

HASIL CEK_60030480

by Ardi Pujiyanta4 60030480

Submission date: 16-Jun-2021 07:06PM (UTC+0700)

Submission ID: 1607453960

File name: pengenalan_citra_objek_sederhana.pdf (840.09K)

Word count: 1864

Character count: 10728

PENGENALAN CITRA OBJEK SEDERHANA DENGAN JARINGAN SARAF TIRUAN METODE PERCEPTRON

1 **Ardi Pujiyanta**
Program Studi Teknik Informatika, Fakultas Teknik Industri
Universitas Ahmad Dahlan Yogyakarta

ABSTRAK

Konsep bangunan dan benda-benda yang ada di sekeliling didasarkan dan dipengaruhi oleh konsep objek sederhana atau sering disebut geometri ruang tiga dimensi, yaitu memiliki panjang, lebar dan tinggi. Namun, dalam rancangan dan penggambarannya menggunakan gambar berdimensi dua saja. Sehingga pada konsep penggambarannya membutuhkan visualisasi yang lebih detail. Diharapkan jaringan syaraf tiruan metode perceptron dapat mengenali gambar yang sesuai dengan bentuk aslinya.

Pada penelitian ini metode jaringan saraf yang digunakan adalah metode perceptron untuk mengenali citra objek sederhana. Objek yang digunakan yaitu bentuk bangun ruang yang terdiri dari kubus, kerucut, tabung, prisma, dan limas dengan berbagai jenisnya. Perangkat lunak yang digunakan pada pembuatan aplikasi ini adalah Borland Delphi 7.0.

Dari hasil pelatihan dan pengujian jaringan saraf tiruan perceptron dapat mengenali pola dengan rata-rata 75,25 % dengan prosentase terendah yaitu 50,75 % dan prosentase tertinggi yaitu 92,65 %. Dengan prosentase yang cukup baik tersebut, sistem dapat digunakan untuk mengenali citra objek sederhana.

Kata kunci : *Objek sederhana, Jaringan syaraf tiruan, Perceptron.*

1. LATAR BELAKANG MASALAH

Konsep bangunan dan benda-benda yang ada di sekeliling didasarkan dan dipengaruhi oleh konsep geometri ruang tiga dimensi, yaitu memiliki panjang, lebar dan tinggi. Namun, dalam rancangan dan penggambarannya menggunakan gambar berdimensi dua saja. Orang yang merancang bangunan harus merancang dan menghitung dengan sangat teliti segala sesuatunya agar didapatkan suatu bangunan yang indah dan stabil. Mereka harus membuat gambar, sketsa ataupun model bangunan tersebut. Untuk mencapai hal tersebut, memerlukan kemampuan berpikir yang berkaitan dengan pengetahuan tentang keruangan (dimensi tiga).

Objek sederhana merupakan bagian dari geometri ruang, yang merupakan dasar dan konsep dari benda yang ada. Objek sederhana pada dasarnya didapat dari benda-benda konkret dengan melakukan abstraksi dan idealisasi. Abstraksi adalah proses memperhatikan dan menentukan sifat, atribut ataupun karakteristik khusus

yang penting saja dengan mengesampingkan hal-hal yang berbeda yang tidak penting. Sebagai contoh, dari benda konkret seperti potongan bambu atau kaleng minuman, proses berabstraksi terjadi ketika kita memperhatikan lalu mendapatkan hal-hal yang sama dari benda-benda tersebut dengan mengesampingkan hal yang berbeda yang tidak penting. Sehingga yang diperhatikan adalah bentuknya yang sama, yang pada akhirnya didapat kesimpulan bahwa bentuk dari 2 benda tersebut adalah sama yaitu yang disebut sebagai tabung. Sedang proses idealisasi adalah proses menganggap segala sesuatu dari benda-benda konkret itu ideal. Contohnya batang bambu yang tidak rata harus dianggap rata [13].

Manusia dapat mengetahui suatu ruang dari ciri dasarnya, yang diistilahkan sebagai [aksioma](#). Tetapi hal ini dipengaruhi oleh pengalaman atau mungkin intuitifnya atau strategi pemetaan dalam berpikir [13]. Sedangkan bagi komputer, untuk mengenali dan membedakan suatu citra diperlukan data-data fisik yaitu bentuk dari citra tersebut dan data-data lain seperti penambahan tekstur dan warna sehingga komputer akan mudah mengenalinya, namun dengan menggunakan metode yang ada dalam jaringan syaraf tiruan diharapkan komputer dapat mengenal citra objek sederhana ini.

2. BATASAN MASALAH

Berdasarkan latar belakang masalah diatas, maka permasalahan dibatasi pada masalah :

- a. Bentuk objek sederhana yang digunakan meliputi bentuk-bentuk objek : kubus, kerucut, tabung, prisma, dan limas dengan berbagai jenisnya.
- b. Metode yang digunakan pada penelitian ini adalah algoritma Jaringan Syaraf Tiruan *Perceptron*.
- c. Input data dibatasi hanya untuk citra yang berekstensi BMP (bitmap).

3. RUMUSAN MASALAH

Berdasarkan latar belakang masalah diatas, maka dapat diambil suatu rumusan masalah sebagai berikut :

- a. Dapatkah jaringan saraf tiruan metode *Perceptron* mengenali citra objek sederhana.

- b. Bagaimana arsitektur jaringan saraf tiruan yang baik agar dapat mengenali citra objek sederhana.

4. TUJUAN PENELITIAN

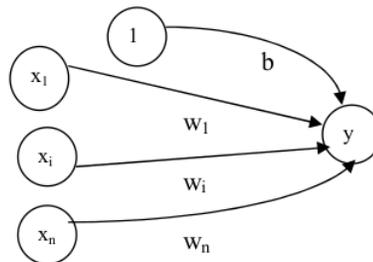
Tujuan dari penelitian ini adalah untuk mempelajari struktur dan parameter jaringan *Perceptron* yang optimal untuk pengenalan citra objek sederhana dan mengetahui performa dari metode pengenalan ini, yang selanjutnya bisa terbangun sistem pengenalan objek sederhana dengan metode tersebut.

5. LANDASAN TEORI

5.1. Arsitektur *perceptron*

Output dari unit assosiator adalah biner vektor. Vektor tersebut dikatakan sebagai sinyal input terhadap sinyal output atau unit response. Oleh karena bobot-bobot dari assosiator ke unit output dapat diubah-ubah, maka yang akan diperhatikan hanya pada bagian layer tersebut.

Tujuan dari jaringan ini adalah mengklasifikasikan setiap pola *input* ke dalam kelas tertentu. Apabila *outputnya* +1, maka *input* yang diberikan termasuk kelas tertentu, sebaliknya jika *outputnya* -1, maka *input* yang diberikan tidak masuk dalam kelas tertentu. Arsitektur *perceptron* digambarkan seperti yang tampak pada gambar 1 berikut :



Gambar 1. Arsitektur *perceptron* sederhana

Keterangan :

$x_1 \dots x_i \dots x_n$ = neuron input

y = neuron output

b = bias

w_1, w_i, w_n = bobot

5.2. Pelatihan *perceptron*

Bobot koneksi dari unit assosiator ke unit response (atau *output*) ditentukan melalui pelatihan (*learning rate*) *perceptron*. Untuk setiap *input* training jaringan akan menghitung response dari unit *output*, kemudian jaringan akan menentukan apakah suatu error terjadi pada pola tersebut dengan cara membandingkan *output* hasil perhitungan dengan nilai targetnya. Jaringan tersebut akan membedakan error antara *output* hasil perhitungan 0 dengan target -1 atau *output*nya +1 dengan target -1.

Dalam kedua kasus tersebut tanda dari error menunjukkan bahwa bobot koneksi harus diubah dalam arah yang dinyatakan oleh nilai target. Namun demikian hanya bobot-bobot pada koneksi dari unit pengiriman sinyal selain 0 ke unit *output* yang akan disesuaikan nilainya, karena hanya sinyal tersebut yang menambah error. Jika error tidak terjadi maka bobot-bobot tersebut tidak akan diubah tetapi sebaliknya jika suatu error terjadi untuk pola input pelatihan tertentu, bobot-bobot akan diubah menurut rumus :

$$w_i \text{ (new)} = w_i \text{ (old)} + \alpha t x_i$$

Keterangan :

x_i = *input* ke - i

t = target yang nilainya +1 atau -1

α = kecepatan belajar (*learning rate*) yaitu $0 < \alpha \leq 1$

w = bobot

jika error tidak terjadi, maka bobot-bobot tidak akan berubah.

5.3. Algoritma pelatihan *perceptron*

Algoritma *perceptron* yang digunakan mempunyai susunan sebagai berikut :

Langkah 1: a. Inisialisasi bobot dan bias (agar sederhana bobot dan bias mula-mula dibuat = 0)

b. Tentukan *learning rate* dengan nilai $0 < \alpha \leq 1$. Untuk penyederhanaan, α diberi nilai = 1

Langkah 2 : Selama ada elemen vektor masukan yang respon unit keluarannya tidak sama dengan target, lakukan :

a. Set aktivasi unit masukan $X_i = S_i$ ($i = 1, \dots, n$)

b. Hitung respon unit keluaran : $net = \sum x_i w_i + b$

$$y = f(net) = \begin{cases} 1 & \text{jika } net > \theta \\ 0 & \text{jika } -\theta \leq net \leq \theta \\ -1 & \text{jika } net < -\theta \end{cases}$$

Langkah 3 : Perbaiki bobot pola yang mengandung kesalahan ($y \neq t$) menurut persamaan :

$$w_i \text{ (baru)} = w_i \text{ (lama)} + \Delta w \text{ (} i = 1, \dots, n \text{) dengan } \Delta w = \alpha t x_i$$

$$b \text{ (baru)} = b \text{ (lama)} + \Delta b \text{ dengan } \Delta b = \alpha t$$

6. METODE PENELITIAN

6.1. Subjek Penelitian

Pada penelitian ini yang menjadi subjek penelitian adalah pengenalan citra objek sederhana dengan menggunakan jaringan syaraf tiruan *perceptron*. Sistem yang dirancang dibatasi pada :

1. Gambar yang digunakan adalah gambar objek sederhana yang meliputi kubus, kerucut, tabung, prisma, dan limas.
2. Masukan *file* dengan format grafik .bmp dan dapat menampilkan pola objek sederhana pada layar monitor.

6.2. Analisis Kebutuhan

Analisis kebutuhan merupakan analisis yang dibutuhkan dalam membuat aplikasi yang berupa *input*, *output*, fungsi-fungsi atau tahap operasi yang dibutuhkan dan antarmuka yang diinginkan.

Klasifikasi pola pengenalan objek sederhana berbasis jaringan syaraf tiruan dilakukan dengan beberapa variabel masukan antara lain :

- a. Citra pola input : Pola objek yang ingin dikenali.
- b. Citra pola *output* : Pola objek yang digunakan sebagai Pembanding (target)

7. HASIL DAN PEMBAHASAN

7.1. Proses Pengolahan Gambar (*Image Processing*)

Proses ini merupakan proses yang akan dijalankan apabila *user* memilih tombol *Ambil Citra*, pada proses ini gambar objek akan diubah sehingga akan

1 menghasilkan gambar yang hanya bernilai 0 atau 1 saja yaitu untuk mempermudah mengetahui apakah *pixel* tersebut “bernilai” atau tidak. Pengolahan citra untuk menghasilkan citra menjadi citra biner ini menggunakan proses *thresholding*.

7.2. Proses Pengolahan Jaringan Syaraf (*Neural Network*)

1 Pada tahap ini akan membuat sebuah obyek *neural network* yang terdiri dari satu *hidden layer*, dan satu *output layer*. Jumlah *hidden layer* ditentukan berdasarkan hasil percobaan, dimana jumlah *hidden layer* lebih dari 1 akan menyebabkan proses pelatihan tidak menghasilkan bobot yang stabil sehingga proses pelatihan menjadi lambat. Bobot yang dihasilkan oleh jaringan syaraf ini dapat disimpan kedalam *file* atau kemudian dapat dibaca dari *file* yang telah diisi bobot-bobot hasil pelatihan.

Yang dibutuhkan pada analisis sistem dalam pembuatan perangkat lunak meliputi perancangan sistem seperti :

7.3. Data Input Pelatihan

1 Data *input* untuk pelatihan berupa data gambar objek sederhana yang antara lain meliputi : kubus, kerucut, tabung, prisma, dan limas dengan berbagai jenisnya. *File* citra untuk *input* dan *output* yang akan dilatih adalah sama dalam format *.bmp*.

Tabel 1. Contoh *Input* Pelatihan

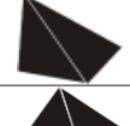
No	Nama <i>file</i> citra	Citra
1	kubus.bmp	
2	kerucut.bmp	
3	limas_segi3_beraturan.bmp	

4	limas_segi3_sembarang.bmp	
5	limas_segi4_beraturan.bmp	

7.4. Pasangan Input Pengujian

Pasangan input untuk pengujian berupa data *sample* citra objek sederhana sebanyak 25 buah yaitu 20 gambar yang sudah dilatih dan 5 gambar yang belum dilatih sebelumnya.

Tabel 2. *Input* pengujian

No	Nama <i>file</i> citra	Citra
1	kubus.bmp	
2	kerucut.bmp	
3	limas_segi3_beraturan.bmp	
4	limas_segi3_sembarang.bmp	
5	limas_segi4_beraturan.bmp	

Hasil pelatihan disajikan pada tabel 3 berikut :

Tabel 3. Hasil pelatihan

No	Nama <i>file</i> gambar yang dilatih		Hasil Pelatihan (nama <i>file</i>)
	Input	Output	
1	kubus.bmp	kubus.bmp	kubus.dat

2	kerucut.bmp	kerucut.bmp	kerucut.dat
3	limas_segi3_beraturan.bmp	limas_segi3_beraturan.bmp	limas_segi3_beraturan.dat
4	limas_segi3_sembarang.bmp	limas_segi3_sembarang.bmp	limas_segi3_sembarang.dat
5	limas_segi4_beraturan.bmp	limas_segi4_beraturan.bmp	limas_segi4_beraturan.dat
6	limas_segi4_sembarang.bmp	limas_segi4_sembarang.bmp	limas_segi4_sembarang.dat
7	limas_segi5_beraturan.bmp	limas_segi5_beraturan.bmp	limas_segi5_beraturan.dat
8	limas_segi5_sembarang.bmp	limas_segi5_sembarang.bmp	limas_segi5_sembarang.dat
9	prisma.bmp	prisma.bmp	prisma.dat
10	prisma_segi3_beraturan.bmp	prisma_segi3_beraturan.bmp	prisma_segi3_beraturan.dat
11	prisma_segi3_miring.bmp	prisma_segi3_miring.bmp	prisma_segi3_miring.dat
12	prisma_segi3_tegak.bmp	prisma_segi3_tegak.bmp	prisma_segi3_tegak.bmp
13	prisma_segi4_miring.bmp	prisma_segi4_miring.bmp	prisma_segi4_miring.dat
14	Prisma_segi4_tegak.bmp	Prisma_segi4_tegak.bmp	Prisma_segi4_tegak.dat
15	prisma_segi5_beraturan.bmp	prisma_segi5_beraturan.bmp	prisma_segi5_beraturan.dat
16	prisma_segi5_miring.bmp	prisma_segi5_miring.bmp	prisma_segi5_miring.dat
17	prisma_segi5_tegak.bmp	prisma_segi5_tegak.bmp	prisma_segi5_tegak.dat
18	prisma_terpancung_1.bmp	prisma_terpancung_1.bmp	prisma_terpancung_1.dat
19	prisma_terpancung_2.bmp	prisma_terpancung_2.bmp	prisma_terpancung_2.dat
20	tabung.bmp	tabung.bmp	tabung.dat

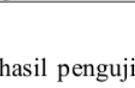
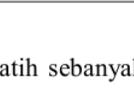
7.5. Pengujian

Proses pengujian atau pengenalan gambar merupakan tahap dimana aplikasi melakukan pengenalan citra sesuai dengan citra yang diinputkan. Pola input yang dimasukkan diolah dan diproses, kemudian dibandingkan dengan pola yang telah dilatih sebelumnya yang hasilnya berupa bobot-bobot koneksi antar sel, sehingga memberikan keluaran akhir berupa gambar yang dikenali, prosentase kebenarannya dan nama *file* citra yang terkenal.

Adapun hasil dari pengujian terhadap citra objek sederhana adalah sebagai mana pada tabel 4 berikut :

Tabel 4. Data hasil pengujian

No	Input		Target		Prosentase	Keterangan
1	kubus.bmp		kubus.bmp		90,24 %	Dikenali

2	limas_segi4_sembarang.bmp		limas_segi4_sembarang.bmp		60,75 %	Dikenali
3	prisma_segi3_miring.bmp		prisma_segi3_miring.bmp		71,5 %	Dikenali
4	prisma_segi3_tegak.bmp		prisma_segi3_tegak.bmp		80,35 %	Dikenali
5	prisma_terpancung_2.bmp		prisma_terpancung_2.bmp		80,56 %	Dikenali
6	tabung.bmp		tabung.bmp		92,65 %	Dikenali
7	kerucut_miring.bmp		kerucut.bmp		-	Tidak dikenali
8	kubus_miring.bmp		kubus.bmp		-	Tidak dikenali
9	limas_miring.bmp		limas_segi4_beraturan.bmp		-	Tidak dikenali

Dari hasil pengujian untuk *file* yang sudah dilatih sebanyak 20 gambar dan *file* yang belum dilatih sebanyak 5 gambar, maka dapat disimpulkan bahwa :

1. Program mampu mengenali citra objek sederhana yang sudah dilatih sebelumnya.
2. Prosentase citra terkenali dihitung dengan rumus *mean* untuk data tunggal yaitu 72,25 % dengan prosentase terendah yaitu 60,75 % dan prosentase tertinggi yaitu 92,65 %.

8. KESIMPULAN

Setelah dilakukan penelitian Pengenalan Citra Objek Sederhana menggunakan Metode *Perceptron*, maka dapat diambil kesimpulan :

1. Jaringan syaraf tiruan metode *Perceptron* dapat digunakan untuk masalah pengenalan citra objek sederhana.
2. Hasil pengenalan yang dilakukan oleh jaringan syaraf tiruan mempunyai prosentase ketepatan yang cukup baik, yaitu rata-ratanya 72,25 % dengan prosentase terendah yaitu 50,75 % dan prosentase tertinggi yaitu 92,65 %.

9. DAFTAR PUSTAKA

- [1] **Bose**, *Neural Network Fundamentals With Graph, Algoritms and Application*, McGraw Hill, New York, 1996.
- [2]. **Dayhoff, J.,E.,1992** *Neural Network Architectures, An Introduction*, Van Nostrad Reinhol, New York.
- [3]. **Fausett, L., 1994** *Fundamentals of Neural Network, Architectures, Algorithm, and Applications*, Preintice-Hall, Inc, New York.
- [4]. **Foster, George, .,1996** *Financial Statement Analysis*, Second Edition, Preintice Hall International Inc.
- [5]. **Haryadi, 2003**. *Pengenalan Pola Alphabet Menggunakan Jaringan Syaraf Tiruan Perseptron*. Skripsi Informatika-S1 UAD, Yogyakarta
- [6]. **Kartalopoulos, stammatis, 1996**, *Understanding Neural Networks And Fuzzy Logic*, IEEE PRESS, New York.
- [7] **Kusumadewi, Sri. 2004**. *Membangun Jaringan Syaraf Tiruan menggunakan Matlab dan Excel Link*. Graha Ilmu, Yogyakarta.
- [8] **Schalkof, Robert J., 1992**, *Pattern Recognition: Statistical, Structural and Neural Approaches*, John Wiley and Sons, Inc Canada.
- [9] **Subiyanto, 2000**. *Aplikasi Jaringan Syaraf Tiruan sebagai Metode Alternatif Prakiraan Beban Jangka Pendek*. Makalah Nomor 29, Tahun VI, Januari 2000. Elektro Indonesia.
- [10] **Wibowo, Catur. 2004**. *Pengenalan Objek Pola 2 Dimensi (Planar) dengan Jaringan Syaraf Tiruan*. Skripsi Informaika-S1 UAD, Yogyakarta.
- [11] www.doc.ic.ac.uk/~nd/surprise_96/journal/vol4/cs11/report.html
- [12] <http://library.gunadarma.ac.id/files/disk1/2>
- [13] www.matematika.id.wikiaz.biz/geometri

HASIL CEK_60030480

ORIGINALITY REPORT

12%

SIMILARITY INDEX

12%

INTERNET SOURCES

1%

PUBLICATIONS

1%

STUDENT PAPERS

PRIMARY SOURCES

1

media.neliti.com

Internet Source

12%

Exclude quotes On

Exclude matches Off

Exclude bibliography Off