \mathbf{p}_{uvw} \quad (9.31)$$

$$\mathbf{R}_{y,\phi} = \begin{bmatrix} \mathbf{i}_x \cdot \mathbf{i}_u & \mathbf{i}_x \cdot \mathbf{j}_v & \mathbf{i}_x \cdot \mathbf{k}_w \\ \mathbf{j}_y \cdot \mathbf{i}_u & \mathbf{j}_y \cdot \mathbf{j}_v & \mathbf{j}_y \cdot \mathbf{k}_w \\ \mathbf{k}_z \cdot \mathbf{i}_u & \mathbf{k}_z \cdot \mathbf{j}_v & \mathbf{k}_z \cdot \mathbf{k}_w \end{bmatrix} = \begin{bmatrix} \cos \phi & 0 & \sin \phi \\ 0 & 1 & 0 \\ -\sin \phi & 0 & \cos \phi \end{bmatrix} = \begin{bmatrix} C\phi & 0 & S\phi \\ 0 & 1 & 0 \\ -S\phi & 0 & C\phi \end{bmatrix}$$

9.2.1.5 Rotation About the OZ -Axis

For *case 3* (Rotation of the $OUVW$ system about the OZ -axis by an angle θ) (Fig. 9.10), the rotation $\mathbf{R}_{z,\theta}$ can be derived as follows

$$\mathbf{p}_{xyz} = \mathbf{R}_{z,\theta} \mathbf{p}_{uvw} \quad (9.32)$$

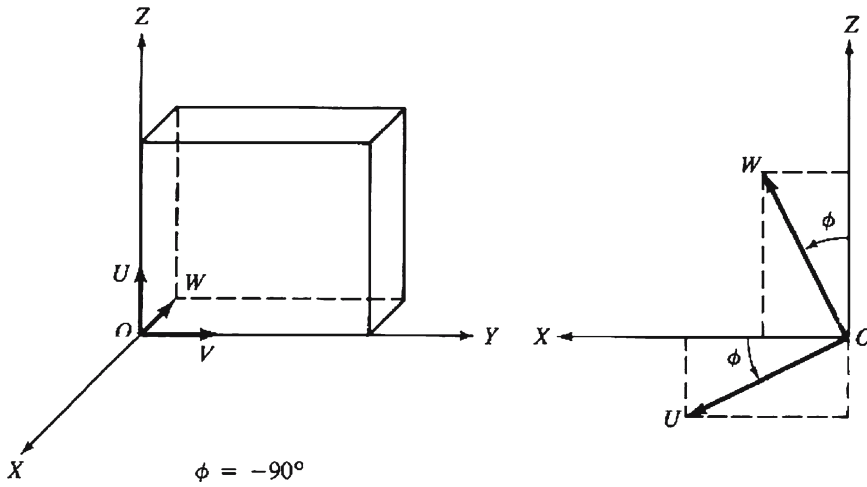


Fig. 9.9 Rotation of the $OUVW$ system about the OY -axis

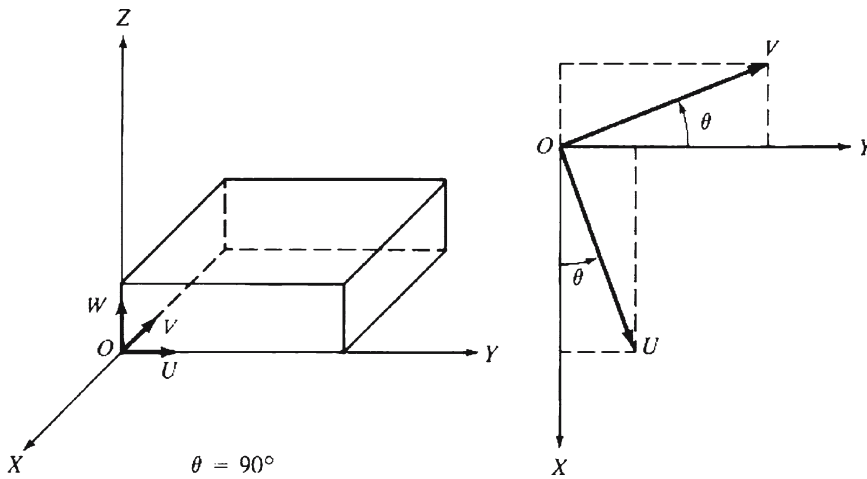


Fig. 9.10 Rotation of the $OUVW$ system about the OZ -axis

$$R_{z,y,\theta} = \begin{bmatrix} \mathbf{i}_x \cdot \mathbf{i}_u & \mathbf{i}_x \cdot \mathbf{j}_v & \mathbf{i}_x \cdot \mathbf{k}_w \\ \mathbf{j}_y \cdot \mathbf{i}_u & \mathbf{j}_y \cdot \mathbf{j}_v & \mathbf{j}_y \cdot \mathbf{k}_w \\ \mathbf{k}_z \cdot \mathbf{i}_u & \mathbf{k}_z \cdot \mathbf{j}_v & \mathbf{k}_z \cdot \mathbf{k}_w \end{bmatrix} = \begin{bmatrix} \cos \theta & -\sin \theta & 0 \\ \sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} C\theta & -S\theta & 0 \\ S\theta & C\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Example 9.1

With respect to the $OUVW$ coordinate system, a point \mathbf{q} in 3-D space is given to be $\mathbf{q}_{uvw} = (3, 6, 4)^T$. Find the equivalent points with respect to the $OXYZ$ coordinate

system, if there has been (1) a 30° rotation about the OX -axis, (2) a 120° rotation about the OY -axis, and (3) a 45° rotation about the OZ -axis (*hint: there will be three different rotation matrices*).

Solution:

(1) With a 30° rotation about the OX -axis, the equivalent point is $\mathbf{q}_{xyz} = \mathbf{R}_{x,30^\circ} \mathbf{q}_{uvw}$ and using the matrix form of Eq. 9.30, where $\alpha = 30^\circ$,

$$\mathbf{R}_{x,30^\circ} = \begin{bmatrix} \mathbf{i}_X \cdot \mathbf{i}_U & \mathbf{i}_X \cdot \mathbf{j}_V & \mathbf{i}_X \cdot \mathbf{k}_W \\ \mathbf{j}_Y \cdot \mathbf{i}_U & \mathbf{j}_Y \cdot \mathbf{j}_V & \mathbf{j}_Y \cdot \mathbf{k}_W \\ \mathbf{k}_Z \cdot \mathbf{i}_U & \mathbf{k}_Z \cdot \mathbf{j}_V & \mathbf{k}_Z \cdot \mathbf{k}_W \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos 30^\circ & -\sin 30^\circ \\ 0 & \sin 30^\circ & \cos 30^\circ \end{bmatrix}$$

$$\mathbf{R}_{x,30^\circ} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0.8660 & -0.5 \\ 0 & 0.5 & 0.8660 \end{bmatrix}$$

$$\therefore \mathbf{q}_{xyz} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0.8660 & -0.5 \\ 0 & 0.5 & 0.8660 \end{bmatrix} \begin{bmatrix} 3 \\ 6 \\ 4 \end{bmatrix} = \begin{bmatrix} 3 \\ 3.1962 \\ 6.4641 \end{bmatrix}$$

(2) With a 120° rotation about the OY -axis, the equivalent point is $\mathbf{q}_{xyz} = \mathbf{R}_{y,120^\circ} \mathbf{q}_{uvw}$ and using the matrix form of Eq. 9.31, where $\phi = 120^\circ$,

$$\mathbf{R}_{y,120^\circ} = \begin{bmatrix} \mathbf{i}_X \cdot \mathbf{i}_U & \mathbf{i}_X \cdot \mathbf{j}_V & \mathbf{i}_X \cdot \mathbf{k}_W \\ \mathbf{j}_Y \cdot \mathbf{i}_U & \mathbf{j}_Y \cdot \mathbf{j}_V & \mathbf{j}_Y \cdot \mathbf{k}_W \\ \mathbf{k}_Z \cdot \mathbf{i}_U & \mathbf{k}_Z \cdot \mathbf{j}_V & \mathbf{k}_Z \cdot \mathbf{k}_W \end{bmatrix} = \begin{bmatrix} \cos 120^\circ & 0 & \sin 120^\circ \\ 0 & 1 & 0 \\ -\sin 120^\circ & 0 & \cos 120^\circ \end{bmatrix}$$

$$\mathbf{R}_{y,120^\circ} = \begin{bmatrix} -0.5 & 0 & 0.8660 \\ 0 & 1 & 0 \\ -0.8660 & 0 & -0.5 \end{bmatrix}$$

$$\therefore \mathbf{q}_{xyz} = \begin{bmatrix} -0.5 & 0 & 0.8660 \\ 0 & 1 & 0 \\ -0.8660 & 0 & -0.5 \end{bmatrix} \begin{bmatrix} 3 \\ 6 \\ 4 \end{bmatrix} = \begin{bmatrix} 1.9641 \\ 6 \\ -4.5981 \end{bmatrix}$$

(c) With a 45° rotation about the OZ -axis, the equivalent point is $\mathbf{q}_{xyz} = \mathbf{R}_{z,45^\circ} \mathbf{q}_{uvw}$ and using the matrix form of Eq. 9.32, where $\theta = 45^\circ$,

$$\mathbf{R}_{z,45^\circ} = \begin{bmatrix} \mathbf{i}_X \cdot \mathbf{i}_U & \mathbf{i}_X \cdot \mathbf{j}_V & \mathbf{i}_X \cdot \mathbf{k}_W \\ \mathbf{j}_Y \cdot \mathbf{i}_U & \mathbf{j}_Y \cdot \mathbf{j}_V & \mathbf{j}_Y \cdot \mathbf{k}_W \\ \mathbf{k}_Z \cdot \mathbf{i}_U & \mathbf{k}_Z \cdot \mathbf{j}_V & \mathbf{k}_Z \cdot \mathbf{k}_W \end{bmatrix} = \begin{bmatrix} \cos 45^\circ & -\sin 45^\circ & 0 \\ \sin 45^\circ & \cos 45^\circ & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$R_{z,45^\circ} = \begin{bmatrix} 0.7071 & -0.7071 & 0 \\ 0.7071 & 0.7071 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\therefore \mathbf{q}_{xyz} = \begin{bmatrix} 0.7071 & -0.7071 & 0 \\ 0.7071 & 0.7071 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 3 \\ 6 \\ 4 \end{bmatrix} = \begin{bmatrix} -2.1213 \\ 6.3639 \\ 4 \end{bmatrix}$$

9.2.1.6 Combination of the Rotation Matrices

The rotation matrices formulated in the previous sections are based on rotation about a single axis. A rotation matrix can be developed to include all rotations of the $OUVW$ body-attached frame about the OX -axis, OY -axis, and OZ -axis, simultaneously. The *order* or *sequence* of rotation about each axis is of critical importance. In other words, a rotation of 30° about the OX -axis then 60° about the OY -axis and 45° about the OZ -axis is *not* equivalent to a rotation of 60° about the OY -axis then a 30° about the OX -axis and a 45° about the OZ -axis. It is also possible for the body-attached reference frame $OUVW$ to rotate about its own axis. It is then necessary to understand that since both coordinate systems are coincident initially, the resulting rotation matrix is a 3×3 identity matrix. Second, if the $OUVW$ system is rotated about one of the principal axes of the $OXYZ$ system, then *premultiply* the resultant rotation matrix with a corresponding rotation matrix. Likewise, if the $OUVW$ system is rotated about any of its principal axes, *postmultiply* the resultant matrix with a corresponding rotation matrix.

Consider the following rotation sequences:

α° about OX -axis $\rightarrow \phi^\circ$ about OY -axis $\rightarrow \theta^\circ$ about OZ -axis.

We may generate the rotation matrix for each case as follows:

$$R = R_{z,\theta} R_{y,\phi} R_{x,\alpha} \begin{bmatrix} C\theta & -S\theta & 0 \\ S\theta & C\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} C\phi & 0 & S\phi \\ 0 & 1 & 0 \\ -S\phi & 0 & C\phi \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & C\alpha & -S\alpha \\ 0 & S\alpha & C\alpha \end{bmatrix}$$

$$R = \begin{bmatrix} C\phi C\theta & S\alpha S\phi C\theta - C\alpha S\theta & C\alpha S\phi C\theta + S\alpha S\theta \\ C\phi S\theta & C\alpha C\theta + S\alpha S\phi S\theta & C\alpha S\phi S\theta + S\alpha C\theta \\ -S\phi & S\alpha S\phi & C\alpha C\phi \end{bmatrix}$$

Example 9.2

Once again referring to *Example 9.1*, formulate the rotation matrix if the given rotations of the $OUVW$ system were in sequence, that is, a 30° rotation about the OX -axis, then a 120° rotation about the OY -axis, and finally a 45° rotation about the OZ -axis. What is the resulting location of the point $\mathbf{q}_{uvw} = (3, 6, 4)^T$ with respect to

the $OXYZ$ -axis after the sequence of rotation (*hint: there is only one rotation matrix computed*).

Solution:

$$\mathbf{R} = R_{z,\theta} R_{y,\phi} R_{x,\alpha} \begin{bmatrix} C\theta & -S\theta & 0 \\ S\theta & C\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} C\phi & 0 & S\phi \\ 0 & 1 & 0 \\ -S\phi & 0 & C\phi \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & C\alpha & -S\alpha \\ 0 & S\alpha & C\alpha \end{bmatrix}$$

Since $\alpha = 30^\circ$, $\phi = 120^\circ$, and $\theta = 45^\circ$, we have

$$\begin{aligned} \mathbf{R}_{uvw} &= R_{z,45} \circ R_{y,120} \circ R_{x,30} \\ &= \begin{bmatrix} C45^\circ & -S45^\circ & 0 \\ S45^\circ & C45^\circ & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} C120 & 0 & S120 \\ 0 & 1 & 0 \\ -S120 & 0 & C120 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & C30^\circ & -S30^\circ \\ 0 & S30^\circ & C30^\circ \end{bmatrix} \\ \mathbf{R}_{uvw} &= \begin{bmatrix} C120^\circ C45^\circ & S30^\circ S120^\circ C45^\circ - C30^\circ S45^\circ & C30^\circ S120^\circ C45^\circ + S30^\circ S45^\circ \\ C120^\circ S45^\circ & C30^\circ C45^\circ + S30^\circ S120^\circ S45^\circ & C30^\circ S120^\circ S45^\circ + S30^\circ C45^\circ \\ -S120^\circ & S30^\circ S120^\circ & C30^\circ C120^\circ \end{bmatrix} \\ \mathbf{R}_{uvw} &= \begin{bmatrix} -0.3536 & -0.3062 & 0.8839 \\ -0.3536 & 0.9186 & 0.1768 \\ -0.8660 & -0.25 & -0.4330 \end{bmatrix} \end{aligned}$$

Therefore, the point \mathbf{q}_{uvw} with respect to the $OXYZ$ -axis after the sequence of rotation can be found by Eq. 9.14, where

$$\mathbf{q}_{xyz} = \mathbf{R}_{uvw} \mathbf{q}_{uvw} = \begin{bmatrix} -0.3536 & -0.3062 & 0.8839 \\ -0.3536 & 0.9186 & 0.1768 \\ -0.8660 & -0.25 & -0.4330 \end{bmatrix} \begin{bmatrix} 3 \\ 6 \\ 4 \end{bmatrix} = \begin{bmatrix} 0.6378 \\ 5.1578 \\ -5.8301 \end{bmatrix}$$

(Note that multiplying the rotation matrix result in Example 1.1 (3), (2), and (1), respectively, will result in the rotation matrix above.)

9.3 Homogenous Representation

The representation of an n -component position vector by an $(n + 1)$ -component vector is called homogeneous coordinate representation [13]. The transformation of an n -dimensional vector is executed in the $(n + 1)$ -dimensional space and the physical n -dimensional is thus obtained by dividing the homogeneous coordinates by the $(n + 1)$ th coordinate, u . Given a position vector $\mathbf{p} = (p_x, p_y, p_z)^T$ in 3-D space, the homogenous coordinate representation can be depicted by an augmented vector

$\mathbf{p} = (up_x, up_y, up_z, u)^T$. The fourth component of the homogenous matrix can then be considered a scaling factor or multiplier. When $u = 1$,

$$\mathbf{p} = (p_x/u, p_y/u, p_z/u, 1)^T = (p_x, p_y, p_z, u)^T \quad (9.33)$$

The transformed homogenous coordinates of a position vector are the same as the physical coordinates of the vector. It is most useful in practice to let $u = 1$ where the homogeneous representation can be used equivalently as the physical coordinates vector. This will aid in fully defining all the possible transformations that can occur to a position vector. The homogeneous transformation matrix, \mathbf{T} (also known as the T-matrix), can be expressed as a composition of four submatrices

$$\mathbf{T} = \begin{bmatrix} \text{Rotation}_{(3 \times 3)} & \text{Positional}_{(3 \times 1)} \\ \text{Perspective}_{(1 \times 3)} & \text{Scaling}_{(1 \times 1)} \end{bmatrix} \quad (9.34)$$

The composition of these four submatrices can be expressed as a 4×4 matrix. The following are the 4×4 representation of the submatrices:

• **Basic Homogeneous Rotational Matrices:**

$$\begin{aligned} \mathbf{T}_{R(x,\alpha)} &= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & C & -S & 0 \\ 0 & S & C & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \\ \mathbf{T}_{R(y,\phi)} &= \begin{bmatrix} C & 0 & S & 0 \\ 0 & 1 & 0 & 0 \\ -S & 0 & C & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \mathbf{T}_{R(z,\theta)} = \begin{bmatrix} C & -S & 0 & 0 \\ S & C & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \end{aligned} \quad (9.35)$$

(Note: $C = \cos\theta$ and $S = \sin\theta$.)

• **Basic Homogeneous Translation Matrix:**

$$\mathbf{T}_{\text{trans}} = \begin{bmatrix} 1 & 0 & 0 & \Delta_x \\ 0 & 1 & 0 & \Delta_y \\ 0 & 0 & 1 & \Delta_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (9.36)$$

Another transformation will be necessary to define scaling. Scaling of a coordinates system is the proportion that the transformed coordinate system has to the original coordinate system (reference coordinate system). Scaling is performed at each axis. If we define the scaling factor along each axis as s_x , s_y , and s_z , then the transformation matrix is

$$\mathbf{p}' = \mathbf{s}^T \mathbf{p} = [s_x s_y s_z] \mathbf{p} \quad (9.37)$$

where \mathbf{s} is the scaling vector for the x , y , z components of the original coordinate system. The 4×4 matrix representation of the scaling factor known as the basic homogenous scaling matrix is

$$\mathbf{T}_s = \begin{bmatrix} S_x & 0 & 0 & 0 \\ 0 & S_y & 0 & 0 \\ 0 & 0 & S_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (9.38)$$

Therefore, the appropriate combination of any of the 4×4 homogenous transformation matrices maps a vector denoted in the homogenous coordinates in relation to the $OUVW$ coordinate system (body-attached frame) to the reference coordinate system $OXYZ$ and can be given by

$$\mathbf{T} = \begin{bmatrix} n_x & s_x & a_x & p_x \\ n_y & s_y & a_y & p_y \\ n_z & s_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (9.39)$$

9.4 Forward or Direct Kinematics

Assuming that the length of the robot links and the joint angles are all known, then the position of the end effector with respect to the reference coordinate system can be computed using forward kinematics analysis. Vector and matrix algebra are utilized to develop a systematic and generalized approach to describe and represent the location of the link of a robot arm with respect to a fixed reference frame. Since the links of a robot arm are allowed to rotate and/or translate with respect to a reference coordinate frame, a body-attached coordinate frame will be established along the joint axis for each link. This will significantly simplify computation.

To analyze forward kinematics, vector analysis and transformation must be discussed. The end effector can be displaced within its workspace through rotations and translations of the links with respect to the reference coordinate frame. We shall also significantly minimize computations of link transformations by utilizing matrix algebra. In 1955, the duo of Denavit and Hartenberg² proposed a method by using matrix algebra to describe and represent the spatial geometry of the links of a robot arm with respect to a fixed reference frame.

²The Denavit and Hartenberg research and proposals were published in the 1955 edition of the American Society of Mechanical Engineers (ASME) *Journal of Applied Mechanics*. The famous technique is widely used in robotics and referred to as the D-H model.

The direct kinematics problem is reduced to finding a transformation matrix that relates the body-attached coordinate frame, that is, the $OUVW$ coordinate system, to the reference coordinate frame, in this case the $OXYZ$ coordinate system. A 3×3 rotation matrix is used to describe the rotational operations of the body-attached frame with respect to the reference frame. The homogenous coordinates are used to represent position vectors in a 3-D space, and the rotation matrices are expanded to 4×4 homogenous transformation matrices to include the translational operations of the body-attached coordinate frame in order to position and orient the rigid body in space.

If we consider an orthonormal coordinate system composed of unit vectors (\mathbf{x}_i , \mathbf{y}_i , \mathbf{z}_i), then we can use this system to denote the base coordinate frame, all links, and its corresponding joint axis, where $i = 1, 2, \dots, n$ and n represents the number of degrees of freedom. When the joint i is set in motion via an actuator, link i is displaced in accordance to the motions of link $i - 1$. Given that the i th coordinate system is fixed in link i , it is displaced along with the link i . Note that a rotary joint has one degree of freedom and every (\mathbf{x}_i , \mathbf{y}_i , \mathbf{z}_i) coordinate frame of a robot arm corresponds to joint $i + 1$ and is stationary in relation to link i . Also the n th coordinate frame will be displaced along with the n th link, which incidentally is the end effector. We have previously labeled the 0th link as the base link and hence the 0th coordinate frame is then the base coordinate frame denoted as (\mathbf{x}_0 , \mathbf{y}_0 , \mathbf{z}_0).

According to the three rules given by Fu et al. [13], every coordinate frame is determined and established on the basis of the following:

1. The \mathbf{z}_{i-1} axis lies along the axis of motion of the i th joint.
2. The \mathbf{x}_i axis is normal to the \mathbf{z}_{i-1} axis, and pointing away from it.
3. The \mathbf{y}_i axis completes the right-handed coordinate system as required.

From these rules, the location of the base coordinate frame 0 can be placed at any region within the base so far \mathbf{z}_0 axis lies along the axis of motion of the first joint (see Fig. 9.11).

The D-H representation of a rigid link depends on four geometric parameters associated with each link. These four parameters completely describe any revolute or prismatic joint [13]. They are defined as follows:

- θ_i = the joint angle from \mathbf{x}_{i-1} axis to the \mathbf{x}_i axis about the \mathbf{z}_{i-1} axis (using the right-hand rule)
- d_i = the distance from the origin of the $(i - 1)$ th coordinate frame to the intersection of the \mathbf{z}_{i-1} axis with the \mathbf{x}_i axis along the \mathbf{z}_{i-1} axis
- a_i = the offset distance from the intersection of the \mathbf{z}_{i-1} axis with the \mathbf{x}_i axis to the origin of the i th frame along the \mathbf{x}_i axis (or the shortest distance between the \mathbf{z}_{i-1} and \mathbf{z}_i axes)
- α_i = the offset angle from the \mathbf{z}_{i-1} axis to the \mathbf{z}_i axis about the \mathbf{x}_i axis (using the right-hand rule)
- In the case of a rotary joint, d_i , a_i , and α_i are the joint parameters which will be constant for a robot while θ_i is the joint variable that changes when link i is displaced or rotated in relation to the link $i - 1$.

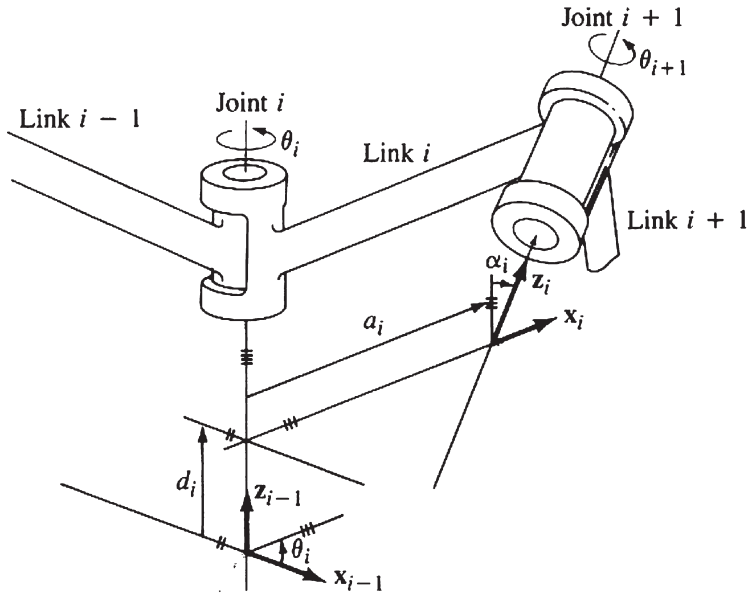


Fig. 9.11 Link coordinate system and its parameters (Courtesy-Fu et al. [13])

- In the case of a prismatic joint, θ_i , a_i , and α_i are the joint parameters and remain constant for a robot, while d_i is the joint variable.

Upon determining the D-H coordinate system for each link, a homogenous transformation matrix can easily be developed relating the i th coordinate frame to the $(i - 1)$ th coordinate frame. Consequently, the point \mathbf{r}_i denoted in the i th coordinate system can be denoted in the $(i - 1)$ th coordinate system as \mathbf{r}_{i-1} . This can be achieved by the following (refer to Fig. 9.12):

- Rotate about the \mathbf{z}_{i-1} axis an angle of θ to align the \mathbf{x}_{i-1} axis with the \mathbf{x}_i axis. The \mathbf{x}_{i-1} axis is parallel to \mathbf{x}_i and directed to the same direction.
- Translate along \mathbf{z}_{i-1} axis a distance of d_i to bring the \mathbf{x}_{i-1} and \mathbf{x}_i axes into coincidence.
- Translate along the \mathbf{x}_i axis a distance of a_i to bring the two origins as well as the x -axis into coincidence.
- Rotate about the \mathbf{x}_i axis an angle of α_i to bring the two coordinate systems into coincidence.

We can thus represent each of these four processes by a basic homogenous rotation-translation matrix. Multiplying these matrices will result in a composite homogenous transformation matrix, ${}^{i-1}\mathbf{A}_i$, known as the D-H transformation matrix for adjacent coordinate frames, i and $i - 1$. We then have

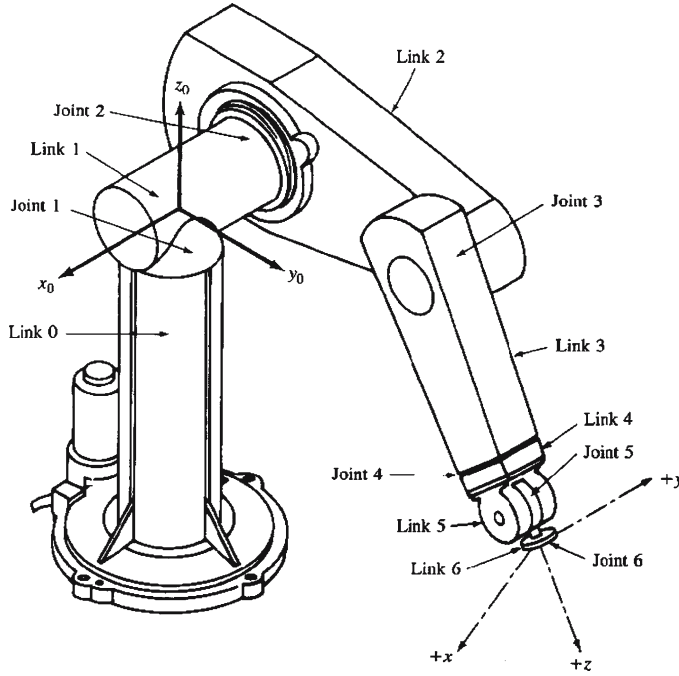


Fig. 9.12 A PUMA robot arm with joints and links (Courtesy-Fu et al. [13])

$${}^{i-1}\mathbf{A}_i = \mathbf{T}_{z,d} \cdot \mathbf{T}_{z,\theta} \cdot \mathbf{T}_{x,\alpha} \cdot \mathbf{T}_{x,\alpha} \quad (9.40)$$

$${}^{i-1}\mathbf{A}_i = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos\theta_i & -\sin\theta_i & 0 & 0 \\ \sin\theta_i & \cos\theta_i & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & a_i \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\alpha_i & -\sin\alpha_i & 0 \\ 0 & \sin\alpha_i & \cos\alpha_i & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^{i-1}\mathbf{A}_i = \begin{bmatrix} \cos\theta_i & -\cos\alpha_i \sin\theta_i & \sin\alpha_i \sin\theta_i & a_i \cos\theta_i \\ \sin\theta_i & \cos\alpha_i \sin\theta_i & -\sin\alpha_i \sin\theta_i & a_i \sin\theta_i \\ 0 & \sin\alpha_i & \cos\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (9.41)$$

If we consider a revolute joint where α_i , a_i , and d_i are constant and θ_i is the joint variable for a revolute joint, then the inverse of the above transformation matrix can be found to be

$$[{}^{i-1}\mathbf{A}_i]^{-1} = {}^i\mathbf{A}_{i-1} = \begin{bmatrix} \cos\theta_i & \sin\theta_i & 0 & -a_i \\ -\cos\alpha_i \sin\theta_i & \cos\alpha_i \sin\theta_i & \sin\alpha_i & -d_i \sin\alpha_i \\ \sin\alpha_i \sin\theta_i & -\sin\alpha_i \cos\theta_i & \cos\alpha_i & -d_i \cos\alpha_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (9.42)$$

Likewise for a prismatic joint, the joint variable is d_i , while α_i , a_i , and θ_i are the constants.

$${}^{i-1}\mathbf{A}_i = \mathbf{T}_{z,\theta} \cdot \mathbf{T}_{z,d} \cdot \mathbf{T}_{x,\alpha} = \begin{bmatrix} \cos\theta_i & -\cos\alpha_i \sin\theta_i & \sin\alpha_i \sin\theta_i & 0 \\ \sin\theta_i & \cos\alpha_i \cos\theta_i & -\sin\alpha_i \cos\theta_i & 0 \\ 0 & \sin\alpha_i & \cos\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (9.43)$$

The inverse matrix for the prismatic joint is

$$[{}^{i-1}\mathbf{A}_i]^{-1} = {}^i\mathbf{A}_{i-1} = \begin{bmatrix} \cos\theta_i & \sin\theta_i & 0 & 0 \\ \cos\alpha_i \sin\theta_i & \cos\alpha_i \cos\theta_i & \sin\alpha_i & d_i \sin\alpha_i \\ \sin\alpha_i \sin\theta_i & -\sin\alpha_i \cos\theta_i & \cos\alpha_i & -d_i \cos\alpha_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (9.44)$$

From the $[{}^{i-1}\mathbf{A}_i]^{-1}$ matrix we can relate a point \mathbf{p}_i at rest in link i , and expressed in homogeneous coordinates with respect to the coordinate system i , to the coordinate system $i-1$ established at link $i-1$ by

$$\mathbf{p}_{i-1} = {}^{i-1}\mathbf{A}_i \mathbf{p}_i \quad (9.45)$$

where $\mathbf{p}_{i-1} = (\mathbf{x}_{i-1}, \mathbf{y}_{i-1}, \mathbf{z}_{i-1}, 1)^T$ and $\mathbf{p}_i = (\mathbf{x}_i, \mathbf{y}_i, \mathbf{z}_i, 1)^T$

9.5 Reverse of Indirect Kinematics

In inverse kinematics analysis, if the position of the end effector with respect to the reference coordinate system is known, then the robot link parameters, variables, and joint angles can be calculated. It is the inverse of direct kinematics and can be solved using proven methods, namely, inverse transform, decoupling technique, [AU1] inverse transform technique, screw algebra, dual matrices, dual quaternion, and a host of other techniques. We shall concentrate on the *Inverse Transform Technique*, which was developed by Peiper in 1968.

If we have the transformation matrix ${}^0\mathbf{A}_6$ indicating the global position and the orientation of the end effector of a six degree of freedom robot in the base frame 0,

then we can proceed to determining the joint variables and links positions. Assume that the geometry and individual transformation matrices ${}^0\mathbf{A}_1(q_1)$, ${}^1\mathbf{A}_2(q_2)$, ${}^2\mathbf{A}_3(q_3)$, ${}^3\mathbf{A}_4(q_4)$, ${}^4\mathbf{A}_5(q_5)$, and ${}^5\mathbf{A}_6(q_6)$ are given as the joint variables. Note that the inverse kinematics is attempting to determine the elements of vector \mathbf{q} when a transformation is given as a function of the joint variables $q_1, q_2, q_3, \dots, q_n$, then we have, ${}^0\mathbf{T}_n = {}^0\mathbf{A}_1(q_1) {}^1\mathbf{A}_2(q_2) {}^2\mathbf{A}_3(q_3) {}^3\mathbf{A}_4(q_4) \dots {}^{n-1}\mathbf{A}_n(q_n)$. Therefore, the problem is to find what the required values of joint variables are to reach a desired point in a desired orientation. The articulated arm transformation matrix derived from forward kinematics for a six degree of freedom robot is given as [15]

$$\mathbf{T}_6 = \begin{bmatrix} n_x & s_x & a_x & p_x \\ n_y & s_y & a_y & p_y \\ n_z & s_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} = {}^0\mathbf{A}_1 {}^1\mathbf{A}_2 {}^2\mathbf{A}_3 {}^3\mathbf{A}_4 {}^4\mathbf{A}_5 {}^5\mathbf{A}_6 \quad (9.46)$$

The inverse kinematics problem is given as the following:

$$\begin{aligned} {}^1\mathbf{T}_6 &= {}^0\mathbf{A}_1^{-1} {}^0\mathbf{A}_6 \\ {}^2\mathbf{T}_6 &= {}^1\mathbf{A}_2^{-1} {}^1\mathbf{A}_1^{-1} {}^1\mathbf{A}_6 \\ {}^3\mathbf{T}_6 &= {}^2\mathbf{A}_3^{-1} {}^2\mathbf{A}_2^{-1} {}^2\mathbf{A}_1^{-1} {}^2\mathbf{A}_6 \\ {}^4\mathbf{T}_6 &= {}^3\mathbf{A}_4^{-1} {}^3\mathbf{A}_3^{-1} {}^3\mathbf{A}_2^{-1} {}^3\mathbf{A}_1^{-1} {}^3\mathbf{A}_6 \\ {}^5\mathbf{T}_6 &= {}^4\mathbf{A}_5^{-1} {}^4\mathbf{A}_4^{-1} {}^4\mathbf{A}_3^{-1} {}^4\mathbf{A}_2^{-1} {}^4\mathbf{A}_1^{-1} {}^4\mathbf{A}_6 \\ \mathbf{I} &= {}^5\mathbf{A}_6^{-1} {}^5\mathbf{A}_5^{-1} {}^5\mathbf{A}_4^{-1} {}^5\mathbf{A}_3^{-1} {}^5\mathbf{A}_2^{-1} {}^5\mathbf{A}_1^{-1} {}^5\mathbf{A}_6 \end{aligned} \quad (9.47)$$

9.6 Robot Vision

Robot vision may be defined as the process of extracting, characterizing, and interpreting information from images of a 3-D world. This process is also known as computer or machine vision and subdivided into six principal areas: (1) sensing, (2) preprocessing, (3) segmentation, (4) description, (5) recognition, and (6) interpretation. The use of machine vision is motivated by the continuing need to increase the flexibility and scope of applications of robotic systems [6]. Generally, there are three tasks involved in robot vision, namely, *image transformation* or *image processing*, *image analysis*, and *image interpretation*. Image analysis is the collection of processes in which a captured image that is prepared by image processing is analyzed in order to extract information about the image and to identify objects or facts [20]. Niku further defines *image processing* as the collection of routines and techniques that improve, simplify, enhance, or, otherwise, alter an image.

9.6.1 Image Transformation/Image Processing

Image transformation involves the conversion of light energy in digital data for processing. Images are converted using equipment such as cameras, photodiodes array, charge-injection device (CID) array, and charged-coupled device (CCD) array. A 3-D image processing involves operations that require motion detection, depth measurement, remote sensing, relative positioning, and navigation. *Spatial digitization* is a process in which intensities of light at each pixel location are read while *sampling* is a technique whereby the more pixels that are present and individually read, the better the resolution of the camera and the image. These two processes are important in image recognition and edge detection.

9.6.1.1 Histogram

A histogram is a representation of the total number of pixels of an image at each gray level. Histograms are applied in a variety of processes which can be used to determine the cutoff point when an image should be transformed to binary levels such as thresholding. These are also used for noise reduction by determining what the noisy gray level is in order to attempt to remove or neutralize the noise.

9.6.1.2 Thresholding

Thresholding is the process of dividing an image into different portions by selecting a certain grayness level as a threshold, comparing each pixel value with the threshold, and then assigning the pixel to the different portions or levels, depending on whether the pixel's grayness level is below threshold(off) or above the threshold(on) [20].

9.6.1.3 Connectivity

Connectivity is used to determine whether adjacent pixels are related to one another. It does so by ascertaining the properties of each pixel. These properties include but are not limited to pixel region, texture, and object of interest. To establish connectivity of neighboring pixels, a connectivity path must be determined (Fig. 9.13).

The 3-D connectivity between voxels (volume cells) can range from 6 to 26 while the three basic connectivity paths for 2-D image processing and analysis are ± 4 -connectivity or $\times 4$ -connectivity, $H6$ or $V6$ connectivity, and 8-connectivity. Refer to Fig. 9.13 for the following definitions:

<i>a</i>	<i>b</i>	<i>c</i>
<i>d</i>	<i>p</i>	<i>e</i>
<i>f</i>	<i>g</i>	<i>h</i>

Fig. 9.13 Neighborhood connectivity

+4-connectivity: When a pixel is analyzed only with respect to the four pixels immediately above, below, to the left, and to the right of p (b, g, d, e), it is a *+4-connectivity*. For a pixel $p(x,y)$, the relevant pixels are

$$(x + 1,y),(x - 1,y),(x,y - 1),(x,y + 1); \quad (9.48)$$

×4-connectivity: A pixel's p 's relationship is analyzed only with respect to the four pixels immediately across from it diagonally on four sides $p(a, c, f, h)$. For a pixel $p(x,y)$, the relevant pixels are

$$(x + 1,y + 1),(x + 1,y - 1),(x - 1,y + 1),(x - 1,y - 1); \quad (9.49)$$

H6-connectivity: A pixel's p 's relationship is analyzed only with respect to the six neighboring pixels on the two rows immediately above and below p (a, b, c, f, g, h). For a pixel $p(x,y)$, the relevant pixels are

$$(x - 1,y + 1),(x,y - 1),(x + 1,y + 1),(x - 1,y - 1),(x,y - 1),(x + 1,y - 1); \quad (9.50)$$

V6-connectivity: A pixel's p 's relationship is analyzed only with respect to the six neighboring pixels on the two columns immediately to the right and to the left of p (a, d, f, c, e, h). For a pixel $p(x,y)$, the relevant pixels are

$$(x - 1,y + 1),(x - 1,y),(x - 1,y - 1),(x + 1,y + 1),(x + 1,y),(x + 1,y - 1); \quad (9.51)$$

8-connectivity: A pixel's p 's relationship is analyzed with respect to all eight pixels surrounding it (a, b, c, d, e, f, g, h). For a pixel $p(x,y)$, the relevant pixels are

$$(x - 1,y - 1),(x,y - 1),(x + 1,y - 1),(x - 1,y),(x + 1,y),(x - 1,y + 1),(x,y + 1),(x + 1,y + 1); \quad (9.52)$$

9.6.1.4 Noise Reduction

In any vision system, noise (unwanted visual distortions) is always a major obstacle to overcome. Many noise reduction algorithms have been written but no matter

how effective and efficient they are, 100% efficiency is almost unattainable. Noise in images is generated by inaccurate data processing from electronic components, lens scratches, dust deposits, faulty memory storage devices, illumination, and a host of others. Noise reduction and filtering techniques are generally divided into two categories-*frequency-related* and *spatial-domain* [20]. *Frequency-related* techniques operate a Fourier transform of the signal, while the *spatial-domain* techniques operate on the image at the pixel level, either locally or globally. Some techniques in noise reduction are image averaging, edge detection, neighborhood averaging, etc.

9.6.1.5 Image Averaging

Image averaging employs a technique whereby a number of images of the same scene are averaged together. Consider a case where an image acquisition is referred to as $A(x,y)$. The image can be divided into two distinct groups, the first being $I(x,y)$, which represents the desired image, and $N(x,y)$, which represents the random noise. The desired image can be found by the following equation:

$$A(x,y) = I(x,y) + N(x,y) \quad (9.53)$$

$$\frac{\sum_n A(x,y)}{n} = \frac{\sum_n I(x,y) + N(x,y)}{n} = \frac{\sum_n I(x,y)}{n} + \frac{\sum_n N(x,y)}{n} = I(x,y) \quad (9.54)$$

If the term $\sum_n \frac{N(x,y)}{n} = 0$, then Eq. 9.54 will be written as

$$\frac{\sum_n A(x,y)}{n} = \frac{\sum_n I(x,y) + N(x,y)}{n} = \frac{\sum_n I(x,y)}{n} = I(x,y) \quad (9.55)$$

9.6.1.6 Edge Detection

A step edge in an image is an image intensity contour across which the brightness of the image changes abruptly. These are frequently associated with the projection of actual object boundaries in the scene [11]. Edges provide a compact representation of the image and can further provide an expressive representation of the salient image feature as well. Edge detection is also necessary in subsequent processes such as segmentation and object recognition. Without the techniques of edge detection to improve image quality, it is extremely difficult, if not impossible, to find overlapping parts.

9.6.1.7 Neighborhood Averaging

This technique of noise reduction has a significant disadvantage because it reduces the sharpness of the image. To partially eliminate this, other averaging filters such as the Gaussian averaging filter, also called the mild isotropic low-pass filter, can be employed. This filter improve the image quality but to a limited extent.

9.6.1.8 Hough Transform

The Hough transform is a technique used to determine the geometric relationship between different pixels on a line, including the slope of a line (Niku et al.). Consider a problem of detecting straight lines in an image from a collection of edge l (edge elements) measurements (x_i, y_i) . If the edge is represented as $y = mx + b$, the measurement (x_i, y_i) provides support for the set of lines that pass through (x_i, y_i) , or equivalently, for the set of values (m, b) that satisfy $y_i = mx_i + b$. The Hough transform provides a voting scheme to combine individual (x_i, y_i) measurements to obtain (m, b) [11]. The Hough-transformation expressed as an integral transformation is [25]

$$H(\mathbf{p}) = \iint_D f(x, y) \delta(g(x, y, \mathbf{p})) dx dy \quad (9.56)$$

This maps all the information of an edge-filtered image $\mathbf{F}(x, y)$ to the parameter space of the search pattern. The elements of $\mathbf{F}(x, y)$ are the gradient magnitude $f(x, y)$ and the gradient direction $\phi(x, y)$. The adaptation is performed through the description $g(x, y, \mathbf{p})$, where \mathbf{p} is a parameter vector that is to be determined by the feature extraction process and $H(\mathbf{p})$ is the Hough accumulator which points to corresponding features in the image.

9.6.2 Vision and Image Processing for Autonomous Robots

Vision systems for autonomous mobile robots must unify the requirements and demands of the very challenging disciplines, namely [24]

1. computer vision and image processing and
2. robotics and embedded systems.

9.6.2.1 Timeliness Constraint

Since autonomous robots in a relatively complex environment perform “simple” tasks similar to obstacle detection and object tracking, they must have image-processing software capable of calculating and processing the data at a high frequency such as 30 Hz or more. Whenever possible, image-processing operations

should be executed in parallel in order to fully exploit the available resources such as dual-CPU boards and hyperthreading and multicore processor technologies.

9.6.2.2 Fixed Frame Rate Image Streams

In most cases, images are acquired at a fixed frame rate. If there is a fluctuation in data acquire in a dynamic environment, trigger devices should be built to evaluate and compensate for these changes.

9.6.2.3 Development Model

Developing a model for autonomous robots poses its own challenges. Since the robot vision is performed on live image streams, which are recorded by a moving robot platform, the development starts on sets of test images and recorded test image streams. If the application domain implies nondeterminism, or if the robot's actions affect the quality of sensor data by inducing effects like motion blur, the vision system needs to be tested rigorously in a realistic environment.

9.6.2.4 Depth Measurement with Vision Systems

There are two methods of depth measurement in a vision system.

1. The first method uses the range finders in unison with a vision system and image-processing technique. In this combination, the scenes acquired are analyzed with respect to the data received by the range finders about the distances of different portions of an environment or the location of particular objects in that environment.
2. The second method makes use of binocular or stereo vision. In this method, there are two cameras simultaneously capturing a scene which, when processing analyzed, provides a perspective view of the environment in order to ascertain the depth.

9.6.3 Position Visual Servoing (Robot Visual Control)

Visual feedback to control a robot is commonly termed *visual servoing*. In terms of manipulation, one of the main motivations for incorporating vision in the control loop was the demand for increased flexibility of robotic systems [16]. In *open-loop* robot control, initialization represents the extraction of features and characteristics that are used to directly generate the control sequence. In this case, there is no online interaction between the robot and the environment. Conversely, in the close-loop, control of a robot system vision is used as the integral sensory component consisting of tracking and control. The visual servoing system consists of *initialization*,

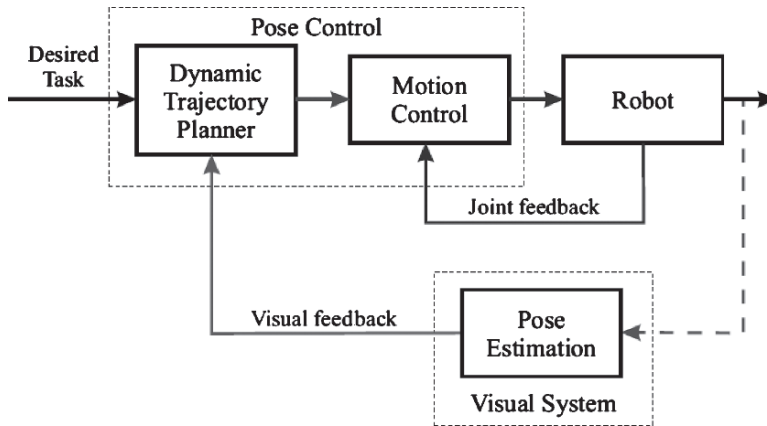


Fig. 9.14 Block scheme of a position-based visual servoing algorithm [19]

tracking, and *control*. *Initialization* occurs when the visual servoing sequence is initialized. *Tracking*, as the name implies, continuously updates the location of features and characteristics used for robot control while *robot controls* are based on the sensory input and the controls sequence of generated signals. Figure 9.14 depicts a typical position-based servoing scheme for an industrial robot.

This algorithm requires the estimation of the *pose* (position) of the target object with respect to a reference frame by using the vision system. The estimated pose is then fed back to a pose controller which performs two main operations, namely, *pose control* and *pose estimation* [19]. The main obstacle of the position-based algorithms is the real-time estimation of the pose of target objects from visual measurements.

9.6.3.1 Pose Control

Pose control is performed through an inner–outer control loop. The inner loop implements motion control, which is an independent joint control or any kind of joint space or task space control. The *dynamic trajectory planner* block (outer loop) calculates the trajectory for the end effector on the basis of the current object pose and task desired.

9.6.3.2 Pose Estimation

Pose estimation algorithm provides the measurement of the target object pose. The use of a multi-camera system requires the adoption of intelligent and computationally efficient strategies for the management of highly redundant information whereby a large number of objects image features from multiple points of view *estimation* [19].

Real-time constraints must be placed on these tasks and therefore all available visual information cannot be extracted and interpreted.

9.7 Conclusion

As discussed, there are numerous applications of autonomous robots primarily in the manufacturing and production industries. Other industries, such as the military, healthcare, and space, are gradually noticing the economic and innovative potential of applications and development of autonomous robots. Its utilization considerably improves the versatility and perceptual ability in tasks, thus expanding its relevance.

This chapter briefly highlighted the overview of autonomous robots and also illustrated some fundamental theories in robotics. Future research is geared toward improving robot autonomy by increasing its decision-making ability through the development of various innovative techniques in vision, control and sensing, and computing. Exceptional advances in robotics have been witnessed in the area of *advanced medical robotics*. Operations research methodologies are also implemented harmoniously with autonomous robots to improve efficiency and productivity in *flexible manufacturing systems*.

References

- 1 Agrawal, A. (1986), Robot Eye-In-Hand Using Fibre Optics, International Trends in Manufacturing Technology: Robot Sensors, Vol. 1-Vision, pp. 115–126.
- 2 Ansoerge, D., Koller, A. (1996), Intelligent Decentralized Planning and Complex Strategies for Negotiation in Flexible Manufacturing Environments, Proceedings of the Sixth International Symposium on Robotics and Manufacturing, Vol. 6, pp. 1–6, May 28–30.
- 3 Asada, M., Hosoda, K., Suzuki, S. (1998), Vision-Based Behavior and Learning and Development for Emergence of Robot Intelligence, Robotics Research: The Eighth International Symposium, pp. 327–338.
- 4 Baird, H. S., Morris, J. (1986), Precise Robotic Assembly using Vision in the Hand, International Trends in Manufacturing Technology: Robot Sensors, Vol. 1-Vision, pp. 85–111.
- 5 Birk, A. (1996), Robotic Control Via Stimulus Response Learning, Proceedings of the Sixth International Symposium on Robotics and Manufacturing, Vol. 6, pp. 13–18, May 28–30.
- 6 Chang, T. C., Wysk, R. A., Wang, H. P. (2006), Computer-Aided Manufacturing, Pearson Education, Inc.
- 7 Chatila, R., Khatib, M. (1998), Interleaving Motion Planning and Execution for Mobile Robots, Robotics Research: The Eighth International Symposium, pp. 124–135.
- 8 Choomuang, R., Afzulpurkar, N. (2005), Hybrid Kalman Filter/Fuzzy Logic Based Position Control of Autonomous Mobile Robot, International Journal of Advanced Robotics Systems Vol. 2, No 3, pp. 197–208.
- 9 Corke, P. I. (1996), Visual Control of Robots: High-Performance Visual Servoing, John Wiley & Sons, New York.
- 10 Dillman, R., Weckesser, P. (1998), Exploration of Unknown Environments with a Mobile Robot using Multisensorfusion, Robotics Research: The Eighth International Symposium, pp. 215–236.

- 11 Dudek, G., Jenkin, M. (2000), *Computational Principles of Mobile Robotics*, Cambridge University Press.
- 12 Faraz, A., Payandeh, S. (2000), *Engineering Approaches to Mechanical and Robotic Design for Minimally Invasive Surgeries*, Kluwer Academic Publishers.
- 13 Fu, K. S., Gonzalez, R. C., Lee, C. S. G. (1987), *Robotics: Control, Sensing, Vision and Intelligence*, McGraw-Hill Book Co., New York.
- 14 Ikuta, K. (1998), Robot Assisted Surgery and Training for Future Minimally Invasive Therapy, *Robotics Research: The Eighth International Symposium*, pp. 299–316.
- 15 Jazar, R. N. (2007), *Theory of Applied Robotics: Kinematics, Dynamics, and Control*, Springer LLC, 2007.
- 16 Kragic, D., Christensen, H. I., Center for Autonomous Systems, Numerical Analysis and Computer, Stockholm, Sweden.
- 17 Lahouar, S., Zeghloul, S., Romdhane, L. (2006), Real-Time Path Planning for Multi-DoF Manipulators in Dynamic Environment, *International Journal of Advanced Robotic Systems*, Vol. 3, No. 2, pp. 125–132.
- 18 Lambert, A., Lévêque, O. (1996), Planning Autonomous Vehicle Displacement Mission in Closed Loop Form, *Proceedings of the Sixth International Symposium on Robotics and Manufacturing*, Vol. 6, pp. 385–392, May 28–30.
- 19 Lippiello, V., Siciliano, B. (2006), Position-Based Visual Servoing in Industrial Multi-Robot Cells Using a Hybrid Camera Configuration, *IEEE*, Revised August.
- 20 Niku, S. B. (2001), *Introduction to Robotics Analysis, Systems, Applications*, Prentice-Hall International, Upper Saddle River, NJ.
- 21 Pinson, L.J. (1986), Robot Sensors: An Evaluation of Imaging Sensors, *International Trends in Manufacturing Technology: Robot Sensors*, Vol. 1-Vision, pp. 15–63.
- 22 Rizzi, A. A., Hollis, R. L. (1998), Opportunities for Increased Intelligence and Autonomy in Robotic Systems for Manufacturing, *Robotics Research: The Eighth International Symposium*, pp. 141–151.
- 23 Topalova, I. (2005), Modular Adaptive System Based on a Multi-Stage Neural Structure for Recognition of 2D objects of Discontinuous Production, *International Journal of Advanced Robotics Systems*, Vol. 2, No 1. pp. 45–51.
- 24 Utz, H., Kaufmann, U., Mayer, G., Kraetzschmar, G.K. (2006), VIP-A Framework-Based Approach to Robot Vision, *International Journal of Advanced Robotics Systems*, Vol. 3, No 1. pp. 67–72.
- 25 Vincze, M., Hager, G. D. (2000), *Robust Vision for Vision-Based Control of Motion*, The Institute of Electrical and Electronics Engineers, Inc., New York.