

## BAB 2

### LANDASAN TEORI

#### 2.1 Teori-teori Dasar

Untuk dapat mengerti tentang kinematika dalam bidang industri pengecatan mobil yang berhubungan dengan robotika, terdapat beberapa pengetahuan umum dalam matematika yang perlu diketahui dan harus diperhatikan.

##### 2.1.1 Definisi Robot

Istilah "*robot*" berasal dari perkataan Czech "*robota*", yang berarti "kerja". Kamus besar *Webster* memberikan definisi mengenai robot, yaitu "Sebuah peralatan otomatis yang dapat melakukan pekerjaan seperti apa yang dilakukan oleh manusia".

##### 2.1.2 Klasifikasi Umum Robot

Berdasarkan fungsinya robot dapat diklasifikasikan ke dalam 2 jenis bagian yang besar, yaitu :

- a. *Industrial Robot*, digunakan pada sektor industri manufaktur.  
Seperti membantu pekerjaan manusia di industri dalam proses perakitan kendaraan, mengelas, mengecat, dan sebagainya.
- b. *Service Robot*, merupakan robot yang berfungsi untuk membantu pekerjaan manusia dalam membersihkan rumah, membantu dokter melakukan operasi, menjadi pemandu wisata, dan lain-lain.

Berdasarkan sifat mobilitasnya, robot diklasifikasikan ke dalam 2 jenis bagian yang besar, yaitu :

- a. *Fixed Robot*, robot yang memiliki ruang kerja (*spatial space*) yang terbatas, dimana bagian dasarnya (*Base*) diletakkan pada sebuah benda yang tetap seperti panel atau lantai.
- b. *Mobile Robot*, robot yang memiliki ruang kerja yang cukup luas, dimana bagian dasarnya diletakkan pada sebuah alat yang bergerak, seperti : roda/ban atau kaki.

### 2.1.3 Hukum Asimov

Bermula dari saintis/penulis Amerika yang berkebangsaan Rusia yang bernama Isaac Asimov (lahir pada 2 Jan. 1920, meninggal 6 Apr. 1992). Asimov begitu meminati tulisan-tulisan ilmiah dan sains. Contohnya, *I Robot* (1950), *The Foundation Trilogy* (1951-52), dan *Foundation's Edge* (1982) yang telah memenangkan Anugerah Hugo dan Nebula. Hukum Asimov mempunyai 3 "Hukum Robotik" yang juga dikenal sebagai "Hukum Zeroth":

▪ Hukum I

Sebuah robot tidak boleh melukai atau bertindak kepada manusia.

▪ Hukum II

Sebuah robot harus patuh terhadap perintah yang di instruksikan manusia kecuali perintah tersebut bertentangan dengan Hukum I.

▪ Hukum III

Sebuah robot harus menjaga eksistensi dirinya dari kepunahannya kecuali perintah tersebut bertentangan dengan Hukum I, Hukum II.

### 2.1.4 Matematika

Pada bagian ini akan dijelaskan mengenai vektor, matriks, operasi matriks dan transpose matriks yang berhubungan dengan industri pengecatan mobil.

#### 2.1.4.1 Vektor

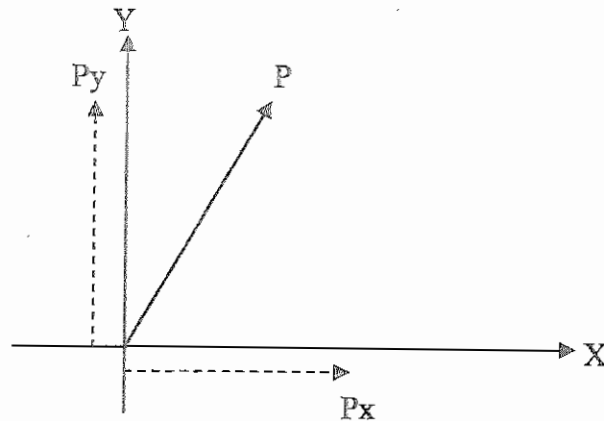
Berdasarkan (Gere & Greaver, 1983, p3) , berikut ini adalah bagian-bagian yang penting tentang vektor :

Vektor sering digunakan dalam banyak bidang matematika murni maupun terapan, fisika dan ilmu teknik. Vektor didefinisikan sebagai besaran yang memiliki panjang dan arah. Vektor sering dinyatakan secara geometris oleh sebuah garis dengan tanda panah pada ujungnya. Panjang garis menunjukkan besarnya vektor, dan tanda panah yang menunjukkan arah vektor.

Jika vektor dinyatakan secara geometris dengan sepotong garis berarah, maka perlu diperkenalkan suatu sistem koordinat sebagai penunjuk arah skala untuk panjangnya. Vektor yang terletak pada bidang memerlukan dua bilangan untuk

menyatakan sembarang vektor : satu untuk besarnya dan satu lagi untuk arahnya (sudut yang dibentuk oleh satu sumbu koordinat). Sedangkan vektor yang terletak dalam ruang berdimensi tiga memerlukan tiga bilangan untuk menggambarkan sembarang vektor : satu bilangan menyatakan besarnya vektor dan dua bilangan yang lainnya menyatakan kedudukannya terhadap koordinat.

Sebagai ilustrasi, perhatikan vektor Pada gambar 2.1 **P** (vektor dicetak dengan huruf tebal untuk membedakannya dengan besaran skalar) adalah sebuah vektor pada bidang yang dapat dipecah terhadap sumbu x dan y, masing-masing sebesar  $P_x$  dan  $P_y$ ,  $P_x$  dan  $P_y$  sendiri, masing-masing adalah vektor yang juga komponen vektor **P**. Di sini jelas bahwa sebuah bilangan tunggal tidak cukup untuk menyatakan vektor ini, sebab harus ada besar dan arahnya.



Gambar 2.1 Vektor **P** pada bidang

Berdasarkan (Hadley, 1961, p16-p17), Perhatikan vektor pada gambar 2.2 terdapat titik ( $P_x$ ,  $P_y$ ,  $P_z$ ) yaitu titik dimana kepala dari vektor berakhir, vektor **P** sudah dinyatakan secara lengkap. Besar/panjangnya adalah

$$|P| = \sqrt{P_x^2 + P_y^2 + P_z^2} \quad (2-1)$$

dan arahnya, ditentukan oleh kedua sudut  $\theta_1$  dan  $\theta_2$  dimana

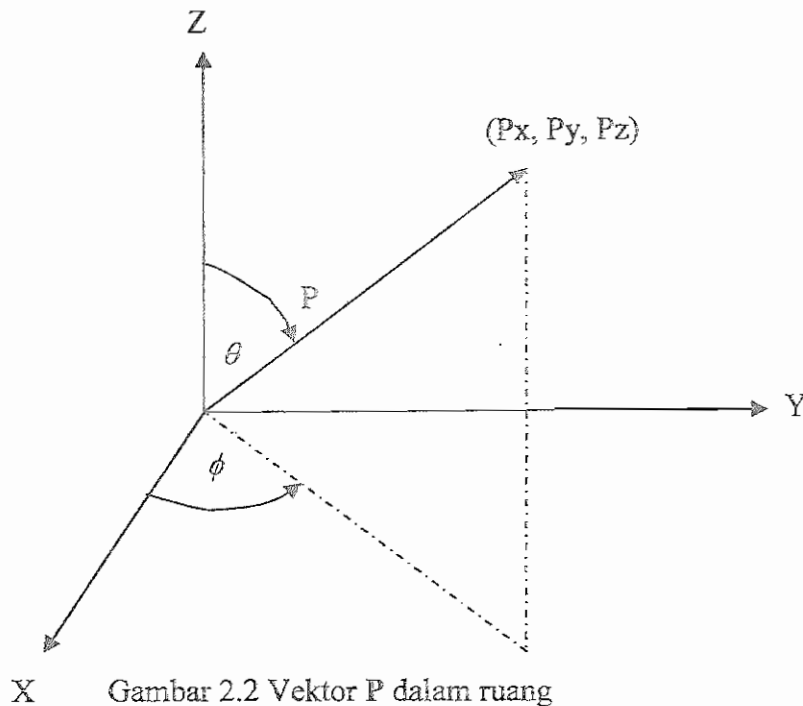
$$\tan \theta = \frac{P_y}{P_x} \quad (2-2)$$

$$\cos \phi = \frac{P_z}{\sqrt{P_x^2 + P_y^2 + P_z^2}} \quad (2-3)$$

selanjutnya hubungan antara vektor dengan titik dapat ditulis sebagai berikut :

$$P = (P_x, P_y, P_z) \quad (2-4)$$

Persamaan (2-4) menandakan bahwa  $P$  adalah sebuah vektor yang digambar dari titik asal sampai titik  $(P_x, P_y, P_z)$  dengan koordinat  $P_x$ ,  $P_y$ , dan  $P_z$ . Hubungan antara vektor dan titik dalam ruang merupakan dasar penting karena nilai praktisnya dalam perhitungan-perhitungan vektor. Vektor juga merupakan kunci untuk perluasan yang lebih abstrak dari konsep vektor.



Gambar 2.2 Vektor  $P$  dalam ruang

#### 2.1.4.2 Matriks

Berdasarkan (James M. Gere & William Weaver, Jr, 1981, p13-14), berikut adalah bagian-bagian penting tentang matriks :

Bentuk yang paling umum dari suatu matriks adalah susunan bilangan-bilangan yang berbentuk persegi panjang yang dapat digambarkan sebagai berikut:

$$A = \begin{bmatrix} A_{11} & A_{12} & \cdots & A_{1n} \\ A_{21} & A_{22} & \cdots & A_{2n} \\ \vdots & \cdots & \cdots & \vdots \\ A_{m1} & A_{m2} & \cdots & A_{mn} \end{bmatrix} \quad (2-5)$$

Susunan matriks di atas disebut sebuah matriks  $m$  kali  $n$  (ditulis  $m \times n$ ) karena memiliki  $m$  baris dan  $n$  kolom. Sebagai aturan, kurung siku [ ], kurung biasa ( ), atau bentuk || ||

digunakan sebagai batasan dari bilangan-bilangan yang terdapat dalam suatu matriks.

*Ordo* (atau *ukuran*) dalam sebuah persamaan matriks ditentukan oleh banyaknya baris dan kolom; maka, matriks  $A$  pada persamaan (2-4) mempunyai ordo  $m$  dan  $n$ , biasanya ditulis  $m \times n$ . *Matriks bujur sangkar* adalah matriks yang jumlah baris dan kolomnya sama ( $m = n$ ) dan dikatakan berada  $n$ . Elemen-elemen dari matriks bujur sangkar mulai dari ujung kiri atas sampai ujung kanan bawah secara diagonal (yaitu elemen  $A_{11}, A_{12}, \dots, A_{nn}$ ) disebut juga *diagonal utama* matriks. Dan elemen-elemen dari kiri ke bawah sampai kanan atas ( $A_{n1}, \dots, \dots$ , dan  $A_{1n}$ ) dinamakan *diagonal kedua*.

Karena matriks pada umumnya merupakan susunan bilangan-bilangan yang berdimensi dua, maka diperlukan dua subskrip untuk menyatakan setiap elemennya. Dengan perjanjian, subskrip pertama berkaitan dengan baris, dan yang kedua dengan kolom.

#### 2.1.4.3 Operasi Matriks

Berdasarkan (Hadley, 1961, p52-59), (Gere & Greaver, 1983, p14-21), berikut adalah bagian-bagian yang penting tentang operasi matriks:

Perkalian matriks skalar adalah bila sebuah matriks  $A$  dan sebuah skalar  $\lambda$ , hasil perkalian  $\lambda$  dan  $A$ , ditulis  $\lambda A$ , didefinisikan sebagai

$$\lambda A = \begin{bmatrix} \lambda A_{11} & \lambda A_{12} & \cdots & \lambda A_{1n} \\ \lambda A_{21} & \lambda A_{22} & \cdots & \lambda A_{2n} \\ \vdots & \cdots & \cdots & \vdots \\ \lambda A_{m1} & \lambda A_{m2} & \cdots & \lambda A_{mn} \end{bmatrix} \quad (2-6)$$

Aturan-aturan operasi matematika (seperti penjumlahan dan perkalian) untuk matriks hampir intuitif dan telah dirumuskan sedemikian rupa agar berguna untuk perhitungan-perhitungan praktis. Hubungan yang paling sederhana adalah kesamaan dua matriks. Agar dua buah matriks dapat dikatakan sama, maka harus berordo sama dan elemen-elemen yang bersesuaian harus sama. Jika  $A = B$ , dimana  $A$  dan  $B$  haruslah berlaku hubungan  $A_{ij} = B_{ij}$ .

### Penjumlahan Matriks

Penjumlahan dua buah matriks A dan B dapat berlangsung jika kedua matriks berordo sama. Jumlah dua buah matriks adalah matriks lain yang berordo sama yang elemen-elemennya merupakan jumlah dari elemen-elemen yang bersesuaian dari kedua matriks asal. Dengan kata lain, jika  $A + B = C$  maka tiap elemen C akan mempunyai bentuk :

$$A_{ij} + B_{ij} = C_{ij} \quad (2-7)$$

### Perkalian Matriks

Perkalian antar matriks adalah bila diberikan matriks  $m \times n$  A dan  $n \times r$  B, maka hasil kali AB didefinisikan sebagai matriks  $m \times r$  C, yang elemen-elemennya dihitung dari elemen-elemen dari A, B menurut

$$C_{ij} = \sum_{k=1}^n a_{ik} b_{kj}, \quad i = 1, \dots, m, j = 1, \dots, r. \quad (2-8)$$

Ada banyak cara untuk mengalikan dua buah matriks antara lain, misalnya kita dapat mengalikan sebuah matriks A yang berordo  $m \times n$  dengan matriks B yang berordo  $n \times p$ . Hasil kalinya berupa matriks C yang berordo  $m \times p$ , yang dinyatakan sebagai berikut :

$$A \cdot B = C \quad (2-9)$$

atau

$$A = \begin{bmatrix} A_{11} & A_{12} & \dots & A_{1n} \\ A_{21} & A_{22} & \dots & A_{2n} \\ \vdots & \dots & \dots & \vdots \\ A_{m1} & A_{m2} & \dots & A_{mn} \end{bmatrix} \begin{bmatrix} B_{11} & B_{12} & \dots & B_{1n} \\ B_{21} & B_{22} & \dots & B_{2n} \\ \vdots & \dots & \dots & \vdots \\ B_{m1} & B_{m2} & \dots & B_{mn} \end{bmatrix} = \begin{bmatrix} C_{11} & C_{12} & \dots & C_{1n} \\ C_{21} & C_{22} & \dots & C_{2n} \\ \vdots & \dots & \dots & \vdots \\ C_{m1} & C_{m2} & \dots & C_{mn} \end{bmatrix} \quad (2-10)$$

Perkalian dilakukan dengan mengambil perkalian-dalam antar baris-baris dari matriks A dan kolom-kolom dari matriks B. Elemen  $C_{ij}$  yang terletak pada baris dan kolom pertama matriks C berasal dari perkalian-dalam antara baris pertama matriks A dengan kolom pertama matriks B, sehingga :

$$C_{11} = A_{11} B_{11} + A_{12} B_{21} + \dots + A_{1n} B_{n1} \quad (2-11)$$

Elemen  $C_{ip}$  berasal dari perkalian-dalam antara baris pertama matriks A dengan kolom ke-p matriks B, sehingga :

$$C_{1p} = A_{11} B_{1p} + A_{12} B_{2p} + \dots + A_{1n} B_{np} \quad (2-12)$$

Elemen-elemen baris ke-dua matriks C dihitung dari perkalian-dalam antara baris ke-dua matriks A dengan setiap kolom matriks B secara bergantian, yakni :

$$\begin{aligned} C_{21} &= A_{21} B_{11} + A_{22} B_{21} + \dots + A_{2n} B_{n1} \\ C_{22} &= A_{21} B_{12} + A_{22} B_{22} + \dots + A_{2n} B_{n2} \\ &\dots \quad \dots \quad \dots \quad \dots \quad \dots \quad \dots \quad \dots \quad \dots \quad \dots \\ C_{2p} &= A_{21} B_{1p} + A_{22} B_{2p} + \dots + A_{2n} B_{np} \end{aligned} \quad (2-13)$$

Cara untuk menghitung elemen-elemen matriks C ini (atau *perkalian matriks*) menunjukkan bahwa supaya perkalian ini dapat berlangsung, maka banyaknya kolom matriks A harus selalu sama dengan banyaknya baris matriks B. Matriks C akan selalu mempunyai banyak baris sama dengan matriks A dan banyak kolom yang sama dengan matriks B. Hubungan kesesuaian untuk perkalian matriks ini dapat dinyatakan dengan bagan sebagai berikut :

$$[m \times n] [n \times p] = [m \times p] \quad (2-14)$$

Perkalian matriks kebanyakan tidak komutatif, yakni :

$$AB \neq BA \quad (2-15)$$

#### 2.1.4.4 Transpose Matriks

Berdasarkan (Hadley, 1961, p65), transpose dari matriks  $A = \|a_{ij}\|$  adalah matriks yang dibentuk dari A dengan mempertukarkan baris-baris dan kolom-kolom sehingga baris  $i$  dari matriks A menjadi kolom  $i$  dari matriks transpose. Transpose dinotasikan dengan  $A^T$ .

$$A^T = \|a_{ji}\| \text{ jika } A = \|a_{ij}\| \quad (2-16)$$

Penjelasan mengenai vektor, matriks dan trigonometri dapat dipelajari dari buku-buku matematika yang umum.

Berdasarkan (Gere & Veaver, 1983, p21), transpose matriks adalah sebuah matriks baru yang dibentuk dengan menukar baris-baris dan kolom-kolom dari matriks asal; yaitu baris pertama dari matriks asal menjadi kolom pertama dari transpose matriks, baris kedua menjadi kolom kedua dan seterusnya. Jika matriks A adalah matriks asal yang berordo  $m \times n$ , seperti yang tampak di bawah ini

$$A = \begin{bmatrix} A_{11} & A_{12} & \cdots & A_{1n} \\ A_{21} & A_{22} & \cdots & A_{2n} \\ \vdots & \cdots & \cdots & \vdots \\ A_{m1} & A_{m2} & \cdots & A_{mn} \end{bmatrix} \quad (2-17)$$

maka transpose matriksnya, ditulis  $A^T$ , akan berordo  $n \times m$  :

$$A^T = \begin{bmatrix} A_{11} & A_{21} & \cdots & A_{n1} \\ A_{12} & A_{22} & \cdots & A_{n2} \\ \vdots & \cdots & \cdots & \vdots \\ A_{n1} & A_{n2} & \cdots & A_{nm} \end{bmatrix} \quad (2-18)$$

## 2.2 Teori-teori khusus yang Berhubungan dengan Topik Yang Dibahas

Teori khusus dalam kinematika robot pengecatan mobil berhubungan dengan geometri dari gerakan. Berikut ini beberapa dasar-dasar teori robot industri dan kinematika.

### 2.2.1 Definisi Robot Industri

Organisasi standart dunia melalui ISO 8373 mendefinisikan robot industri sebagai “sebuah manipulator dengan tiga atau lebih sumbu, yang dikontrol secara otomatis, yang dapat diprogram ulang, dengan banyak tujuan, dimana diletakkan pada tempat yang tetap atau dapat bergerak untuk keperluan dan aplikasi-aplikasi otomasi industri”. Di samping itu, *British Robot Association* memberikan definisi mengenai robot industri, yaitu “sebuah peralatan yang dapat diprogram ulang yang dirancang untuk sekaligus memanipulasi dan memindahkan benda, peralatan atau alat khusus untuk proses manufaktur, melalui gerakan dengan pemrogramman variabel untuk melakukan tugas manufaktur khusus”.

([http : //vacdefs.net/~industrial robotis/storys.htm](http://vacdefs.net/~industrial_robotis/storys.htm)).

Demikian halnya dengan *Japanese Industrial Robot Association* yang memberikan definisi robot menjadi enam tingkatan (Fu et al, 1987, p1-6), yaitu :

- a. Peralatan manual (*manual handling devices*).
- b. Peralatan mengangkat dan meletakkan obyek (*pick and place device*).



- c. Manipulator yang variabelnya dapat diprogram secara berurutan (*programmable variable sequence manipulator*).
- d. Robot yang diajar secara manual (*robot taught manually*).
- e. Robot yang dikendalikan dengan sebuah bahasa pemrograman (*robot controlled by a programming language*).
- f. Robot yang dapat bereaksi terhadap lingkungannya (*robots which can react to their environment*).

### 2.2.2 Karakteristik Robot Industri

Berdasarkan James L. Fuller (1991, p37-38), berikut adalah karakteristik-karakteristik dari robot industri :

- a. *Hand*, atau tangan dalam robotika dikenal sebagai *gripper*, *end-effector*, atau *end-of arm*. Di dalam tangan tersebut terdapat peralatan-peralatan mekanik yang bisa digerakkan dan dilekatkan pada ujung manipulator. Tangan robot dapat memegang dan memindahkan objek serta bisa juga melakukan proses manufaktur seperti mengelas, mengecat, dan sebagainya.
- b. *Wrist*, merupakan penghubung antara tangan (*hand*) dan lengan (*arm*) pada robot, *Wrist* dapat bergerak dengan tiga pergerakan, antara lain :
  - *Pitch*, pergerakan *joint* naik dan turun.
  - *Yaw*, pergerakan *joint* menyamping kiri dan kanan.
  - *Roll*, pergerakan *joint* memutar.
- c. *Arm*, atau lengan, merupakan sarana untuk berinteraksi langsung dengan objek. Bentuk dari lengan ini akan dapat berpengaruh terhadap pergerakan, lintasan dan luas jangkauan dari robot. Secara mekanis lengan robot terbagi menjadi dua bagian utama, yaitu :
  - *Link*, merupakan bagian lengan.
  - *Joint*, merupakan titik pertemuan antara dua *link*. *Joint* itu sendiri terdiri dari beberapa macam, yaitu *prismatic joint*, *planar joint*, *spherical joint*, *revolute joint*, *screw joint*, dan *cylindrical joint*.
- d. *Shoulder*, berfungsi untuk mendukung atau membantu pergerakan lengan.

- e. *Lifting Power*, merupakan gaya angkat dari lengan robot yang bergantung pada berat, bentuk dan objek yang akan dipindahkan.
- f. *Repeatability*, untuk dapat menggantikan peran manusia dengan sempurna, sebuah robot harus dapat melakukan tugas atau pekerjaan secara berulang-ulang dengan presisi yang tinggi atau baik.
- g. *Automatic Control System*, digunakan untuk menjalankan instruksi-instruksi yang tersimpan di dalam memori robot. Tanpa *Automatic Control System*, robot seperti sebuah peralatan kendali jarak jauh.  
([http : //vacdefs.net/~industrial robotis.htm](http://vacdefs.net/~industrial/robotis.htm)).

### 2.2.3 Komponen Robot Industri

Berdasarkan (James L. Fuller, 1991, p37-38), robot industri mempunyai dua komponen yang utama, yaitu :

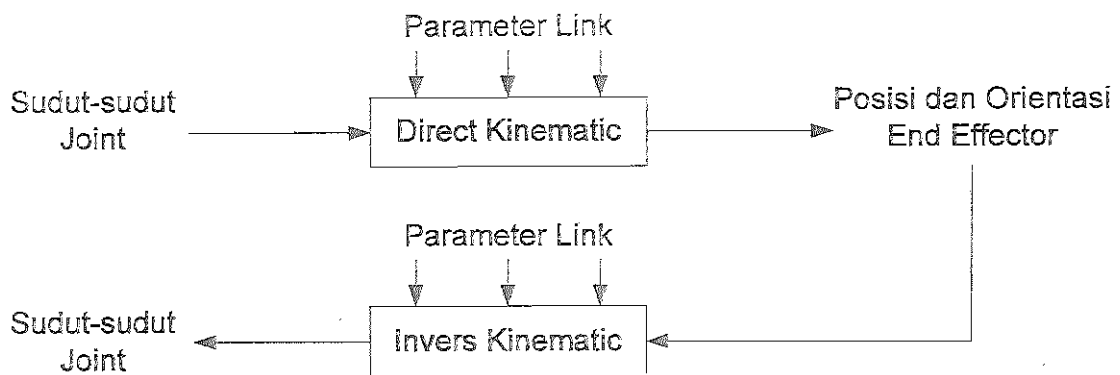
- a. *Physical Part/Anatomy*, terbentuk dari empat bagian, yaitu :
  - *Mechanical Part/Manipulator*, adalah suatu struktur peralatan mekanik yang mempunyai fungsi tertentu, misalnya untuk menjepit, berputar ataupun bergeser dan lain sebagainya. *Manipulator* biasanya dibuat dalam bentuk lengan.
  - *Controller*, yaitu bagian dari sistem yang bertugas mengatur semua kegiatan yang terjadi pada robot. *Controller* pada robot dapat menggunakan mikrokontroler atau mikroprosesor dan berisi kontrol program dan *task* program.
  - *Power Supplay*, berfungsi memberikan energi kepada manipulator untuk dapat bergerak. Energi yang diberikan bisa berupa elektrik, hidrolik, atau pneumatik.
  - *Vechile*, komponen ini dapat digunakan ataupun tidak (*optional*). Berfungsi untuk membawa bagian dasar (*base*) dari robot ke arah tempat robot bekerja. Bagian dasar robot (*base*) bisa diletakkan pada sebuah benda tetap seperti panel atau lantai, juga bisa dilekatkan pada sebuah alat gerak seperti roda/ban atau kaki.

- b. *Learned Behavior/Task Program*, berfungsi sebagai data pada *control program* untuk melaksanakan pekerjaan yang diberikan.

#### 2.2.4 Kinematika Dasar

Berdasarkan (Fu, Gonzales dan Lee, 1987, p6-13), berikut adalah pengertian tentang permasalahan-permasalahan yang terdapat pada kinematika :

Terdapat dua permasalahan dalam mempelajari kinematika, permasalahan pertama adalah *direct kinematika*. *Direct Kinematics* berhubungan dengan mencari solusi untuk mendapatkan posisi dan orientasi dari *tool frame* dalam hal ini *end effector* yang relatif terhadap *base frame* dengan memberikan sudut-sudut dari *joint* ( $\theta_i$ ). Permasalahan yang kedua adalah *invers kinematics*, berhubungan dengan mencari solusi untuk menemukan besarnya sudut-sudut *joint* ( $\theta_i$ ) dengan memberikan posisi dan orientasi dari *end effector*.



Gambar 2.3 *Direct Kinematics* dan *Invers Kinematics*

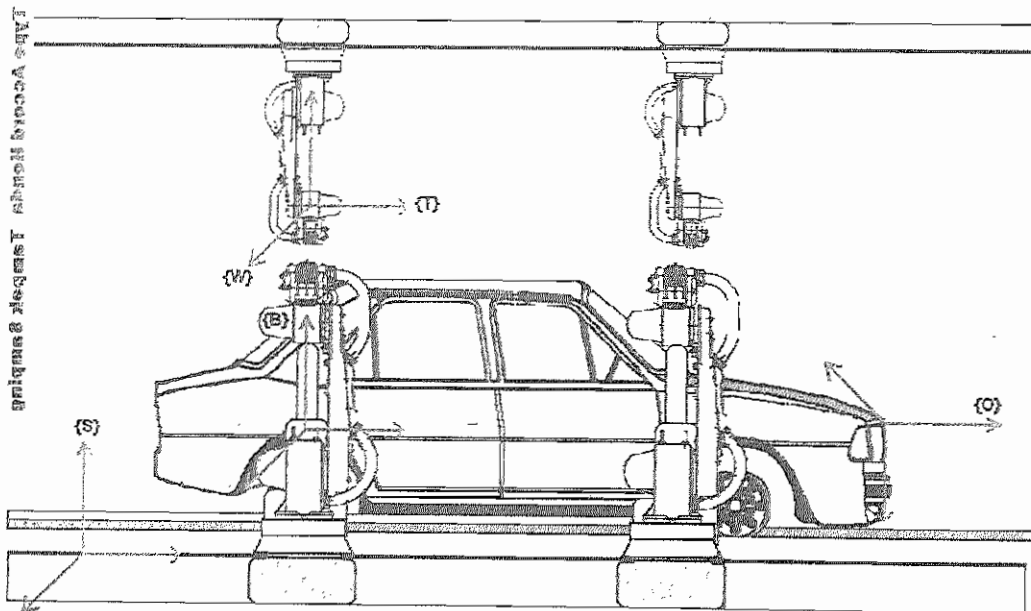
#### Definisi Posisi, Orientasi, dan frame

Berdasarkan (Craig, 1989, p99-101), ada beberapa istilah *frame* yang sering dipakai dalam robotika dalam industri, yaitu *Station Frame*, *Base Frame*, *Wrist Frame*, *Tool Frame*, dan *Goal Frame*. *Station Frame* (S) berada pada tempat dilakukannya aktivitas/pekerjaan, seringkali disebut sebagai *Universal Frame*, dimana semua gerakan yang dilakukan oleh robot dibuat mengacu pada sistem koordinat ini. Berdasarkan gambar 2.4, *Station Frame* terletak pada sudut tumpuan (tanah). *Frame* (S) ini ditetapkan dengan memperhatikan *Base Frame* (B), yaitu  ${}^B T_S$ . *Base Frame* (B) berada

bagian dasar (bagian yang tidak bergerak) dari robot, seringkali disebut juga sebagai *frame* (0).

*Wrist Frame* (B) terletak pada *link* terakhir manipulator (di pergelangan lengan robot) dan selalu mengikuti perpindahan yang terjadi pada *link* tersebut, biasanya *wrist frame* didefinisikan relatif terhadap *base frame*, yaitu  $(W) = {}^W T_w$ . *Tool Frame* (T) terletak pada ujung dari peralatan yang dipergunakan oleh robot, dikenal sebagai *end effector*. *Frame* (T) ini selalu mengacu pada *frame* (W). *End effector* digunakan untuk menggambarkan tangan atau alat yang melekat pada *wrist*. *End effector* dapat dibagi menjadi 2 kategori : *Gripper* dan *tools*. *Gripper* digunakan untuk memegang suatu objek dan menahannya selama siklus kerja robot berlangsung. Ada beberapa variasi dalam metode menahan sebagai tambahan dari memegang bagian antara dua atau lebih jari, termasuk di dalamnya penggunaan mangkok penyedot, magnet, kait, dan sekop. Sedangkan *tools* digunakan sebagai *end effector* dalam aplikasi industri pengecatan mobil di mana robot diperlukan untuk menyemprotkan cat pada mobil.

*Goal Frame* (G) merupakan lokasi tujuan tempat dimana peralatan yang dipegang oleh robot ingin dipindahkan. *Frame* (T) harus dibawa sampai berhimpitan dengan *frame* (G). *Goal Frame* (G) selalu ditetapkan dengan mengacu pada *station frame* (S).



Gambar 2.4 *Frame-frame* yang dimanipulasi dengan sebuah obyek mobil sedan

## Macam-macam Gerakan

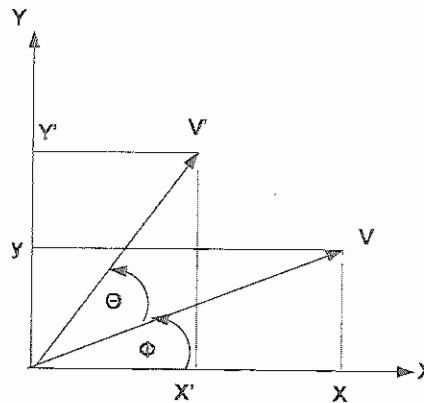
Berdasarkan (Carig, 1989, p32-37), gerakan bisa dibagi menjadi gerak translasi, gerak rotasi, dan transformasi. Gerak translasi adalah gerak perpindahan dari suatu titik ke titik yang lain secara lurus di dalam ruangan. Gerak rotasi adalah gerak perpindahan suatu titik ke titik lain secara melingkar di dalam ruangan. Transformasi adalah gerakan gabungan dari gerak translasi dan gerak rotasi.

### 2.2.5 Direct Kinematics

Pada bagian ini akan dijelaskan mengenai matriks rotasi, transformasi *homogeneous*, dan konsep Denavit-Hartenberg.

Hal ini dimaksudkan dalam memahami konsep Denavit-Hartenberg memerlukan pembahasan transformasi *homogenous* dimana untuk memahaminya memerlukan pembahasan tentang matriks rotasi dan translasi baik dalam dua ataupun tiga dimensi.

#### 2.2.5.1 Rotasi Dalam 2 Dimensi



Gambar 2.5 Vektor V berotasi menjadi V'

Gambar 2.5 menggambarkan vektor  $V = \begin{pmatrix} x \\ y \end{pmatrix}$  adalah posisi dari sebuah titik. Bila ditulis dengan koordinat polar menjadi

$$V = \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} r \cos \phi \\ r \sin \phi \end{pmatrix} \quad (2-19)$$

bila terjadi rotasi sebesar  $\theta$ , V menjadi V' dimana

$$V' = \begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} r \cos(\theta + \phi) \\ r \sin(\theta + \phi) \end{pmatrix} = \begin{pmatrix} r(\cos \theta \cos \phi - \sin \theta \sin \phi) \\ r(\sin \theta \sin \phi + \cos \theta \cos \phi) \end{pmatrix}$$

$$V' = \begin{pmatrix} x \cos \theta - y \sin \theta \\ x \sin \theta + y \cos \theta \end{pmatrix} = \begin{pmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} \quad (2-20)$$

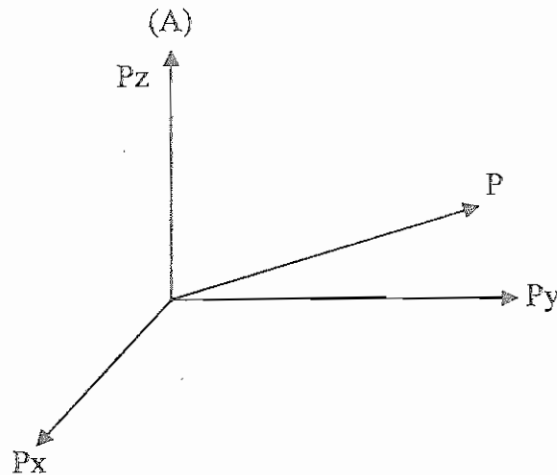
sehingga

$$V' = R \cdot V \quad (2-21)$$

dengan

$$R = \text{matriks rotasi} = \begin{pmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{pmatrix} \quad (2-22)$$

Ketika sebuah *frame* (sistem koordinat) dibentuk maka sebuah titik dalam sistem koordinat tersebut bisa direpresentasikan melalui vektor posisi 3x1. Vektor posisi tersebut harus mempunyai informasi tentang sistem koordinat yang telah terbentuk.



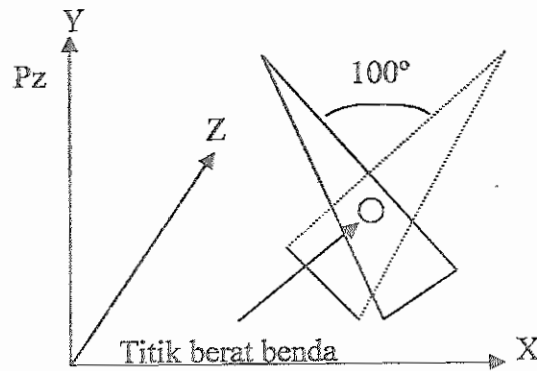
Gambar 2.6 Vektor P relatif terhadap sistem koordinat A

Gambar 2.6 merepresentasikan sebuah sistem koordinat (A),  ${}^A P$  merupakan vektor posisi dalam ruang. Komponen-komponen dari vektor tersebut adalah Px, Py, Pz:

$${}^A P = \begin{bmatrix} Px \\ Py \\ Pz \end{bmatrix} \quad (2-23)$$

Selain posisi, dalam *direct kinematics* juga harus ditentukan orientasi untuk memberikan informasi tentang pengarahannya sebuah titik dalam sistem koordinat. Orientasi dapat dipresentasikan menggunakan matriks 3x3. Matriks ini digunakan untuk

mengambarkan operasi rotasional suatu koordinat yang relatif terhadap sistem koordinat yang lain. Pengertian rotasi sendiri adalah pergerakan yang mengakibatkan adanya perubahan pada orientasi. Jika diibaratkan suatu benda berotasi terhadap sumbu Z sebesar  $100^\circ$ , maka posisi dari titik berat benda tetap, tetapi orientasinya berubah  $100^\circ$  terhadap sumbu Z. Perhatikan gambar 2.7.



Gambar 2.7 Benda yang berotasi

### 2.2.5.2 Rotasi Dalam 3 Dimensi

Gambar 2.8 menggambarkan sistem koordinat  $OUVW$  berotasi terhadap sistem koordinat referensi  $OXYZ$ . Titik  $P$  dapat direpresentasikan melalui tiap-tiap koordinatnya dengan mengacu kepada sistem koordinat  $OUVW$  dan  $OXYZ$ , sehingga didapat sebuah persamaan :

$$P_{UVW} = (P_U, P_V, P_W)^T \text{ dan } P_{XYZ} = (P_X, P_Y, P_Z)^T \quad (2-24)$$

dimana  $P_{XYZ}$  dan  $P_{UVW}$  merepresentasikan titik  $P$  yang sama dalam ruang melalui dua sistem koordinat yang berbeda.

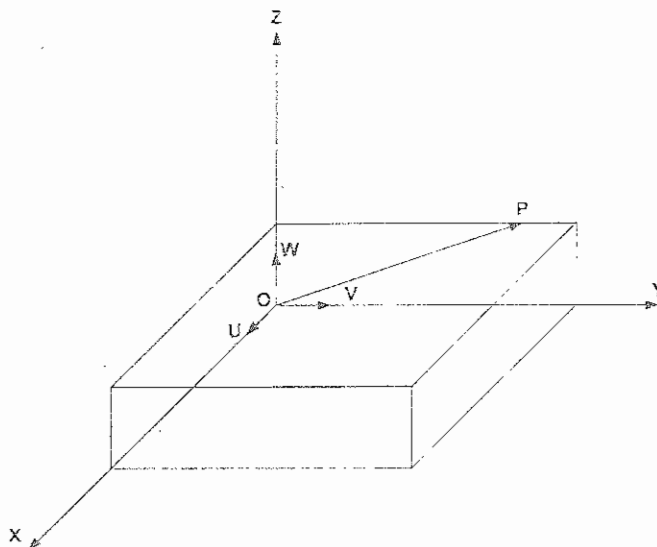
Untuk mendapatkan sebuah transformasi matriks  $3 \times 3$   $R$  yang terbentuk oleh koordinat  $P_{UVW}$  yang mengacu pada sistem koordinat  $OXYZ$  setelah sistem koordinat  $OUVW$  berotasi. Maka dapat dibentuk persamaan sebagai berikut :

$$P_{XYZ} = R P_{UVW} \quad (2-25)$$

dimana titik  $P_{UVW}$  berotasi bersamaan dengan sistem koordinat  $OUVW$ .

dengan menguraikan vektor  $P_{UVW}$ , maka didapat persamaan,

$$P_{UVW} = P_U i_U + P_V j_V + P_W k_W \quad (2-26)$$



Gambar 2.8 Koordinat OUVW relatif terhadap sistem koordinat OXYZ

selanjutnya dengan menggunakan perkalian skalar, persamaan (2-25) menjadi,

$$\begin{aligned} P_X &= i_X \cdot P = i_X \cdot i_U P_U + i_X \cdot j_V P_V + i_X \cdot k_W P_W \\ P_Y &= i_Y \cdot P = i_Y \cdot i_U P_U + i_Y \cdot j_V P_V + i_Y \cdot k_W P_W \\ P_Z &= i_Z \cdot P = i_Z \cdot i_U P_U + i_Z \cdot j_V P_V + i_Z \cdot k_W P_W \end{aligned} \quad (2-27)$$

dimana  $P_X$ ,  $P_Y$ , dan  $P_Z$  merupakan komponen-komponen vektor  $P$  sepanjang sumbu  $OX$ ,  $OY$ ,  $OZ$ . Dalam bentuk matriks dapat dituliskan sebagai berikut,

$$\begin{bmatrix} P_X \\ P_Y \\ P_Z \end{bmatrix} = \begin{bmatrix} i_X \cdot i_U & i_X \cdot j_V & i_X \cdot k_W \\ j_Y \cdot i_U & j_Y \cdot j_V & j_Y \cdot k_W \\ k_Z \cdot i_U & k_Z \cdot j_V & k_Z \cdot k_W \end{bmatrix} \begin{bmatrix} P_U \\ P_V \\ P_W \end{bmatrix} \quad (2-28)$$

dari perkalian diatas, matriks  $R$  dapat ditulis sebagai berikut,

$$R = \begin{bmatrix} i_X \cdot i_U & i_X \cdot j_V & i_X \cdot k_W \\ j_Y \cdot i_U & j_Y \cdot j_V & j_Y \cdot k_W \\ k_Z \cdot i_U & k_Z \cdot j_V & k_Z \cdot k_W \end{bmatrix} \quad (2-29)$$

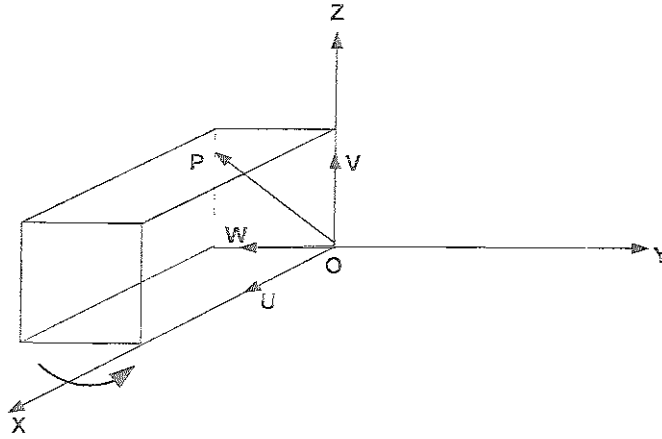
Selanjutnya dengan menggunakan persamaan (2-26), dapat dibentuk matriks rotasi  $3 \times 3$  sesuai dengan perputarannya terhadap suatu sumbu yaitu sumbu koordinat referensi  $OX$ ,  $OY$ ,  $OZ$ . Transformasi matriks  $R_x$ ,  $\alpha$  dan dapat diperoleh dari konsep matriks transformasi diatas,

$$P_{XYZ} = R_x \cdot \alpha \cdot P_{UVW} \quad (2-30)$$

dimana,  $i_X = i_U$  dan dalam bentuk matriks ditulis,



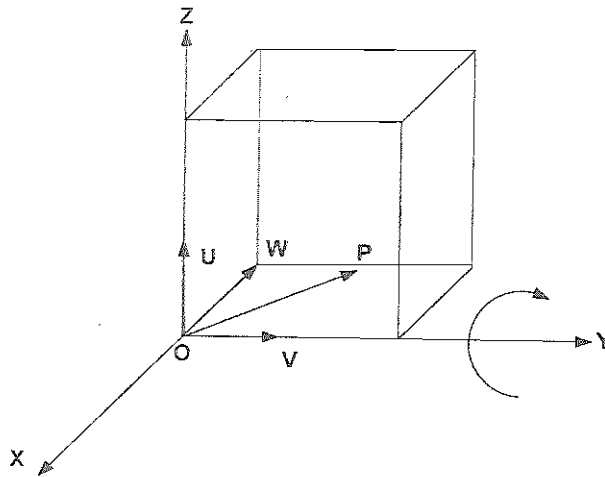
$$R_{x,\alpha} = \begin{bmatrix} i_X \cdot i_U & i_X \cdot i_V & i_X \cdot i_W \\ j_Y \cdot i_U & j_Y \cdot i_V & j_Y \cdot i_W \\ k_Z \cdot i_U & k_Z \cdot i_V & k_Z \cdot i_W \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \alpha & -\sin \alpha \\ 0 & \sin \alpha & \cos \alpha \end{bmatrix} \quad (2-31)$$



Gambar 2.9 Koordinat OUVW berotasi terhadap sumbu OX

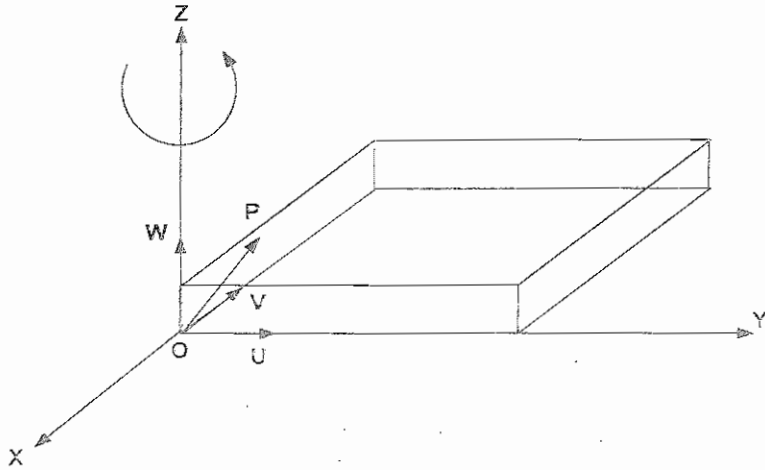
lalu dengan cara yang sama pula matriks rotasi 3x3 untuk rotasi sumbu OY dengan sudut  $\phi$  dan sumbu OZ dengan sudut  $\theta$  dapat dicari, hasilnya adalah,

$$R_{y,\phi} = \begin{bmatrix} \cos \phi & 0 & \sin \phi \\ 0 & 1 & 0 \\ -\sin \phi & 0 & \cos \phi \end{bmatrix} \quad (2-32)$$



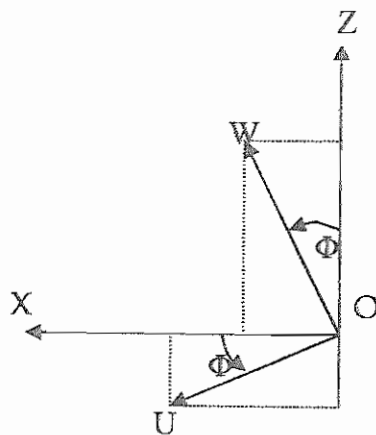
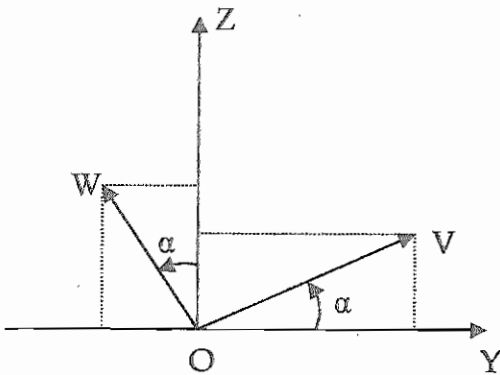
Gambar 2.10 Koordinat OUVW terhadap sumbu OY

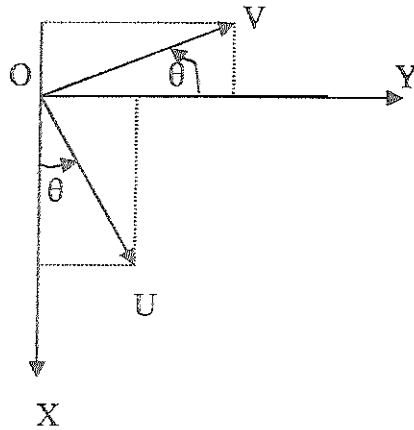
$$R_{z,\theta} = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (2-33)$$



Gambar 2.11 Koordinat GUVW berotasi terhadap sumbu OZ

Arah rotasi berlawanan jarum jam tanda  $\alpha$ ,  $\theta$ , dan  $\Phi$  adalah positif sedangkan arah rotasi searah dengan arah gerak jarum jam tanda  $\alpha$ ,  $\theta$ , dan  $\Phi$  adalah negatif. Matriks-matriks  $R_{x,\alpha}$  persamaan (2-30),  $R_{y,\phi}$  persamaan (2-31), dan  $R_{z,\theta}$  persamaan (2-32) diatas dapat disebut sebagai matriks rotasi dasar.

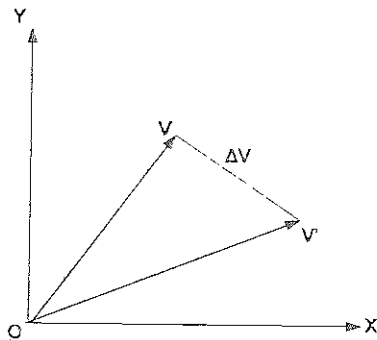




Gambar 2.12 Sistem-sistem koordinat yang berputar

### 2.2.5.3 Translasi Dalam 2 Dimensi

Gambar 2.13 menggambarkan posisi sebuah titik  $(x, y)$  dinyatakan dengan sebuah vektor  $V$ . Titik ini kemudian berpindah menjadi  $(x', y')$  dan dinyatakan dengan sebuah vektor  $V'$ .



Gambar 2.13 Vektor  $V$  bertranslasi menjadi  $V'$

Misalnya perubahan vektor  $\Delta V$  yang merupakan perubahan dalam arah  $x$  sebesar  $\Delta x$ , dan dalam arah  $y$  sebesar  $\Delta y$ , maka

$$V' = V + \Delta V \quad (2-34)$$

jika ditulis dalam komponen menjadi,

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} \Delta x \\ \Delta y \end{pmatrix}$$

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} x + \Delta x \\ y + \Delta y \end{pmatrix} \quad (2-35)$$

#### 2.2.5.4 Taranslasi Dalam 3 Dimensi

Seperti halnya dengan translasi 2 dimensi, untuk translasi 3 dimensi hanya menambahkan komponen  $z$ , sehingga persamaan (2-34) menjadi,

$$\begin{pmatrix} x' \\ y' \\ z' \end{pmatrix} = \begin{pmatrix} x + \Delta x \\ y + \Delta y \\ z + \Delta z \end{pmatrix} \quad (2-36)$$

jika ditulis dalam bentuk matriks persamaan (2-35) diatas menjadi,

$$\begin{pmatrix} x' \\ y' \\ z' \\ 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & \Delta x \\ 0 & 1 & 0 & \Delta y \\ 0 & 0 & 1 & \Delta z \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} \quad (2-37)$$

Matriks  $\begin{pmatrix} 1 & 0 & 0 & \Delta x \\ 0 & 1 & 0 & \Delta y \\ 0 & 0 & 1 & \Delta z \\ 0 & 0 & 0 & 1 \end{pmatrix}$  disebut matriks translasi. Nilai 1 pada pojok kanan bawah

hanya merupakan operator matematika pembantu saja agar penulisan dalam bentuk matriks bisa dilakukan. Selain itu jika bukan translasi saja tetapi juga melakukan rotasi maka matriks rotasi dan translasi bisa digabungkan menjadi satu dan biasanya disebut matriks transformasi *homogenous*.

#### 2.2.5.5 Matriks Transformasi Homogeneous

Berdasarkan (Fu, Gonzales dan Lee, 1987, p27-29), berikut adalah bagian-bagian yang penting tentang transformasi *homogeneous*.

Matriks rotasi 3x3 yang sebelumnya diperkenalkan tidak dapat digunakan untuk menunjukkan pergeseran dari suatu posisi (translasi) dan penskalaannya, untuk itu dibutuhkan sebuah matriks baru yang dapat merepresentasikan pergeseran sekaligus penskalaan.

Matriks transformasi *homogeneous* adalah sebuah matriks 4x4 dimana matriks ini dapat memetakan sebuah vektor posisi yang diekspresikan dalam koordinat *homogenous* dari suatu sistem koordinat ke sistem koordinat lainnya. Sebuah matriks transformasi *homogeneous* terdiri dari 4 submatriks.

$$T = \begin{bmatrix} R_{3 \times 3} & I & P_{3 \times 3} \\ - & I & - \\ f_{1 \times 3} & I & S_{1 \times 3} \end{bmatrix} = \begin{bmatrix} \text{matriks} & I & \text{vektor} \\ \text{rotasi} & & \text{posisi} \\ - & & - \\ \text{transformasi} & I & \text{skala} \\ \text{perspektif} & & \end{bmatrix} \quad (2-38)$$

dimana submatriks  $3 \times 3$  yang terletak di kiri atas merepresentasikan matriks rotasi, submatriks  $3 \times 1$  dibagian kanan atas merepresentasikan vektor posisi dari sistem koordinat asal yang dirotasi mengacu pada sistem koordinat referensi. Submatriks  $1 \times 3$  dibagian bawah kiri merepresentasikan transformasi perspektif, dan terakhir submatriks  $1 \times 1$  yang terletak dibagian kanan bawah adalah matriks yang merepresentasikan faktor penskalaan.

Selanjutnya sebuah matriks rotasi  $3 \times 3$  bisa diperluas menjadi matriks transformasi *homogeneous*  $4 \times 4$  yang dilambangkan dengan *Trot* ( $T$ ). Persamaan (2-27) sampai dengan persamaan (2-29) dapat diekspresikan sebagai matriks transformasi *homogeneous*,

$$T_{x,\alpha} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos \alpha & -\sin \alpha & 0 \\ 0 & \sin \alpha & \cos \alpha & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (2-39)$$

$$T_{y,\Phi} = \begin{pmatrix} \cos \Phi & 0 & \sin \Phi & 0 \\ 0 & 1 & 0 & 0 \\ -\sin \Phi & 0 & \cos \Phi & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (2-40)$$

$$T_{z,\theta} = \begin{pmatrix} \cos \theta & -\sin \theta & 0 & 0 \\ \sin \theta & \cos \theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (2-41)$$

Matriks-matriks rotasi  $4 \times 4$  diatas disebut juga matriks rotasi *homogeneous* dasar. Submatriks  $1 \times 3$  yang merepresentasikan transformasi perspektif berguna pada komputer vision dan kalibrasi dari sebuah kamera. Semua elemen dari submatriks ini diset sama dengan nol untuk mengindikasikan bahwa perspektif/sudut pandang tidak berubah karena tidak menggunakan kamera. Pada matriks rotasi *homogeneous* faktor penskalaan diset 1, karena diasumsikan tidak terjadi pembesaran pada objek jika tidak menggunakan kamera.

### 2.2.5.6 Konsep Denavit-Hartenberg

Manipulator mekanik terdiri atas sejumlah bagian tubuh yang dinamakan sebagai *link*. Kemudian sejumlah *joint* digunakan untuk menghubungkan *link-link* tersebut. Setiap *joint* mewakili satu derajat kebebasan.

Untuk mendepelintikan hubungan translasi dan rotasi antara *link-link* yang berdekatan, Denavit dan Hartenberg (1955) memperkenalkan rancangan sebuah metode matriks yang secara sistematis membangun sebuah sistem koordinat dari masing-masing *link*.

Sebuah sistem koordinat kartesian yang saling ortonormal ( $X_i, Y_i, Z_i$ ) dapat dibangun untuk setiap *link* berdasarkan poros *joint*-nya, dimana  $i = 1, 2, 3, \dots, n$  ( $n$  = jumlah derajat kebebasan) ditambah dengan koordinat *base*. Ketika setiap *joint* putar (*revolute joint*) memiliki satu derajat kebebasan, maka koordinat ( $X_i, Y_i, Z_i$ ) ditujukan untuk *joint*  $i+1$  dan berada pada *link*  $i$ . Ketika *actuator* mengaktifkan *joint*  $i$  maka *link*  $i$  akan mengalami pergerakan dengan mengacu pada *link*  $i$  maka koordinat tersebut akan bergerak bersama *link*  $i$ . Dengan demikian koordinat ke- $n$  bergerak atau berpindah bersama dengan *link*  $n$ . Koordinat *base* didefinisikan sebagai *frame* 0 ( $X_0, Y_0, Z_0$ ). Dengan demikian sebuah robot yang memiliki enam poros putar, maka akan memiliki tujuh buah koordinat (*frame*) yang diberikan nama ( $X_0, Y_0, Z_0$ ), ( $X_1, Y_1, Z_1$ ), ..., ( $X_6, Y_6, Z_6$ ). Setiap koordinat dibuat berdasarkan setiap tiga buah aturan dibawah ini :

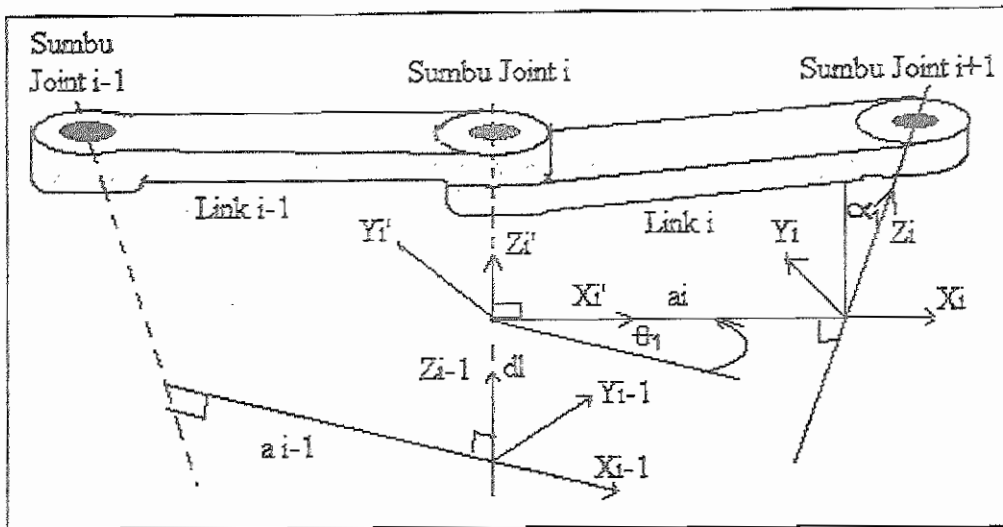
1. Sumbu  $Z_{i-1}$  berada di sepanjang poros pergerakan dari *joint* ke- $i$ .
2. Sumbu  $X_i$  normal terhadap sumbu  $Z_{i-1}$  dan merujuk menjauh darinya.
3. Sumbu  $Y_i$  melengkapi sistem koordinat yang diperlukan berdasarkan aturan tangan kanan.

Dengan aturan ini, *frame* 0 bisa diletakkan dimana saja selama sumbu  $Z_0$  berada disepanjang sumbu pergerakan *joint* pertama. Koordinat yang terakhir (*frame* ke- $n$ ) dapat diletakkan dimana saja pada lengan robot selama sumbu  $X_n$  normal terhadap sumbu  $Z_{n-1}$ .

Parameter-parameter yang digunakan untuk konsep Denavit-Hartenberg berdasarkan (Fu, Gonzales dan Lee, 1987, p36=37) adalah :

$\theta_i$  adalah sudut pada *joint* dari sumbu  $X_{i-1}$  ke sumbu  $X_i$ , dengan sumbu  $Z_{i-1}$  sebagai porosnya (perputarannya menggunakan aturan tangan kanan).

- $d_i$  adalah jarak dari pusat sumbu koordinat ke (i-1) ke perpotongan sumbu  $Z_{i-1}$  dengan sumbu  $X_i$  sepanjang sumbu  $Z_{i-1}$ .
- $a_i$  adalah jarak/panjang perpotongan sumbu  $Z_{i-1}$  dengan sumbu  $X_i$  ke pusat sumbu koordinat ke-i sepanjang sumbu  $X_i$  (atau jarak terpendek antara sumbu  $Z_{i-1}$  dan sumbu  $Z_i$ ).
- $\alpha_i$  adalah jarak sudut dari sumbu  $Z_{i-1}$  ke sumbu  $Z_i$  dengan poros sumbu  $X_i$  (perputarannya dengan menggunakan aturan tangan kanan).



Gambar 2.14 Parameter kinematik menurut Denavit-Hartenberg (dikutip dari Sciavicco dan Siciliano, 1996, p39)

Dalam menetapkan koordinat setiap *link* maka digunakan konversi dan Denavit-Hartenberg. Konversi ini menjelaskan empat parameter  $\theta_i$ ,  $d_i$ ,  $a_i$ , dan  $\alpha_i$  dimana *joint* (i) dapat dirotasikan, jika menggunakan *joint* putar (*revolute joint*) maka,  $d_i$ ,  $a_i$ ,  $\alpha_i$  konstan sedangkan  $\theta_i$  merupakan variabel dan jika menggunakan *joint* geser (*prismatic joint*) maka yang konstan adalah parameter,  $\theta_i$ ,  $a_i$ ,  $\alpha_i$ , sedangkan  $d_i$  merupakan variabel. Gambar 2.14 hanya menampilkan *joint* yang dapat berputar dan tidak dapat bergeser. Dengan memperhatikan konversi Denavit-Hartenberg, matriks transformasi Denavit-Hartenberg untuk sistem koordinat yang berdekatan  ${}^{i-1}A_i$ , yaitu sistem koordinat i dan sistem koordinat i-1 untuk *joint* putar.

$${}^{i-1}A_i = T_{z,d}T_{x,\theta}T_{x,a}T_{x,\alpha}$$

$$\begin{aligned}
&= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos \theta_i & -\sin \theta_i & 0 & 0 \\ \sin \theta_i & \cos \theta_i & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & a_i \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos \alpha_i & -\sin \alpha_i & 0 \\ 0 & \sin \alpha_i & \cos \alpha_i & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
&= \begin{bmatrix} \cos \theta_i & -\cos \alpha_i \sin \theta_i & \sin \alpha_i \sin \theta_i & a_i \cos \theta_i \\ \sin \theta_i & \cos \alpha_i \cos \theta_i & -\sin \alpha_i \cos \theta_i & a_i \sin \theta_i \\ 0 & \sin \alpha_i & \cos \alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2-42)
\end{aligned}$$

Matriks transformasi *homogeneous*  ${}^0T_i$  menunjukkan lokasi dari sistem koordinat ke-i dengan mengacu ke sistem koordinat dasar dan merupakan perkalian dari matriks-matriks transformasi koordinat suksesif  ${}^{i-1}A_i$ . Kalau *joint* yang dipakai dibatasi hanya 3 maka  ${}^0T_3$  yang dicari merupakan perkalian dari  ${}^0A_1$ ,  ${}^1A_2$ , dan  ${}^2A_3$ .

$${}^0T_3 = {}^0A_1 {}^1A_2 {}^2A_3 \quad (2-43)$$

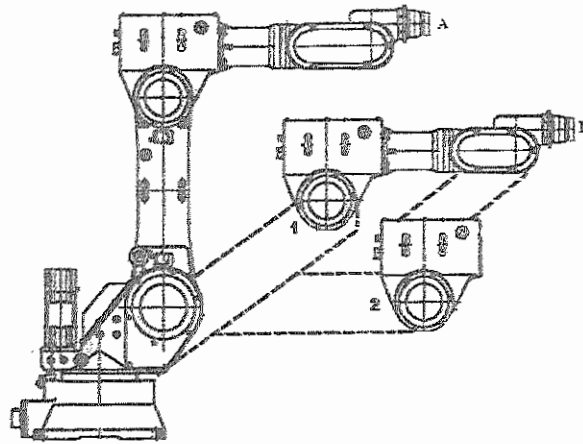
### 2.2.6 Invers Kinematics

*Invers Kinematics* diperlukan pada pengendalian posisi dan orientasi dari *end effector* robot untuk mencapai suatu objek di dalam sistem koordinat. Berdasarkan posisi dan orientasi ( ${}^0T_i$ ) yang ingin dicapai, *Invers Kinematics* digunakan untuk mendapatkan posisi dan orientasi tersebut (Fu, Gonzales dan Lee, 1987, p52).

Sebelum melakukan pencarian nilai  $\theta_i$ , perlu diketahui dahulu kemungkinan adanya solusi. Kemungkinan untuk mendapatkan solusi sangat penting untuk diketahui, sehingga perhitungan untuk mencari solusi tidak perlu dilakukan apabila tidak ada jaminan bisa mendapatkan solusi. Ada tidaknya solusi berhubungan erat dengan Area Jangkauan (*Workspace*) robot. Apabila posisi dan orientasi (titik tujuan) dari *end effector* berada di dalam area jangkauan, maka sekurang-kurangnya terdapat satu solusi (Craig, 1989, p114-116).

Apabila solusinya ada, maka kemungkinan lain bisa terjadi adalah solusinya lebih dari satu (multi solusi). Hal ini menjadi masalah karena sistem dituntut hanya bisa memilih satu solusi saja (Craig, 1989, p116-118).





Gambar 2.15 Dua solusi untuk satu posisi

Sebagai contoh bisa dilihat pada gambar 2.15, berdasarkan gambar tersebut, *end effector* semula berada pada posisi A dan diinginkan pindah ke posisi B. Untuk mencapai posisi B ada dua macam konfigurasi gerakan manipulator yang bisa dilakukan (ditunjukkan dengan garis putus-putus 1 dan 2). Kedua solusi tersebut sama-sama bisa mendapat posisi yang sama.

Bila terdapat lebih dari satu solusi, maka akan sangat baik untuk memilih solusi dengan gerakan paling sedikit pada *joint-joint*, berdasarkan contoh gambar 2.10 diatas, maka garis putus-putus paling atas (1) merupakan pilihan terbaik, karena posisi (2) dapat membuat motor penggerak yang ada pada robot akan menjadi panas dan cepat rusak. Namun apabila ada halangan atau rintangan ketika memilih solusi dengan gerakan yang paling minimal, maka solusi dengan pergerakan yang lebih jauh bisa dipilih. Hal ini akan menunjukkan perlunya menghitung semua solusi yang mungkin dilakukan.

Banyaknya jumlah solusi ditentukan oleh jumlah *joint* yang dimiliki manipulator, dan fungsi dari parameter-parameter *link* ( $d_i$ ,  $a_i$ , dan  $\alpha_i$  pada manipulator dengan *joint* putar dan  $\alpha_i$ ,  $a_i$ , dan  $\theta_i$  pada manipulator dengan *joint* prismatic) yang tidak sama dengan nol. Semakin banyak parameter *link* yang bernilai nol, maka semakin banyak pula cara atau solusi untuk mencapai suatu titik tujuan. Oleh sebab itu dalam *software* simulasi yang akan dibuat nantinya adalah menggunakan (1) solusi.

Berdasarkan (Fu, Gonzales dan Lee, 1987, p53), ada dua cara untuk mencari solusi *Inverse Kinematics* : teknik transformasi *Inverse* untuk mencari sudut Euler, dan pendekatan geometris.

### Teknik Transformasi *Inverse* untuk Solusi Sudut Euler

Berdasarkan (Fu, Gonzales dan Lee, 1987, p53-60), solusi dicari dengan menggunakan konsep dasar dari teknik transformasi *Inverse* untuk mencari sudut Euler. Karena matriks rotasi 3x3 dapat ditulis dengan persamaan sudut Euler ( $\Phi$ ,  $\theta$ ,  $\psi$ ) sesuai dengan persamaan di bawah ini,

$$\begin{bmatrix} n_x & s_x & a_x \\ n_y & s_y & a_y \\ n_z & s_z & a_z \end{bmatrix} = R_{z,\Phi} R_{u,\theta} R_{w,\psi} \quad (2-44)$$

$$\begin{bmatrix} n_x & s_x & a_x \\ n_y & s_y & a_y \\ n_z & s_z & a_z \end{bmatrix} = \begin{bmatrix} C\Phi C\psi - S\Phi C\theta S\psi & -C\Phi S\psi - S\Phi C\theta C\psi & S\Phi S\theta \\ S\Phi C\psi + C\Phi C\theta S\psi & -S\Phi S\psi + C\Phi C\theta C\psi & -C\Phi S\theta \\ S\theta S\psi & S\theta C\psi & C\theta \end{bmatrix} \quad (2-45)$$

Dimana :

$n$  = *normal vector*, orthogonal terhadap *end effector*.

$s$  = *sliding vector*, menunjuk *end effector* bergerak (membuka atau menutup).

$a$  = *approach vector*, menunjuk ke arah luar *end effector*.

$C\theta = \cos \theta$ ,  $S\theta = \sin \theta$ ,  $C\Phi = \cos \Phi$ ,  $S\Phi = \sin \Phi$ ,  $C\psi = \cos \psi$ ,  $S\psi = \sin \psi$ .

Pada persamaan diatas (2-42) jika ruas kiri sama dengan ruas kanan, maka diperoleh

$$n_x = C\Phi C\psi - S\Phi C\theta S\psi \quad (2-46a)$$

$$n_y = S\Phi C\psi + C\Phi C\theta S\psi \quad (2-46b)$$

$$n_z = S\theta S\psi \quad (2-46c)$$

$$s_x = -C\Phi S\psi - S\Phi C\theta C\psi \quad (2-46d)$$

$$s_y = -S\Phi S\psi + C\Phi C\theta C\psi \quad (2-46e)$$

$$s_z = S\theta C\psi \quad (2-46f)$$

$$a_x = S\Phi S\theta \quad (2-46g)$$

$$a_y = -C\Phi S\theta \quad (2-46h)$$

$$a_z = C\theta \quad (2-46i)$$

Kemudian gunakan persamaan (2-46f), (2-46h), dan (2-46i) untuk mencari nilai  $\Phi$ ,  $\theta$ ,  $\psi$ , maka diperoleh

$$\theta = \cos^{-1}(a_z) \quad (2-47)$$

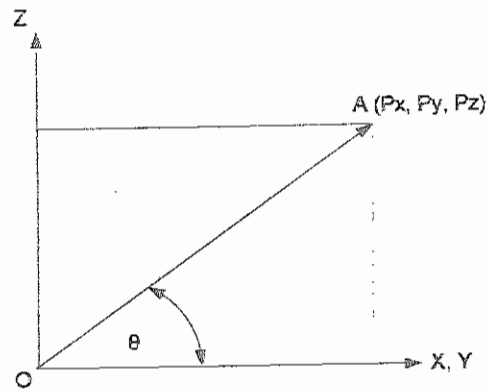
$$\psi = \cos^{-1}\left(\frac{s_z}{S\theta}\right) \quad (2-48)$$

$$\Phi = -\cos^{-1}\left(\frac{-a_y}{S\theta}\right) \quad (2-49)$$

Ada dua jenis pendekatan yang bisa digunakan untuk mendapatkan solusi *Invers Kinematics*, yaitu pendekatan numerik dan pendekatan analitik. Pada pendekatan numerik, dilakukan perhitungan berulang-ulang untuk mendapatkan suatu solusi pendekatan. Sedangkan pada pendekatan analitik, dilakukan analisis matematis tanpa melakukan perhitungan yang berulang-ulang untuk mendapatkan jawaban yang eksak (tepat). Pendekatan analitik dibagi menjadi dua macam, yaitu cara *algebraic* dan cara geometris (Craig, 1989, p118-120).

### Pendekatan Geometris

Cara geometris ini didapat berdasarkan sistem koordinat *link* dan geometri tangan manusia. Pada pendekatan ini dibagi menjadi 2 macam solusi yaitu solusi untuk posisi dan untuk orientasi. Solusi untuk posisi menyelesaikan 3 *joint* pertama sedangkan untuk solusi orientasinya menyelesaikan 3 *joint* terakhir. Dalam penyelesaiannya menggunakan trigonometri untuk mendapatkan sudut-sudutnya. Berdasarkan (Fu, Gonzales dan Lee, 1987, p60-69), cara geometris, solusi dicari dengan menerapkan ilmu-ilmu geometri dan hukum-hukum trigonometri. Sebagai contoh :

Gambar 2.16 Sudut *joint* 1

Untuk mencari  $\theta_i$ , bisa digunakan ilmu-ilmu geometri, sehingga didapatkan :

$$\sin \theta_i = \frac{P_z}{\sqrt{(P_x^2 + P_y^2 + P_z^2)}} \quad (2-50)$$

$$\cos \theta_i = \frac{\sqrt{(P_x^2 + P_y^2)}}{\sqrt{(P_x^2 + P_y^2 + P_z^2)}} \quad (2-51)$$

$$\theta_i = \arctan 2 (\sin \theta_i, \cos \theta_i) \quad (2-52)$$

### Pendekatan *Algebraic*

Berdasarkan (Craig, 1989, p131-136), pada cara *algebraic*, solusi dicari dengan cara melakukan penurunan persamaan-persamaan kinematika dari matriks transformasi *homogeneous* sehingga didapat sudut-sudut *joint* yang dibutuhkan.

Sebagai contoh, adalah :

$${}^0T_3 = {}^0A_1 {}^1A_2 {}^2A_3 \quad (2-53)$$

dimana :

$${}^0T_1 = {}^0A_1$$

$${}^1T_2 = {}^1A_2$$

$${}^2T_3 = {}^2A_3$$

Untuk mencari besarnya pergerakan sudut pada setiap *joint*, bisa dilakukan dengan memindahkan elemen dari berbagai kiri, sehingga menjadi :

$$({}^0T_1)^{-1} {}^0T_3 = {}^1T_2 {}^2T_3 \quad (2-54)$$

$$({}^0T_1)^{-1} {}^0T_3 = {}^1T_3 \quad (2-55)$$

dengan menggunakan persamaan di atas, maka bisa dicari besarnya sudut-sudut *joint* yang dibutuhkan untuk mencapai suatu posisi.

Apabila masih ada sudut *joint* lain yang belum didapatkan, maka pindahkan lagi elemen di bagian kanan ke bagian kiri, sehingga menjadi :

$$({}^0T_1)^{-1} ({}^1T_2)^{-1} {}^0T_3 = {}^2T_3 \quad (2-56)$$

$$({}^0T_2)^{-1} {}^0T_3 = {}^2T_3 \quad (2-57)$$

dengan bantuan persamaan di atas, maka solusi untuk *Invers Kinematics* bisa didapatkan.

### Pendekatan Numerik

Pada robot yang distandarisasi dapat dirangkai dalam struktur dengan beragam derajat kebebasan dan bentuk untuk tugas yang berbeda. Pada umumnya robot manipulator yang didesain, solusi *inverse kinematics* diturunkan sendiri dan disimpan sebagai fungsi pustaka dalam pengendalian suatu robot. Bagaimanapun untuk modul sistem yang dapat dipasang, untuk jumlah geometri dan derajat kebebasan bisa didapat dari modul yang meningkat secara eksponensial. Pendekatan numerik tidak mengutamakan diperolehnya jawaban yang eksak (tepat), tetapi mengusahakan perumusan metode yang menghasilkan jawaban pendekatan yang berbeda dari jawaban yang eksak sebesar suatu nilai yang dapat diterima berdasarkan pertimbangan praktis, tetapi cukup dapat memberikan penghayatan pada persoalan yang dihadapi (Harijono Djojodiharjo, 2000, p3).

Pendekatan numerik memanfaatkan kemampuan komputer untuk melakukan perhitungan secara berulang-ulang di dalam mendapatkan solusi *Invers Kinematics* komputer akan terus menghitung semua kemungkinan solusi secara berulang-ulang sampai didapatkan suatu solusi yang sesuai untuk sudut-sudut *joint* yang dibutuhkan agar bisa mencapai posisi dan orientasi yang diinginkan.

Cara numerik relatif lebih mudah, namun menghabiskan lebih banyak waktu karena sifat perulangannya (Craig, 1989, p119-120).