

PENGENALAN BENTUK BANGUN GEOMETRIS DUA DIMENSI DENGAN MENGGUNAKAN METODE DESKRIPSI DAN PENCOCOKAN

Yulia

Fakultas Teknologi Industri, Jurusan Teknik Informatika – Universitas Kristen Petra
e-mail : yulia@petra.ac.id

Agustinus Noertjahyana

Fakultas Teknologi Industri, Jurusan Teknik Informatika – Universitas Kristen Petra
e-mail : agust@petra.ac.id

ABSTRAK: Deskripsi dan pencocokan merupakan suatu metode dalam ilmu kecerdasan buatan, yang digunakan untuk mengenali keberadaan suatu objek dalam suatu gambar dengan mencocokkan gambar tersebut dengan sekumpulan gambar lain sebagai gambar acuan.

Penelitian ini menyajikan suatu sistem yang dapat mengenali bentuk bangun geometris dua dimensi dengan cara deskripsi dan pencocokan (*describe and match*) dimana bangun geometris dua dimensi yang digunakan di sini dibatasi untuk bentuk lingkaran, segiempat, segitiga, dan garis.

Tiga bentuk permasalahan yang akan dibahas dalam penelitian ini yaitu *Library Matching Problem*, *Describe and Transform Problem* serta *Describe, Match and Transform Problem*.

Kata kunci: Deskripsi dan pencocokan, kecerdasan buatan, bangun geometris dua dimensi.

ABSTRACT: *Describe and match is a method in artificial intelligence, which used to recognize the existence of an object in a picture by matching the picture with a group of other picture as a reference.*

This research presents a system to recognize two-dimensional geometric structure through describe and match where the two-dimensional geometric structures which are used here are limited on circle, square, triangle, and line.

Three problems, which we will discuss in this research, are Library Matching Problem, Describe and Transform Problem and Describe, Match and Transform Problem.

Keywords: *Describe and match, artificial intelligence, two-dimensional geometric structure.*

1. PENDAHULUAN

Salah satu konsep pengenalan pola adalah metode Deskripsi dan Pencocokan. Dalam metode ini suatu bentuk bangun geometris pada gambar akan dideskripsikan dengan cara mendeskripsikan hubungan antar objek pada gambar. Setelah itu hasil pendeskripsian akan digunakan untuk dicocokkan dengan hasil pendeskripsian gambar yang lain.

2. ALGORITMA

2.1 Deskripsi

2.1.1 Deskripsi Suatu Objek

Deskripsi suatu objek digunakan untuk mengetahui properti seperti warna cat, corak

cat, sudut kemiringan, ukuran, serta posisi dari suatu objek.

2.1.2 Deskripsi Antar Objek

Deskripsi antar dua objek digunakan untuk mengetahui hubungan antar objek dalam suatu layar. Deskripsi tersebut meliputi :

Di Atas

$$Di\ Atas(Objek1,Objek2)=\begin{cases} True & \text{jika}(Y_1 < Y_2) \\ False & \text{selainnya} \end{cases}$$

Di Bawah

$$Di\ Bawah(Objek1,Objek2)=\begin{cases} True & \text{jika}(Y_1 > Y_2) \\ False & \text{selainnya} \end{cases}$$

Di Kiri

$$Di\ Kiri(Objek1,Objek2)=\begin{cases} True & \text{jika}(X_1 < X_2) \\ False & \text{selainnya} \end{cases}$$

Di Kanan

$$Di\ Kanan(Objek1,Objek2) = \begin{cases} True & \text{jika}(X_1 > X_2) \\ False & \text{selainnya} \end{cases}$$

Keterangan :

X_1, Y_1 = titik pusat dari Objek1

X_2, Y_2 = titik pusat dari Objek2

Di Dalam

DiDalam(Objek1, Objek2)

$$= \begin{cases} True & \text{jika}\{(X_{11} \leq X_{22}) \text{ dan } (X_{11} \geq X_{21}) \\ & \text{dan } (X_{12} \leq X_{22}) \text{ dan } (X_{12} \geq X_{21}) \text{ dan} \\ & (Y_{11} \leq Y_{22}) \text{ dan } (Y_{11} \geq Y_{21}) \text{ dan } (Y_{12} \leq Y_{22}) \\ & \text{dan } (Y_{12} \geq Y_{21})\} \\ False & \text{selainnya} \end{cases}$$

X_{11}, Y_{11} = titik terkecil dari Objek1

X_{12}, Y_{12} = titik terbesar dari Objek1

X_{21}, Y_{21} = titik terkecil dari Objek2

X_{22}, Y_{22} = titik terbesar dari Objek2

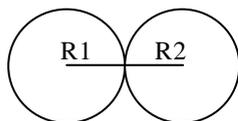
Menempel

Pertama-tama dicari persamaan garis dari semua sisi pada objek 1 dan objek 2, jika objek 1 atau objek 2 tersebut bukan lingkaran, tetapi jika objek 1 atau objek 2 tersebut lingkaran maka dicari persamaan lingkarannya.

Setelah persamaan garis atau lingkaran diketahui maka :

- Jika kedua objek adalah lingkaran :

Contoh :



Gambar 1. Deskripsi Menempel antara Dua Lingkaran

Dicari titik tengah kedua objek, kemudian dicari jarak dua buah titik tersebut (misal diberi nama Jarak). Jika Jarak = $R_1 + R_2$ maka dapat dikatakan bahwa objek1 menempel pada objek 2.

Menempel (Objek1,Obtek2)

$$= \begin{cases} True & \text{jika } J = R_1 + R_2 \\ False & \text{selainnya} \end{cases}$$

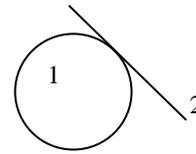
J = Jarak antara Objek1 dengan Objek2

R_1 = Jari-jari Objek1

R_2 = Jari-jari Objek2

- Jika objek 1 adalah lingkaran dan objek 2 bukan lingkaran :

Contoh :



Gambar 2. Deskripsi Menempel Lingkaran dan Bukan Lingkaran

Pertama-tama dicari jarak antar titik pusat lingkaran dengan titik-titik ujung garis. Kemudian diperiksa apakah kedua ujung garis tersebut ada di luar lingkaran. Jika kedua ujung garis tersebut di luar lingkaran maka dicari jarak antara titik pusat lingkaran dengan garis. Jika jarak tersebut sama dengan panjang jari-jari lingkaran maka dapat dikatakan bahwa dua objek tersebut saling menempel.

Menempel(Objek1,Objek2)

$$= \begin{cases} True & \text{jika}(J_1 > R) \text{ dan } (J_2 > R) \text{ dan } (J_3 = R) \\ False & \text{selainnya} \end{cases}$$

J_1 = Jarak antara titik ujung pertama dari Objek2 dengan titik pusat Objek1

J_2 = Jarak antara titik ujung kedua dari Objek2 dengan titik pusat Objek1

J_3 = Jarak terdekat dari Objek2 dengan titik pusat Objek1

R = Jari-jari Objek1

- Jika kedua objek bukan lingkaran
Antara sisi-sisi pada objek 1 dan objek 2 diperiksa, apakah ada di antara sisi dari objek1 yang mempunyai persamaan garis yang sama dengan sisi dari objek 2. Jika ada maka dapat dikatakan bahwa kedua objek saling menempel.

Menempel (Objek1,Objek2)

$$= \begin{cases} True & \text{jika}(Y_1 = Y_2) \text{ dan } (M_1 = M_2) \text{ dan} \\ & (X_1 = X_2) \text{ dan } (C_1 = C_2) \\ False & \text{selainnya} \end{cases}$$

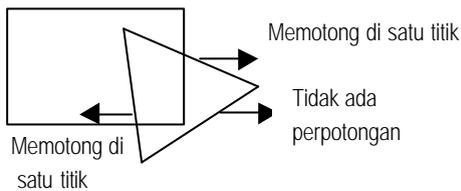
$Y_1 = M_1 X_1 + C_1 \rightarrow$ persamaan garis dari Objek1

$Y_2 = M_2 X_2 + C_2 \rightarrow$ persamaan garis dari Objek2

Memasuki

Suatu objek (1) dikatakan memasuki objek yang lain (2) jika ada bagian dari objek 1 yang memasuki objek 2. Hal ini dinyatakan dengan notasi Memasuki (Objek1, Objek2).

Dengan memeriksa hubungan setiap garis pada sisi-sisi kedua objek, misalnya:



Gambar 3. Deskripsi Memasuki

Sedangkan untuk mencari titik potong adalah : misalnya ada 2 garis yaitu garis 1 dan garis 2. Persamaan garis1 $Y_1 = M_1 X_1 + C_1$ dan persamaan garis2 $Y_2 = M_2 X_2 + C_2$, maka titik potongnya adalah :

Untuk koordinat X :

$$I = C_1 + \frac{1}{M_2} * M_1 \quad X = \frac{I - C_2}{M_2 + \frac{1}{M_2}}$$

Untuk koordinat Y

$$I = C_1 + \frac{1}{M_2} * M_1 \quad X = -\frac{1}{M_2} * X + I$$

Jika diketahui ada lebih dari atau sama dengan 1 garis dari objek 1 yang berpotongan di satu titik pada garis dari objek 2, maka dapat dikatakan bahwa objek 1 memasuki objek 2.

Memasuki (Objek1,Objek2)

$$= \begin{cases} True & \text{jika } N \geq 1 \\ False & \text{selainnya} \end{cases}$$

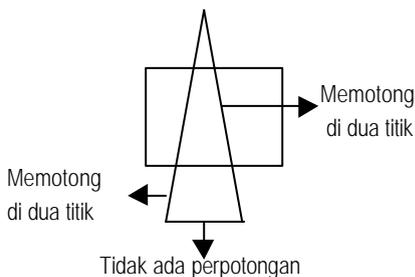
N = jumlah sisi dari Objek1 yang memotong sisi dari Objek2 di satu titik

Menembus

Suatu objek(1) dikatakan menembus objek(2) yang lain jika objek 1 menembus objek 2 (memasuki objek 2 sampai ada bagian dari objek 1 yang keluar dari objek 2) .

Aturannya :

Dengan memeriksa hubungan setiap garis pada sisi-sisi kedua objek, misalnya:



Gambar 4. Deskripsi Menembus

Kemudian dicari titik potong antara dua garis. Jika diketahui ada lebih dari atau sama dengan 2 (dua) garis dari objek 1 yang berpotongan di dua titik pada garis dari objek 2, maka dapat dikatakan bahwa objek 1 menembus objek 2.

Menembus(Objek1,Objek2)

$$= \begin{cases} True & \text{jika } N \geq 2 \\ False & \text{selainnya} \end{cases}$$

N = jumlah sisi dari Objek1 yang memotong sisi dari Objek2 di dua titik.

Ada Jarak

Suatu objek(1) dikatakan ada jarak objek(2) yang lain jika objek 1 tersebut mempunyai jarak dengan objek 2.

AdaJarak(Objek1,Objek2)

$$= \begin{cases} True & \text{jika } \text{menempel}(Objek1,Objek2) = False \\ & \text{dan } \text{memasuki}(Objek1,Objek2) = False \\ & \text{dan } \text{menembus}(Objek1,Objek2) = False \\ & \text{dan } \text{DiDalam}(Objek1,Objek2) = False \\ False & \text{selainnya} \end{cases}$$

2.2 Transformasi

Pada bagian ini, dijelaskan jenis-jenis transformasi yang terjadi antara objek-objek di layar asal dengan objek-objek di layar hasil transformasi, dimana transformasi tersebut meliputi :

Diputar / Dirotasikan

Suatu Objek A di layar asal dikatakan dirotasikan menjadi Objek A` di layar hasil transformasi, jika sudut kemiringan dari Objek A berbeda dengan sudut kemiringan Objek A`.

Dihapus

Suatu Objek A di layar yang satu dikatakan dihapus di layar yang lain, jika Objek A tersebut dari ada di layar asal menjadi tidak ada di layar hasil transformasi.

Ditambah

Transformasi ini terjadi jika suatu objek dari tidak ada di layar asal menjadi ada di layar hasil transformasi.

Diperbesar

Suatu Objek A di layar asal dikatakan diperbesar menjadi Objek A` di layar hasil transformasi, jika ukuran sisi-sisi dari Objek

A` lebih besar n kali daripada ukuran sisi-sisi Objek A.

Diperkecil

Suatu Objek A di layar asal dikatakan diperkecil menjadi Objek A` di layar hasil transformasi, jika ukuran sisi-sisi dari Objek A` lebih kecil n kali daripada ukuran sisi-sisi Objek A.

Di-Shearing

Suatu Objek A di layar asal dikatakan di-*shearing* menjadi Objek A` di layar hasil transformasi, jika ukuran sisi dari Objek A` berbeda dengan ukuran sisi Objek A dan perbandingan perbedaan ukuran antar sisi tersebut tidak sama.

Berubah warna cat

Suatu Objek A di layar asal dikatakan berubah warna cat menjadi Objek A` di layar hasil transformasi, jika warna cat dari Objek A berbeda dengan warna cat dari Objek A`.

Berubah corak cat

Suatu Objek A di layar asal dikatakan berubah corak cat menjadi Objek A` di layar hasil transformasi, jika corak cat dari Objek A berbeda dengan corak cat dari Objek A`.

Translasi

Suatu Objek A di layar asal dikatakan bertranslasi menjadi Objek A` di layar hasil transformasi, jika posisi Objek A terhadap objek acuan di layar asal berbeda dengan posisi dari Objek A` terhadap objek acuan di layar hasil transformasi.

Berubah Bentuk

Suatu Objek A di layar asal dikatakan berubah bentuk menjadi Objek A` di layar hasil transformasi, jika tipe bentuk Objek A di layar asal berbeda dengan tipe bentuk dari Objek A` di layar hasil transformasi.

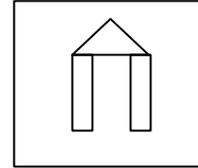
2.3 Pencocokan

Dalam makalah ini, teknik pencocokan digunakan untuk mencocokkan hasil deskripsi tiap layar dan transformasi antar layar yang telah dilakukan.

Teknik ini hanya digunakan untuk *Library Matching Problem* dan *Describe, Transform and Match Problem*. Tetapi untuk tiap masalah tersebut, penggunaan teknik ini berbeda.

3. LIBRARY MATCHING PROBLEM

Pada *library matching problem* ini dibutuhkan dua proses. Proses pertama adalah *proses pendeskripsian* objek-objek pada layar dan proses kedua adalah *proses pencocokan* hasil deskripsi tadi dengan hasil deskripsi bentuk-bentuk yang ada di *library*. Jika ada hasil pencocokan deskripsi tersebut mirip, maka sistem akan menyebutkan jenis bangun tersebut.



Gambar 5. Gapura

Prosedur analogi pada contoh gambar di atas adalah sebagai berikut:

Prosedur deskripsi:

Deskripsi suatu objek

Meliputi tipe, top, left, posisi titik-titik sudut, panjang, tinggi, sisi atas, sisi bawah, sudut, warna cat, corak cat dari tiap objek dalam gambar.

Deskripsi antar objek

1. Ada 3 objek, yaitu 2 kotak dan 1 segitiga.
2. 2 kotak tersebut mendukung atau menempel pada segitiga.
3. Segitiga di atas kotak 1 dan kotak 2
4. Antara 2 kotak tersebut ada jarak.

Prosedur pencocokan:

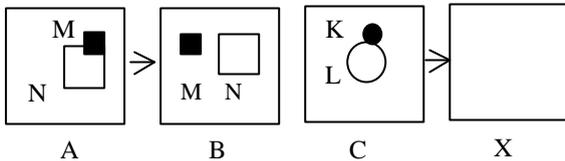
Hasil dari deskripsi pada layar tersebut kemudian dicocokkan dengan deskripsi semua bentuk gambar pada file *library*. Ternyata bentuk tersebut paling sesuai dengan deskripsi bentuk 'Gapura', dimana deskripsi gapura adalah :

1. Ada 3 objek, yaitu 2 kotak dan 1 segitiga.
2. 2 kotak tersebut mendukung atau menempel pada segitiga.
3. Segitiga di atas kotak 1 dan kotak 2
4. Antara 2 kotak tersebut ada jarak.

Karena deskripsi gambar tersebut paling mirip dengan deskripsi gapura, maka bentuk bangun tersebut disebut 'Gapura'.

4. DESCRIBE AND TRANSFORM PROBLEM

Describe and Transform Problem adalah suatu masalah diberikan dimana dari masalah yang diberikan pada layar A,B dan C, diminta untuk memberikan solusinya di layar X.



Gambar 6. Describe and Transform Problem

Di sini diketahui bahwa dari A bisa menjadi B dan dari C menjadi X, tetapi X tidak diketahui dan komputer diminta untuk memberikan jawabannya.

Pada *describe and transform problem* ini diperlukan dua proses. Yang pertama adalah *proses deskripsi* objek-objek yang ada di layar A, B dan C. Proses yang kedua adalah *proses transformasi* objek-objek di layar A ke objek-objek di layar B. Hasil dari transformasi dari layar A ke layar B ini dijadikan dasar untuk menggambar objek-objek di layar X, dimana transformasi yang terjadi pada masing-masing objek dari layar A ke B juga dilakukan pada objek-objek dari layar C ke layar X.

Prosedur analogi pada contoh gambar di atas adalah sebagai berikut:

Prosedur deskripsi :

Untuk *proses deskripsi*, yang dilakukan adalah deskripsi diri sendiri dan deskripsi antar objek. Salah satu contohnya untuk deskripsi layar A di atas adalah:

Deskripsi suatu objek

Meliputi tipe, top, left, posisi titik-titik sudut, panjang, tinggi, sisi atas, sisi bawah, sudut, warna cat, corak cat dari tiap objek dalam gambar.

Deskripsi antar objek

- Kotak M di atas kotak N
- Kotak M di kanan kotak N
- Kotak M ada jarak dengan kotak N

Prosedur transformasi:

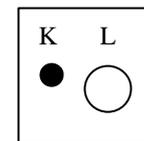
Salah satu contoh *prosedur transformasi* untuk contoh di atas adalah transformasi

yang terjadi dari layar A ke layar B, yaitu: N ditranslasikan sebesar jarak J dan sudut S terhadap M.

Untuk setiap transformasi yang terjadi akan mengeluarkan nilai hasil transformasi tersebut. Misalnya untuk kasus di atas, transformasi yang terjadi adalah translasi. Hasil dari translasi adalah jarak dan beda sudut suatu objek dari objek acuan.

Misal :
N ditranslasikan
(jarak(j), sudut(s))

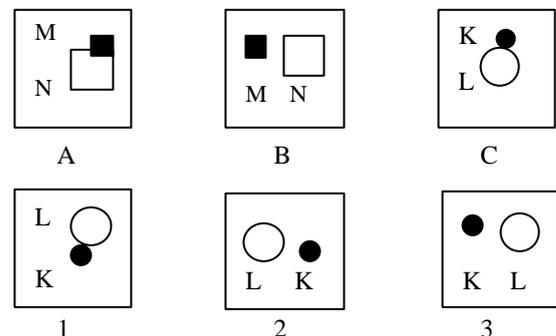
Berarti N ditranslasikan dengan beda jarak sebesar j dari objek acuan (untuk kasus ini objek acuan adalah M) dan dengan perbedaan sudut sebesar s dari objek acuan. Hasil dari *prosedur transformasi* tersebut di atas, dijadikan dasar untuk menggambar objek-objek X di layar X. Hasil dari layar X adalah :



Gambar 7. Hasil Describe and Transform Problem

5. DESCRIBE, TRANSFORM AND MATCH PROBLEM

Describe, Transform and Match Problem adalah suatu masalah dimana dari masalah yang ada, diberikan alternatif jawaban yang dapat dipilih.



Gambar 8. Describe, Transform and Match Problem

Di sini masalahnya adalah memilih jawaban pada layar 1..3, dimana jika A menjadi B, maka C menjadi X (salah satu di antara layar 1..3).

Pada masalah ini diperlukan tiga proses. Yang pertama adalah *proses deskripsi* objek-objek yang ada di layar A, B, C, 1, 2 dan 3. Proses yang kedua adalah *proses transformasi* objek-objek di layar A ke objek-objek di layar B serta transformasi objek-objek di layar C ke layar 1, 2 dan 3. Sedangkan proses yang ketiga adalah *proses pencocokan* hasil transformasi layar A ke layar B dengan hasil transformasi layar C ke layar 1, 2 dan 3. Hasil pencocokan yang paling cocok yang dipilih sebagai jawaban. Prosedur analogi pada contoh gambar di atas adalah sebagai berikut:

Prosedur deskripsi :

Deskripsi suatu objek

Meliputi tipe, top, left, posisi titik-titik sudut, panjang, tinggi, sisi atas, sisi bawah, sudut, warna cat, corak cat dari tiap objek dalam gambar.

Deskripsi antar objek

Kotak M di atas kotak N

Kotak M di kiri kotak N

Kotak M ada jarak dengan kotak N

Prosedur transformasi :

Salah satu contoh *prosedur transformasi* untuk contoh di atas adalah transformasi yang terjadi dari layar A ke layar B, yaitu : N ditranslasikan sebesar jarak J dan sudut S terhadap M.

Untuk setiap transformasi yang terjadi akan mengeluarkan nilai hasil transformasi tersebut. Misalnya untuk kasus di atas, transformasi yang terjadi adalah translasi. Hasil dari translasi adalah jarak dan beda sudut suatu objek dari objek acuan.

Misal :

N ditranslasikan(jarak (j), sudut(s))

Berarti N ditranslasikan dengan beda jarak sebesar j dari objek acuan (untuk kasus ini objek acuan adalah M) dan dengan perbedaan sudut sebesar s dari objek acuan.

Proses transformasi pada contoh di atas adalah :

Transformasi dari layar A ke B

N ditranslasikan(jarak, sudut)

Transformasi dari layar C ke 1

K ditranslasikan(jarak,sudut)

Transformasi dari layar C ke 2

K ditranslasikan(jarak,sudut)

Transformasi dari layar C ke 3

K ditranslasikan(jarak,sudut)

Prosedur pencocokan :

Untuk *proses pencocokan* ini selain mencocokkan jenis transformasi, juga mencocokkan hasil dari transformasi. Jika terjadi kesamaan transformasi seperti contoh di atas, maka yang dijadikan jawaban adalah yang mempunyai hasil translasi yang paling mendekati translasi dari layar A ke layar B.

Setelah proses pencocokan tersebut dapat diketahui bahwa jawaban yang sesuai adalah transformasi dari layar C ke layar 3, sehingga layar 3 adalah jawaban dari contoh masalah tersebut.

5. KESIMPULAN

Hasil pembuatan penelitian ini dapat digunakan untuk mengenali bentuk geometris dua dimensi, dimana untuk mengenali bentuk diperlukan proses deskripsi dan pencocokan sedang untuk menganalogikan perubahan bentuk diperlukan proses deskripsi, transformasi dan pencocokan.

Tujuan dari pembuatan penelitian ini adalah untuk pengembangan ilmu kecerdasan buatan, dimana program ini berusaha mengaplikasikan konsep-konsep yang ada pada Kecerdasan Buatan, khususnya konsep pengenalan pola (*pattern matching*) pada benda-benda geometris. Dengan ini penulis membuktikan bahwa konsep *describe and match* dapat diaplikasikan.

DAFTAR PUSTAKA

1. Levine I. R, Drange E.D., Edelson B., *Artificial Intelligence and Expert Systems*, McGraw-Hill International Editions, 1991.
2. Rich E. dan Kight K., *Artificial Intelligence*, McGraw-Hill, Inc, 1991
3. Tong C. dan Sriram D., *Artificial Intelligence in Engineering Design*, Academic Press, Inc, 1992.
4. Winston, P. H., *Artificial Intelligence*, Addison-Wesley Publishing Company (Reading, Massachusetts), 1984.