



UNIVERSITAS INDONESIA

***PROBABILISTIC NEURAL NETWORK* TEROPTIMASI UNTUK
PENGENALAN WAJAH BERSPEKTRUM GABUNGAN
INFRA MERAH DAN CAHAYA TAMPAK DENGAN
NORMALISASI DAN KOMPENSASI ILUMINASI**

SKRIPSI

**HARRY BIAN PRAMUDIA
0706267761**

**FAKULTAS TEKNIK
PROGRAM STUDI TEKNIK ELEKTRO
DEPOK
JUNI 2011**



UNIVERSITAS INDONESIA

***PROBABILISTIC NEURAL NETWORK* TEROPTIMASI UNTUK
PENGENALAN WAJAH BERSPEKTRUM GABUNGAN
INFRA MERAH DAN CAHAYA TAMPAK DENGAN
NORMALISASI DAN KOMPENSASI ILUMINASI**

SKRIPSI


Diajukan sebagai salah satu syarat untuk memperoleh gelar Sarjana Teknik


**HARRY BIAN PRAMUDIA
0706267761**

**FAKULTAS TEKNIK
PROGRAM STUDI TEKNIK ELEKTRO
DEPOK
JUNI 2011**

HALAMAN PERNYATAAN ORISINALITAS

**Skripsi ini adalah hasil karya saya sendiri,
dan semua sumber baik yang dikutip maupun dirujuk
telah saya nyatakan dengan benar.**



Nama : Harry Bian Pramudia
NPM : 0706267761
Tanda Tangan : 
Tanggal : 15 Juni 2011


HALAMAN PENGESAHAN

Skripsi ini diajukan oleh :

Nama : Harry Bian Pramudia
NPM : 0706267761
Program Studi : Teknik Elektro
Judul Skripsi : *Probabilistic Neural Network* Teroptimasi untuk
Pengenalan Wajah Berspektrum Gabungan Infra Merah dan
Cahaya Tampak dengan Normalisasi dan Kompensasi
Illuminasi

Telah berhasil dipertahankan di hadapan Dewan Penguji dan diterima sebagai bagian persyaratan yang diperlukan untuk memperoleh gelar Sarjana Teknik pada Program Studi Teknik Elektro, Fakultas Teknik, Universitas Indonesia.

DEWAN PENGUJI

Pembimbing :
Prof. Dr.Eng. Drs. Benyamin Kusumoputro M.Eng. ()

Penguji 1 :
Dr. Abdul Halim M.Eng. ()

Penguji 2 :
Ir. Aries Subiantoro M.SEE. ()

Ditetapkan di : Depok

Tanggal : 4 Juli 2011

KATA PENGANTAR

Puji dan syukur ke hadirat Tuhan Yang Maha Esa, karena atas rahmat dan karunia-Nya penulis dapat menyelesaikan Laporan Skripsi ini. Penyusunan Skripsi ini tidak semata-mata dihasilkan dari usaha penulis saja, tetapi juga berkat bantuan, dukungan, dan saran yang diberikan orang-orang lain di sekeliling penulis selama masa penelitian dan penyusunan Skripsi. Oleh karena itu, penulis bermaksud untuk mengucapkan terima kasih kepada :

- 1) Prof. Dr.Eng. Drs. Benyamin Kusumoputro M.Eng., selaku dosen pembimbing yang telah menyediakan waktu, tenaga, dan pikiran untuk mengarahkan penulis dalam penyusunan Skripsi ini;
- 2) Orang tua dan keluarga besar penulis yang telah banyak mendukung dan berjuang demi pendidikan penulis;
- 3) Stephen yang telah membantu penulis dalam mengambil data penelitian dan menjadi tempat penulis bertukar pikiran;
- 4) Dimas dan Alifia selaku teman sebangunan yang selalu memberikan saran yang membangun;
- 5) Ardy, Arif, Dimas, Faisal, Irwansah, Kidut, Novri, Ryan, Rizky, Rudi, Sandi, Stephen, Templang, Victor, Wicak, Wilman, Yasil, Yuddy dan Yudo yang telah bersedia untuk menjadi responden untuk data penelitian;
- 6) Para sahabat Elektro dan anggota Tim Robotika UI yang selalu memberikan dukungan dan semangat;

Penulis menyadari bahwa dalam penulisan karya ilmiah ini tidak terlepas dari kekurangan dan kesalahan. Oleh karena itu Penulis mengharapkan saran dan kritik yang membangun pada Skripsi ini. Semoga karya ilmiah ini dapat bermanfaat.

Depok, Juni 2011

Penulis

**HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI
TUGAS AKHIR UNTUK KEPENTINGAN AKADEMIS**

Sebagai sivitas akademik Universitas Indonesia, saya yang bertanda tangan di bawah ini:

Nama : Harry Bian Pramudia
NPM : 0706267761
Program Studi : Teknik Elektro
Departemen : Teknik Elektro
Fakultas : Teknik
Jenis Karya : Skripsi

Demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada Universitas Indonesia **Hak Bebas Royalti Noneksklusif (*Non-exclusive Royalty- Free Right*)** atas karya ilmiah saya yang berjudul:

***PROBABILISTIC NEURAL NETWORK* TEROPTIMASI UNTUK
PENGENALAN WAJAH BERSPEKTRUM GABUNGAN
INFRA MERAH DAN CAHAYA TAMPAK DENGAN
NORMALISASI DAN KOMPENSASI ILUMINASI**

beserta perangkat yang ada. Dengan Hak Bebas Royalti Noneksklusif ini Universitas Indonesia berhak menyimpan, mengalihmedia/formatkan, mengelola dalam bentuk pangkalan data (*database*), merawat, dan memublikasikan tugas akhir saya selama tetap mencantumkan nama saya sebagai penulis/pencipta dan sebagai pemilik Hak Cipta.

Demikian pernyataan ini saya buat dengan sebenarnya.

Dibuat di : Depok

Pada tanggal : 15 Juni 2011

Yang menyatakan



(Harry Bian Pramudia)

ABSTRAK

Nama : Harry Bian Pramudia
Program Studi : Teknik Elektro
Judul : *Probabilistic Neural Network* Teroptimasi untuk Pengenalan Wajah Berspektrum Gabungan Infra merah dan Cahaya tampak dengan Normalisasi dan Kompensasi Iluminasi

Backpropagation (BP) memiliki performa yang baik dalam mengklasifikasi citra-citra wajah bertingkat iluminasi seragam. Namun untuk citra wajah yang bertingkat iluminasi beragam seperti pada aplikasi kamera pengintai maka BP akan kesulitan dalam mempelajari dan mengenalinya. Skripsi ini menggunakan metode *Probabilistic Neural Network* Teroptimasi (OPNN) sebagai Sistem Pengenal Wajah untuk spektrum gabungan infra merah dan cahaya tampak dengan intensitas yang berubah-ubah. Skripsi ini juga menggunakan metode Normalisasi dan Kompensasi Iluminasi untuk mengurangi dampak variasi iluminasi pada citra. Hasil penelitian menunjukkan bahwa performa OPNN untuk mengenali wajah akan meningkat jika Data *Train* yang digunakan berisi citra dengan tingkat iluminasi yang beragam, dimana Tingkat Rekognisi rata-rata OPNN 18.36% lebih tinggi dari BP.

Kata kunci:

Probabilistic Neural Network Teroptimasi, *Backpropagation*, Normalisasi dan Kompensasi Iluminasi, infra merah, cahaya tampak.

ABSTRACT

Name : Harry Bian Pramudia
Study Program : Electrical Engineering
Title : Optimized Probabilistic Neural Network for Visible Light and Infrared combined Spectrum Face Recognition with Illumination Compensation and Normalization

Backpropagation (BP) has a good performance in classifying face images with uniform illumination level. But Backpropagation have difficulty in learning and recognizing face images with varied illumination level such in surveillance camera. This thesis uses Optimized Probabilistic Neural Network (OPNN) method as Face Recognition System for the joint spectrum of infrared and visible light with varying intensity. This thesis also uses Illumination Normalization and Compensation method to reduce the impact of illumination variance on the image. The research shows that OPNN performance to recognize face will increase if Train Data used contains images with varying levels of illumination, which recognition rate of OPNN is 18.36% higher than BP.

Keywords :

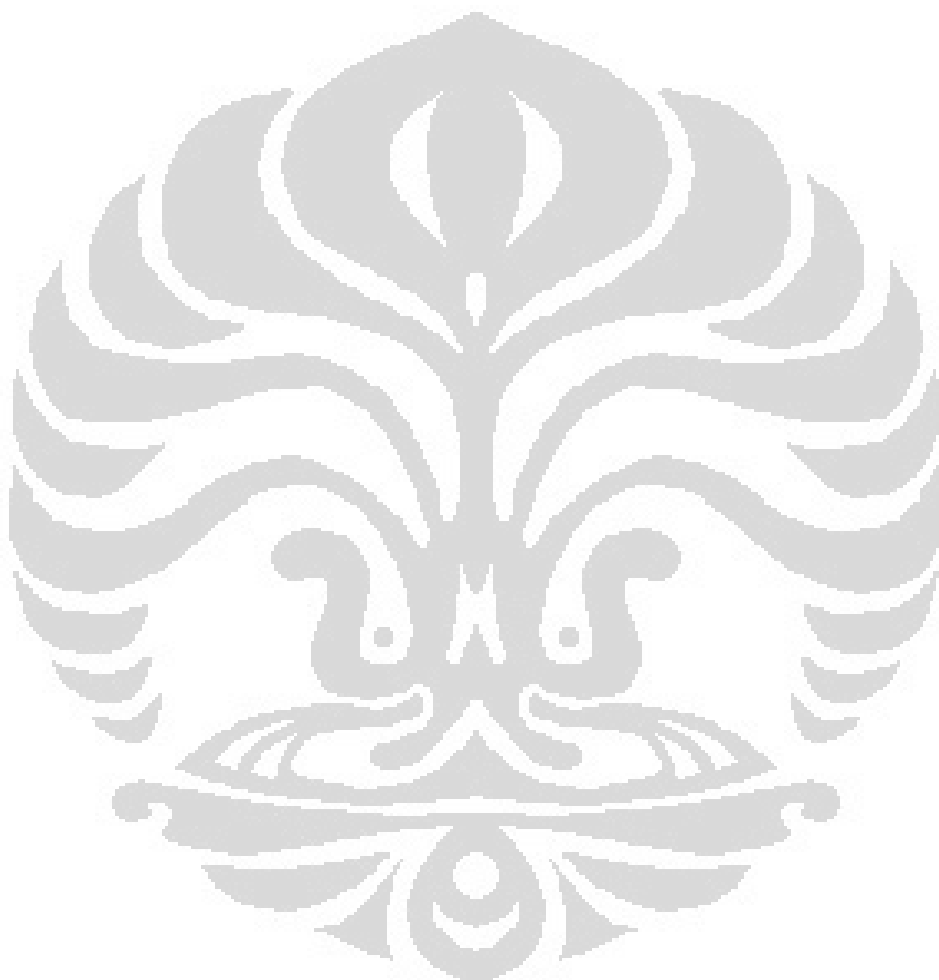
Optimized Probabilistic Neural Network, Backpropagation, Illumination Normalization and Compensation, infrared, visible light.

DAFTAR ISI

HALAMAN PERNYATAAN ORISINALITAS.....	ii
HALAMAN PENGESAHAN.....	iii
KATA PENGANTAR	iv
HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI TUGAS AKHIR UNTUK KEPENTINGAN AKADEMIS	v
ABSTRAK	vi
ABSTRACT.....	vii
DAFTAR ISI.....	viii
DAFTAR GAMBAR	xi
DAFTAR TABEL.....	xiv
BAB 1 PENDAHULUAN	1
1.1. Latar Belakang.....	1
1.2. Perumusan Masalah.....	2
1.3. Tujuan Penelitian.....	2
1.4. Batasan Masalah	2
1.5. Manfaat Penelitian.....	3
1.6. Sistematika Penulisan Laporan.....	3
BAB 2 <i>PROBABILISTIC NEURAL NETWORK</i> TEROPTIMASI, NORMALISASI DAN KOMPENSASI ILUMINASI	5
2.1. <i>Probabilistic Neural Network</i>	5
2.2. <i>Principal Component Analysis</i>	8
2.3. Metode Gram-Schmidt	11
2.4. Algoritma Ortogonal untuk optimasi PNN.....	13
2.5. Normalisasi dan Kompensasi Iluminasi	15
2.5.1. Transformasi Logaritma Citra.....	16
2.5.2. <i>Discrete Cosine Transform</i> (DCT)	17
2.5.3. Normalisasi dan Kompensasi Iluminasi Citra dalam Ruang DCT Logaritma	17
BAB 3 DATA PENELITIAN DAN PENALAAAN PARAMETER.....	19
3.1. Perangkat Pendukung Penelitian	19
3.2. Proses pengambilan Data.....	21
3.3. Penalaan <i>Smoothing Parameter</i> PNN Standar	22
3.3.1. Tujuan Percobaan.....	22

3.3.2.	Skema Percobaan	22
3.3.3.	Hasil dan Analisa Percobaan.....	24
3.4.	Percobaan Optimasi PNN.....	29
3.4.1.	Tujuan Percobaan.....	29
3.4.2.	Skema Percobaan	29
3.4.3.	Hasil dan Analisa Percobaan.....	31
3.5.	Penalaan $\log \mu$ pada Normalisasi Iluminasi	36
3.5.1.	Tujuan Percobaan.....	36
3.5.2.	Skema Percobaan	36
3.5.3.	Hasil dan Analisa Percobaan.....	37
3.6.	Normalisasi Iluminasi Citra berspektrum gabungan Infra merah dan Cahaya tampak	38
BAB 4 OPNN UNTUK PENGENALAN WAJAH BERSPEKTRUM GABUNGAN INFRA MERAH DAN CAHAYA TAMPAK.....		39
4.1.	PNN dengan Data Asli dan Data Ternormalisasi dan Terkompensasi Iluminasi	39
4.1.1.	Tujuan Percobaan.....	39
4.1.2.	Skema Percobaan	40
4.1.3.	Hasil dan Analisa Percobaan dengan Data Asli.....	41
4.1.4.	Hasil dan Analisa Percobaan dengan Data Ternormalisasi Iluminasi.....	44
4.1.5.	Hasil dan Analisa Percobaan dengan Data Ternormalisasi dan Terkompensasi Iluminasi	48
4.2.	Pengurutan Tingkat Kepentingan Set Data	54
4.3.	OPNN menggunakan Data <i>Train</i> bertingkat iluminasi campuran.....	55
4.3.1.	Tujuan Percobaan.....	55
4.3.2.	Skema Percobaan	55
4.3.3.	Hasil dan Analisa Percobaan OPNN.....	58
4.3.4.	Komparasi OPNN dengan PNN Standar dan BP.....	61
4.3.5.	Komparasi Data Asli dengan Data Ternormalisasi dan Terkompensasi Iluminasi	67
BAB 5 PENUTUP		69
5.1.	Kesimpulan.....	69

5.2. Saran	70
DAFTAR PUSTAKA	71
LAMPIRAN A	73
LAMPIRAN B	76



DAFTAR GAMBAR

Gambar 2.1. Arsitektur PNN. sumber: (Mao, 2000).....	5
Gambar 2.2. Kurva PDF pada Klasifikasi Bayesian sumber: (Khanifa, 2010).....	8
Gambar 2.3. Rekonstruksi gambar menjadi matriks kolom.....	9
Gambar 2.4. Proyeksi u_2 pada ortogonal W_1	11
Gambar 2.5. Proyeksi u_3 pada ortogonal W_2	12
Gambar 3.1. Kamera Infra merah	19
Gambar 3.2. 1 buah lampu pijar 100 watt, 2 buah lampu TL 36 watt, 1 buah Dimmer 500 watt.....	20
Gambar 3.3. Lux meter	20
Gambar 3.4. Contoh Set Data 1 hingga 7 dalam format RGB (baris ke-1)	22
Gambar 3.5. Skema Percobaan PNN Standar	23
Gambar 3.6. Skema Percobaan BP	24
Gambar 3.7. Tingkat Rekognisi PNN Standar masing-masing Set Data dengan variasi <i>smoothing parameter</i>	27
Gambar 3.8. Prosedur Optimasi Struktur Neuron PNN.....	30
Gambar 3.9. Struktur Neuron Iterasi ke-2 (optimal) dan ke-15.....	33
Gambar 3.10. Tingkat Rekognisi kelas ke-3 seiring perubahan jumlah neuron ...	33
Gambar 3.11. Tingkat Rekognisi kelas ke-12 seiring perubahan jumlah neuron .	34
Gambar 3.12. Proses Normalisasi Iluminasi.....	36
Gambar 3.13. Skema Percobaan PNN Standar dengan Normalisasi Iluminasi	36
Gambar 3.14. Citra sampel yang mengalami perubahan $\log \mu$ dari 0 hingga 300	37
Gambar 3.15. Tingkat Rekognisi PNN terhadap perubahan $\log \mu$	37
Gambar 4.1. Prosedur Normalisasi dan Kompensasi Iluminasi.....	40
Gambar 4.2. Skema Percobaan PNN dengan tiap macam Data <i>Train</i>	40
Gambar 4.3. Skema Percobaan BP dengan tiap macam Data <i>Train</i>	41
Gambar 4.4. Tingkat Rekognisi Data <i>Test</i> PNN oleh masing-masing Data <i>Train</i>	42
Gambar 4.5. Tingkat Rekognisi Data <i>Test</i> BP oleh masing-masing Data <i>Train</i> ..	43
Gambar 4.6. Tingkat Rekognisi Data <i>Test</i> rata-rata PNN dan BP oleh masing- masing Data <i>Train</i>	44

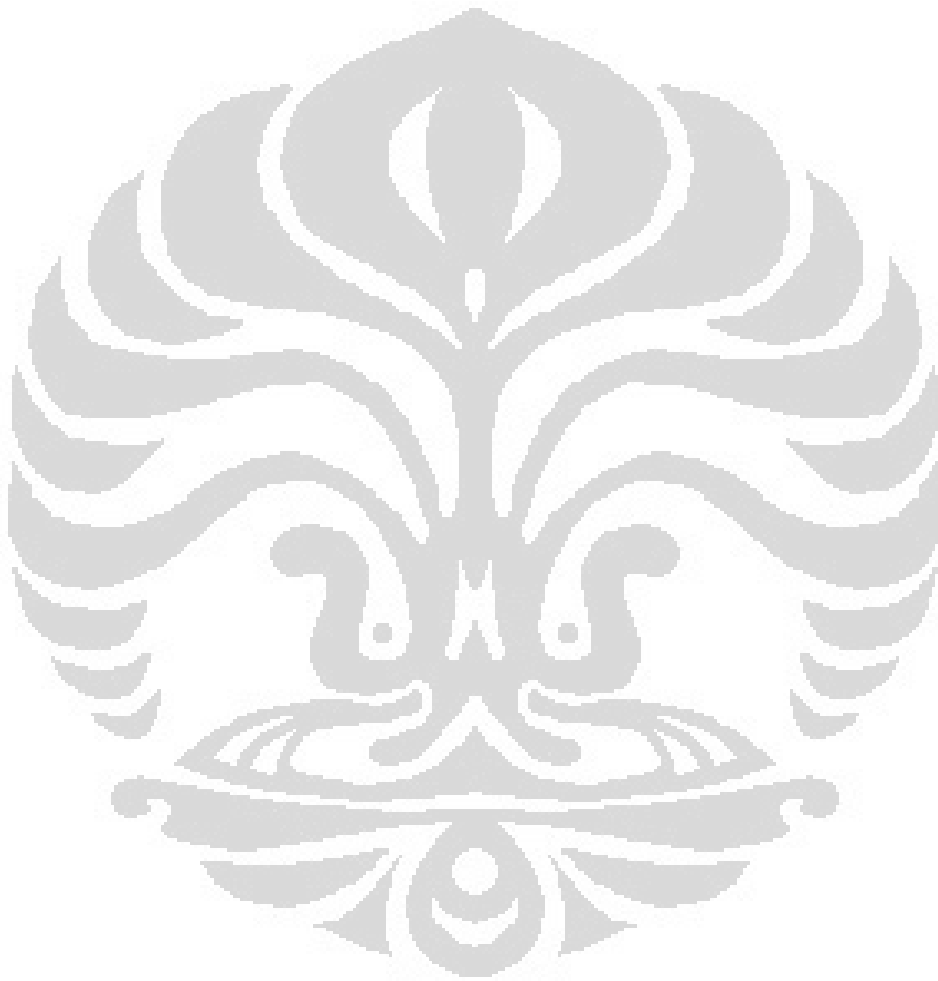
Gambar 4.7. Tingkat Rekognisi Data <i>Test</i> PNN oleh masing-masing Data <i>Train</i> (menggunakan NI).....	45
Gambar 4.8. Tingkat Rekognisi Data <i>Test</i> BP oleh masing-masing Data <i>Train</i> (menggunakan NI).....	46
Gambar 4.9. Tingkat Rekognisi Data <i>Test</i> rata-rata PNN dan BP oleh masing- masing Data <i>Train</i> (menggunakan NI).....	47
Gambar 4.10. Contoh perubahan frekuensi cutoff pada Set Data 0 (baris ke-1), Set Data 3 (baris ke-2) dan Set Data 6 (baris ke-2).....	49
Gambar 4.11. Tingkat Rekognisi Data <i>Test</i> rata-rata PNN oleh masing-masing Data <i>Train</i> (menggunakan NI+KI) berdasarkan frekuensi cutoff yang digunakan	50
Gambar 4.12. Tingkat Rekognisi Data <i>Test</i> PNN oleh masing-masing Data <i>Train</i> (menggunakan NI+KI dengan frekuensi cutoff 2)	51
Gambar 4.13. Tingkat Rekognisi Data <i>Test</i> BP oleh masing-masing Data <i>Train</i> (menggunakan NI+KI dengan frekuensi cutoff 2)	52
Gambar 4.14. Tingkat Rekognisi Data <i>Test</i> rata-rata PNN dan BP oleh masing- masing Data <i>Train</i> (menggunakan NI+KI dengan frekuensi cutoff 2)	53
Gambar 4.15. Prosedur Optimasi PNN	56
Gambar 4.16. Skema Percobaan OPNN	57
Gambar 4.17. Skema Percobaan PNN Standar	57
Gambar 4.18. Skema Percobaan BP	57
Gambar 4.19. Tingkat Rekognisi Data <i>Train</i> dan Data <i>Test</i> rata-rata OPNN oleh masing-masing Data <i>Train</i>	58
Gambar 4.20. Tingkat Rekognisi Data <i>Test</i> OPNN oleh masing-masing Data <i>Train</i>	60
Gambar 4.21. Tingkat Rekognisi Data <i>Train</i> PNN, OPNN dan BP oleh masing- masing Data <i>Train</i>	61
Gambar 4.22. Tingkat Rekognisi Data <i>Test</i> PNN oleh masing-masing Data <i>Train</i>	62
Gambar 4.23. Tingkat Rekognisi Data <i>Test</i> BP oleh masing-masing Data <i>Train</i>	63
Gambar 4.24. Tingkat Rekognisi Data <i>Test</i> rata-rata PNN, OPNN dan BP oleh masing-masing Data <i>Train</i>	64

Gambar 4.25. Waktu *Training* OPNN dan BP oleh masing-masing Data *Train*.. 65

Gambar 4.26. Waktu *Testing* PNN, OPNN dan BP oleh masing-masing Data
Train 66

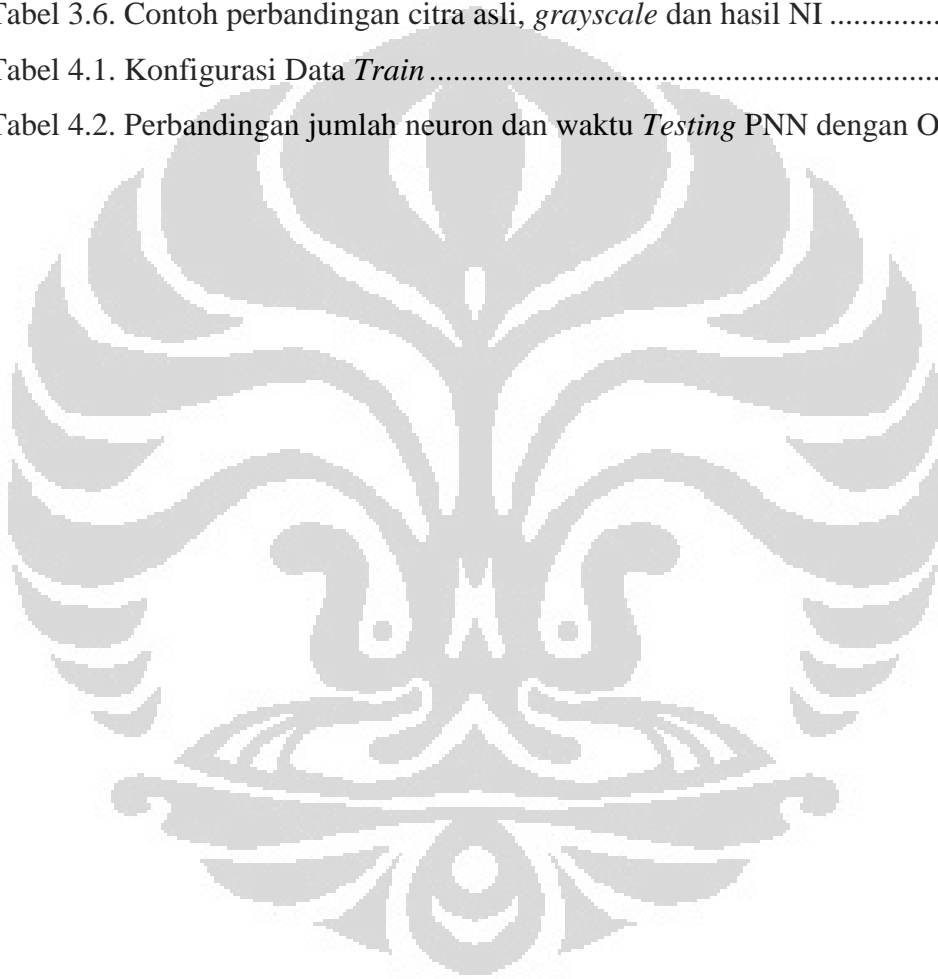
Gambar 4.27. Tingkat Rekognisi Data *Train* dan Data *Test* OPNN oleh masing-
masing Data *Train* (Data Asli tanpa Normalisasi dan Kompensasi Iluminasi) 67

Gambar 4.28. Tingkat Rekognisi Data *Test* rata-rata OPNN untuk Data Asli dan
Data NI+KI oleh masing-masing Data *Train* 68



DAFTAR TABEL

Tabel 3.1. Spesifikasi Kamera Infra merah.....	19
Tabel 3.2. Konfigurasi Intensitas Cahaya tampak tiap Set Data.....	21
Tabel 3.3. Perbandingan performa PNN Standar dengan BP	28
Tabel 3.4. Tingkat Rekognisi OPNN setiap iterasi.....	31
Tabel 3.5. Perbandingan PNN Standar, OPNN dan BP	35
Tabel 3.6. Contoh perbandingan citra asli, <i>grayscale</i> dan hasil NI	38
Tabel 4.1. Konfigurasi Data <i>Train</i>	56
Tabel 4.2. Perbandingan jumlah neuron dan waktu <i>Testing</i> PNN dengan OPNN	66



BAB 1

PENDAHULUAN

1.1. Latar Belakang

Aplikasi Sistem Pengenal Wajah berbasis *Artificial Neural Network* (ANN) telah banyak digunakan dalam kehidupan sehari-hari, mulai dari sistem Login, Kamera Pengintai, sistem keamanan dll. Pada Aplikasi Kamera Pengintai, Sistem Pengenal Wajah umumnya diintegrasikan untuk mengenali identitas pemilik wajah yang terduplik oleh Kamera, biasanya sistem ini dimanfaatkan untuk pendataan orang-orang yang memasuki suatu ruangan. Pada Sistem *Login* yang biasa terintegrasi dalam *Notebook* dan *Auto Door Lock*, Sistem Pengenal Wajah digunakan sebagai pengambil keputusan apakah seseorang yang wajahnya terduplik oleh kamera akan diberikan otoritas atau tidak untuk memasuki suatu ruangan, menggunakan suatu fasilitas, ataupun masuk ke *Desktop* sebuah komputer.

Tak bisa disangkal bahwa pada aplikasinya, Sistem Pengenal Wajah tidak akan selalu berada pada lingkungan berkondisi pencahayaan baik, apalagi jika kamera dipasang di tempat terbuka yang kondisi pencahayaannya berubah-ubah. Pada aplikasi Kamera Pengintai sebagian besar dilengkapi pemancar infra merah sehingga dapat tetap bekerja dengan baik pada lingkungan yang gelap. Umumnya intensitas intensitas infra merah yang dipancarkan bergantung pada kondisi pencahayaan lingkungan tersebut, saat kondisi terang maka pemancar infra merah tidak aktif, saat kondisi mulai gelap maka pemancar infra merah akan mulai aktif, dan semakin gelap maka intensitas infra merah yang dipancarkan akan semakin maksimum. Akibatnya citra yang dicitra oleh kamera akan memiliki intensitas cahaya yang berubah-ubah, mulai dari intensitas cahaya tampak penuh, intensitas transisi, hingga intensitas cahaya infra merah penuh. Perubahan dominasi cahaya pada objek antara cahaya tampak dengan infra merah dapat mengubah kekontrasan *grayscale* citra yang akan berdampak pada Tingkat Rekognisi Sistem Pengenal Wajah.

Performa baik dari *Backpropagation Neural Network* (BP) telah terbukti dalam mengklasifikasikan citra wajah yang berada dalam kondisi pencahayaan yang baik. Namun saat objek wajah mengalami pencahayaan yang berubah-ubah, maka Tingkat Rekognisi BP umumnya akan menurun.

Melalui Tulisan ini akan diajukan sebuah metode *Probabilistic Neural Network* Teroptimasi (OPNN) sebagai Sistem Pengenal Wajah untuk citra dalam spektrum gabungan cahaya tampak dan infra merah. Di samping itu juga akan dicoba metode Normalisasi dan Kompensasi Iluminasi menggunakan *Discrete Cosine Transform* untuk mengurangi dampak perubahan intensitas cahaya pada citra.

1.2. Perumusan Masalah

Masalah yang akan dibahas pada laporan skripsi ini adalah penggunaan *Probabilistic Neural Network* Teroptimasi (OPNN) untuk mengenali citra wajah yang berada pada spektrum gabungan cahaya tampak dan infra merah yang berubah-ubah, dengan bantuan *Discrete Cosine Transform* (DCT) untuk Normalisasi dan Kompensasi Iluminasi citra wajah.

1.3. Tujuan Penelitian

Tujuan utama dari Penelitian ini adalah Pengembangan Sistem Pengenal Wajah yang *robust* terhadap perubahan intensitas cahaya dan dapat digunakan dalam spektrum gabungan cahaya tampak dan infra merah menggunakan metode *Probabilistic Neural Network* Teroptimasi dan dengan bantuan metode Normalisasi dan Kompensasi Iluminasi.

1.4. Batasan Masalah

Berikut adalah batasan masalah dari percobaan yang dilakukan:

1. Citra wajah baik untuk Data *Train* maupun Data *Test* dicuplik dengan kamera yang sama, dengan posisi, orientasi dan jarak yang sama untuk masing-masing Set Data.
2. Citra wajah yang digunakan seluruhnya adalah wajah tampak depan, sudut kamera frontal terhadap horizontal maupun vertikal wajah.

3. Citra wajah yang akan digunakan terbagi dalam 7 kelompok berdasarkan tingkat Intensitas Cahaya tampak yang digunakan untuk menyinari wajah, mulai yang paling redup hingga yang paling terang.
4. Identitas wajah yang akan dipelajari dan dikenali berjumlah 20 kelas.
5. Data yang akan diklasifikasi merupakan Ekstraksi Fitur wajah yang telah diolah oleh *Pincipal Component Analysis*, dengan dimensi akhir sebanyak 30.

1.5. Manfaat Penelitian

Sesuai dengan Tujuan Penelitian, manfaat utama yang diharapkan dari hasil Penelitian ini adalah:

1. Sistem Pengenal Wajah yang lebih handal dan tahan terhadap perubahan intensitas cahaya.
2. Sistem Pengenal Wajah yang dapat diterapkan secara umum pada Kamera Pengintai dalam spektrum gabungan cahaya tampak dan infra merah yang berubah-ubah, tanpa mengetahui kondisi cahaya apa yang lebih dominan.
3. Optimasi PNN yang dapat memperkecil struktur neuron dan mempercepat waktu komputasi.
4. Dapat digunakan sebagai referensi untuk aplikasi Sistem Pengenal Wajah lebih lanjut untuk Kamera Pengintai, khususnya untuk citra wajah yang proses pencuplikannya tidak frontal.
5. Dapat diterapkan dalam sistem keamanan.

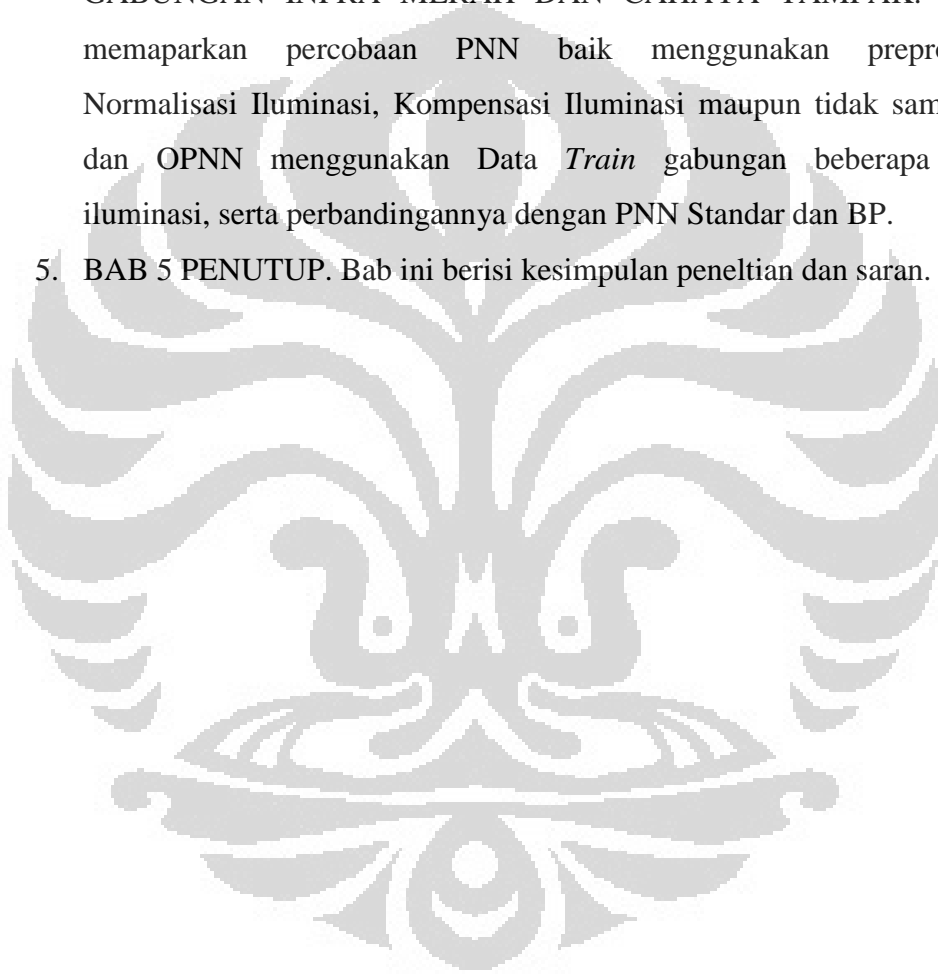
1.6. Sistematika Penulisan Laporan

Laporan ini terdiri dari 5 bab dengan deskripsi tiap bab seperti berikut:

1. BAB 1 PENDAHULUAN. Bab ini menjelaskan latar belakang penelitian, perumusan masalah, tujuan penelitian, manfaat penelitian, dan sistematika penulisan laporan.
2. BAB 2 *PROBABILISTIC NEURAL NETWORK* TEROPTIMASI, NORMALISASI DAN KOMPENSASI ILUMINASI. Bab ini menjelaskan dasar teori PNN dilengkapi dengan teknik optimasinya menggunakan Algoritma Ortogonal. Selain itu, juga akan dibahas dasar teori Normalisasi

dan Kompensasi Iluminasi menggunakan metode *Discrete Cosine Transform*.

3. BAB 3 DATA PENELITIAN DAN PENALAAAN PARAMETER. Bab ini berisi tentang perangkat pendukung penelitian, proses pengambilan data dan penalaan parameter-parameter yang akan digunakan pada percobaan selanjutnya pada Bab 4.
4. BAB 4 OPNN UNTUK PENGENALAN WAJAH BERSPEKTRUM GABUNGAN INFRA MERAH DAN CAHAYA TAMPAK. Bab ini memaparkan percobaan PNN baik menggunakan preprocessing Normalisasi Iluminasi, Kompensasi Iluminasi maupun tidak sama sekali dan OPNN menggunakan Data *Train* gabungan beberapa tingkat iluminasi, serta perbandingannya dengan PNN Standar dan BP.
5. BAB 5 PENUTUP. Bab ini berisi kesimpulan penelitian dan saran.



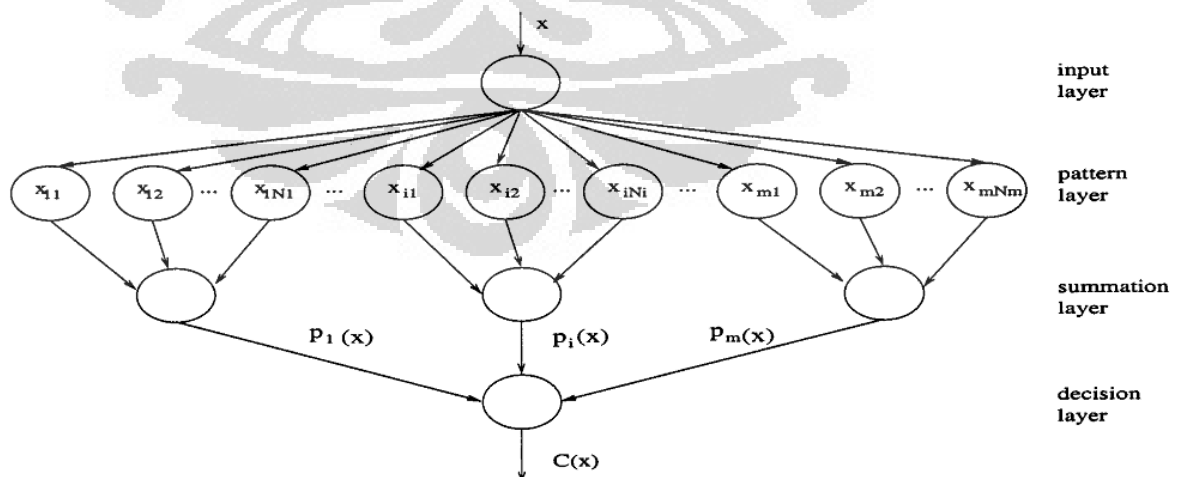
BAB 2

PROBABILISTIC NEURAL NETWORK TEROPTIMASI, NORMALISASI DAN KOMPENSASI ILUMINASI

Bab ini memaparkan tentang teori *Probabilistic Neural Network* (PNN) dan proses optimasi PNN menggunakan Algoritma Ortogonal. Di akhir Bab akan dilakukan pengujian Klasifikasi Citra Wajah menggunakan PNN dan *Probabilistic Neural Network* Teroptimasi (OPNN), serta perbandingan keduanya baik dalam Tingkat Rekognisi, Struktur Neuron dan waktu komputasinya. Setiap percobaan juga akan membandingkan hasilnya dengan metode *Backpropagation* (BP).

2.1. Probabilistic Neural Network

Probabilistic Neural Network (PNN) adalah jenis jaringan saraf tiruan yang dikembangkan oleh Donald Specht pada tahun 1990 (Fausett, 1994) (Masters, 1995) (Mao, 2000). PNN dirancang menggunakan ide dari teori klasik probabilitas, yaitu klasifikasi Bayesian dan Estimator PDF Parzen (Fausett, 1994). PNN telah menjadi salah satu algoritma yang efektif untuk memecahkan banyak permasalahan klasifikasi karena proses pelatihan yang mudah dan adanya fondasi statistik dalam teori estimasi Bayesian (Mao, 2000).



Gambar 2.1. Arsitektur PNN. sumber: (Mao, 2000)

Arsitektur PNN dapat dilihat pada Gambar 2.1., terdiri dari 4 lapisan, yaitu *input layer*, *pattern layer*, *summation layer*, dan *decision layer*. Pada *input layer*, tidak dilakukan operasi apapun dan hanya berperan untuk meneruskan data x ke neuron-neuron dalam *pattern layer*. *Pattern layer* terdiri dari neuron-neuron yang berisi Data *Train* dan dinotasikan sebagai x_{ij} (neuron/data *Train* ke- j dari kelas i), di mana satu neuron merepresentasikan satu data *Train* (Fausett, 1994). Setiap neuron pada lapisan ini akan menghasilkan keluaran $\phi_{ij}(x)$ yang merupakan derajat kecocokan pola x terhadap neuron x_{ij} dan akan diteruskan ke *summation layer*. Keluaran neuron x_{ij} merupakan hasil operasi dari persamaan (2.1) (Mao, 2000) (Tian, 2000).

$$\phi_{ij}(x) = \frac{1}{(2\pi)^{d/2} \sigma^d} \frac{1}{N_i} \exp \left[-\frac{(x - x_{ij})^T (x - x_{ij})}{2\sigma^2} \right] \quad (2.1)$$

di mana :

- x = data *Testing*
- i = indeks kelas
- j = indeks neuron
- x_{ij} = neuron ke- j dari kelas i
- $\phi_{ij}(x)$ = keluaran neuron ke- j dari kelas i
- d = dimensi data *Testing*
- σ = *smoothing parameter*

Pada *summation layer*, dihitung *maximum likelihood* data x diklasifikasikan ke dalam kelas C_i dengan menjumlahkan seluruh keluaran neuron-neuron yang berasal dari kelas yang sama pada *pattern layer* seperti yang dijabarkan dalam persamaan (2.2) (Mao, 2000).

$$p_i(x) = \sum_{j=1}^{N_i} \phi_{ij}(x)$$

$$i = 1, 2, 3, \dots, m \quad (2.2)$$

di mana :

- $p_i(x)$ = nilai *maximum likelihood* data x diklasifikasikan ke kelas C_i
- N_i = jumlah neuron kelas i
- $\phi_{ij}(x)$ = keluaran neuron ke- j dari kelas i

m = jumlah kelas

Pada lapisan terakhir, yaitu *decision layer*, akan ditentukan kepada kelas manakah data x diklasifikasikan. Lapisan ini hanya terdiri dari satu neuron yang menerima masukan dari setiap neuron di *summation layer*. Cara pengambilan keputusan dalam lapisan ini sesuai dengan teori klasifikasi Bayesian, di mana keputusan diambil berdasarkan nilai *maximum likelihood* dari masing-masing kelas. Data x akan diklasifikasikan ke dalam kelas yang mempunyai nilai *maximum likelihood* paling besar.

$$C(x) = \arg \max_{i=1,2,3,\dots,m} \{p_i(x)\} \quad (2.3)$$

di mana :

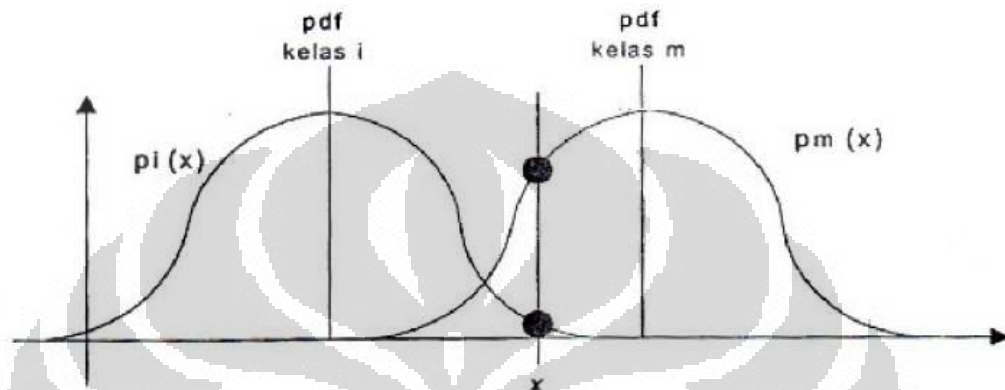
$C(x)$ = kelas estimasi x

Nilai *smoothing parameter* (σ) merupakan parameter yang penting karena menentukan lebar kurva Gaussian yang terbentuk pada setiap titik sampel (Jatmiko, 2000). Pemilihan nilai σ akan mempengaruhi fungsi probabilitas. Nilai *smoothing parameter* yang terlalu kecil akan membuat setiap kasus sampel sangat berpengaruh sehingga dapat menghilangkan manfaat dari informasi kelompok. Jika nilai σ terlalu besar, maka sampel data akan terlalu kabur di mana informasi detil dari kerapatan akan hilang dan akan mengganggu estimasi dari kerapatan. Pemilihan nilai σ yang sesuai sangat diperlukan karena akan mempengaruhi kinerja dari algoritma PNN dalam proses klasifikasi suatu pola. Nilai σ yang sesuai dapat berbeda-beda, bergantung pada data yang digunakan (Mao, 2000).

Sistem klasifikasi yang digunakan dalam penentuan keputusan pada jaringan ini merupakan sistem klasifikasi yang mengikuti konsep Bayesian. Dalam klasifikasi Bayesian, nilai PDF membentuk suatu persebaran data dari suatu kelas. Keputusan dari suatu data dilihat dari probabilitas terbesar dari data tersebut sesuai dengan persebaran nilai PDF-nya. Data tes dimasukkan ke dalam kelas yang memiliki derajat kecocokan PDF yang lebih besar untuk data tes tersebut. Untuk lebih jelasnya, dapat dilihat pada Gambar 2.2.

Pada Gambar 2.2., didefinisikan dua buah kelas yang akan menjadi hasil klasifikasi suatu data, yaitu kelas i dan kelas m . Kedua kelas tersebut memiliki persebaran PDF seperti dapat dilihat pada Gambar 2.2. Selanjutnya, akan diuji

suatu data x untuk kemudian ditentukan apakah data x masuk ke dalam kelas i atau kelas m . Berdasar perhitungan, akan diperoleh derajat kecocokan x terhadap kelas i ($p_i(x)$) dan juga derajat kecocokan x terhadap kelas m ($p_m(x)$). Hasilnya, diperoleh bahwa $p_m(x)$ lebih besar dari $p_i(x)$. Oleh karena itu, data x dimasukkan ke dalam kelas m .




Gambar 2.2. Kurva PDF pada Klasifikasi Bayesian
sumber: (Khanifa, 2010)

2.2. *Principal Component Analysis*

Principal Component Analysis (PCA) adalah suatu metode statistik untuk mentransformasi vektor-vektor ke dalam ruang eigen yang berbasis ortogonal. Ruang berbasis ortogonal adalah ruang yang tidak memiliki korelasi antar dimensinya yang disebut *Principal Components* (PCs). Seperti yang terlihat pada Bab Pendahuluan, setiap data (foto) memiliki dimensi 40×30 atau 1200 dimensi. Dimensi sebesar ini akan memperlambat komputasi PNN dan Jaringan Syaraf Tiruan lainnya. Penelitian ini menggunakan PCA sebagai alat untuk mereduksi dimensi data. Dalam ruang eigen kita dapat memotong dimensi berdasarkan nilai eigen dari tiap basis penyusun ruang eigen tersebut atau yang disebut vektor eigen. Dengan PCA maka data baru akan memiliki dimensi lebih kecil dengan tetap meminimalkan pengurangan variansi nilai antar data, sehingga informasi yang menjadi ciri khas tiap data hampir tidak mengalami perubahan yang signifikan. Oleh karena itu PCA sering disebut sebagai proses Ekstraksi Fitur.

1. *Menyusun matriks data*

Setiap foto berdimensi 40x30 disusun menjadi vektor kolom berdimensi 1200. Lalu vektor kolom semua data yang ada digabung menjadi 1 matriks.



$$= \begin{bmatrix} x_{11} & \cdots & x_{1,30} \\ \vdots & \ddots & \vdots \\ x_{40,1} & \cdots & x_{40,30} \end{bmatrix} = \begin{bmatrix} x_{1,1} \\ \vdots \\ x_{40,1} \\ \vdots \\ x_{1,30} \\ \vdots \\ x_{40,30} \end{bmatrix}$$

Gambar 2.3. Rekonstruksi gambar menjadi matriks kolom

$$database = \begin{bmatrix} x_{1,1} & x_{2,1} & \cdots & x_{t,1} \\ \vdots & \vdots & \ddots & \vdots \\ x_{1,1200} & x_{2,1200} & \cdots & x_{t,1200} \end{bmatrix} \quad (2.4)$$

2. Normalisasi z-score

Normalisasi z-score bertujuan untuk meletakkan nilai rata-rata tiap dimensi di titik nol dan mengatur agar tiap dimensi memiliki nilai variansi yang seragam. Normalisasi z-score dapat dilakukan dengan pengurangan tiap dimensi data dengan rata-rata dimensi seluruh data dan membaginya dengan standar deviasi dimensi seluruh data. Tiap dimensi dilakukan perhitungan z-score yang terpisah.

$$\mu = \frac{\sum_{i=1}^n X_i}{n} \quad \sigma = \sqrt{\frac{\sum_{i=1}^n (X_i - \mu)^2}{n-1}} \quad z = \frac{X - \mu}{\sigma} \quad (2.5)$$

3. Menghitung matriks kovarians

$$C_z = \frac{Z^T Z}{(n-1)} \quad (2.6)$$

4. Menghitung vektor eigen dan nilai eigen dari matriks kovarians

Jika C adalah suatu matriks persegi $n \times n$, yang dalam hal ini adalah matriks kovarians, maka vektor tak-nol \mathbf{x} pada suatu ruang vektor R^n disebut sebagai suatu vektor eigen dari C jika $C\mathbf{x}$ adalah suatu penggandaan vektor dari \mathbf{x} yaitu:

$$C\mathbf{x} = \lambda\mathbf{x} \text{ atau } (\lambda I - C)\mathbf{x} = \mathbf{0} \quad (2.7)$$

λ adalah nilai eigen dari C , dan \mathbf{x} adalah vektor eigen dari C yang berpadanan dengan λ . Semua Nilai eigen dan vektor eigen merupakan kumpulan solusi dari persamaan tersebut.

5. Bentuk matriks transformasi

Nilai-nilai eigen memiliki nilai yang bervariasi. Vektor eigen yang memiliki nilai eigen terbesar merupakan *Principal Component* pertama dari suatu set data. Vektor eigen (*Principal Component, PC*) akan diurutkan dari yang memiliki nilai eigen terbesar hingga terkecil. *PC* yang memiliki nilai eigen kecil dapat dibuang. Hal ini memang akan menyebabkan kurangnya informasi data, namun *PC* yang nilai eigennya kecil tidak berpengaruh besar sehingga dapat diabaikan. Dengan demikian, set data akhir akan memiliki dimensi yang lebih kecil daripada set data awal. Jika set data awal memiliki n dimensi dan kita hanya memilih p vektor eigen terbesar, maka set data akhir pun hanya akan memiliki p dimensi.

Matriks transformasi (M) dibentuk dari vektor-vektor eigen yang dipilih berdasarkan besarnya nilai eigennya.

$$M = (Veig_1 \quad Veig_2 \quad \dots \quad Veig_n)^T \quad (2.8)$$

6. Menurunkan Matriks Data Baru

Vektor eigen akan diurutkan dari yang bernilai eigen terbesar hingga terkecil. Matriks transformasi akan dibentuk dari kumpulan vektor eigen yang memiliki nilai eigen terbesar. Nilai eigen sebanding dengan tingkat kepentingan basis tersebut sehingga basis dengan nilai eigen yang kecil dapat dihilangkan karena tidak akan berpengaruh banyak pada informasi penting dalam data. Pada akhirnya matriks data akhir didapat dengan mengalikan matriks data hasil normalisasi z-score dengan matriks transformasi.

$$MatriksDataBaru = M \times Z \quad (2.9)$$

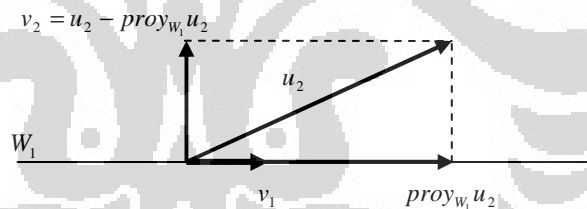
Dimana M adalah matriks transformasi dan Z adalah matriks data hasil normalisasi z-score. *MatriksDataBaru* adalah set data terakhir atau database terbaru yang telah berhasil dimampatkan dimensinya.

2.3. Metode Gram-Schmidt

Proses Gram-Schmidt adalah proses untuk mengubah sembarang basis menjadi basis ortogonal (Anton, 2000). Basis-basis ortogonal dapat diubah menjadi basis-basis ortonormal dengan membagi masing-masing basis ortogonal dengan norma basis tersebut. Bila V adalah sembarang ruang hasil kali dalam tak-nol berdimensi n dan $U = \{u_1, u_2, \dots, u_n\}$ adalah sembarang basis untuk V , maka langkah untuk menghasilkan suatu basis ortogonal $\{v_1, v_2, \dots, v_n\}$ untuk V adalah sebagai berikut (Anton, 2000):

1. Anggap $v_1 = u_1$.
2. Suatu vektor v_2 yang ortogonal terhadap v_1 diperoleh dengan menghitung komponen u_2 yang ortogonal terhadap ruang W_1 yang terentang oleh v_1 .

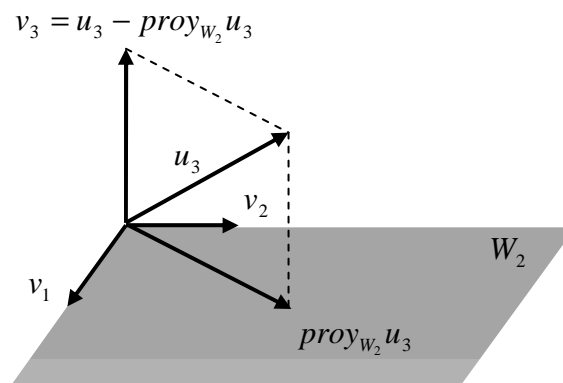
$$\begin{aligned} v_2 &= u_2 - \text{proy}_{W_1} u_2 \\ &= u_2 - \frac{\langle u_2, v_1 \rangle}{\|v_1\|^2} v_1 \end{aligned} \quad (2.10)$$



Gambar 2.4. Proyeksi u_2 pada ortogonal W_1

3. Suatu vektor v_3 yang ortogonal terhadap v_1 dan v_2 diperoleh dengan menghitung komponen u_3 yang ortogonal terhadap ruang W_2 yang terentang oleh v_1 dan v_2 .

$$\begin{aligned} v_3 &= u_3 - \text{proy}_{W_2} u_3 \\ &= u_3 - \frac{\langle u_3, v_1 \rangle}{\|v_1\|^2} v_1 - \frac{\langle u_3, v_2 \rangle}{\|v_2\|^2} v_2 \end{aligned} \quad (2.11)$$



Gambar 2.5. Proyeksi u_3 pada ortogonal W_2

4. Suatu vektor v_4 yang ortogonal terhadap v_1 , v_2 , dan v_3 dapat diperoleh dengan menghitung komponen u_4 yang ortogonal terhadap ruang W_3 yang terentang oleh v_1 , v_2 , dan v_3 .

$$\begin{aligned}
 v_4 &= u_4 - \text{proy}_{W_3} u_4 \\
 &= u_4 - \frac{\langle u_4, v_1 \rangle}{\|v_1\|^2} v_1 - \frac{\langle u_4, v_2 \rangle}{\|v_2\|^2} v_2 - \frac{\langle u_4, v_3 \rangle}{\|v_3\|^2} v_3
 \end{aligned} \tag{2.12}$$

...

...

...

Langkah ini diteruskan hingga sebanyak n langkah sehingga didapatkan himpunan vektor-vektor ortogonal $\{v_1, v_2, \dots, v_n\}$.

Keterangan : $\langle a, b \rangle$ = hasil kali dalam (*inner product*) vektor a dan b

Dengan proses Gram-Schmidt dapat pula dihitung vektor-vektor ortonormal dengan membagi vektor-vektor ortogonal yang didapat dengan norma masing-masing vektor.

$$q_i = \frac{v_i}{\|v_i\|} \tag{2.13}$$

2.4. Algoritma Ortogonal untuk optimasi PNN

Salah satu cara untuk optimasi PNN adalah dengan cara mereduksi struktur neuronnya dengan bantuan Algoritma Ortogonal. Algoritma Ortogonal merupakan metode untuk mengurutkan neuron-neuron setiap kelas berdasarkan tingkat kepentingannya, dimana Tingkat Kepentingan neuron merupakan tolak ukur seberapa representatif neuron tersebut untuk mewakili kelasnya. Dengan kata lain, neuron dengan tingkat kepentingan tertinggi dapat dianggap sebagai neuron yang paling mewakili kelasnya. Algoritma Ortogonal merupakan bagian dari proses *Training* dari PNN karena proses ini bertujuan untuk mendapatkan struktur paling optimal untuk PNN.

Algoritma Ortogonal dapat diterapkan dengan langkah-langkah sebagai berikut (Mao, 2000):

1. Anggap nilai σ yang paling optimal telah ditentukan.
2. Untuk mengukur kedekatan Neuron ke- j dalam kelas ke- i dengan neuron-neuron di dekatnya maka perlu dihitung nilai derajat kecocokan neuron tersebut terhadap seluruh neuron dalam kelas ke- i . Oleh karena itu seluruh Data *Train* dijadikan neuron pada Pattern Layer sekaligus sebagai Data yang diujikan ke PNN untuk mendapatkan nilai derajat kecocokan seluruh Data *Train* terhadap seluruh neuron, yang tergabung dalam matriks Φ_i , dimana jumlah neuron sama dengan jumlah Data *Training* yaitu sebanyak N_i .

$$\Phi_i = \begin{bmatrix} \phi_{i1}(x_{i1}) & \phi_{i2}(x_{i1}) & \cdots & \phi_{iN_i}(x_{i1}) \\ \phi_{i1}(x_{i2}) & \phi_{i2}(x_{i2}) & \cdots & \phi_{iN_i}(x_{i2}) \\ \vdots & \vdots & \ddots & \vdots \\ \phi_{i1}(x_{iN_i}) & \phi_{i2}(x_{iN_i}) & \cdots & \phi_{iN_i}(x_{iN_i}) \end{bmatrix} \quad (2.14)$$

Dimana $\phi_{ij}(x_{ik})$ adalah derajat kecocokan Data *Train* ke- k (x_{ik}) pada neuron ke- j (ϕ_{ij}).

3. Transformasi Ortogonal Linear atau Proses Gram-Schmidt dapat diterapkan untuk mendekomposisi matriks Φ_i menjadi persamaan linier dengan basis yang ortogonal Q .

$$\Phi = QR = [Q_1, Q_2, \dots, Q_{N_i}]R \quad (2.15)$$

Dekomposisi QR untuk matriks Φ_i dilakukan untuk setiap kelas dengan langkah sebagai berikut:

a. Mencari nilai Q_1

- i. Q_1 merupakan basis ortogonal pertama yang akan didapatkan dari neuron terpenting. Q_1 akan didapat dari neuron yang paling mewakili kelas atau dengan kata lain neuron yang paling dekat dengan neuron-neuron lain. Seluruh neuron dapat dijadikan kandidat Q_1 dengan notasi $Q_1^{(\alpha)}$, yang nilainya adalah sama dengan $\phi_{(\alpha)}$.

$$\begin{aligned} Q_1^{(\alpha)} &= \phi_{(\alpha)} \\ \alpha &= 1, 2, \dots, N_i \end{aligned} \quad (2.16)$$

- ii. Tingkat kepentingan masing-masing $Q_1^{(\alpha)}$ dapat dihitung dengan rumus berikut.

$$\Gamma_1^{(\alpha)} = [Q_1^{(\alpha)}]^T Q_1^{(\alpha)} \quad (2.17)$$

Semakin besar nilai Γ berarti semakin dekat neuron-neuron lain dengan neuron kandidat tersebut.

- iii. Q_1 dipilih dari kandidat yang memiliki nilai $\Gamma_1^{(\alpha)}$ terbesar.
- iv. Neuron terpenting Pertama adalah yang bersesuaian dengan $\Gamma_1^{(\alpha)}$ terbesar.

b. Mencari nilai Q_j

- i. Q_j merupakan basis ortogonal ke-j yang akan didapat dari neuron terpenting ke-j.
- ii. Seluruh neuron yang tersisa dapat dijadikan sebagai kandidat untuk mencari Q_j yang dinotasikan sebagai $Q_j^{(\alpha)}$, yang nilainya merupakan proyeksi ortogonal ϕ_{k_α} terhadap vektor yang tegak lurus terhadap seluruh basis ortogonal yang telah didapat sebelumnya (Q_1 sampai Q_{j-1}).

$$\begin{aligned} Q_j^{(\alpha)} &= \phi_{k_\alpha} - \sum_{l=1}^{j-1} r_{l\alpha}^{(\alpha)} Q_l \\ \alpha &= 1, 2, \dots, N_i - j + 1 \end{aligned}$$

$$r_{l\alpha}^{(\alpha)} = \frac{Q_l^T \phi_{k_\alpha}}{Q_l^T Q_l}$$

$$\alpha = 1, 2, \dots, N_i - j + 1$$

$$l < j$$
(2.18)

Dimana ϕ_{k_α} adalah vektor psi yang dijadikan kandidat

iii. Persamaan (2.18) dapat dijabarkan menjadi berikut:

$$Q_2^{(\alpha)} = \phi_{k_\alpha} - \frac{Q_1^T \cdot \phi_{k_\alpha}}{Q_1^T \cdot Q_1} Q_1$$

$$Q_3^{(\alpha)} = \phi_{k_\alpha} - \frac{Q_1^T \cdot \phi_{k_\alpha}}{Q_1^T \cdot Q_1} Q_1 - \frac{Q_2^T \cdot \phi_{k_\alpha}}{Q_2^T \cdot Q_2} Q_2$$

$$Q_4^{(\alpha)} = \phi_{k_\alpha} - \frac{Q_1^T \cdot \phi_{k_\alpha}}{Q_1^T \cdot Q_1} Q_1 - \frac{Q_2^T \cdot \phi_{k_\alpha}}{Q_2^T \cdot Q_2} Q_2 - \frac{Q_3^T \cdot \phi_{k_\alpha}}{Q_3^T \cdot Q_3} Q_3$$
(2.19)

iv. Tingkat kepentingan masing-masing $Q_j^{(\alpha)}$ dapat dihitung dengan rumus yang sama seperti pada pencarian Q_1 .

$$\Gamma_j^{(\alpha)} = [Q_j^{(\alpha)}]^T Q_j^{(\alpha)}$$
(2.20)

v. Q_j sebenarnya dipilih dari kandidat $Q_j^{(\alpha)}$ yang memiliki nilai $\Gamma_j^{(\alpha)}$ terbesar.

vi. Neuron terpenting ke- j adalah yang bersesuaian dengan nilai $\Gamma_j^{(\alpha)}$ terbesar.

2.5. Normalisasi dan Kompensasi Iluminasi

Setiap Citra 2 dimensi dapat dinyatakan oleh matriks $f(x,y)$, di mana x dan y merupakan indeks pixel-pixel secara vertikal dan horizontal, merupakan hasil perkalian antara reflektansi citra $r(x,y)$ dengan iluminasi citra $e(x,y)$ seperti yang terlihat pada persamaan (2.21) (Chen, Er, & Wu, 2006).

$$f(x, y) = r(x, y).e(x, y)$$
(2.21)

Pada citra wajah, reflektansi merepresentasikan pola wajah dan terletak pada frekuensi tinggi dalam domain frekuensi. Reflektansi suatu citra hampir tidak pernah berubah sepanjang bentuk fisik objek dalam citra tidak berubah. Sedangkan, iluminasi merepresentasikan tingkat pencahayaan pada setiap pixel

citra. Tingkat iluminasi pada citra mengalami perubahan yang perlahan sepanjang bidang citra, oleh karena itu variasi iluminasi terletak pada frekuensi rendah dalam domain frekuensi citra (Chen, Er, & Wu, 2006). Nilai iluminasi suatu citra pada setiap pixel bervariasi bergantung pada kondisi pencahayaan seperti tingkat keterangan, sudut datang cahaya ataupun bayangan benda yang menghalanginya. Untuk mendapatkan citra yang baik untuk tujuan sistem pengenalan wajah, kita perlu mendapatkan nilai murni reflektansi citra dengan iluminasi yang seragam pada setiap pixel. Proses penghapusan variasi iluminasi pada suatu citra sehingga iluminasi seluruh pikselnya seragam disebut dengan Kompensasi Iluminasi (KI). Kompensasi Iluminasi dalam domain spasial dapat dilakukan dengan cara menyamakan variabel $e(x,y)$ untuk setiap pixel. Hal ini sulit dilakukan karena kita harus mengetahui nilai $e(x,y)$ (setiap pixel) untuk membuatnya seragam.

2.5.1. Transformasi Logaritma Citra

Dalam domain Logaritma persamaan (2.21) bertransformasi menjadi persamaan (2.22).

$$\log f(x, y) = \log r(x, y) + \log e(x, y) \quad (2.22)$$

Jika dalam domain Spasial KI dilakukan dengan operasi pembagian maka dalam domain Logaritma Kompensasi dilakukan dengan operasi pengurangan $e(x,y)$ dengan kompensator $\epsilon(x,y)$. Persamaan (2.23) adalah proses kompensasi dalam persamaan matematikanya.

$$\begin{aligned} \log f'(x, y) &= \log r(x, y) + \log e'(x, y) \\ \log f'(x, y) &= \log r(x, y) + \log e'(x, y) - \epsilon(x, y) \\ \log f'(x, y) &= \log f(x, y) - \epsilon(x, y) \end{aligned} \quad (2.23)$$

dimana

$\log f'(x, y)$ = Nilai total piksel citra wajah dalam ruang logaritma

$\log r(x, y)$ = Nilai reflektansi piksel citra wajah dalam ruang logaritma

$\log e(x, y)$ = Nilai iluminasi piksel citra wajah yang bervariasi dalam ruang logaritma

$\log e'(x, y)$ = Nilai iluminasi piksel citra wajah yang seragam dalam ruang logaritma

$\in (x, y)$ = Kompensator

Karena citra akan diproses lebih lanjut dalam Ruang Logaritma maka citra harus mengalami sedikit penyesuaian saat dilakukan Transformasi Logaritma seperti pada persamaan (2.24), sehingga citra dalam ruang Logaritma tetap berada pada *range* nilai 0-255 yaitu nilai kuantisasi citra digital 8 bit.

$$F(x, y) = \frac{255}{\log(256)} \log(f(x, y) + 1) \quad (2.24)$$

2.5.2. Discrete Cosine Transform (DCT)

Discrete Cosine Transform (DCT) merupakan transformasi dari domain spasial ke domain frekuensi. DCT memiliki 4 tipe dan yang akan digunakan dalam KI berikut adalah DCT tipe 2. Transformasi DCT 2D dari gambar $f(x,y)$ adalah seperti persamaan (2.25) dan inversnya adalah seperti persamaan (2.26).

$$C(u, v) = \alpha(u)\alpha(v) \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x, y) \times \cos\left[\frac{\pi(2x+1)u}{2M}\right] \cos\left[\frac{\pi(2y+1)v}{2N}\right] \quad (2.25)$$

$$f(x, y) = \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} \alpha(u)\alpha(v)C(u, v) \times \cos\left[\frac{\pi(2x+1)u}{2M}\right] \cos\left[\frac{\pi(2y+1)v}{2N}\right] \quad (2.26)$$

dimana

$$\alpha(u) = \begin{cases} \frac{1}{\sqrt{M}}, & u = 0 \\ \sqrt{\frac{2}{M}}, & u = 1, 2, \dots, M-1 \end{cases} \quad \alpha(v) = \begin{cases} \frac{1}{\sqrt{N}}, & v = 0 \\ \sqrt{\frac{2}{N}}, & v = 1, 2, \dots, N-1 \end{cases}$$

2.5.3. Normalisasi dan Kompensasi Iluminasi Citra dalam Ruang DCT Logaritma

Tingkat Iluminasi pada citra mengalami perubahan yang perlahan sepanjang bidang citra, oleh karena itu variasi iluminasi terletak pada frekuensi rendah dalam domain frekuensi citra (Chen, Er, & Wu, 2006). Kompensasi Iluminasi dalam domain DCT dapat dilakukan dengan menghilangkan koefisien frekuensi rendah dalam matriks $C(u,v)$ yaitu matriks koefisien DCT citra. Penghilangan sebanyak n frekuensi rendah serupa dengan operasi pengurangan dalam persamaan (2.27).

$$\begin{aligned}
 F'(x, y) &= \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} E(u, v) - \sum_{i=1}^n E(u_i, v_i) \\
 &= F(x, y) - \sum_{i=1}^n E(u_i, v_i)
 \end{aligned} \tag{2.27}$$

dimana

$$E(u, v) = \alpha(u)\alpha(v)C(u, v) \cos\left[\frac{\pi(2x+1)u}{2M}\right] \cos\left[\frac{\pi(2y+1)v}{2N}\right] \tag{2.28}$$

Jika variasi iluminasi berada pada komponen frekuensi rendah dan $F'(x, y)$ disetarakan dengan nilai total piksel citra wajah dalam Ruang Logaritma seperti pada persamaan (2.23), maka $\sum_{i=1}^n E(u_i, v_i)$ dapat diserupakan sebagai Kompensator, sehingga penghilangan koefisien DCT frekuensi rendah akan sama dengan Proses Kompensasi variasi iluminasi citra. Banyaknya frekuensi rendah yang akan dihilangkan disebut dengan Frekuensi Cutoff.

Koefisien frekuensi nol pada matriks DCT $C(0,0)$ disebut komponen DC yang menyatakan tingkat pencahayaan pada seluruh permukaan gambar. Proses KI menghapus komponen frekuensi rendah citra namun tidak menghapus komponen frekuensi nol ini (komponen DC) karena penghapusan komponen DC membuat citra hitam total atau seluruh nilai pikselnya nol. Nilai Komponen DC terdekomposisi menjadi persamaan (2.29).

$$C(0,0) = \log \mu \cdot \sqrt{MN} \tag{2.29}$$

dimana M dan N adalah dimensi citra. Setiap citra berdimensi berapapun akan memiliki tingkat pencahayaan yang sama jika $\log \mu$ setiap citra bernilai sama. Proses penyetaraan nilai komponen DC untuk semua gambar disebut dengan proses Normalisasi Iluminasi (NI).

BAB 3

DATA PENELITIAN DAN PENALAAAN PARAMETER

3.1. Perangkat Pendukung Penelitian

Penelitian ini menggunakan perangkat berikut dalam pengambilan data ataupun pemrosesan data:

- a. Kamera Infra merah



Gambar 3.1. Kamera Infra merah

Tabel 3.1. Spesifikasi Kamera Infra merah

Vendor	LYD
Sensor	CMOS
LED	Near Infra Red
Jumlah Led	12 buah
Intensitas min	0 Lux
Intensitas max	145 Lux
Led Switching	Auto

Kamera Infra merah digunakan untuk mengambil foto wajah dengan alasan bahwa kamera ini dapat dipakai dalam kondisi cahaya apapun dan dapat menerima Cahaya tampak maupun Cahaya Infra merah. Kamera ini memiliki *Light Dependent Resistor* (LDR) sebagai referensi untuk mengatur Intensitas Pemancar Infra merah yang dikeluarkan. Intensitas Infra merah yang dipancarkan berbanding terbalik dengan Intensitas Cahaya tampak lingkungan sekitarnya.

b. Lampu dan Dimmer

Lampu digunakan untuk menerangi wajah saat pengambilan data dan Dimmer digunakan untuk mengatur intensitas cahaya tampak dari lampu sebagai variasi data wajah yang akan dicuplik.



Gambar 3.2. 1 buah lampu pijar 100 watt,
2 buah lampu TL 36 watt, 1 buah Dimmer 500 watt

c. Lux meter

Lux meter digunakan untuk mengukur besar intensitas cahaya yang diterima wajah.



Gambar 3.3. Lux meter

d. MATLAB R2009a

Penelitian dilakukan menggunakan m-file pada MATLAB R2009a untuk menyusun program Sistem Pengenal Wajah PNN maupun *Backpropagation*. Sedangkan komputer yang digunakan berbasis Intel Core2Duo 2.55 GHz.

3.2. Proses pengambilan Data

Pengambilan data dilakukan dalam ruangan yang hampir gelap total sehingga pencahayaan objek hampir seluruhnya bergantung pada Lampu Pijar atau Lampu TL yang digunakan, serta latarnya berwarna hitam polos. Metode pengambilan data yaitu dengan merekam video yang dicuplik oleh Kamera Infra merah dalam format kompresi MPEG 2.

Jumlah objek yang akan diambil datanya adalah 20 orang. Setiap orang diminta untuk berekspresi di depan kamera selama video direkam dan ekspresi berubah-ubah seperti senyum, sedih, membuka mulut serta variasi gerakan mata dan alis. Semua variasi ekspresi tersebut dianggap cukup mewakili ekspresi normal manusia. Setiap orang akan diambil 7 macam set data dengan perbedaan Tingkat Iluminasi. Set Data 1 hanya menggunakan penerangan LED Infra merah, sedangkan Set Data 2 sampai dengan 6 menggunakan lampu pijar sebagai penerangan dengan variasi persentase dimmer. Kamera Infra merah dan lampu pijar berjarak ± 1 meter dari wajah dan dengan sudut tegak lurus terhadap wajah. Set Data 1-6 dianggap sebagai kondisi ideal karena arah datang cahaya tegak lurus. Sedangkan Set Data 7 menggunakan lampu TL dari jarak ± 3 meter dengan arah pencahayaan dari atas kepala, kondisi ini dianggap sebagai kondisi non-ideal atau penerangan yang sesuai dengan kondisi sebenarnya pada aplikasi Kamera Pengintai.

Tabel 3.2. Konfigurasi Intensitas Cahaya tampak tiap Set Data

Set Data	Persentase Dimmer (%)	Lampu TL	Intensitas terukur (Lux)
1	0	-	3.6
2	20	-	4.2
3	40	-	7.2
4	60	-	21.7
5	80	-	54.9
6	100	-	86.1
7	0	aktif	95.4

Dari ketujuh video yang terekam dari masing-masing orang akan dicuplik foto berformat *Bitmap* (*.bmp) sebanyak 30 foto untuk masing-masing video. Foto dilakukan *preprocessing* secara manual, yaitu bagian wajah dari tiap foto dipotong dengan rasio 4:3, lalu memampatkan resolusinya menjadi 40x30 piksel dan mengubahnya menjadi format *grayscale*.



Gambar 3.4. Contoh Set Data 1 hingga 7 dalam format RGB (baris ke-1) dan grayscale (baris ke-2)

3.3. Penalaan *Smoothing Parameter* PNN Standar

3.3.1. Tujuan Percobaan

Percobaan ini bertujuan sebagai berikut:

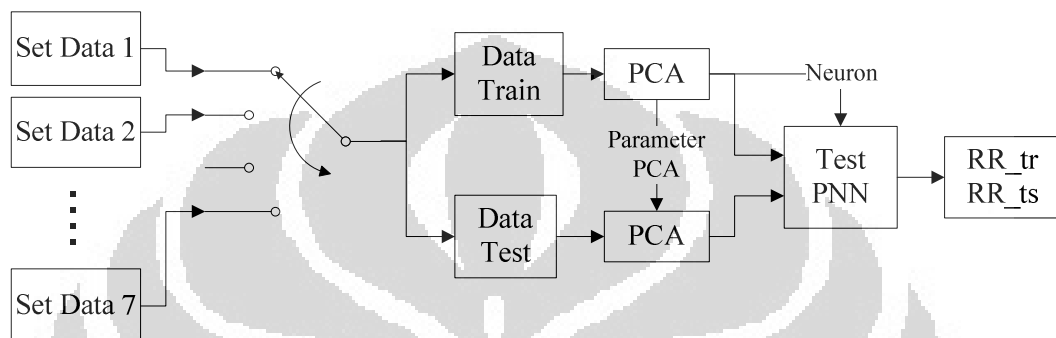
- Menguji performa PNN Standar dalam mengklasifikasikan Data Wajah berspektrum gabungan infra merah dan cahaya tampak.
- Menentukan nilai *smoothing parameter* optimal PNN Standar untuk masing-masing tipe data.
- Membandingkan performa (Tingkat Rekognisi dan waktu komputasi) PNN Standar dengan *Backpropagation*.

3.3.2. Skema Percobaan

Pada Percobaan ini, data yang akan digunakan adalah seluruh Set Data Wajah. Ketujuh macam data ini akan diuji secara terpisah oleh PNN untuk mengetahui apakah data wajah dengan tingkat iluminasi yang berbeda akan memiliki nilai *smoothing parameter* optimal yang berbeda pula. Nilai *smoothing parameter* yang

akan digunakan adalah dari 0.2 sampai 40. Vektor data akan direduksi menjadi 30 dimensi menggunakan PCA. Berikut adalah rangkuman konfigurasi percobaan:

- Set Data : 1 s/d 7
- Isi Set Data : 20 kelas, 30 pola/kelas
- Data *Test* : Data *Train* : 50% : 50%
- PCA : 30



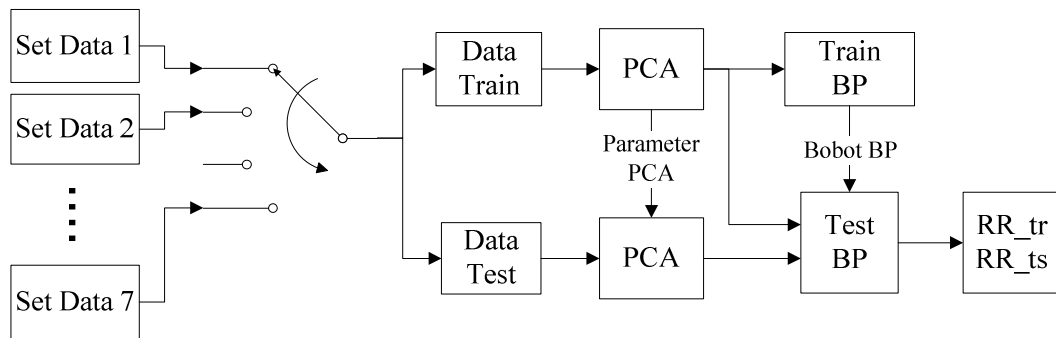
Gambar 3.5. Skema Percobaan PNN Standar

Gambar 3.5. adalah skema percobaan PNN Standar yang akan dilakukan dan berikut adalah konfigurasi yang digunakan:

- Metode : PNN Standar
- *Smoothing Parameter* : [0.2 0.4 0.6 0.8 1 2 3 ... 40]
- Jumlah neuron : 300 (maksimum)

Backpropagation (BP) akan digunakan pula untuk merekognisi Data Wajah tersebut sebagai perbandingan. Gambar 3.6. adalah skema percobaan BP yang akan dilakukan dan berikut adalah konfigurasi BP yang digunakan:

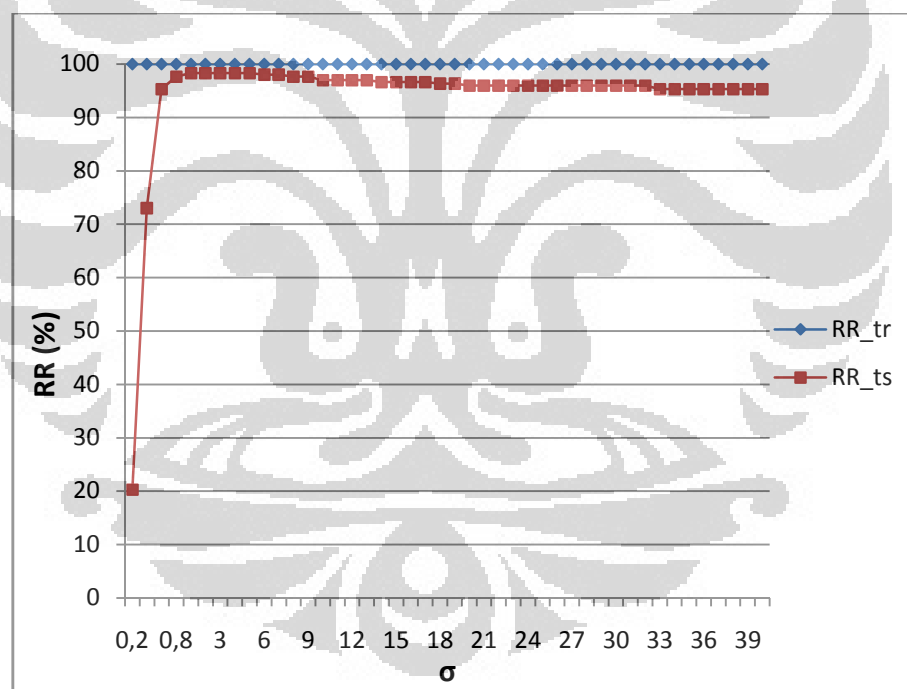
- Metode : *Backpropagation*
- Jumlah Neuron Input : 30
- Jumlah Neuron *Hidden* : 15
- Fungsi Error : Kuadratis
- Laju Pembelajaran : 0.2
- Batas pembelajaran : epoh max 10000 atau error total min 0.01
- Inisialisasi Bobot : Nguyen Widrow



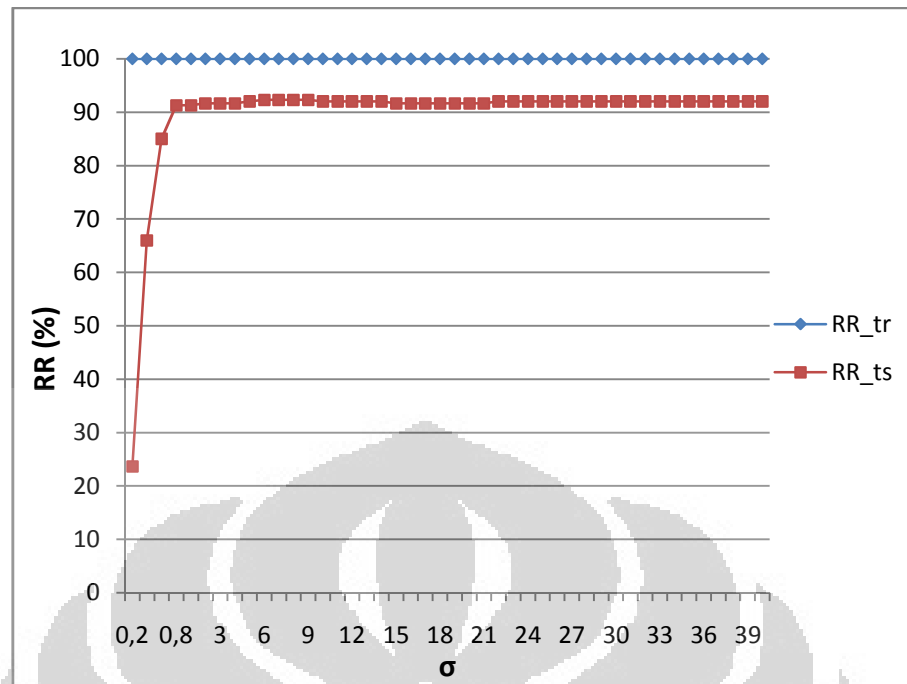
Gambar 3.6. Skema Percobaan BP

3.3.3. Hasil dan Analisa Percobaan

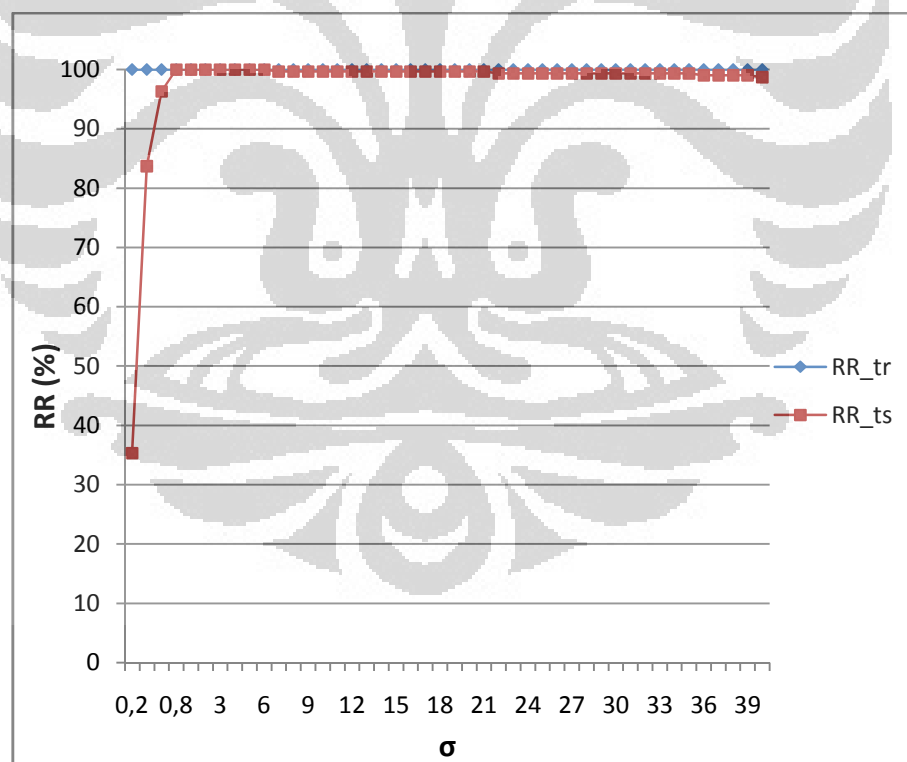
Berikut adalah grafik Tingkat Rekognisi PNN Standar terhadap Data *Train* dan Data *Test* dengan variasi nilai σ untuk Set Data 1 sampai dengan 7.



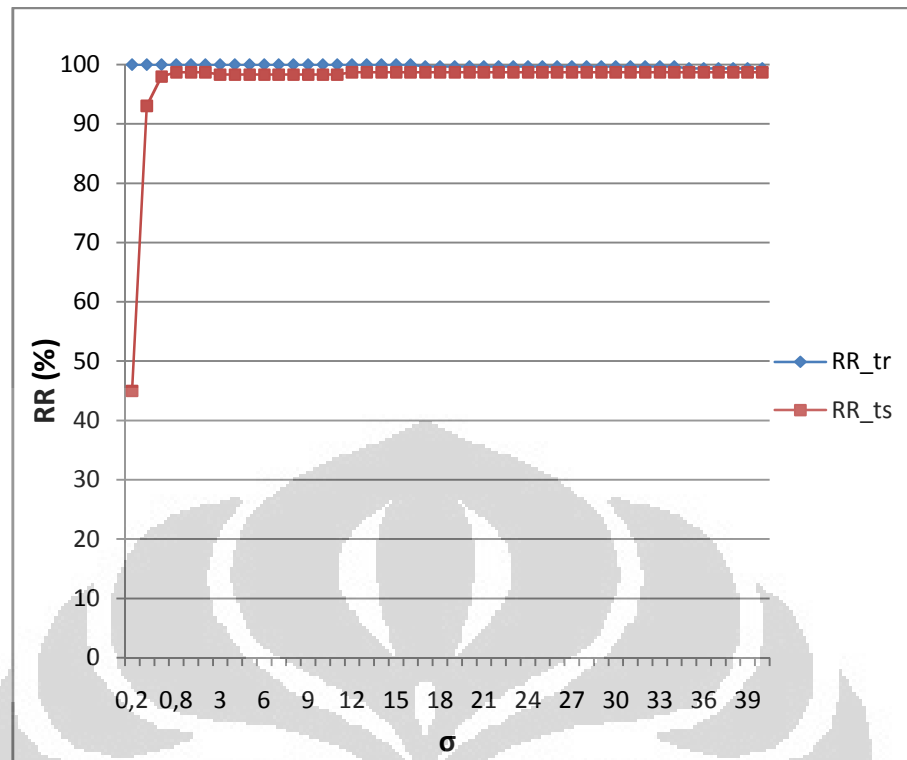
(a) RR Set Data 1



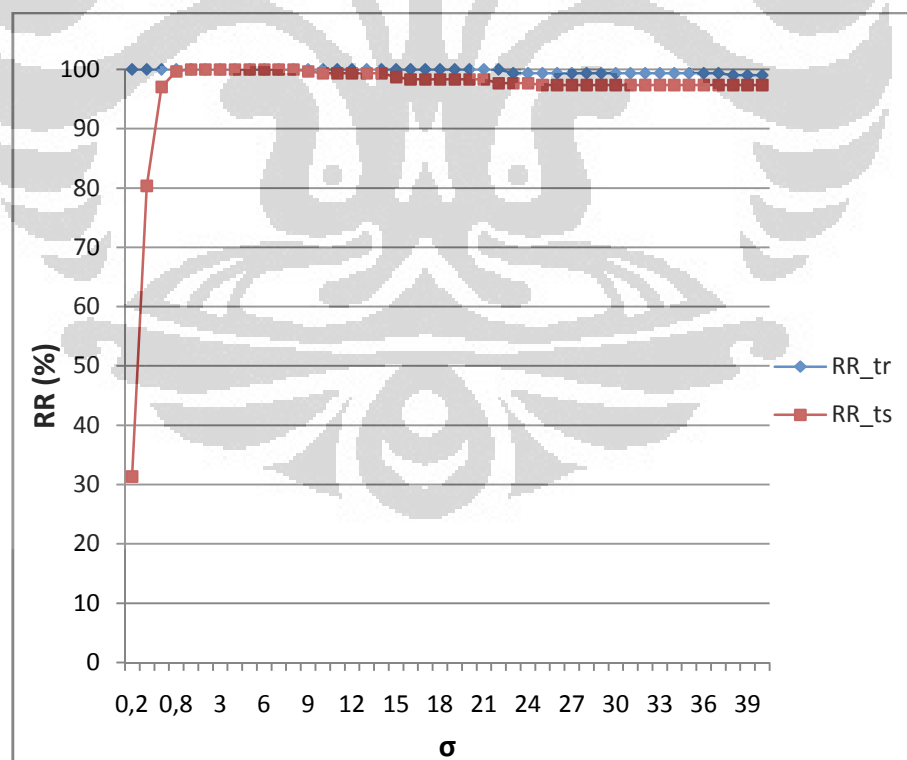
(b) RR Set Data 2



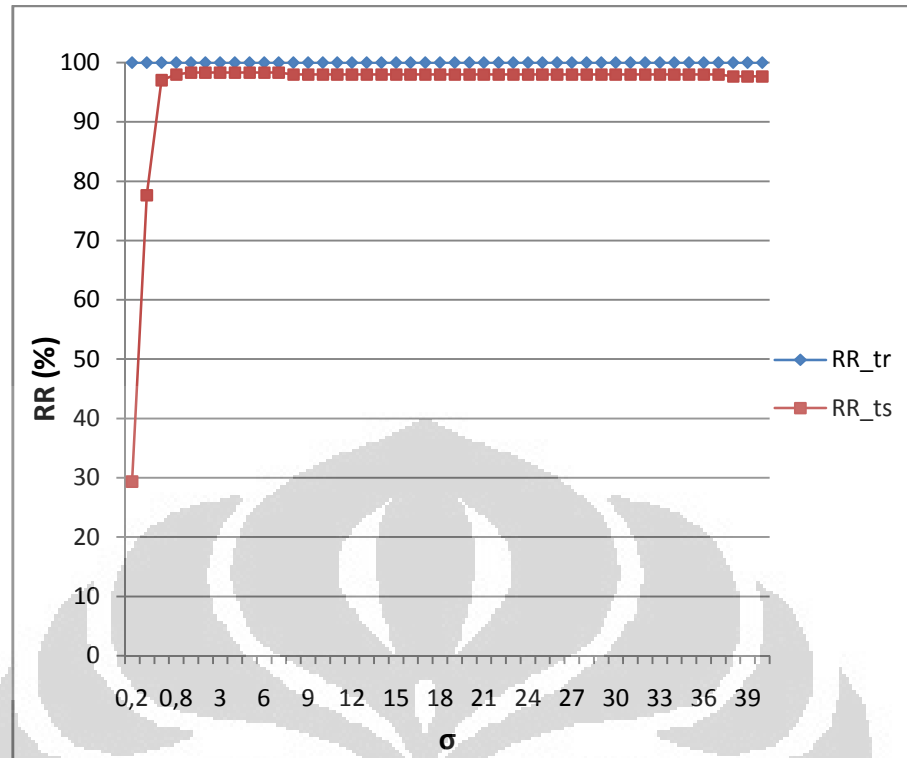
(c) RR Set Data 3



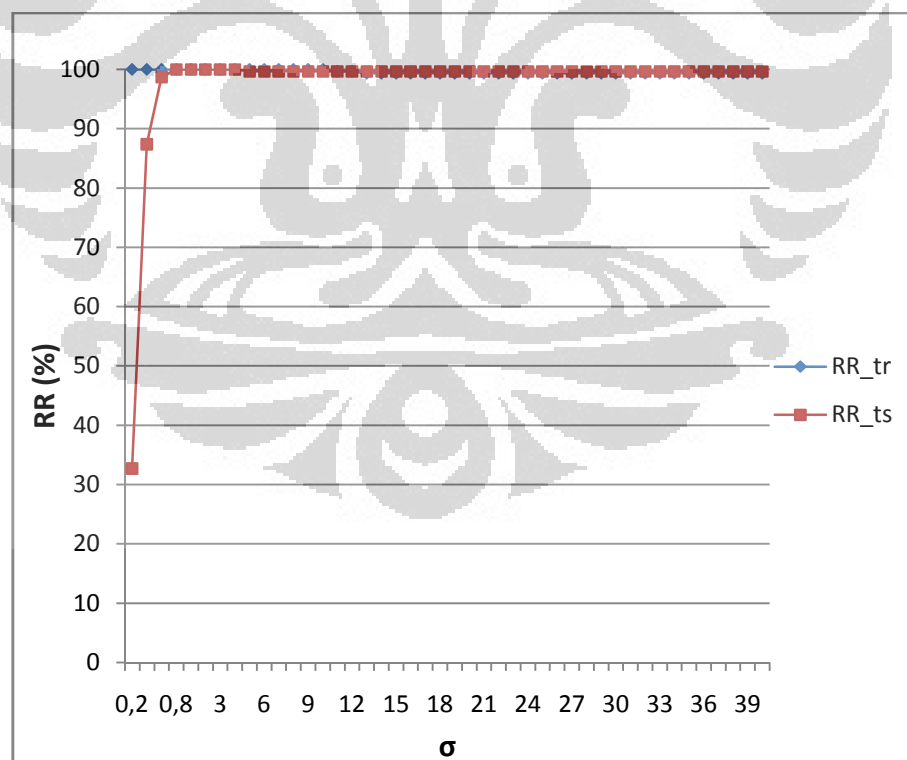
(d) RR Set Data 4



(e) RR Set Data 5



(f) RR Set Data 6



(g) RR Set Data 7

Gambar 3.7. Tingkat Rekognisi PNN Standar masing-masing Set Data dengan variasi *smoothing parameter*

Dari Gambar 3.7 terlihat bahwa Tingkat Rekognisi Data *Train* untuk Set Data 1, 2, 3, 6 selalu bernilai 100% untuk nilai *smoothing parameter* (σ) berapapun, sedangkan untuk Set Data lainnya cenderung menurun jika nilai σ terlalu besar. Tingkat Rekognisi Data *Train* untuk PNN Standar memang seharusnya bernilai 100% karena Data *Train* yang diujikan seluruhnya terdapat pada neuron PNN sehingga dapat dengan mudah mengklasifikasikannya dengan benar. Tingkat Rekognisi Data *Train* yang tidak 100% berarti nilai σ yang sudah terlalu besar sehingga kurva PDF tiap kelas akan terlalu lebar dan saling membaaur sehingga membuat PNN salah klasifikasi. Oleh karena itu kita dapat memperoleh batasan nilai σ optimal dari Tingkat Rekognisi Data *Train* tiap Set Data tersebut, σ maksimum dimana Tingkat Rekognisi Data *Train* tetap 100% yaitu 10.

Sedangkan Tingkat Rekognisi Data *Test* pada setiap Set Data memiliki kurva yang tidak linear. Tingkat Rekognisi Data *Test* umumnya bernilai di bawah 90% pada nilai σ di bawah 0.8 dan bernilai di atas 90% pada nilai σ di atas 0.8. Namun umumnya Tingkat Rekognisi Data *Test* mulai menurun kembali pada nilai σ melebihi 9, walaupun penurunannya tidak begitu tajam. Kurva Tingkat Rekognisi Data *Test* terhadap nilai *smoothing parameter* yang dibentuk oleh tiap Set Data memiliki pola yang mirip. Berdasarkan Tingkat Rekognisi Data *Train* dan Tingkat Rekognisi Data *Test* tiap Set Data dapat disimpulkan nilai *smoothing parameter* optimal dapat diaproksimasi pada rentang $0.8 < \sigma < 9$.

Tabel 3.3. Perbandingan performa PNN Standar dengan BP

Set Data	RR_tr (%)		RR_ts (%)		Waktu <i>Training</i> (detik)		Waktu <i>Testing</i> (detik)	
	PNN	BP	PNN	BP	PNN	BP	PNN	BP
1	100	100	98.33	88.67	-	483.77	2.033	0.173
2	100	100	92	83.67	-	540.97	2.120	0.100
3	100	100	100	99	-	512.88	2.049	0.087
4	100	100	98.33	96	-	529.89	2.250	0.127
5	100	100	100	91	-	507.62	2.061	0.088
6	100	100	98.33	90.33	-	502.81	2.222	0.087
7	100	100	99.67	91.33	-	502.84	2.049	0.089
rerata	100	100	98.09	91.43	-	511.54	2.112	0.107

Tabel 3.3. adalah perbandingan performa PNN Standar dan BP dalam mengklasifikasikan Data Wajah gabungan infra merah dan cahaya tampak, dengan mengambil nilai $\sigma = 5$ untuk PNN. Parameter yang akan dibandingkan adalah Tingkat Rekognisi dan waktu *Training* dan *Testing* (*Testing* untuk 600 Data, Data *Train* dan Data *Test*).

Dari Tabel 3.3. dapat disimpulkan bahwa:

- PNN Standar dan BP memiliki Tingkat Rekognisi Data *Train* yang sama untuk seluruh tipe Data Wajah, yaitu 100%.
- PNN Standar memiliki Tingkat Rekognisi Data *Test* yang lebih baik dari BP untuk semua tipe Data wajah, dengan perbedaan rata-rata adalah 6.67%.
- PNN Standar tidak memiliki fase *Training* pada jaringannya, sedangkan BP memiliki waktu *Training* yang cukup lama yaitu dengan nilai rata 511.54 detik.
- PNN Standar memiliki waktu *Testing* yang lebih lama dari BP karena waktu komputasi PNN Standar bergantung pada jumlah neuron yang digunakan. Waktu *Testing* rata-rata PNN Standar untuk 600 Data adalah 2.112 detik, sedangkan waktu *Testing* rata-rata BP adalah 0.107 detik, sehingga PNN Standar memiliki waktu *Testing* 19.7 kali lebih lama dari BP.

3.4. Percobaan Optimasi PNN

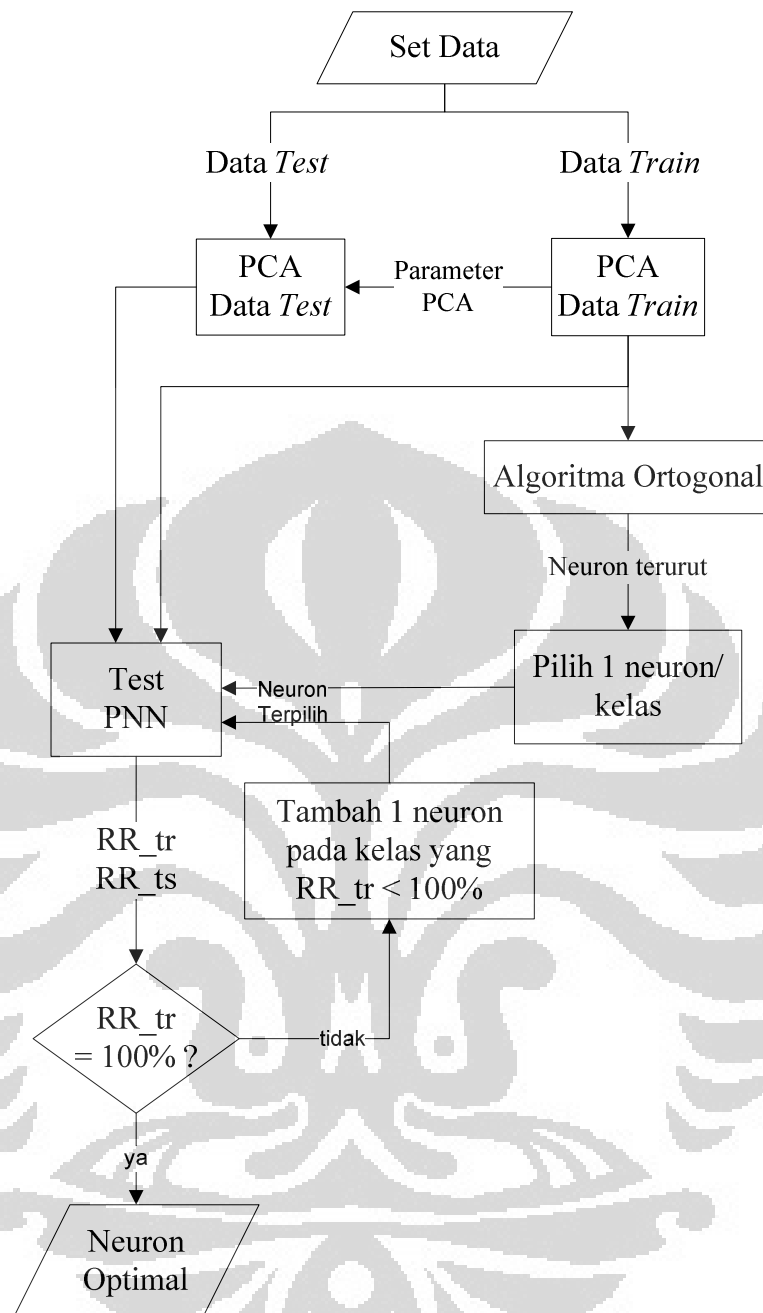
3.4.1. Tujuan Percobaan

Percobaan ini bertujuan sebagai berikut:

- Menguji performa PNN Teroptimasi (OPNN) dalam mengklasifikasikan Data Wajah.
- Membandingkan performa (Tingkat Rekognisi dan waktu komputasi) OPNN dengan PNN Standar dan *Backpropagation*.

3.4.2. Skema Percobaan

Pada Percobaan ini, data yang akan digunakan hanya Set Data 1, karena percobaan ini hanya untuk melihat efek optimasi jaringan PNN terhadap performanya, apakah performanya akan menurun atau tetap atau bahkan mungkin dapat lebih baik. Gambar 3.8. adalah bagan percobaan yang akan dilakukan.



Gambar 3.8. Prosedur Optimasi Struktur Neuron PNN

Pada iterasi pertama neuron yang digunakan pada Pattern Layer hanya berjumlah 1 pada tiap kelas, lalu seluruh *Data Train* dan *Data Test* diklasifikasi menggunakan PNN tersebut. Kelas yang memiliki Tingkat Rekognisi *Data Train* dibawah 100% akan ditambah 1 neuron di Pattern Layer pada iterasi berikutnya. Iterasi berhenti saat Tingkat Rekognisi *Data Train* seluruh kelas telah mencapai 100% atau ada salah satu kelas telah mencapai jumlah neuron maksimum.

Nilai *smoothing parameter* yang akan digunakan adalah 5, dengan alasan bahwa nilai ini masuk ke dalam rentang nilai *smoothing parameter* optimal yang didapat dari percobaan PNN Standar. Seperti pada percobaan PNN Standar, Vektor data akan direduksi menjadi 30 dimensi menggunakan PCA. Berikut adalah rangkuman konfigurasi percobaan:

- Metode : OPNN
- *Smoothing Parameter* : 5
- Set Data : 1
- Isi Set Data : 20 kelas, 30 pola/kelas
- Data *Test* : Data *Train* : 50% : 50%
- PCA : 30

3.4.3. Hasil dan Analisa Percobaan

Berikut adalah tabel perubahan Tingkat Rekognisi OPNN seiring dengan penambahan jumlah neuron yang digunakan.

Tabel 3.4. Tingkat Rekognisi OPNN setiap iterasi

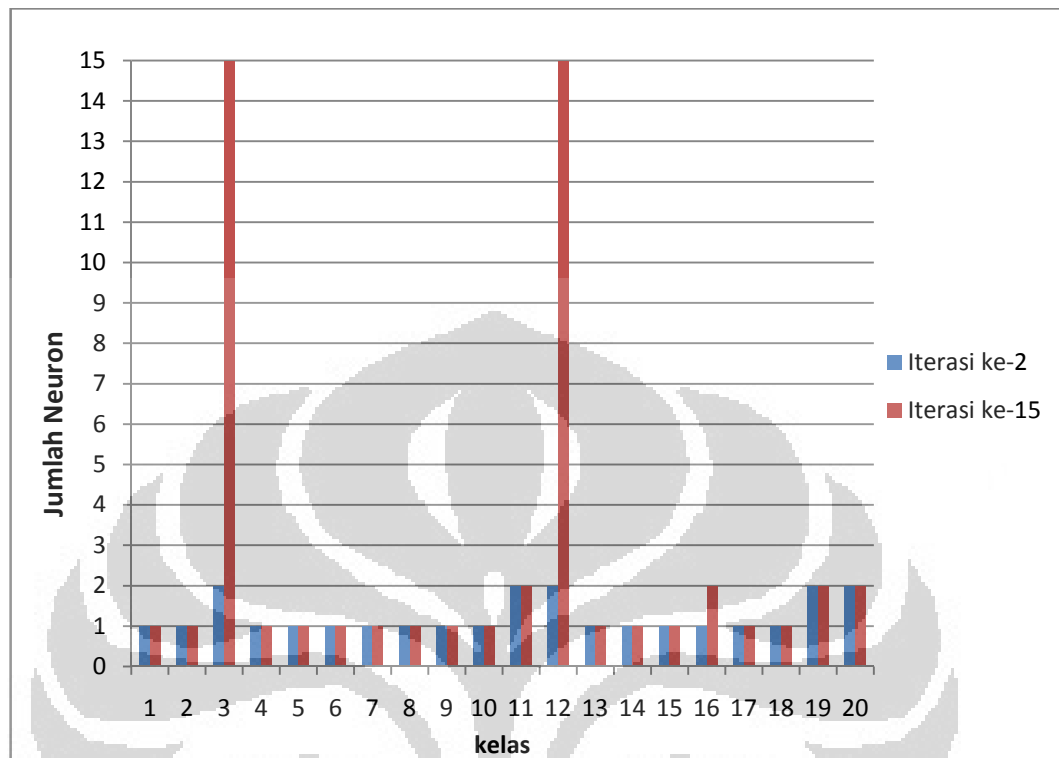
Iterasi ke	Waktu <i>Training</i>	Jumlah neuron	Waktu <i>Testing</i>	RR_tr (%)	RR_ts (%)
1	0,3333	20	0,2303	97	89
2	0,5996	25	0,2663	100	94,33
3	0,8865	28	0,2869	100	96,67
4	11.854	30	0,2988	100	97
5	15.037	32	0,3183	100	98,67
6	18.285	34	0,3248	100	98,67
7	21.654	36	0,3369	100	98,67
8	25.178	38	0,3525	100	98,67
9	28.811	40	0,3632	100	98,67
10	32.562	42	0,3752	100	98,67
11	36.452	44	0,389	100	98,67
12	40.544	46	0,4092	100	98,67
13	44.697	48	0,4153	100	98,67
14	48.962	50	0,4265	100	98,67
15	53.366	52	0,4404	100	98,67

Pada iterasi 1, yaitu saat tiap kelas hanya memiliki 1 neuron, Tingkat Rekognisi Data *Train* telah mencapai 97% dan Tingkat Rekognisi Data *Test*

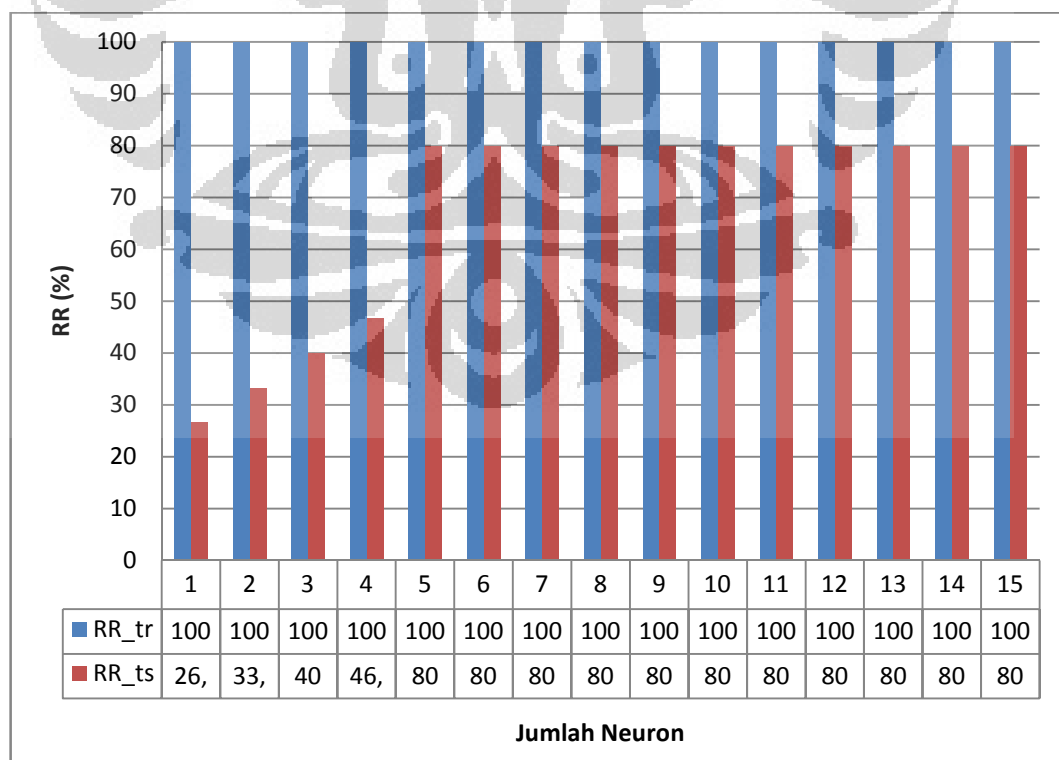
mencapai 89%. Pada iterasi pertama ini tingkat rekognisi PNN sudah cukup baik. Namun untuk lebih meningkatkan Rekognisi PNN, neuron ditambahkan satu per satu pada kelas yang masih memiliki Tingkat Rekognisi Data *Train* di bawah 100%. Akibatnya setiap iterasi penambahan neuron, Tingkat Rekognisi Data *Train* bertambah hingga mengalami saturasi, yaitu sampai iterasi ke 2 dengan jumlah neuron adalah 25 dan Tingkat Rekognisi Data *Train* 100% serta Tingkat Rekognisi Data *Test* 94.33%. Penambahan neuron berikutnya tidak berdampak apapun pada Tingkat Rekognisi Data *Train* karena telah mencapai nilai maksimum sedangkan Tingkat Rekognisi Data *Test* masih meningkat hingga iterasi ke-5. Namun Tingkat Rekognisi Data *Test* tidak dapat dijadikan acuan dalam optimasi Struktur Neuron dengan alasan neuron-neuron telah diurutkan oleh Algoritma Ortogonal berdasarkan kedekatannya dengan Data *Train*, bukan Data *Test*. Seandainya proses optimasi didasarkan pada Tingkat Rekognisi Data *Test* maka performa OPNN hampir menyamai performa PNN yaitu pada iterasi ke-5 dengan Tingkat Rekognisi Data *Train* 100% dan Tingkat Rekognisi Data *Test* 98.67%. Namun Pada aplikasi sesungguhnya Data *Test* tidak termasuk dalam pengetahuan sistem, sehingga proses optimasi ini hanya didasarkan oleh Tingkat Rekognisi Data *Train*. Pada akhirnya struktur neuron paling optimal untuk Set Data 1 ini berdasarkan Tingkat Rekognisi Data *Train* adalah struktur pada iterasi ke-2 dengan jumlah neuron 25.

Gambar 3.9. adalah struktur neuron PNN tiap kelas pada iterasi ke-5 dan ke-15. Dari grafik terlihat bahwa jumlah neuron tiap kelas tidak selalu sama. Hampir semua kelas mencapai Tingkat Rekognisi Data *Test* 100%, yaitu kelas yang jumlah neuronnya sama pada saat iterasi ke-2 dengan ke-15. Namun ada pula kelas yang tidak pernah mencapai Tingkat Rekognisi Data *Test* 100%, yaitu kelas ke-3 dan ke-12 sehingga struktur neuronnya mengalami penambahan terus hingga iterasi ke-15. Untuk kelas yang telah mencapai Tingkat Rekognisi Data *Train* 100% maka penambahan neuron tidak akan berkontribusi apapun untuk meningkatkan Tingkat Rekognisi Data *Train*, sehingga neuron-neuron yang tersisa ini tidak dipakai sehingga struktur neuron menjadi tereduksi dan proses komputasi akan semakin cepat. Struktur Neuron Optimal seperti Gambar 3.9 di

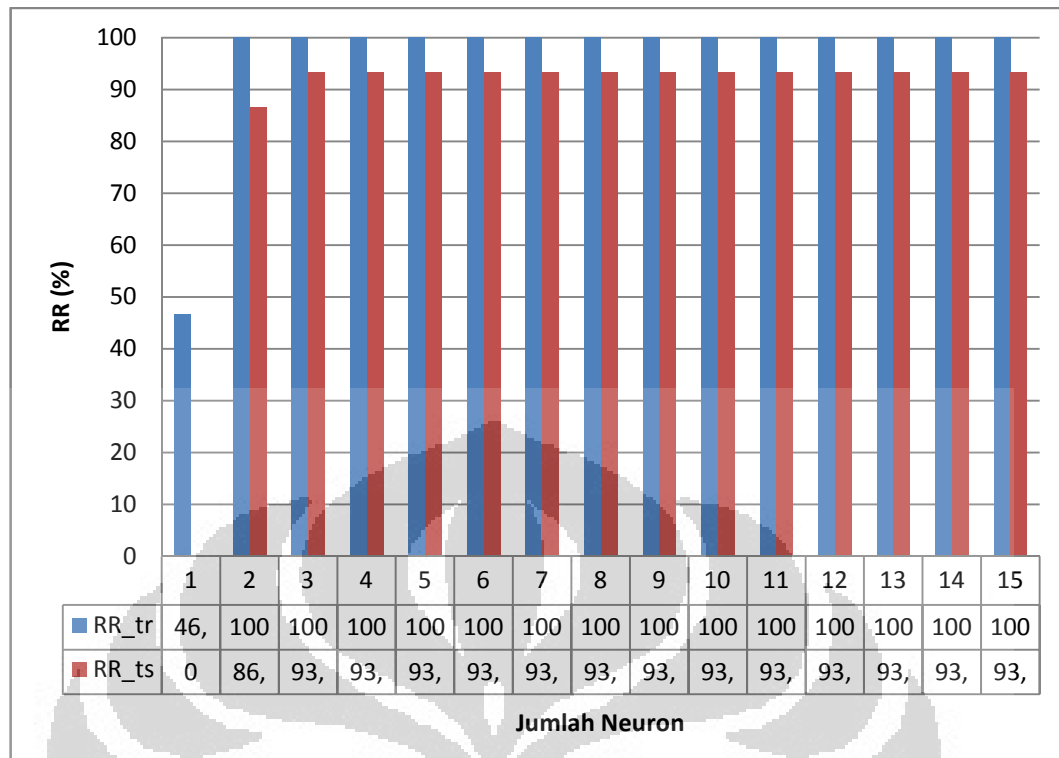
atas belum tentu sama untuk Data yang lain karena pencarian struktur optimal bergantung kualitas Data *Train*.



Gambar 3.9. Struktur Neuron Iterasi ke-2 (optimal) dan ke-15



Gambar 3.10. Tingkat Rekognisi kelas ke-3 seiring perubahan jumlah neuron



Gambar 3.11. Tingkat Rekognisi kelas ke-12 seiring perubahan jumlah neuron

Setelah melalui proses Algoritma Ortogonal, setiap kelas akan memiliki neuron-neuron yang telah terurut, sehingga neuron pertama adalah neuron paling penting dan paling merepresentasikan kelas tersebut. Pada kelas ke-3 dan ke-12, penggunaan hanya 1 buah neuron hanya menghasilkan Tingkat Rekognisi Data *Train* masing-masing 100% dan 46.67% yang berarti neuron terpenting pada kelas ke-3 dan ke-12 tidak cukup untuk merepresentasikan seluruh Data *Train*, inilah alasan mengapa kelas ke-3 dan ke-12 memerlukan lebih dari 1 neuron untuk mencapai Tingkat Rekognisi Data *Train* maksimum.

Struktur Neuron Optimal bergantung pada tingkat representatif neuron terhadap kelas tersebut. Neuron-neuron dalam kelas ke-3 dan ke-12 memiliki tingkat representatif yang tinggi terhadap Data *Train* namun tidak begitu tinggi untuk Data *Test* dibandingkan dengan kelas-kelas lain. Kesimpulan ini dibuktikan oleh ketidakmampuan neuron-neuron kelas ke-3 dan ke-12 untuk mengenali semua Data *Test* walaupun jumlah neuronnya terus ditambah, namun dapat mengenali seluruh Data *Train* hanya dengan 2 neuron.

Pada saat Tingkat Rekognisi Data *Test* kelas ke-3 dan ke-12 telah mencapai maksimum 100%, yaitu pada jumlah neuron 2, penambahan neuron menambah nilai Tingkat Rekognisi Data *Test* namun tidak berpengaruh pada nilai Tingkat Rekognisi Data *Train*, ini berarti neuron-neuron yang tersisa berada berdekatan dengan neuron-neuron yang telah terpilih sebelumnya. Proses optimasi akan membuang neuron-neuron yang tidak berkontribusi dalam meningkatkan Tingkat Rekognisi Data *Train* seperti pada kasus kelas ke-3 dan kelas ke-12 walaupun sebenarnya neuron-neuron tersebut dapat memperbaiki Tingkat Rekognisi Data *Test*. Oleh karena itu, proses optimasi struktur neuron PNN pada dasarnya mereduksi jumlah neuron tiap kelas dengan menjaga Tingkat Rekognisi Data *Train* tetap 100% namun tidak menjamin Tingkat Rekognisi Data *Test* akan maksimum.

Berikut adalah perbandingan performa OPNN dengan PNN Standar dan BP dalam mengklasifikasi Set Data 1.

Tabel 3.5. Perbandingan PNN Standar, OPNN dan BP

	PNN Standar	OPNN	BP
RR_tr	100 %	100 %	100 %
RR_ts	98.33 %	94.33 %	88.67 %
Waktu Training	-	0.6 detik	483.77 detik
Waktu Testing	2.029 detik	0.266 detik	0.173 detik
Jumlah Neuron Pattern Layer	300	25	-

Dari Tabel 3.5. terlihat bahwa dari Tingkat Rekognisi Data *Train* PNN OPNN dapat menyamai PNN Standar maupun BP. Dari segi Tingkat Rekognisi Data *Test* OPNN ternyata lebih buruk dari PNN Standar namun tetap lebih baik dari BP. Dari tabel perbandingan juga terlihat bahwa waktu *Training* OPNN sekitar 806 kali lebih cepat dari BP. Waktu *Testing* OPNN (untuk 600 Data) adalah 7.6 kali lebih cepat dari PNN Standar, dengan jumlah neuronnya hanya 8.33% dari PNN Standar. Namun Waktu *Testing* BP masih paling cepat, yaitu 1.5 kali lebih cepat dari OPNN.

3.5. Penalaan log μ pada Normalisasi Iluminasi

3.5.1. Tujuan Percobaan

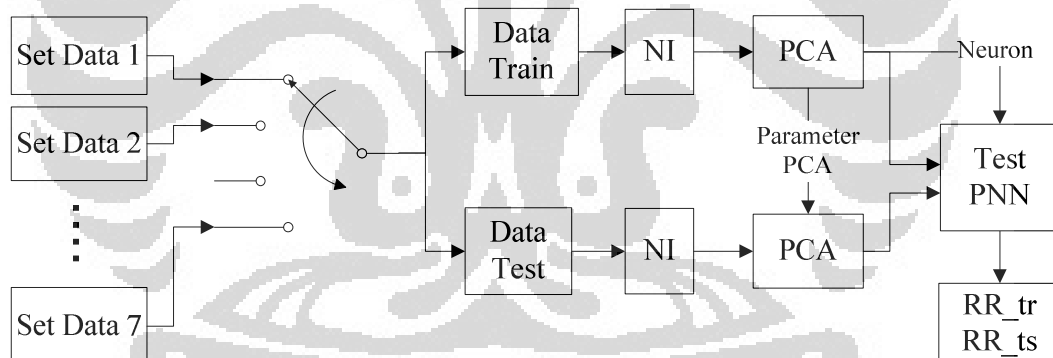
Percobaan ini bertujuan untuk mencari nilai log μ yang tepat untuk Data Wajah pada proses Normalisasi Iluminasi dalam ruang DCT Logaritma. Parameter tepat atau tidaknya nilai log μ dinilai dari Tingkat Rekognisi PNN dalam mengenali Data Wajah tersebut.

3.5.2. Skema Percobaan

Gambar 3.12. adalah prosedur dalam melakukan Normalisasi Iluminasi dan Gambar 3.13. adalah skema percobaan yang dilakukan.



Gambar 3.12. Proses Normalisasi Iluminasi

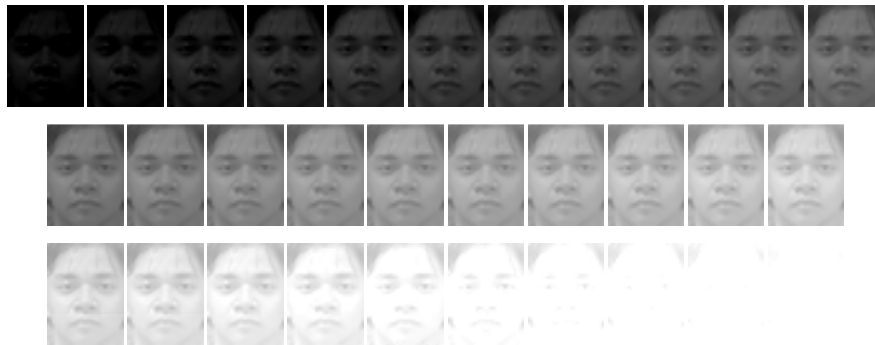


Gambar 3.13. Skema Percobaan PNN Standar dengan Normalisasi Iluminasi

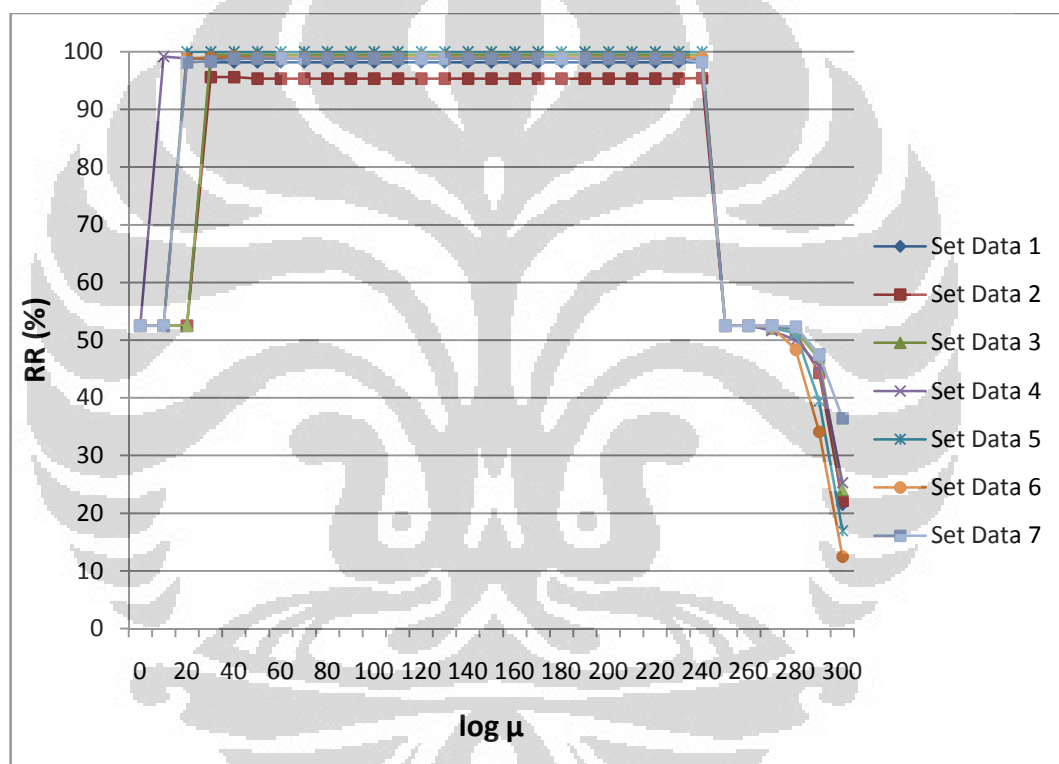
Berikut adalah konfigurasi percobaan :

- Metode : PNN Standar
- Smoothing Parameter : 4
- Set Data : 1 s/d 7
- Data Test : Data Train : 50% : 50%
- PCA : 30
- Nilai log μ : 0 s/d 300 dengan kelipatan 10

3.5.3. Hasil dan Analisa Percobaan



Gambar 3.14. Citra sampel yang mengalami perubahan $\log \mu$ dari 0 hingga 300






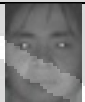





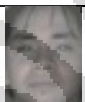


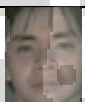
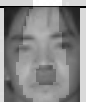

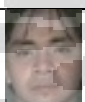


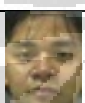
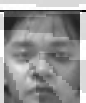

Gambar 3.15. Tingkat Rekognisi PNN terhadap perubahan $\log \mu$

Tingkat Rekognisi bernilai maksimum pada aproksimasi rentang $30 < \log \mu < 240$. Semakin kecil nilai $\log \mu$ akan menyebabkan gambar semakin gelap atau nilai rata-rata piksel gambar akan semakin dekat ke nol, sebaliknya semakin besar nilai $\log \mu$ maka gambar semakin putih atau rata-rata nilai pikselnya semakin mendekati 255 (citra digital 8 bit). Kedua keadaan tersebut akan membuat PNN semakin sulit mengenali identitas wajah. Maka nilai $\log \mu$ optimal berada pada rentang $30 < \log \mu < 240$.

3.6. Normalisasi Iluminasi Citra berspektrum gabungan Infra merah dan Cahaya tampak

Untuk percobaan-percobaan yang akan dilakukan selanjutnya akan digunakan nilai $\log \mu$ 150 untuk setiap Normalisasi Iluminasi citra, dengan alasan nilai $\log \mu$ ini berada pada rentang yang optimal dan secara visual memiliki kecerahan normal.

Tabel 3.6. Contoh perbandingan citra asli, *grayscale* dan hasil NI

Set Data	Citra Asli	Citra Grayscale	Citra hasil NI
1			
2			
3			
4			
5			
6			
7			

Tabel 3.6. memperlihatkan contoh perbandingan citra asli, hasil grayscale dan hasil NI dari ketujuh Set Data yang digunakan dalam Penelitian ini. Hasil NI terlihat memiliki tingkat kecerahan yang sama untuk ketujuh Set Data citra, namun perbedaan kontras ketujuh citra tetap tidak sepenuhnya hilang.

BAB 4

OPNN UNTUK PENGENALAN WAJAH BERSPEKTRUM GABUNGAN INFRA MERAH DAN CAHAYA TAMPAK

Bab ini akan membahas seluruh percobaan yang dilakukan untuk membangun sebuah sistem pengenalan wajah berbasis PNN Teroptimasi untuk berbagai tingkat iluminasi dengan spektrum gabungan Infra merah dan Cahaya tampak. Pertama-tama akan dilakukan percobaan PNN Standar menggunakan berbagai tipe Data *Train* dari Set Data 1 hingga 7 untuk mengenali seluruh Set Data. Percobaan tersebut akan diulang dengan perbedaan yaitu Set Data asli, Set Data Ternormalisasi Iluminasi dan Set Data Ternormalisasi dan Terkompensasi Iluminasi. Lalu dari ketiga hasil percobaan tersebut akan dibandingkan dan dicari yang memiliki performa paling baik untuk seterusnya digunakan dalam percobaan OPNN menggunakan Data *Train* bertingkat iluminasi campuran. Dalam percobaan terakhir tersebut akan dibandingkan performa PNN dan OPNN yang menggunakan kombinasi Data *Train* dari berbagai Set Data yang ada.

4.1. PNN dengan Data Asli dan Data Ternormalisasi dan Terkompensasi Iluminasi

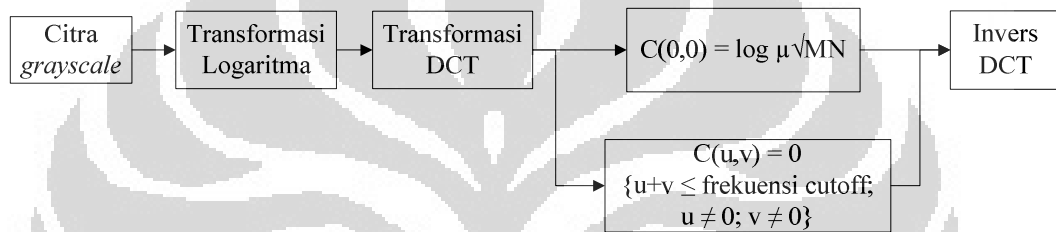
4.1.1. Tujuan Percobaan

Tujuan dari percobaan ini adalah:

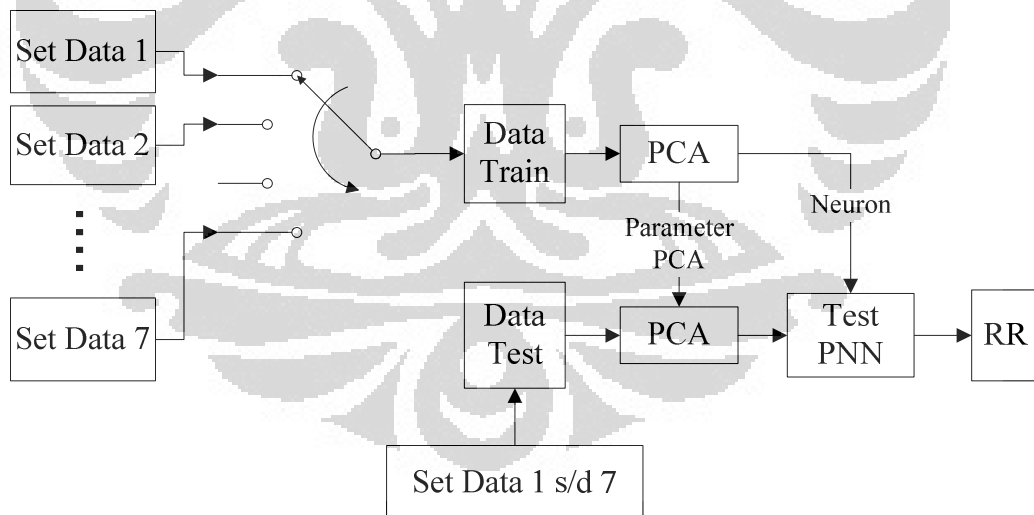
- Menguji performa PNN menggunakan Data *Train* dengan tingkat iluminasi tertentu untuk mengenali Data *Test* dengan tingkat iluminasi berbeda.
- Menerapkan metode Normalisasi Iluminasi pada data wajah dan melihat pengaruhnya pada performa PNN dengan Data *Train* dengan tingkat iluminasi tertentu untuk mengenali Data *Test* dengan tingkat iluminasi berbeda.
- Menerapkan metode NI dan KI pada data wajah dan melihat pengaruhnya pada performa PNN dengan Data *Train* dengan tingkat iluminasi tertentu untuk mengenali Data *Test* dengan tingkat iluminasi berbeda.
- Membandingkan performa PNN dengan BP pada percobaan ini.

4.1.2. Skema Percobaan

Pada percobaan ini, PNN Standar akan diuji untuk mengklasifikasi seluruh Set Data dan BP juga akan digunakan sebagai perbandingan. Data *Train* yang digunakan akan digilir dari Set Data 1 hingga 7, sehingga didapat 7 macam PNN dan BP dengan Data *Train* yang berbeda, yang masing-masing akan diuji untuk merekognisi Data *Test* yang berisi seluruh Set Data (Set Data 1 hingga 7). Percobaan ini akan dilakukan 3 kali yaitu menggunakan Data asli, Data ternormalisasi iluminasi dan Data ternormalisasi sekaligus terkompensasi iluminasi.



Gambar 4.1. Prosedur Normalisasi dan Kompensasi Iluminasi



Gambar 4.2. Skema Percobaan PNN dengan tiap macam Data *Train*

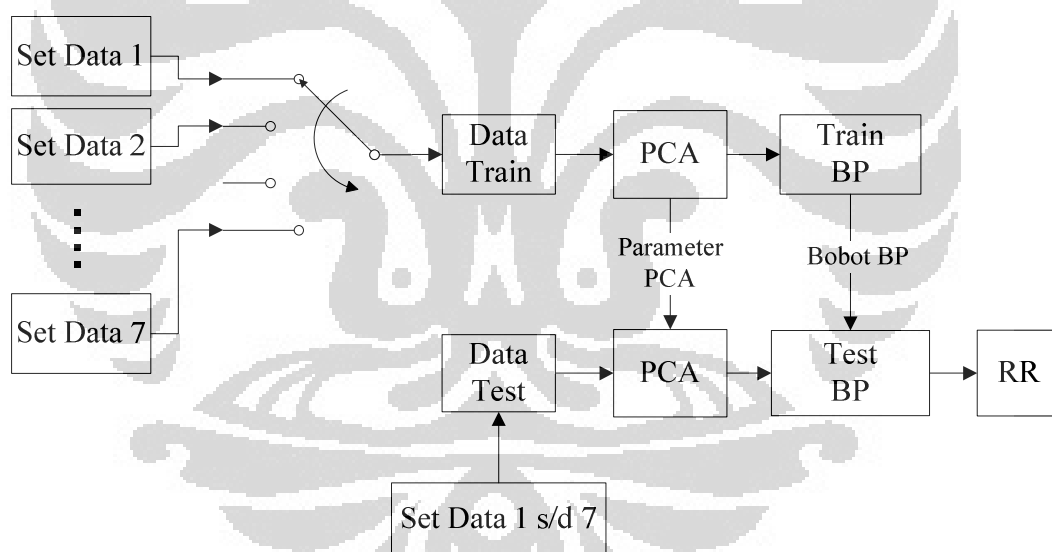
Gambar 4.1. merupakan skema prosedur NI + KI yang dilakukan dalam percobaan ini. Gambar 4.2. adalah skema percobaan PNN yang akan dilakukan dan berikut adalah konfigurasi PNN yang digunakan :

- Metode : PNN Standar

- Smoothing parameter : 5
- Jumlah neuron : 600 (maksimum)
- PCA : 30

Sedangkan Gambar 4.3. adalah bagan percobaan BP yang akan dilakukan dan berikut adalah konfigurasi BP yang digunakan:

- Metode : *Backpropagation*
- Jumlah Neuron Input : 30
- Jumlah Neuron *Hidden* : 15
- Fungsi Error : Kuadratis
- Laju Pembelajaran : 0.2
- Batas pembelajaran : epoh max 10000 atau error total min 0.01
- Inisialisasi Bobot : Nguyen Widrow
- PCA : 30

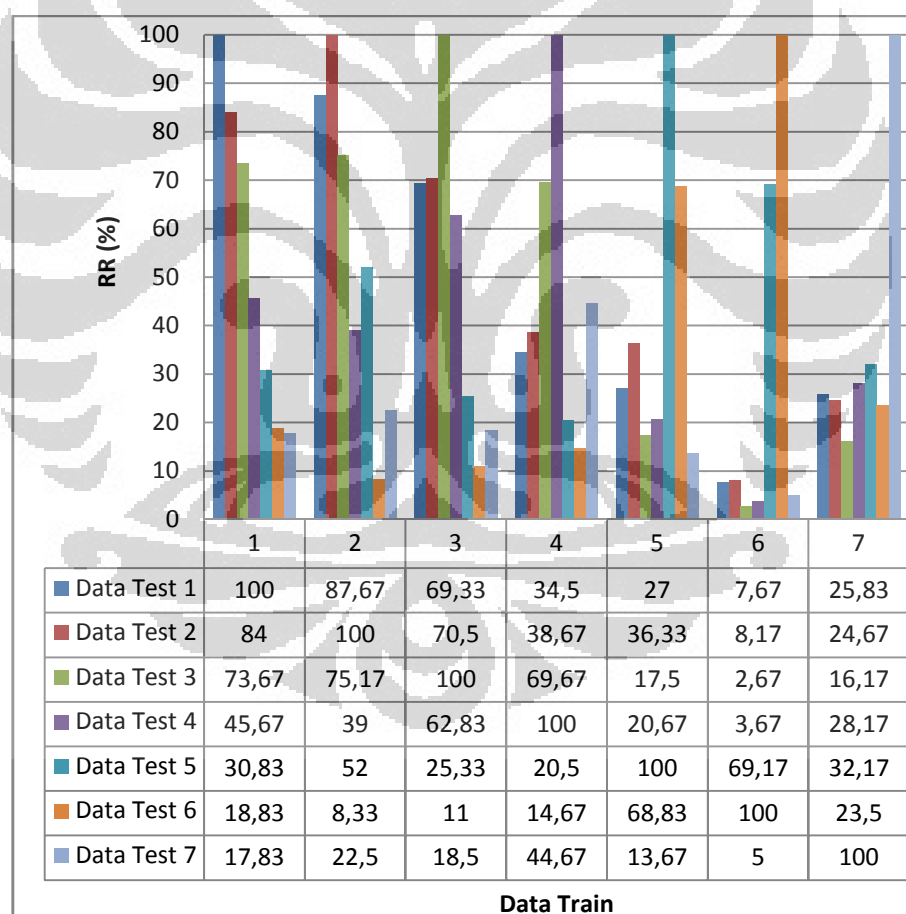


Gambar 4.3. Skema Percobaan BP dengan tiap macam Data *Train*

4.1.3. Hasil dan Analisa Percobaan dengan Data Asli

Dari Gambar 4.4 terlihat bahwa Tingkat Rekognisi PNN dengan masing-masing tipe Data *Train* memiliki nilai yang tidak seragam untuk tiap tipe Data *Test*. Setiap tipe Data *Train* PNN memiliki Tingkat Rekognisi 100% terhadap tipe Data *Test* yang sama, hal ini wajar karena Data *Test* sama persis dengan Data

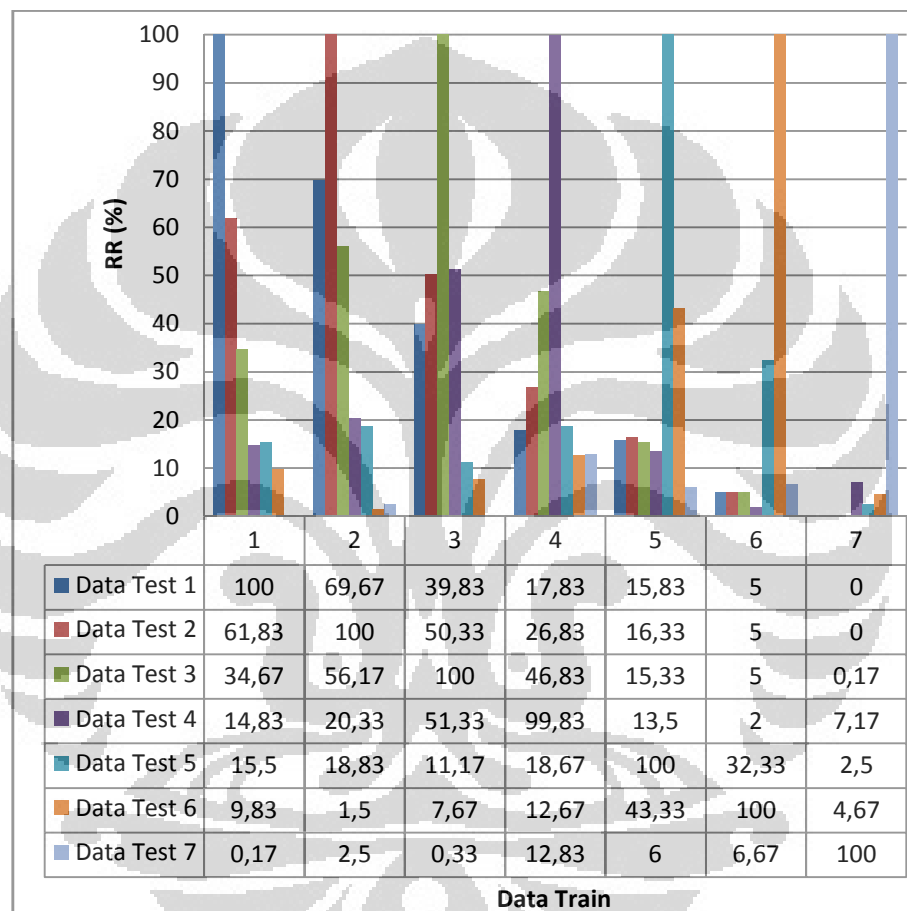
Train. Sedangkan untuk *Data Test* yang berbeda tipe dengan *Data Train* menghasilkan Tingkat Rekognisi PNN yang lebih rendah, dan semakin jauh perbedaan tipe antara *Data Train* dengan *Data Test* maka Tingkat Rekognisi semakin jatuh. Hasil seperti itu disebabkan oleh perbedaan tingkat iluminasi antar tipe Set Data yang mengakibatkan nilai piksel gambar berbeda dan ketika *Data Test* dibandingkan dengan neuron-neuron (*Data Train*) pada Pattern Layer tentu menghasilkan derajat kecocokan ϕ yang rendah sehingga PNN akan salah mengklasifikasikannya. Dari seluruh tipe *Data Train* PNN pada percobaan ini dapat disimpulkan bahwa PNN tidak dapat merekognisi dengan baik *Data Test* yang memiliki tingkat iluminasi yang berbeda dengan *Data Train* yang menjadi neuronnya.



Gambar 4.4. Tingkat Rekognisi *Data Test* PNN oleh masing-masing *Data Train*

Sedangkan Tingkat Rekognisi BP juga memiliki perilaku yang sama dengan PNN. Dari Gambar 4.5, terlihat bahwa Tingkat Rekognisi BP berkisar 100%

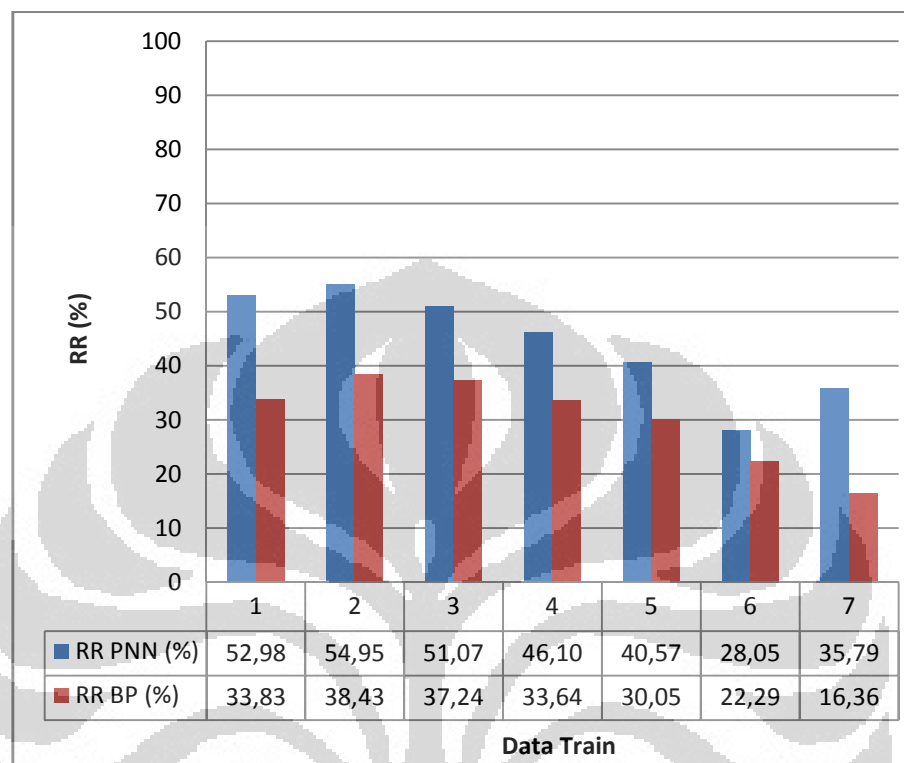
hanya pada Data *Test* yang sama dengan Data *Train* yang digunakan, dengan alasan bahwa BP berhasil mempelajari semua Data *Train*-nya dengan baik pada proses *Training*. Namun pada Data *Test* dengan tipe berbeda, BP tidak dapat mengenalinya dengan baik, terlihat dari Tingkat Rekognisinya yang semakin menurun seiring dengan semakin jauhnya perbedaan tipe antara Data *Test* dengan Data *Train*.



Gambar 4.5. Tingkat Rekognisi Data *Test* BP oleh masing-masing Data *Train*

Dari Gambar 4.6 terlihat bahwa Tingkat Rekognisi rata-rata ketujuh tipe Data *Train* PNN berada pada kisaran 28.05% hingga 54.95%, dimana PNN dengan Data *Train* 2 memiliki Tingkat Rekognisi tertinggi terhadap seluruh tingkat iluminasi Data *Test*. Namun dengan Tingkat Rekognisi hanya 54.95% PNN dengan Data *Train* 2 belum cukup untuk menjadi model umum yang dapat merekognisi seluruh tingkat iluminasi data wajah. Namun secara keseluruhan

Tingkat Rekognisi rata-rata masing-masing tipe Data *Train* PNN jauh lebih baik dari BP, yang berada pada kisaran 16.36% hingga 38.43%.

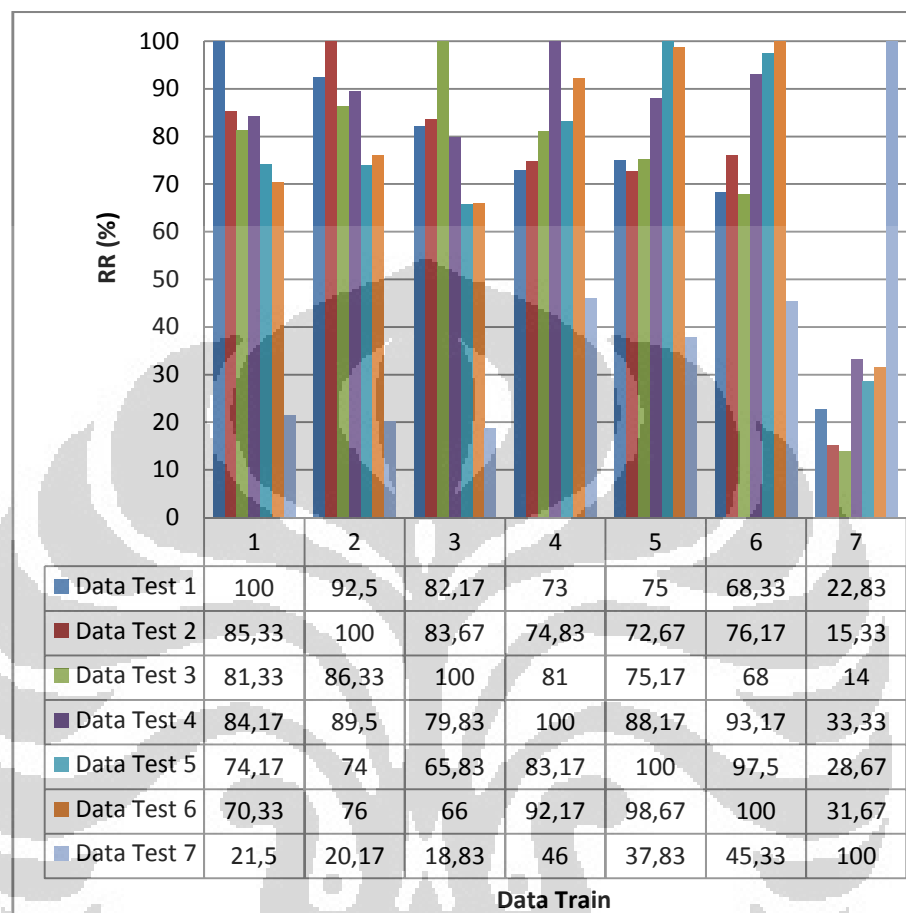


Gambar 4.6. Tingkat Rekognisi Data *Test* rata-rata PNN dan BP oleh masing-masing Data *Train*

4.1.4. Hasil dan Analisa Percobaan dengan Data Ternormalisasi Iluminasi

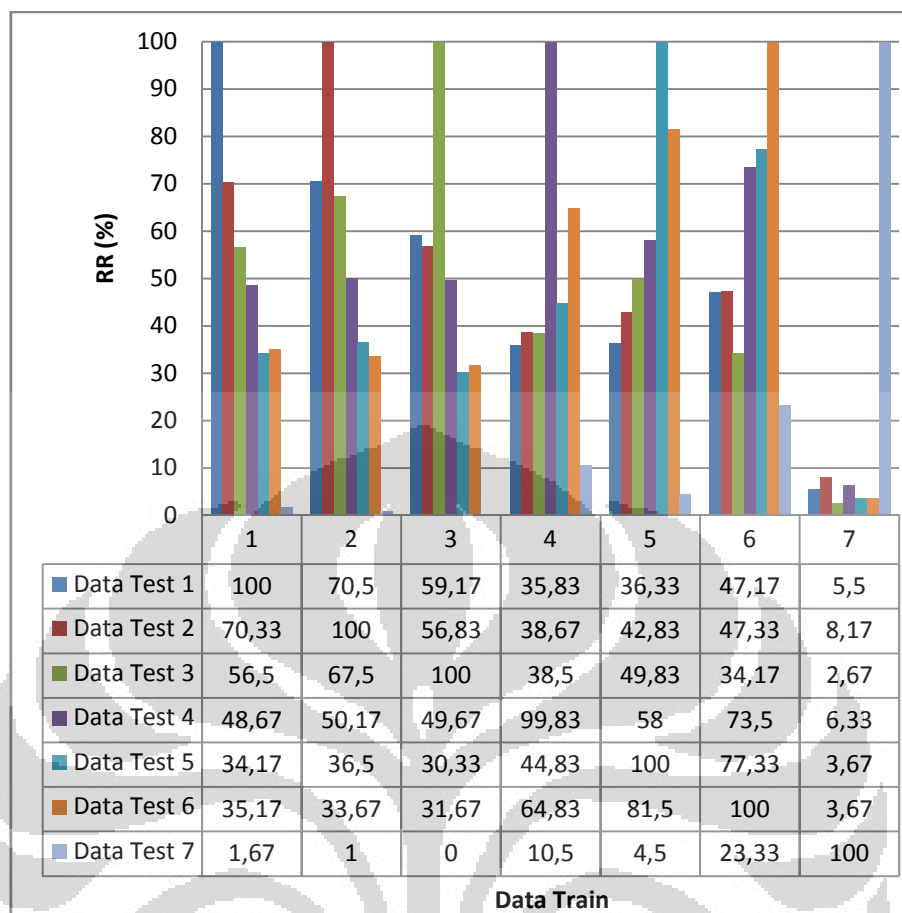
Gambar 4.7 dan Gambar 4.8 masing-masing adalah grafik Tingkat Rekognisi rata-rata tiap tipe Data *Train* PNN dan BP. Secara umum Grafik ketujuh tipe Data *Train* PNN menunjukkan perbaikan dari hasil percobaan sebelumnya. Tingkat Rekognisi ketujuh tipe Data *Train* PNN umumnya meningkat akibat penggunaan metode NI ini. Tingkat Rekognisi rata-rata PNN dengan Data *Train* 1 sampai 6 berada pada kisaran 70.9% hingga 78.6%, namun Tingkat Rekognisi PNN Data *Train* 7 tetap buruk yaitu 35.12%. PNN Data *Train* 7 tidak menunjukkan peningkatan rekognisi dibandingkan percobaan sebelumnya. Jika dilihat dari Tingkat Rekognisinya terhadap masing-masing tipe Data *Test*, PNN Data *Train* 7 hanya memiliki Tingkat Rekognisi pada kisaran 14% hingga 33.33% untuk Data *Test* 1 hingga 6 dan 100% untuk Data *Test* 7. Dengan kata lain PNN Data *Train* 7 tidak dapat mengenali dengan baik Data *Test* yang bertingkat iluminasi berbeda

dengan Data *Train* yang digunakan, walaupun menggunakan Normalisasi Iluminasi.



Gambar 4.7. Tingkat Rekognisi Data *Test* PNN oleh masing-masing Data *Train* (menggunakan NI)

Walaupun Tingkat Rekognisi PNN Data *Train* 1 hingga 6 meningkat dibandingkan percobaan sebelumnya, setiap tipe Data *Train* PNN tetap menunjukkan pola Tingkat Rekognisi yang mirip, yaitu semakin menurun seiring perbedaan tingkat iluminasi Data *Test* dengan Data *Train* yang semakin jauh, sementara Tingkat Rekognisi setiap tipe Data *Train* PNN hanya bernilai 100% pada Data *Test* yang sama. Sehingga dapat disimpulkan bahwa penggunaan metode NI dapat meningkatkan Rekognisi PNN untuk Data *Test* yang bertingkat iluminasi berbeda-beda, namun tetap tidak menghilangkan penurunan Tingkat Rekognisi akibat perbedaan tingkat iluminasi Data *Test* dengan Data *Train*.

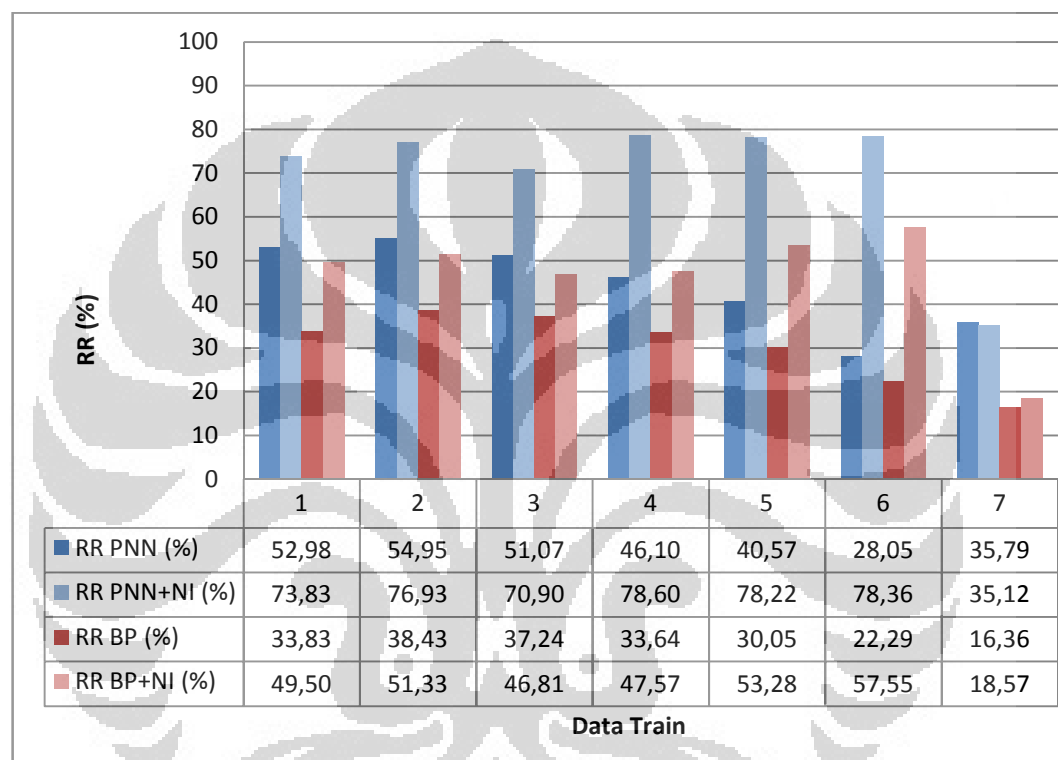


Gambar 4.8. Tingkat Rekognisi Data *Test* BP oleh masing-masing Data *Train* (menggunakan NI)

Sedangkan untuk percobaan BP seperti yang terlihat pada Gambar 4.8, Tingkat Rekognisinya juga memiliki perilaku yang sama dengan PNN, yaitu semakin menurun seiring perbedaan tingkat iluminasi yang semakin berbeda antara Data *Train* dengan Data *Test*. Sehingga dapat disimpulkan bahwa baik PNN maupun BP memiliki performa yang semakin menurun jika Data *Test* memiliki tingkat iluminasi yang semakin berbeda dengan Data *Train*, walaupun telah menggunakan metode NI.

Dari Gambar 4.9 terlihat bahwa secara umum Tingkat Rekognisi PNN tetap lebih baik daripada BP. Tingkat Rekognisi rata-rata PNN Data *Train* 1 hingga 6 berkisar antara 70.9% - 78.6%, sedangkan untuk BP berkisar antara 46.81% - 57.55%. Tingkat Rekognisi rata-rata PNN Data *Train* 7 bernilai 35.12%, sedangkan untuk BP bernilai 18.57%. Performa baik metode NI tidak berlaku pada PNN Data *Train* 7 ini. Jika dilihat dari tipe Data yang digunakan, Set Data 7

memang berbeda dari Set Data yang lain. Set Data 7 diambil menggunakan pencahayaan Lampu TL dari arah atas kepala sedangkan database yang lain menggunakan Lampu Pijar dari arah frontal wajah. Perbedaan arah penyinaran berdampak pada perbedaan variasi iluminasi pada wajah sehingga pada percobaan berikutnya akan ditambahkan metode KI untuk menghilangkan variasi iluminasi pada citra.



Gambar 4.9. Tingkat Rekognisi Data *Test* rata-rata PNN dan BP oleh masing-masing Data *Train* (menggunakan NI)

Dengan menggunakan Data *Test* asli Tingkat Rekognisi rata-rata ketujuh tipe Data *Train* PNN bernilai 44.21%. Metode NI secara umum memperbaiki Tingkat Rekognisi rata-rata PNN Data *Train* 1 hingga 6 menjadi 70.28% dengan peningkatan 26.07% dari percobaan tanpa NI, namun tidak berpengaruh terhadap Tingkat Rekognisi PNN Data *Train* 7 yang hanya bernilai 35.12%.

4.1.5. Hasil dan Analisa Percobaan dengan Data Ternormalisasi dan Terkompensasi Iluminasi

Percobaan kali ini menggunakan 2 jenis preprocessing data yaitu Normalisasi dan Kompensasi Iluminasi. Namun parameter frekuensi cutoff yang tepat untuk proses KI menjadi permasalahan sendiri, sehingga tiap tipe Data *Train* PNN akan diuji berulang kali dengan variasi nilai fekuensi cutoff dari 0 hingga 7, dengan alasan rentang frekuensi tersebut termasuk dalam frekuensi rendah. Nilai frekuensi cutoff yang tepat dinilai dari Tingkat Rekognisi tertinggi oleh PNN.

Secara teori variasi iluminasi citra terletak pada frekuensi rendah dan reflektansi citra terletak pada frekuensi tinggi. Penghapusan frekuensi dimana variasi iluminasi berada tentu akan meningkatkan Rekognisi PNN karena seluruh citra akan berada pada kondisi iluminasi yang sama yang akan mempermudah PNN mengenali identitas objek. Sebaliknya penghapusan frekuensi dimana reflektansi berada akan menurunkan Tingkat Rekognisi PNN karena sebagian pola wajah akan hilang. Tidak ada nilai frekuensi yang terdefinisi umum yang menjadi batasan frekuensi iluminasi dengan frekuensi reflektansi. Percobaan ini hanya mengasumsikan bahwa batas tersebut berada antara frekuensi 0 dan 7, dan mencarinya secara statistik dengan cara menguji citra hasil kompensasi tersebut langsung ke ketujuh tipe Data *Train* PNN. Frekuensi cutoff yang dianggap tepat adalah yang memiliki Tingkat Rekognisi tertinggi oleh PNN, sehingga Tingkat Rekognisi ketujuh tipe Data *Train* PNN tersebut akan dirata-ratakan berdasarkan frekuensi cutoff yang sama.

Dari Gambar 4.10 terlihat bahwa walaupun telah diterapkan metode KI namun variasi iluminasi masih terlihat pada Set Data 7, salah satunya adalah lingkaran gelap di sekitar mata baik pada frekuensi cutoff 0 hingga 7.

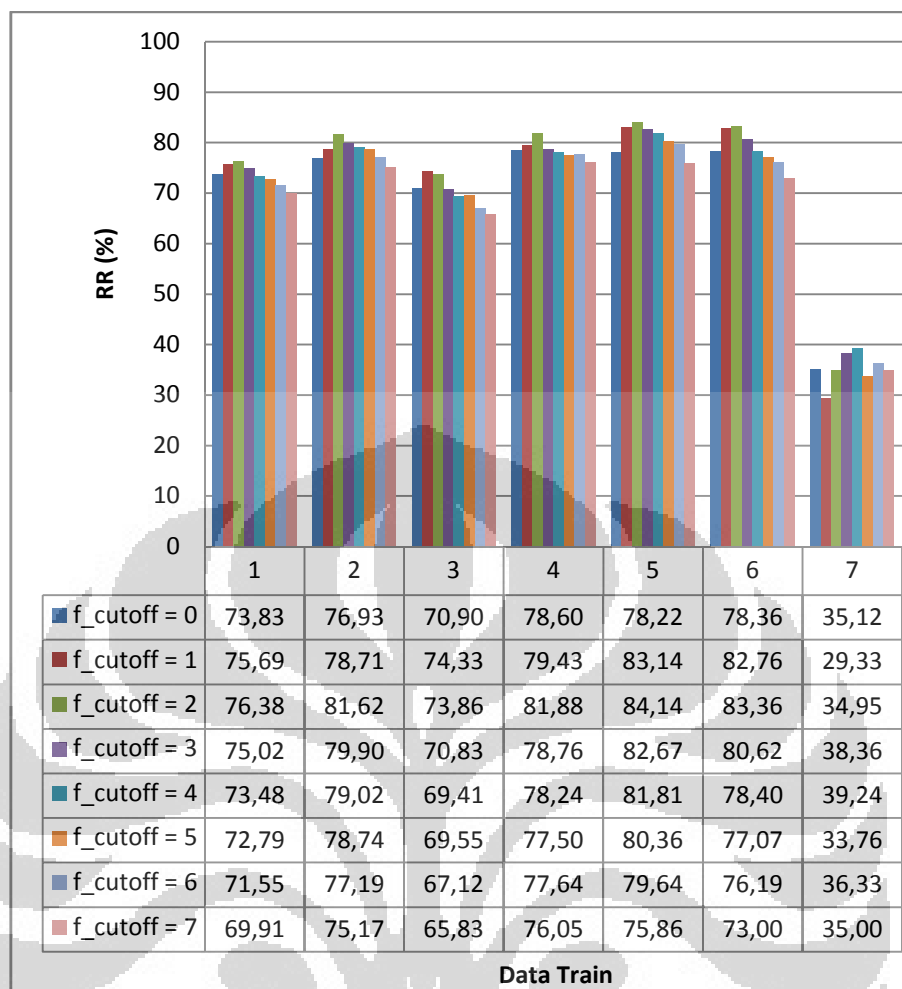




Gambar 4.10. Contoh perubahan frekuensi cutoff pada Set Data 0 (baris ke-1), Set Data 3 (baris ke-2) dan Set Data 6 (baris ke-2)

Gambar 4.11 menunjukkan Tingkat Rekognisi ketujuh tipe Data *Train* PNN terhadap perubahan nilai frekuensi cutoff saat proses KI. Tingkat Rekognisi setiap tipe Data *Train* PNN memiliki nilai yang fluktuatif yang disebabkan karena penghapusan sejumlah frekuensi citra. Frekuensi cutoff 0 sama dengan proses NI biasa karena koefisien DCT yang diubah hanya frekuensi 0, dan akan dijadikan perbandingan dengan frekuensi cutoff yang lain. Untuk rentang nilai frekuensi cutoff 0 hingga 2, Tingkat Rekognisi setiap tipe Data *Train* PNN umumnya naik 2%-5%, kecuali Data *Train* 7. Namun saat frekuensi cutoff lebih besar dari 2 Tingkat Rekognisinya secara umum turun, kecuali Data *Train* 7. Perbedaan pola Tingkat Rekognisi PNN Data *Train* 7 dengan yang lain disebabkan karena Set Data 7 memiliki perlakuan berbeda saat pengambilan data, yaitu arah pencahayaan datang dari atas kepala. Oleh karena itu frekuensi cutoff yang tepat untuk PNN Data *Train* 7 pun berbeda, yaitu frekuensi 4. Namun ternyata penambahan metode KI juga tidak memperbaiki Tingkat Rekognisi PNN Data *Train* 7 secara signifikan, dengan nilai maksimum hanya 39.24% pada frekuensi cutoff 4. Bahkan pada pada frekuensi cutoff 5 Tingkat Rekognisinya kembali menurun.

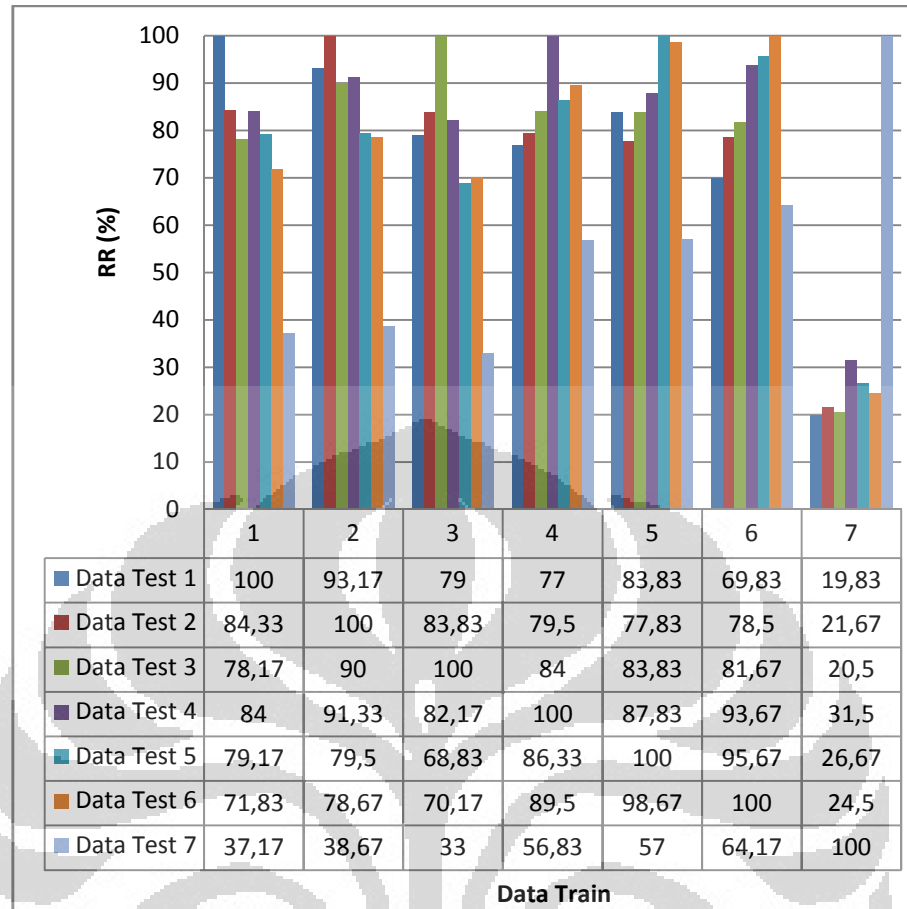
Variasi iluminasi pada Set Data 7 tidak seluruhnya hilang pada proses KI hingga frekuensi cutoff 5, namun sebaliknya yang terjadi adalah Tingkat Rekognisi menurun yang menandakan ada nilai reflektansi yang terhapus dan jika frekuensi cutoff semakin ditingkatkan maka lebih banyak reflektansi yang akan terhapus. Hasil percobaan ini membuktikan bahwa sebenarnya tidak ada nilai batas frekuensi yang jelas antara frekuensi iluminasi dengan frekuensi reflektansi. Kompensasi Iluminasi menggunakan metode DCT ini hanyalah sebuah pendekatan yang mengasumsikan bahwa variasi iluminasi terletak pada frekuensi rendah, namun pada kenyataannya variasi iluminasi mungkin saja terletak pada frekuensi tinggi dimana reflektansi juga berada.



Gambar 4.11. Tingkat Rekognisi Data *Test* rata-rata PNN oleh masing-masing Data *Train* (menggunakan NI+KI) berdasarkan frekuensi cutoff yang digunakan

Jika ketujuh tipe Data *Train* PNN tersebut dirata-ratakan berdasarkan frekuensi cutoff yang sama, Tingkat Rekognisi rata-rata maksimum adalah 73.74% yaitu pada nilai frekuensi cutoff 2. Oleh karena itu frekuensi cutoff yang akan digunakan adalah 2.

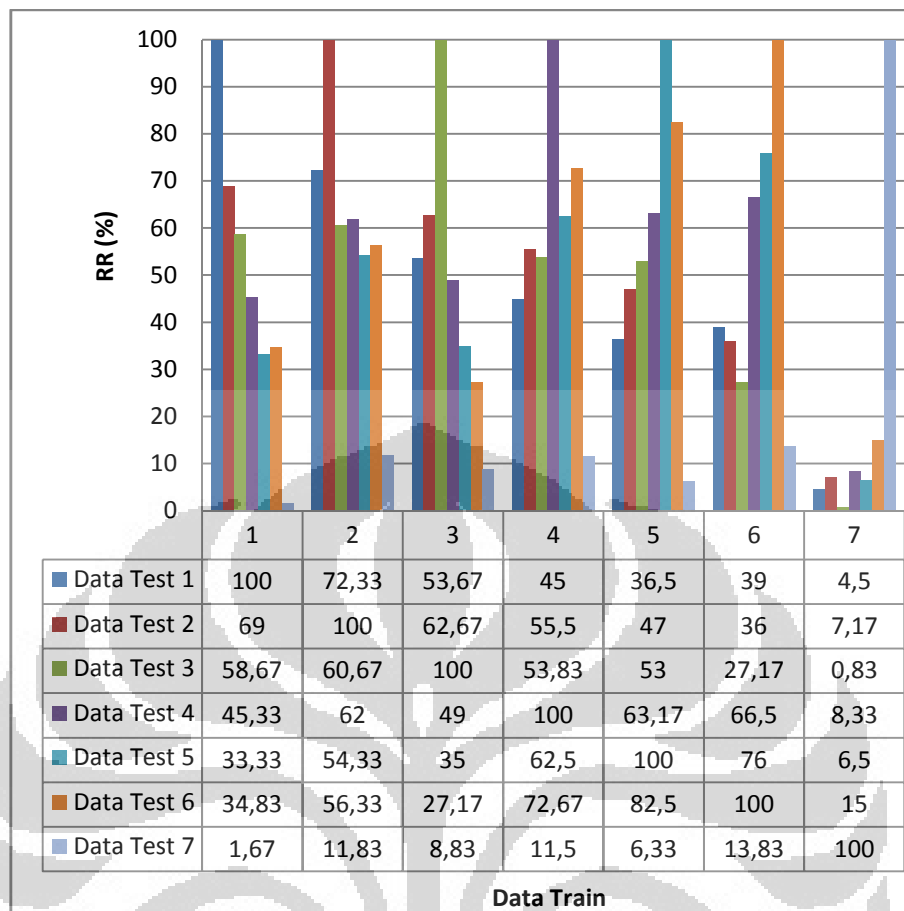
Dari Gambar 4.12 terlihat bahwa hasil percobaan PNN ini tidak banyak mengalami peningkatan Rekognisi untuk semua tipe Data *Train* PNN dibandingkan percobaan sebelumnya. Tingkat Rekognisi masing-masing Data *Train* PNN terhadap berbagai tipe Data *Test* masih memiliki pola yang sama dengan percobaan sebelumnya, yaitu semakin menurun jika perbedaan tingkat iluminasi Data *Test* semakin berbeda dengan Data *Train* yang digunakan.



Gambar 4.12. Tingkat Rekognisi Data *Test* PNN oleh masing-masing Data *Train* (menggunakan NI+KI dengan frekuensi cutoff 2)

Dari Gambar 4.13 terlihat bahwa hasil percobaan BP ini memiliki pola yang sama dengan PNN. Sehingga dapat disimpulkan bahwa metode NI + KI tidak dapat menghilangkan penurunan Tingkat Rekognisi PNN maupun BP terhadap perbedaan tingkat iluminasi antara Data *Test* dengan Data *Train*.

Dari Gambar 4.14 terlihat bahwa Tingkat Rekognisi rata-rata PNN Data *Train* 1 hingga 6 dengan NI+KI berkisar antara 73.86% - 84.14%, sedangkan untuk BP berkisar antara 48.98% - 59.64%. Tingkat Rekognisi rata-rata PNN Data *Train* 7 bernilai 34.95%, sedangkan untuk BP bernilai 20.33%. Seperti percobaan-percobaan sebelumnya Tingkat Rekognisi PNN pada percobaan ini juga selalu lebih baik dari BP untuk tipe Data *Train* manapun. Hal ini semakin membuktikan bahwa PNN jauh lebih baik dari BP dalam klasifikasi.

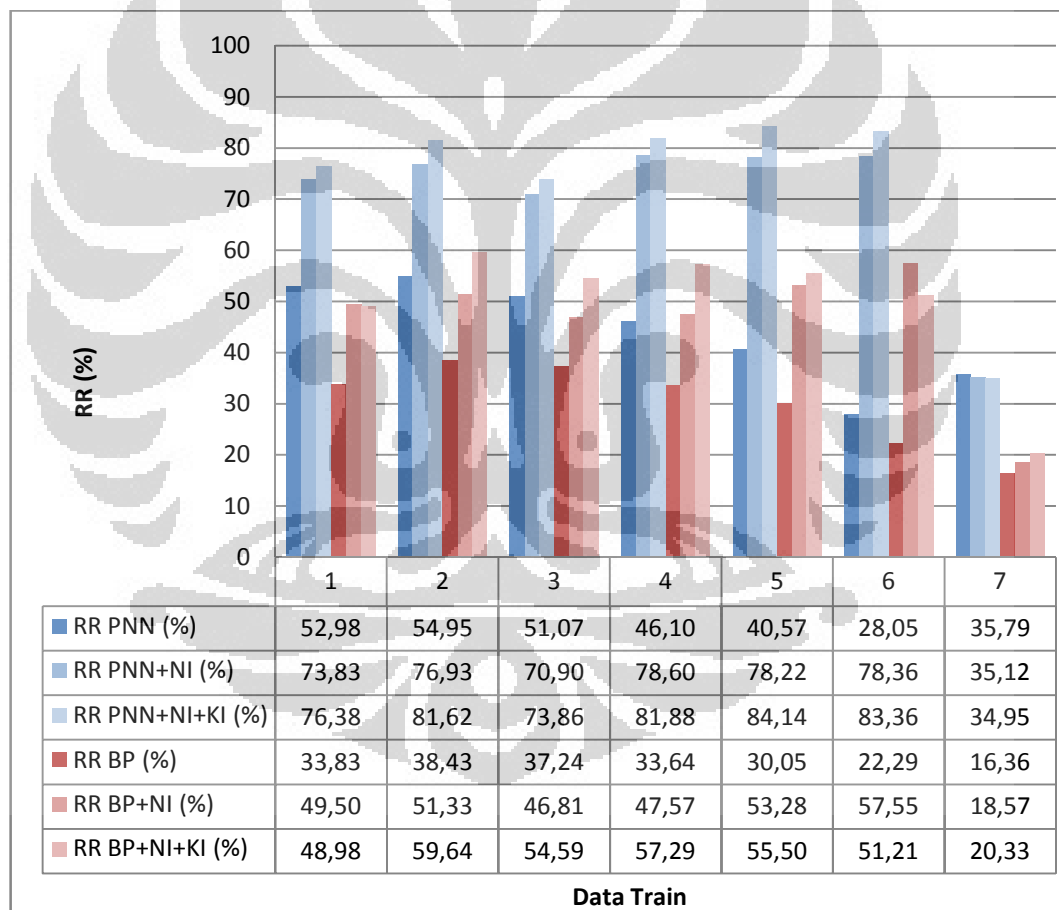


Gambar 4.13. Tingkat Rekognisi Data *Test* BP oleh masing-masing Data *Train* (menggunakan NI+KI dengan frekuensi cutoff 2)

Gabungan metode NI dan KI secara umum memperbaiki Tingkat Rekognisi rata-rata PNN Data *Train* 1 hingga 6 menjadi 73.74% dengan peningkatan 3.46% dari percobaan hanya menggunakan NI, namun tetap tidak berpengaruh terhadap Tingkat Rekognisi PNN Data *Train* 7 yang hanya bernilai 34.95%.

Hasil dari ketiga percobaan menunjukkan bahwa PNN Data *Train* 7 tidak pernah memiliki Rekognisi yang baik. Hal ini disebabkan oleh variasi iluminasi yang tidak sepenuhnya hilang walaupun telah menggunakan metode gabungan NI+KI karena ada beberapa variasi iluminasi yang berada pada frekuensi tinggi yang tidak bisa dihapus tanpa mengubah reflektansi gambar. Selain variasi iluminasi, perbedaan yang paling signifikan antara Set Data 7 dengan lainnya adalah tingkat kekontrasan gambar. Set Data 7 memiliki tingkat kekontrasan paling tinggi. Perbedaan inilah yang paling menyebabkan Tingkat Rekognisi PNN Data *Train* 7 yang rendah. Tingkat Rekognisi PNN Data *Train* 7 yang rendah

bukan mayoritas disebabkan oleh perbedaan sudut arah datang cahaya, namun disebabkan karena perbedaan cahaya yang digunakan. Set Data 7 menggunakan Lampu TL yang berwarna putih sedangkan Set Data 2 hingga 6 menggunakan Lampu Pijar yang berwarna kekuningan. Lampu TL memiliki warna cahaya yang lebih terang dari Lampu Pijar untuk intensitas yang sama. Perbedaan warna cahaya mengakibatkan perbedaan tingkat kekontrasan wajah pada citra yang tercuplik. Tabel 3.1 pada bab 3 menunjukkan bahwa pada Set Data 7 pola wajah terlihat sangat jelas pada citra RGB, grayscale maupun hasil NI. Sedangkan Set Data 1 hingga 6 terlihat lebih buram. Perbedaan ini disebabkan oleh perbedaan warna cahaya yang digunakan, sehingga mempengaruhi performa PNN.



Gambar 4.14. Tingkat Rekognisi Data *Test* rata-rata PNN dan BP oleh masing-masing Data *Train* (menggunakan NI+KI dengan frekuensi cutoff 2)

Dari Gambar 4.14, yang menunjukkan komparasi Tingkat Rekognisi ketiga percobaan tersebut, dapat disimpulkan bahwa metode NI sukses dalam

memperbaiki Tingkat Rekognisi PNN terhadap data wajah dengan tingkat iluminasi bervariasi namun tetap menggunakan spektrum cahaya yang sama, dengan peningkatan rata-rata 26.07%. Hal ini disebabkan proses NI menyeragamkan tingkat iluminasi seluruh citra. Sedangkan metode NI+KI juga memperbaiki Tingkat Rekognisi PNN untuk data wajah yang menggunakan spektrum cahaya yang sama walaupun data wajah menerima pencahayaan dari sudut frontal dan peningkatan rata-ratanya adalah 29.53%. Hal ini disebabkan proses KI dapat menghilangkan sebagian variasi iluminasi pada citra. Walaupun peningkatan yang diberikan KI tidak terlalu signifikan, namun metode KI dapat dianggap sebagai pelengkap metode NI sehingga akan tetap digunakan pada percobaan selanjutnya.

4.2. Pengurutan Tingkat Kepentingan Set Data

Untuk membangun OPNN dengan Data *Train* bertingkat iluminasi campuran maka pertama perlu diketahui Set Data manakah yang paling penting untuk dijadikan Data *Train*. Dari percobaan pada subbab 4.3 telah didapatkan Tingkat Rekognisi PNN masing-masing tipe Data *Train* untuk mengklasifikasi seluruh Data *Test* yang ada dan akan disusun dalam sebuah Matriks *RR*.

$$RR = \begin{bmatrix} RR_{SD1}(SD1) & RR_{SD2}(SD1) & \dots & RR_{SD7}(SD1) \\ RR_{SD1}(SD2) & RR_{SD2}(SD2) & \dots & RR_{SD7}(SD2) \\ \vdots & \vdots & \ddots & \vdots \\ RR_{SD1}(SD7) & RR_{SD2}(SD7) & \dots & RR_{SD7}(SD7) \end{bmatrix} \quad (4.1)$$

Dimana $RR_{SD_i}(SD_j)$ adalah Tingkat Rekognisi PNN menggunakan Set Data ke-*i* sebagai Data *Train* dalam mengklasifikasi Set Data ke-*j* sebagai Data *Test*.

$$RR = \begin{bmatrix} 100 & 93.17 & 79 & 77 & 83.83 & 69.83 & 19.83 \\ 84.33 & 100 & 83.83 & 79.5 & 77.83 & 78.5 & 21.67 \\ 78.17 & 90 & 100 & 84 & 83.83 & 81.67 & 20.5 \\ 84 & 91.33 & 82.17 & 100 & 87.83 & 93.67 & 31.5 \\ 79.17 & 79.5 & 68.83 & 86.33 & 100 & 95.67 & 26.67 \\ 71.83 & 78.67 & 70.17 & 89.5 & 98.67 & 100 & 24.5 \\ 37.17 & 38.67 & 33 & 56.83 & 57 & 64.17 & 100 \end{bmatrix} \% \quad (4.2)$$

Kali ini Algoritma Ortogonal akan kembali digunakan untuk mendekomposisi matriks *RR* menjadi kombinasi linier dari basis-basis ortogonal, yang pada

akhirnya akan digunakan untuk mengurutkan seluruh Set Data berdasarkan tingkat kepentingannya.

$$RR = QR = [Q_1, Q_2, \dots, Q_7]R \quad (4.3)$$

Proses Gram-Schmidt yang dilakukan sama dengan yang diterapkan pada subbab 2.6 dalam mengurutkan tingkat kepentingan neuron PNN. Pada akhirnya didapat urutan Set Data berdasarkan tingkat kepentingannya yaitu Set Data 5,7,3,1,4,2,6.

Set Data terpenting adalah Set Data 5 sehingga PNN dengan Set Data tingkat iluminasi menengah memiliki tingkat adaptasi yang tertinggi terhadap Set Data tingkat iluminasi seluruhnya. Sedangkan Set Data terpenting ke-2 adalah Set Data 7 karena memiliki Tingkat Rekognisi terendah terhadap Set Data lainnya sehingga penting untuk dijadikan Data *Train*. Urutan kepentingan seluruh Set Data berpola selang-seling karena semakin berbeda tingkat iluminasi antar Set Data maka semakin penting Set Data tersebut untuk dijadikan Data *Train*.

4.3. OPNN menggunakan Data *Train* bertingkat iluminasi campuran

4.3.1. Tujuan Percobaan

Percobaan ini bertujuan sebagai berikut:

- Melihat pengaruh dari Data *Train* bertingkat iluminasi campuran terhadap performa OPNN.
- Membandingkan performa OPNN untuk semua kombinasi Data *Train*.
- Membandingkan performa OPNN dengan PNN Standar dan BP pada percobaan ini.

4.3.2. Skema Percobaan

Setiap Set Data akan dibagi 2 yaitu 50% pertama untuk Data *Train* dan 50% sisanya untuk Data *Test*. Ada 7 konfigurasi Data *Train* yang akan menjadi neuron PNN, masing-masing merupakan kombinasi dari 50% tiap Set Data dengan tingkat kepentingan tertinggi hingga terendah.

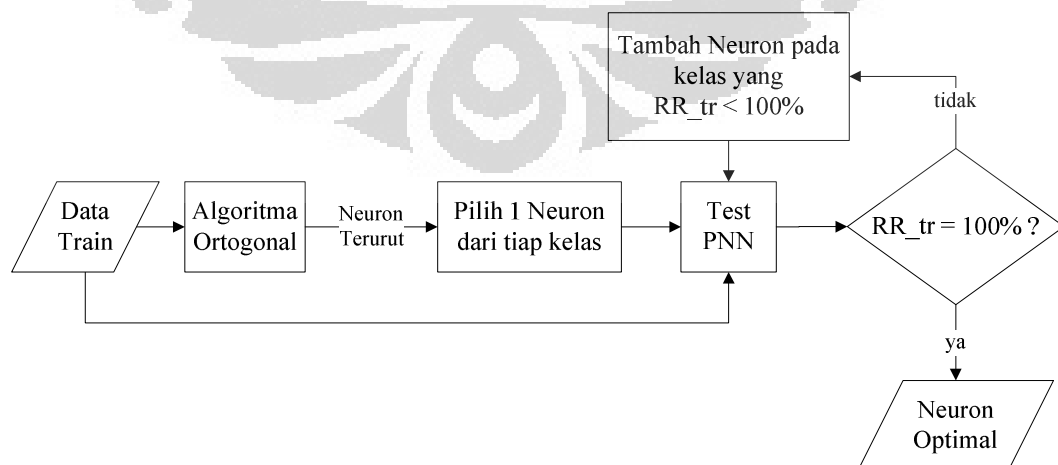
Dengan perbedaan variasi Data *Train* tersebut PNN akan diuji untuk merekognisi seluruh Data *Test* yang merupakan sisa 50% dari seluruh Set Data. Sedangkan preprocessing data yang akan digunakan adalah NI dan KI karena

kedua metode tersebut telah terbukti dapat memperbaiki tingkat Rekognisi pada percobaan sebelumnya. Percobaan ini akan menggunakan metode PNN, OPNN dan BP untuk dibandingkan. PNN dan OPNN akan menggunakan smoothing parameter bernilai 5. Berikut adalah rangkuman konfigurasi data yang digunakan untuk ketiga metode tersebut :

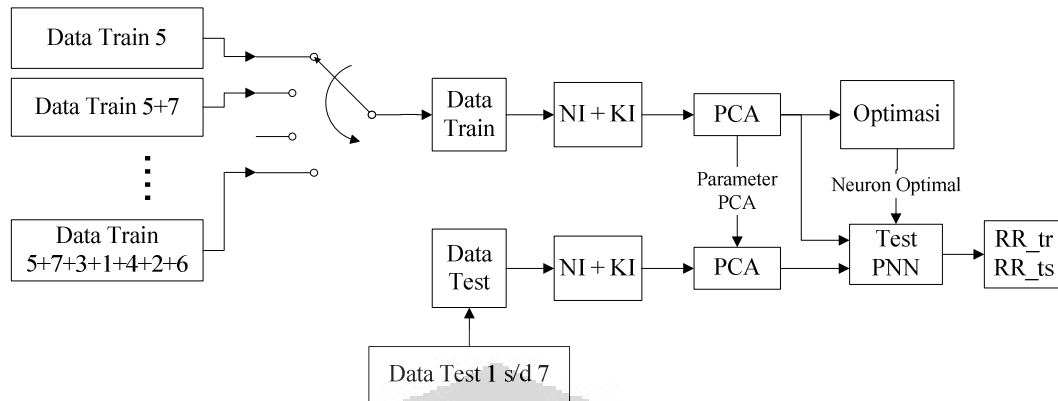
- *Preprocessing* data : Normalisasi dan Kompensasi Iluminasi
- $\text{Log } \mu$: 150
- Frekuensi cutoff : 2
- PCA : 30

Tabel 4.1. Konfigurasi Data *Train*

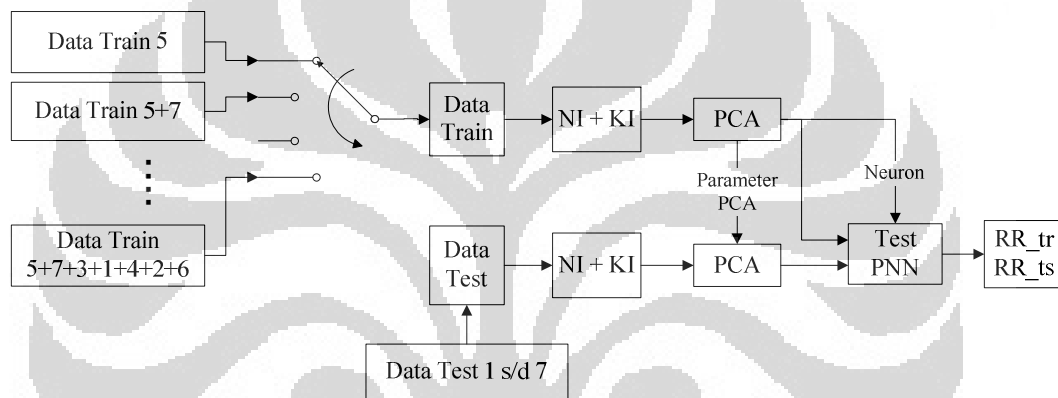
		Set Data							Total
		5	7	3	1	4	2	6	
Data Train	A	300	-	-	-	-	-	-	300
	B	300	300	-	-	-	-	-	600
	C	300	300	300	-	-	-	-	900
	D	300	300	300	300	-	-	-	1200
	E	300	300	300	300	300	-	-	1500
	F	300	300	300	300	300	300	-	1800
	G	300	300	300	300	300	300	300	2100



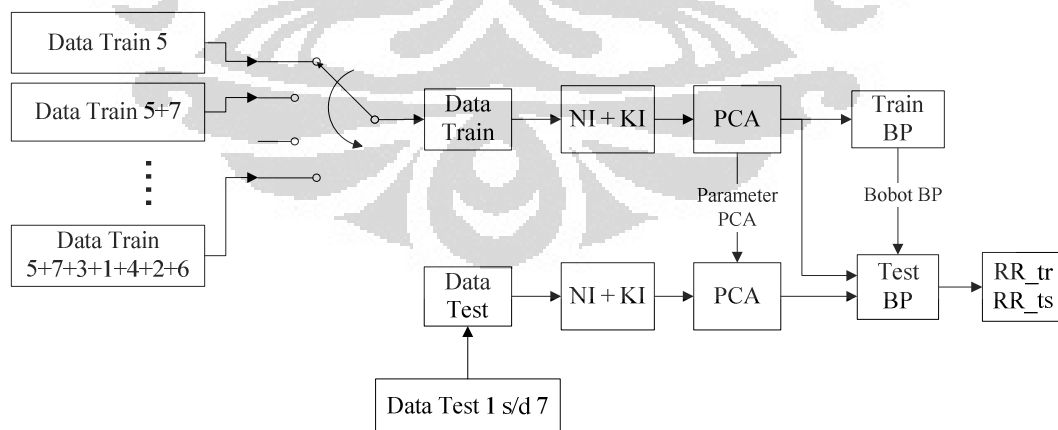
Gambar 4.15. Prosedur Optimasi PNN



Gambar 4.16. Skema Percobaan OPNN



Gambar 4.17. Skema Percobaan PNN Standar



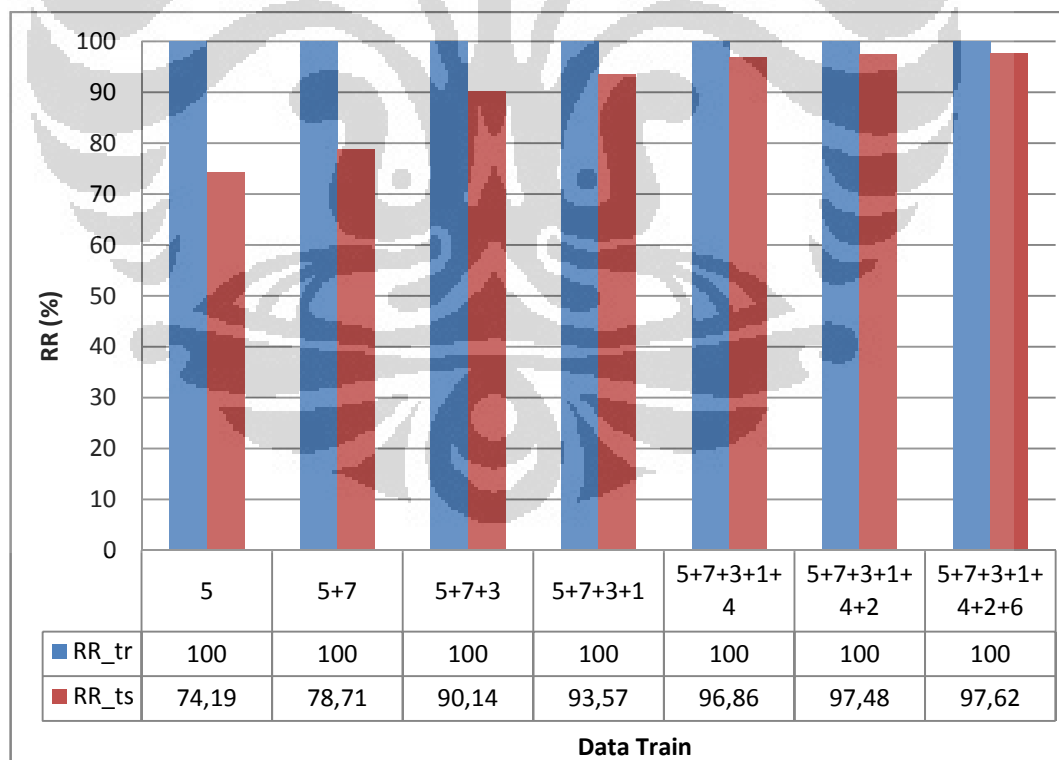
Gambar 4.18. Skema Percobaan BP

Gambar 4.16., Gambar 4.17. dan Gambar 4.18. masing-masing merupakan skema percobaan OPNN, PNN Standar dan BP. Untuk PNN Standar dan OPNN nilai smoothing parameter yang digunakan adalah 5. PNN menggunakan neuron maksimum, yaitu sejumlah *Data Train* yang digunakan, sedangkan OPNN menggunakan neuron optimal yang dicari menggunakan prosedur dalam Gambar 4.15.

Berikut adalah konfigurasi BP yang digunakan:

- Metode : *Backpropagation*
- Jumlah Neuron Input : 30
- Jumlah Neuron *Hidden* : 15
- Fungsi Error : Kuadratis
- Laju Pembelajaran : 0.2
- Batas pembelajaran : epoch max 10000 atau error total min 0.01
- Inisialisasi Bobot : Nguyen Widrow

4.3.3. Hasil dan Analisa Percobaan OPNN



Gambar 4.19. Tingkat Rekognisi *Data Train* dan *Data Test* rata-rata OPNN oleh masing-masing *Data Train*

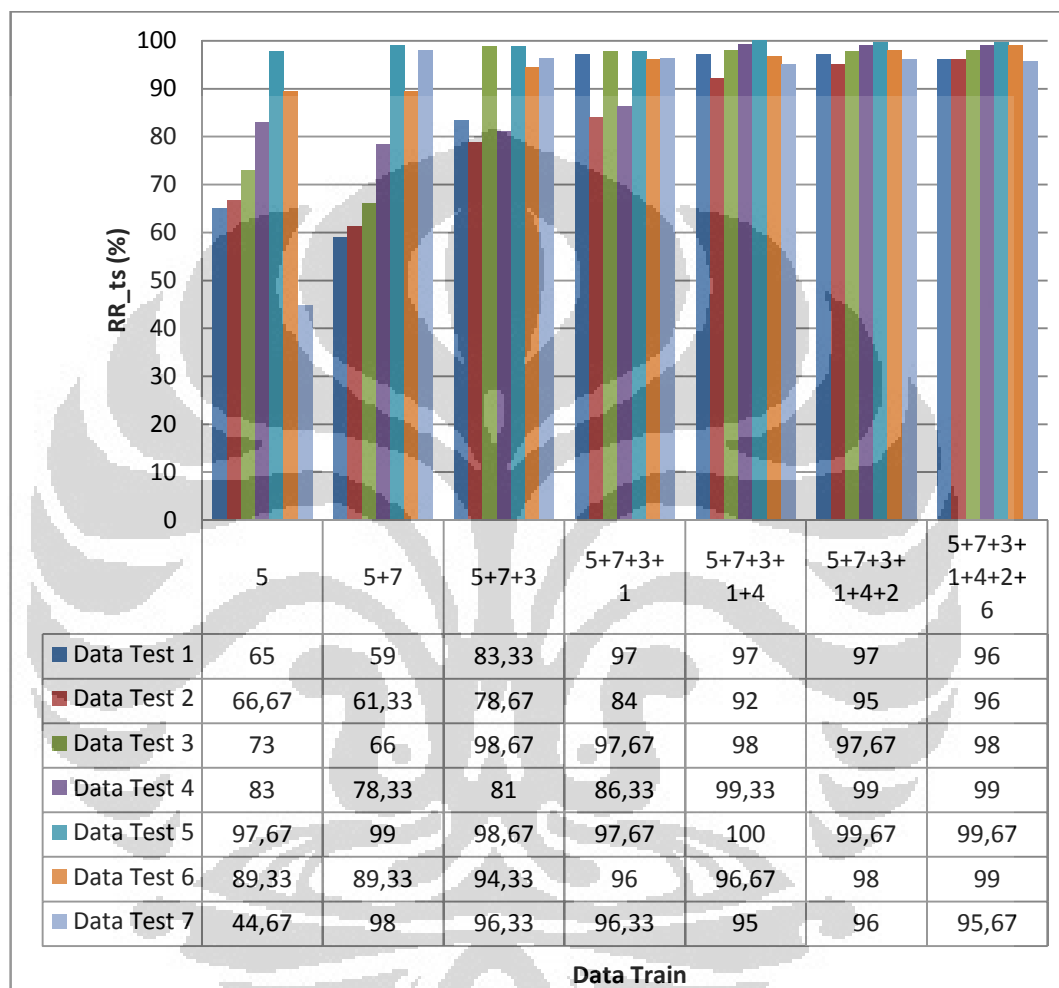
Dari Gambar 4.19. terlihat bahwa Tingkat Rekognisi Data *Train* OPNN selalu bernilai 100%. Hal ini menunjukkan bahwa OPNN tidak memiliki masalah jika Data *Train* berisi citra dengan tingkat iluminasi yang berbeda-beda. Dari Gambar 4.19. juga terlihat perbandingan antara OPNN Data *Train* sejenis dengan OPNN Data *Train* campuran. Jika Data *Train* hanya berisi 1 dari 7 jenis tingkat iluminasi yang menjadi Data *Test* maka Tingkat Rekognisi OPNN akan bernilai sekitar 74.19%, sedangkan jika OPNN memiliki Data *Train* dengan seluruh jenis tingkat iluminasi yang ada pada Data *Test* maka Tingkat Rekognisi OPNN akan bernilai sekitar 97.62%. Dengan demikian dapat disimpulkan bahwa semakin banyak variasi tingkat iluminasi Data *Train* maka Tingkat Rekognisi OPNN akan lebih baik.

Dari Gambar 4.20 terlihat pengaruh kombinasi Data *Train* terhadap Tingkat Rekognisi Data *Test* OPNN. OPNN Data *Train* 5 memiliki Tingkat Rekognisi Data *Test* terkecil, sedangkan OPNN Data *Train* gabungan 5+7 dapat memperbaiki Tingkat Rekognisi Data *Test* untuk Data *Test* 7 sendiri, namun dampaknya juga menurunkan Tingkat Rekognisi Data *Test* untuk Data *Test* 1 hingga 4. Hasil ini disebabkan karena Set Data 7 memiliki tingkat kontras yang berbeda dengan Set Data 1 hingga 4 sehingga keterlibatan Set Data 7 pada Data *Train* sedikit mengacaukan hasil rekognisi OPNN.

Namun kekurangan tersebut dapat diperbaiki dengan menambahkan Set Data 3 pada Data *Train*, yaitu Data *Train* 5+7+3 sehingga Tingkat Rekognisi Data *Test* secara keseluruhan membaik. Penambahan Data *Train* 3 bukan hanya memperbaiki Tingkat Rekognisi Data *Test* 3 sendiri namun juga meningkatkan Tingkat Rekognisi Data *Test* 1, 2, 4, yaitu yang memiliki tingkat iluminasi terdekat. Dan seterusnya semakin banyak tipe Data *Train* ditambahkan maka tingkat rekognisi OPNN terus bertambah.

Dari percobaan ini dapat disimpulkan bahwa Data *Train* yang terdiri dari berbagai tingkat iluminasi hampir tidak menurunkan kemampuan Rekognisi PNN, sebaliknya kondisi tersebut membuat performa PNN lebih baik. Hal ini disebabkan karena PNN menggunakan konsep probabilitas untuk menghitung kecocokan suatu Data *Test* dengan setiap neuron pada *Pattern Layer*. Derajat kecocokan (ϕ) antara Data *Test* dengan neuron dihitung menggunakan persamaan

(2.1) pada bab 2, yang membuat nilai ϕ berbanding terbalik dengan eksponensial dari selisih antara Data *Test* dengan neuron. Sehingga neuron yang berbeda jauh dengan Data *Test* akan menghasilkan ϕ yang sangat kecil, sedangkan neuron yang paling mirip dengan Data *Test* akan menghasilkan ϕ yang sangat besar.



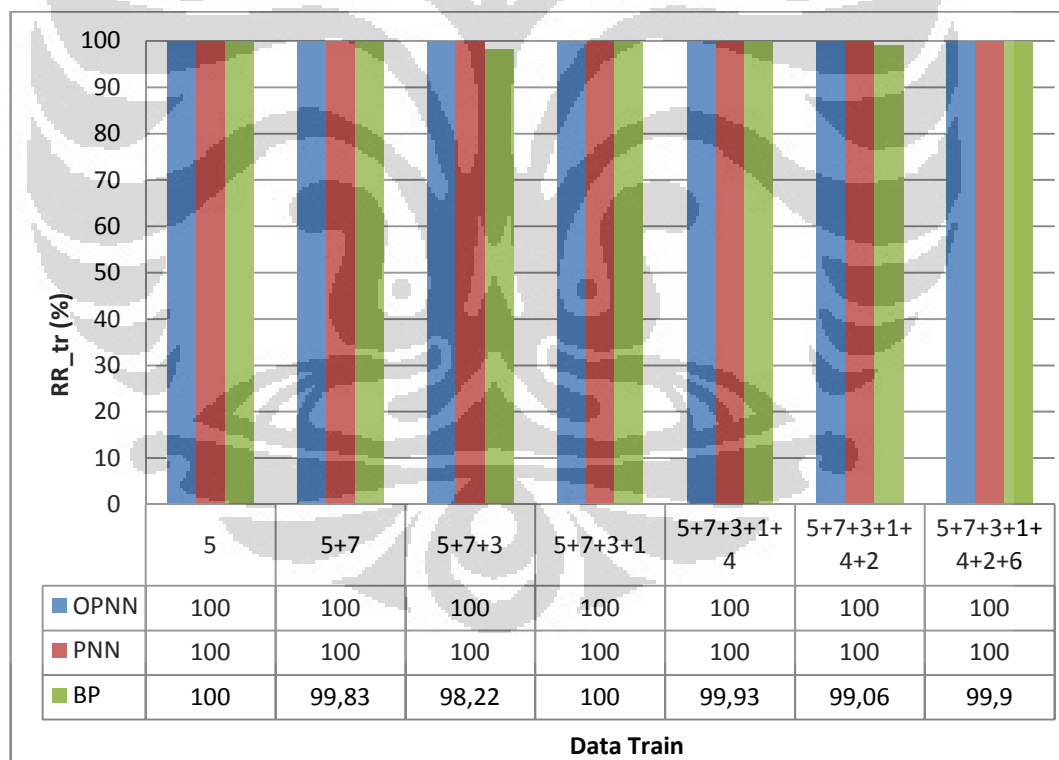
Gambar 4.20. Tingkat Rekognisi Data *Test* OPNN oleh masing-masing Data *Train*

Walaupun nilai PDF Data *Test* (p) pada *Summation Layer* didapat dari penjumlahan seluruh nilai ϕ dari tiap neuron, namun sebenarnya yang berkontribusi terbesar pada nilai PDF tersebut adalah neuron yang paling mirip dengan Data *Test*, sedangkan nilai ϕ dari neuron yang berbeda jauh dengan Data *Test* hampir tidak berkontribusi apapun karena nilai ϕ -nya yang jauh lebih kecil. Hal ini membuat neuron yang terdiri dari berbagai tingkat iluminasi Data *Train* hampir tidak menurunkan kemampuan Rekognisi PNN karena keputusan pada

sistem klasifikasi PNN paling ditentukan oleh neuron yang memiliki derajat kecocokan tertinggi terhadap Data *Test*.

4.3.4. Komparasi OPNN dengan PNN Standar dan BP

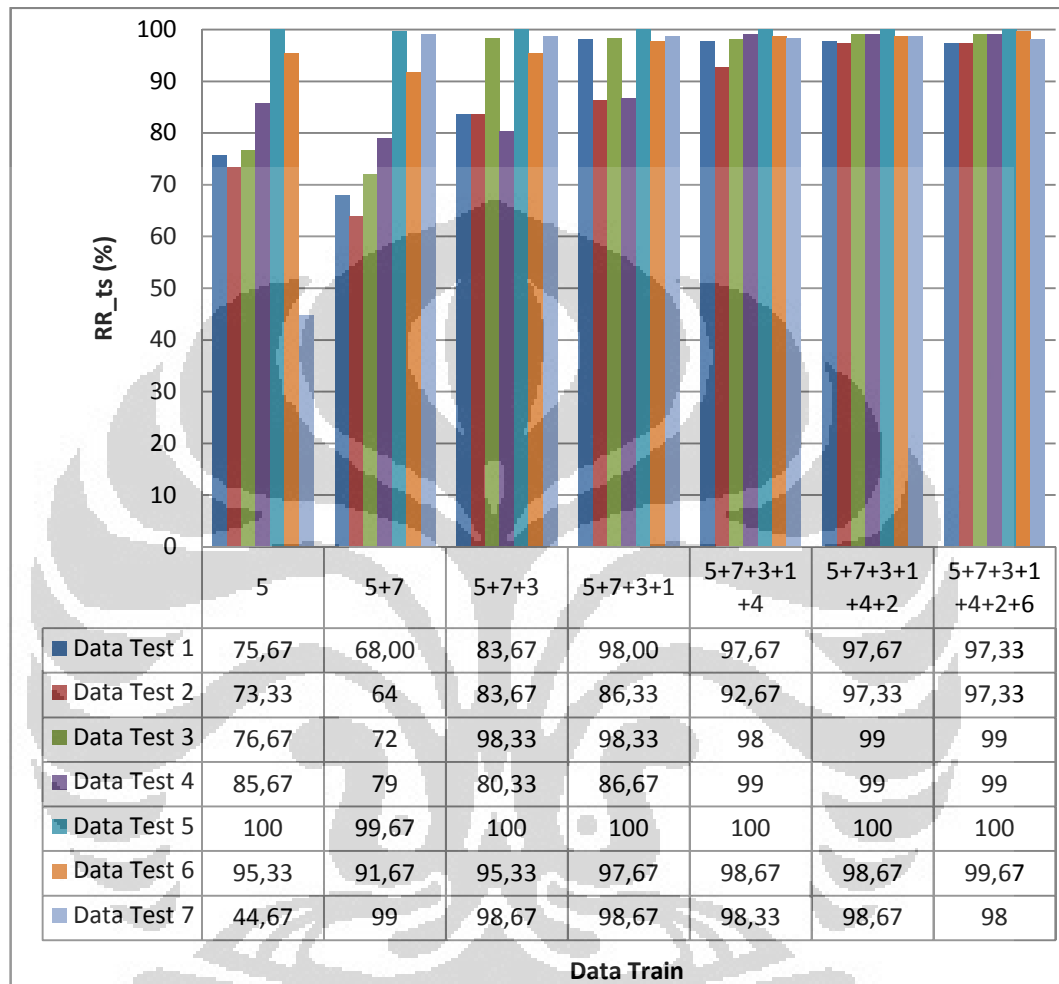
Baik PNN Standar maupun OPNN memiliki Tingkat Rekognisi Data *Train* yang sama yaitu 100% untuk seluruh tipe Data *Train*, walaupun kedua jenis PNN ini memiliki struktur neuron berbeda. Sedangkan Tingkat Rekognisi Data *Train* oleh BP tidak selalu bernilai 100%. Data *Train* 5 memiliki Tingkat Rekognisi Data *Train* 100% karena hanya berisi 1 Set Data atau 1 jenis tingkat iluminasi data, sehingga BP dapat dengan baik mempelajari Data *Train* pada saat *Training*. Namun BP secara umum tidak dapat mempelajari seluruh Data *Train* gabungan dengan baik, terlihat dari Tingkat Rekognisi Data *Train* yang tidak 100% pada beberapa Data *Train* campuran.



Gambar 4.21. Tingkat Rekognisi Data *Train* PNN, OPNN dan BP oleh masing-masing Data *Train*

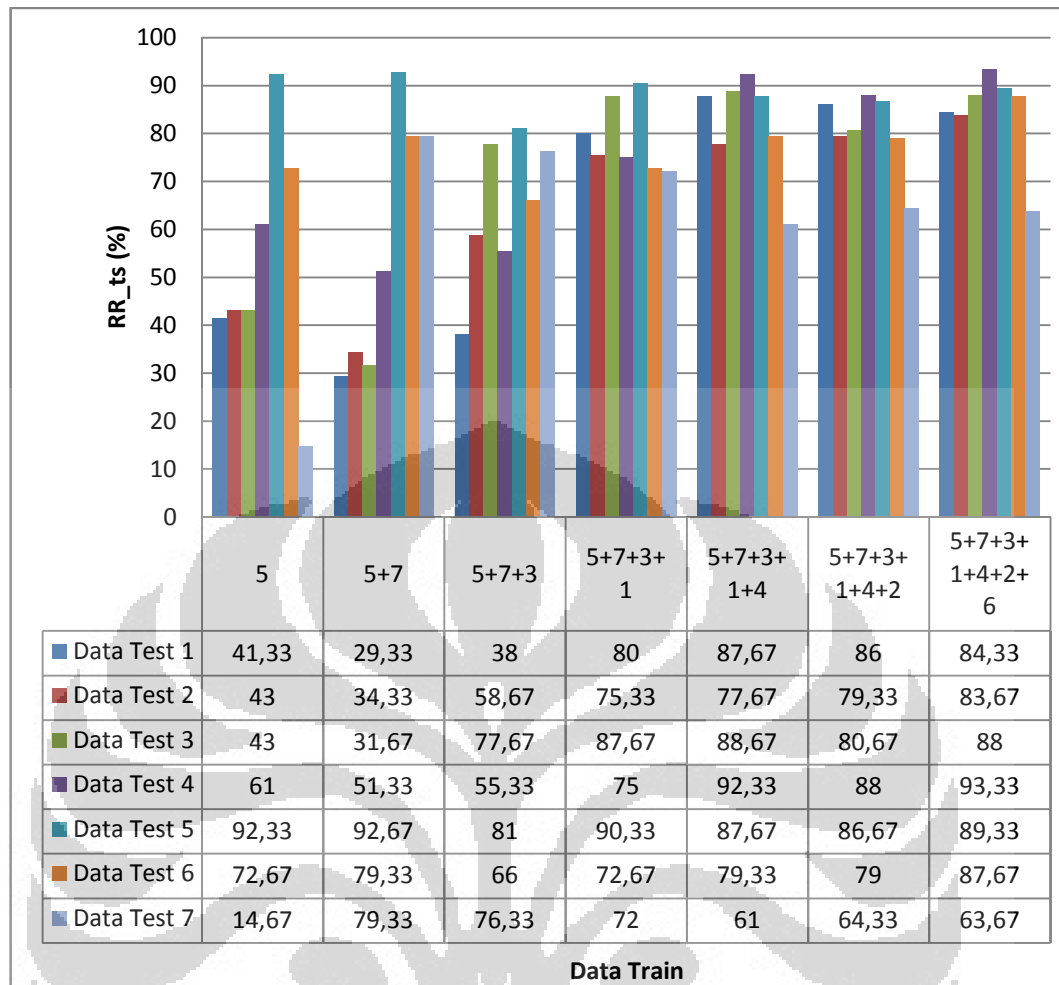
Dari Gambar 4.22 terlihat pengaruh kombinasi Data *Train* terhadap Tingkat Rekognisi Data *Test* PNN, yang sebenarnya tidak berbeda jauh, namun sedikit

lebih baik dari OPNN. Hal ini disebabkan karena struktur neuron maksimum sehingga lebih adaptif terhadap variasi Data *Test* seperti yang telah dijelaskan pada subbab 3.4.



Gambar 4.22. Tingkat Rekognisi Data *Test* PNN oleh masing-masing Data *Train*

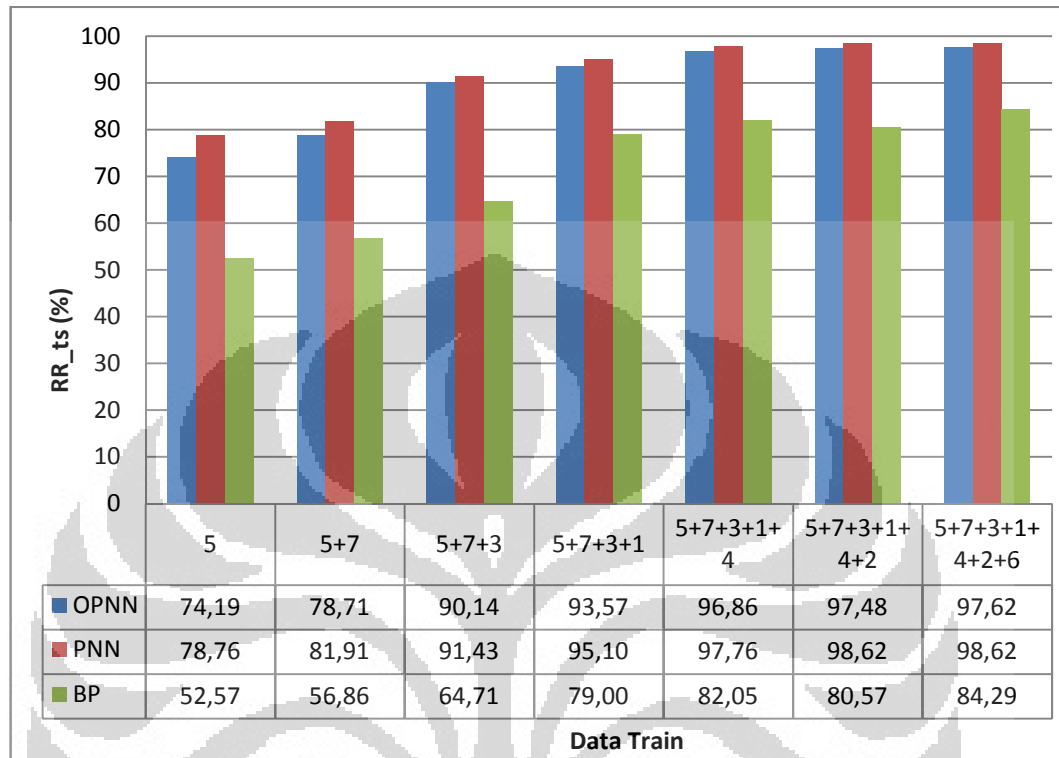
Dari Gambar 4.23 terlihat bahwa gabungan Data *Train* juga berpengaruh baik pada tingkat rekognisi BP, namun secara keseluruhan Tingkat Rekognisi Data *Test* BP lebih buruk dari PNN Standar maupun OPNN. Hal ini disebabkan pada proses *Training* BP yang tidak selesai dalam 10000 epoch. BP akan mengalami kesulitan mempelajari Data *Train* yang memiliki tingkat iluminasi yang beragam, yang ditandai dengan Tingkat Rekognisi BP terhadap Data *Train* yang tidak mencapai 100%.



Gambar 4.23. Tingkat Rekognisi Data *Test* BP oleh masing-masing Data *Train*

Dari Gambar 4.24. terlihat perbandingan Tingkat Rekognisi Data *Test* rata-rata dari ketiga metode yang digunakan dalam percobaan kombinasi Data *Train* ini, yaitu OPNN, PNN Standar dan BP. Grafik menunjukkan metode PNN Standar selalu memiliki tingkat rekognisi tertinggi dibandingkan OPNN dan BP dalam berbagai kombinasi Data *Train*. Sebenarnya perbedaan tingkat rekognisi Data *Test* OPNN dengan PNN tidak berbeda jauh, perbedaan Tingkat Rekognisi Data *Test* rata-rata antar PNN Standar dengan OPNN untuk seluruh jenis kombinasi Data *Train* adalah 1.95%. Perbedaan ini tidak lain disebabkan karena neuron optimal pada OPNN tidak cukup untuk merepresentasikan seluruh Data *Test* sehingga beberapa Data *Test* salah diklasifikasi. Sedangkan Tingkat Rekognisi Data *Test* BP jauh lebih rendah dari OPNN dengan selisih rata-rata untuk berbagai macam

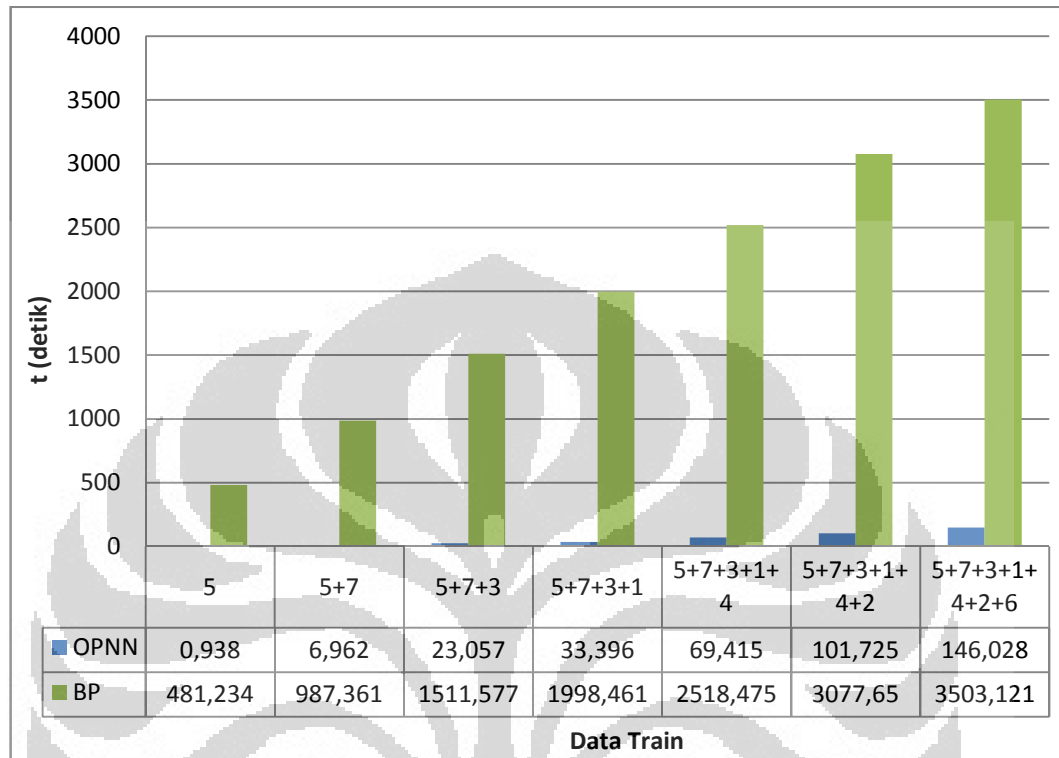
kombinasi Data *Train* sebesar 18.36%, sedangkan selisih rata-rata PNN Standar dengan BP adalah 20.31%.



Gambar 4.24. Tingkat Rekognisi Data *Test* rata-rata PNN, OPNN dan BP oleh masing-masing Data *Train*

Dari Gambar 4.25. terlihat perbandingan waktu *Training* OPNN dan BP. PNN tidak memiliki fase *Training*, sedangkan waktu *Training* OPNN jauh lebih cepat dari BP. Fase *Training* atau optimasi OPNN adalah penerapan Algoritma Ortogonal untuk mengurutkan neuron berdasarkan tingkat kepentingannya dan penalaan struktur neuron optimal. Waktu *Training* OPNN bergantung pada jumlah iterasi pada proses penalaan struktur neuron. Proses penalaan struktur neuron berhenti hingga Tingkat Rekognisi Data *Train* mencapai 100% sehingga proses tersebut sangat bergantung pada kualitas Data *Train*, akibatnya proses *Training* OPNN tiap Data *Train* akan berbeda. Gambar 4.25. juga menunjukkan bahwa semakin banyak Data *Train* maka waktu *Training* OPNN semakin lama dan kurvanya menunjukkan hubungan yang kuadratis. Hal ini disebabkan pada proses optimasi setiap Data *Train* dibandingkan dengan setiap neuron, sedangkan penambahan Data *Train* juga mengakibatkan penambahan neuron sehingga

perbandingan waktu komputasinya menjadi kuadrat dari perbandingan jumlah Data *Train* yang digunakan.

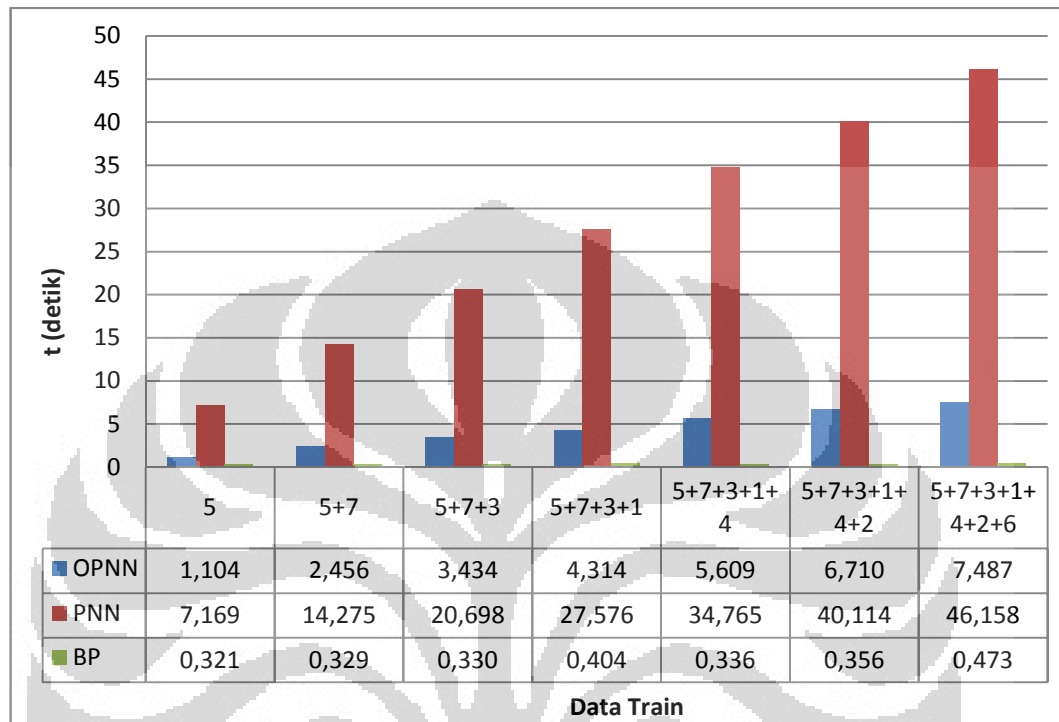


Gambar 4.25. Waktu *Training* OPNN dan BP oleh masing-masing Data *Train*

Sedangkan waktu *Training* BP menunjukkan peningkatan yang linear terhadap penambahan Data *Train*. Hal ini disebabkan BP mempelajari seluruh Data *Train* pada setiap epohnya dan jika jumlah epoh yang diperlukan konstan maka waktu *Training*-nya naik sebanding dengan penambahan Data *Train*. BP memiliki waktu *Training* yang paling lama dibandingkan dengan PNN Standar dan OPNN.

Gambar 4.26. menunjukkan bahwa waktu *Testing* PNN akan meningkat secara linear sebanding dengan penambahan Data *Train* dan begitu pula dengan OPNN. Hal ini disebabkan karena PNN Standar dan OPNN menggunakan Data *Train* sebagai neuron sehingga ukuran jaringan PNN bergantung pada jumlah Data *Train* yang digunakan dan penambahan neuron akan sebanding dengan peningkatan waktu *Testing* tiap Data *Test*. Kelebihan dari BP yaitu memiliki ukuran jaringan yang konstan dan tidak dipengaruhi oleh Data *Train*. Data *Train* hanya berpengaruh pada nilai dari bobot-bobot jaringannya, namun tidak

mengubah jumlah neuron maupun bobotnya sehingga waktu *Testing* BP akan relatif konstan untuk berapapun Data *Train* yang digunakan. BP memiliki waktu *Testing* yang tercepat dibandingkan PNN Standar dan OPNN.



Gambar 4.26. Waktu *Testing* PNN, OPNN dan BP oleh masing-masing Data *Train*

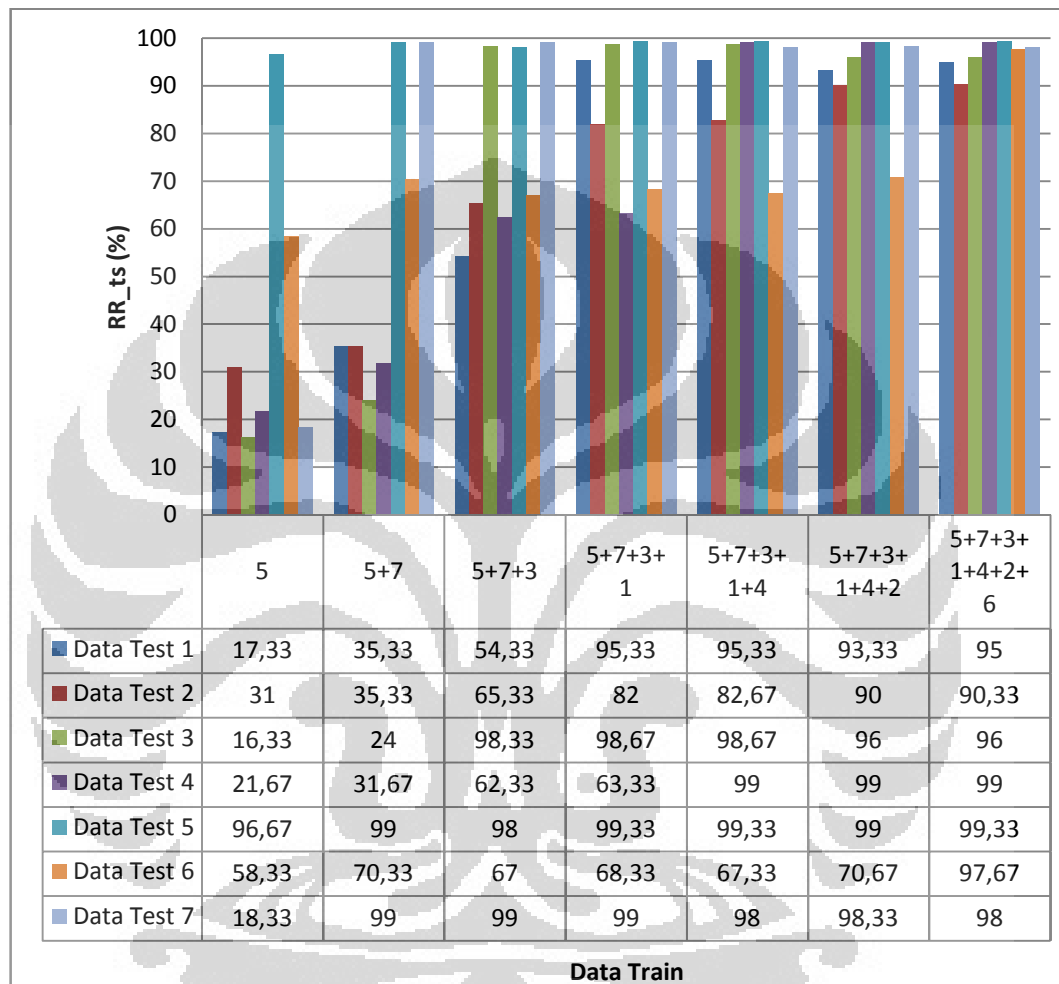
Tabel 4.2. Perbandingan jumlah neuron dan waktu *Testing* PNN dengan OPNN

Data <i>Train</i>	Jumlah Neuron			Waktu <i>Testing</i>		
	PNN	OPNN	%	PNN	OPNN	%
5	300	33	11.0%	7.169	1.104	15.4%
5+7	600	91	15.2%	14.275	2.456	17.2%
5+7+3	900	135	15.0%	20.698	3.434	16.6%
5+7+3+1	1200	172	14.3%	27.576	4.314	15.6%
5+7+3+1+4	1500	232	15.5%	34.765	5.609	16.1%
5+7+3+1+4+2	1800	282	15.7%	40.114	6.710	16.7%
5+7+3+1+4+2+6	2100	313	14.9%	46.158	7.487	16.2%
	Rata-rata		14.5%			16.3%

Tabel 4.2. menunjukkan struktur neuron OPNN optimal rata-rata dapat berkurang menjadi hanya 14.5% dari struktur neuron PNN Standar. Hal ini

menyebabkan waktu *Testing* rata-rata tiap Data *Test* pada OPNN menjadi hanya 16.3% dari PNN Standar atau 6.13 kali lebih cepat dari PNN Standar.

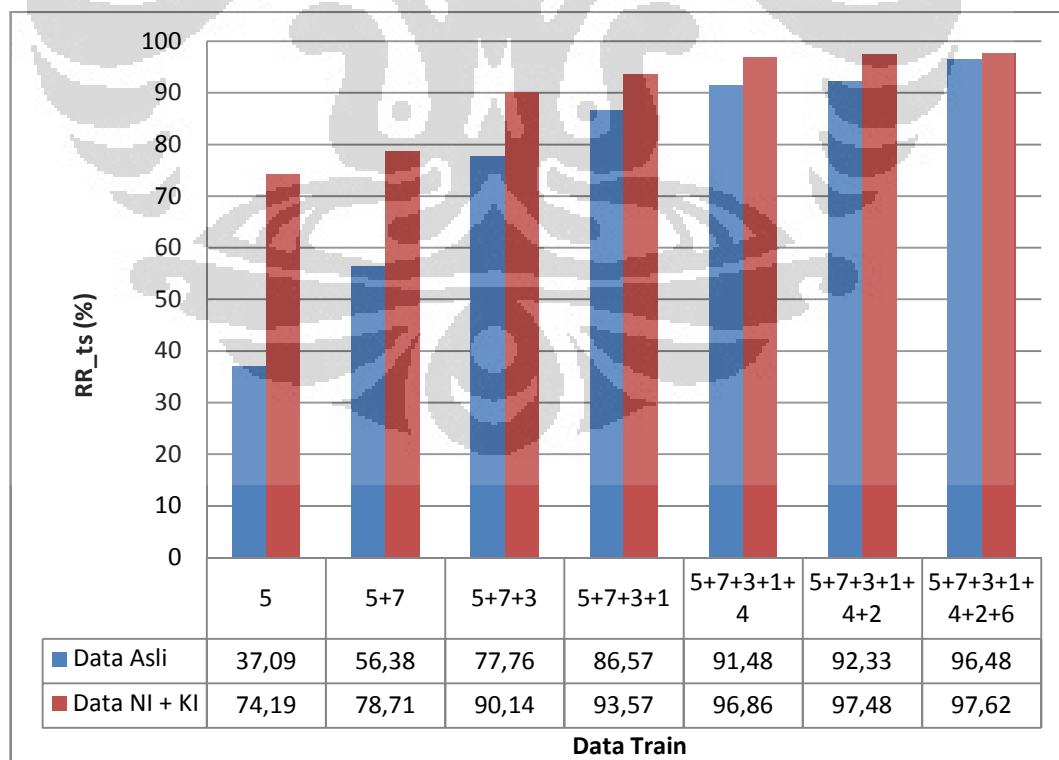
4.3.5. Komparasi Data Asli dengan Data Ternormalisasi dan Terkompensasi Iluminasi



Gambar 4.27. Tingkat Rekognisi Data *Train* dan Data *Test* OPNN oleh masing-masing Data *Train* (Data Asli tanpa Normalisasi dan Kompensasi Iluminasi)

Dari Gambar 4.27. terlihat pengaruh kombinasi Data *Train* terhadap Tingkat Rekognisi Data *Test* OPNN untuk Data Asli tanpa Normalisasi dan Kompensasi Iluminasi (NI+KI). Dari grafik tersebut terlihat bahwa setiap penambahan Data *Train* Tingkat Rekognisi OPNN akan maksimum pada tiap Data *Test* yang bertingkat iluminasi yang sama dengan Data *Train*. Namun Tingkat Rekognisi yang relatif rendah terlihat pada Data *Test* yang bertingkat iluminasi berbeda dengan Data *Train*. Sehingga penggunaan NI+KI pada data menjadi penting agar

OPNN dapat beradaptasi dengan baik pada Data Test yang bertingkat iluminasi berbeda dengan Data Train. Hal ini juga terlihat pada Gambar 4.28. yang menggambarkan perbandingan Tingkat Rekognisi Data Test rata-rata OPNN antara Data Asli dengan Data Ternormalisasi dan Terkompensasi Iluminasi. Untuk Data Asli, Tingkat Rekognisi OPNN bernilai di atas 90% jika 5 dari 7 jenis tingkat iluminasi data yang ada dijadikan Data Train. Sedangkan untuk Data menggunakan NI+KI, cukup menggunakan 3 dari 7 jenis Data Train yang ada sudah mampu mencapai Tingkat Rekognisi di atas 90%. Pada aplikasi sistem pengenalan wajah sebenarnya, untuk mendapatkan segala variasi tingkat iluminasi untuk Data Train tentu akan merepotkan. Oleh karena itu dengan menggunakan NI+KI pada data, walaupun variasi tingkat iluminasi pada Data Train hanya sedikit namun kemampuan OPNN dalam mengenali Data Test yang bertingkat iluminasi berbeda akan lebih baik, sehingga OPNN lebih mudah beradaptasi dengan segala kondisi intensitas pencahayaan, dengan syarat spektrum cahaya yang digunakan tidak berbeda jauh dengan Data Train (dalam hal ini adalah *near infrared* atau cahaya tampak).



Gambar 4.28. Tingkat Rekognisi Data *Test* rata-rata OPNN untuk Data Asli dan Data NI+KI oleh masing-masing Data *Train*

BAB 5 PENUTUP

5.1. Kesimpulan

Berikut adalah kesimpulan dari seluruh percobaan yang telah dilakukan pada penelitian ini :

1. OPNN lebih unggul dari PNN Standar pada segi struktur neuron dan waktu komputasi. Jumlah neuron optimal rata-rata pada OPNN hanya sekitar 14.5% dari jumlah neuron total sehingga waktu *Testing* rata-rata OPNN hanya sekitar 16.3% dari PNN Standar untuk jumlah Data *Test* yang sama.
2. Tingkat Rekognisi Data *Train* OPNN dapat menyamai PNN Standar, yaitu 100%.
3. Kelemahan OPNN dibandingkan PNN Standar yaitu memiliki Tingkat Rekognisi Data *Test* yang sedikit lebih buruk dari PNN Standar, dengan perbedaan sekitar 1.95%.
4. OPNN lebih unggul dari BP baik dari segi Tingkat Rekognisi Data *Train* maupun Tingkat Rekognisi Data *Test*. BP hanya memiliki Tingkat Rekognisi Data *Train* 100% untuk Data *Train* bertingkat iluminasi sejenis, sedangkan OPNN 100% baik untuk Data *Train* bertingkat iluminasi sejenis maupun campuran. Tingkat Rekognisi Data *Test* OPNN sekitar 18.36% lebih baik dari BP.
5. Waktu *Training* OPNN jauh lebih cepat dari BP.
6. Kelemahan OPNN dibandingkan BP yaitu memiliki waktu *Testing* yang lebih lama dari BP, karena waktu *Testing* OPNN berbanding lurus dengan jumlah Data *Train* yang digunakan, sedangkan waktu *Testing* BP konstan.
7. NI sukses dalam memperbaiki Tingkat Rekognisi PNN untuk data wajah dengan tingkat iluminasi beragam namun menggunakan spektrum cahaya yang sama, dengan peningkatan rata-rata 26.07%. Sedangkan NI+KI menghasilkan perbaikan Tingkat Rekognisi PNN dengan peningkatan rata-rata 29.53%.

8. NI + KI masih menyisakan perbedaan tingkat kekontrasan citra yang diakibatkan perbedaan dominasi cahaya Infra merah dan cahaya tampak, yang menyebabkan Tingkat Rekognisi OPNN semakin menurun jika perbedaan tingkat iluminasi antara Data *Train* dengan Data *Test* semakin jauh. Namun kelemahan tersebut dapat ditanggulangi dengan menggunakan Data *Train* bertingkat iluminasi campuran dan semakin banyak variasi tingkat iluminasi Data *Train* maka Tingkat Rekognisinya semakin baik.

5.2. Saran

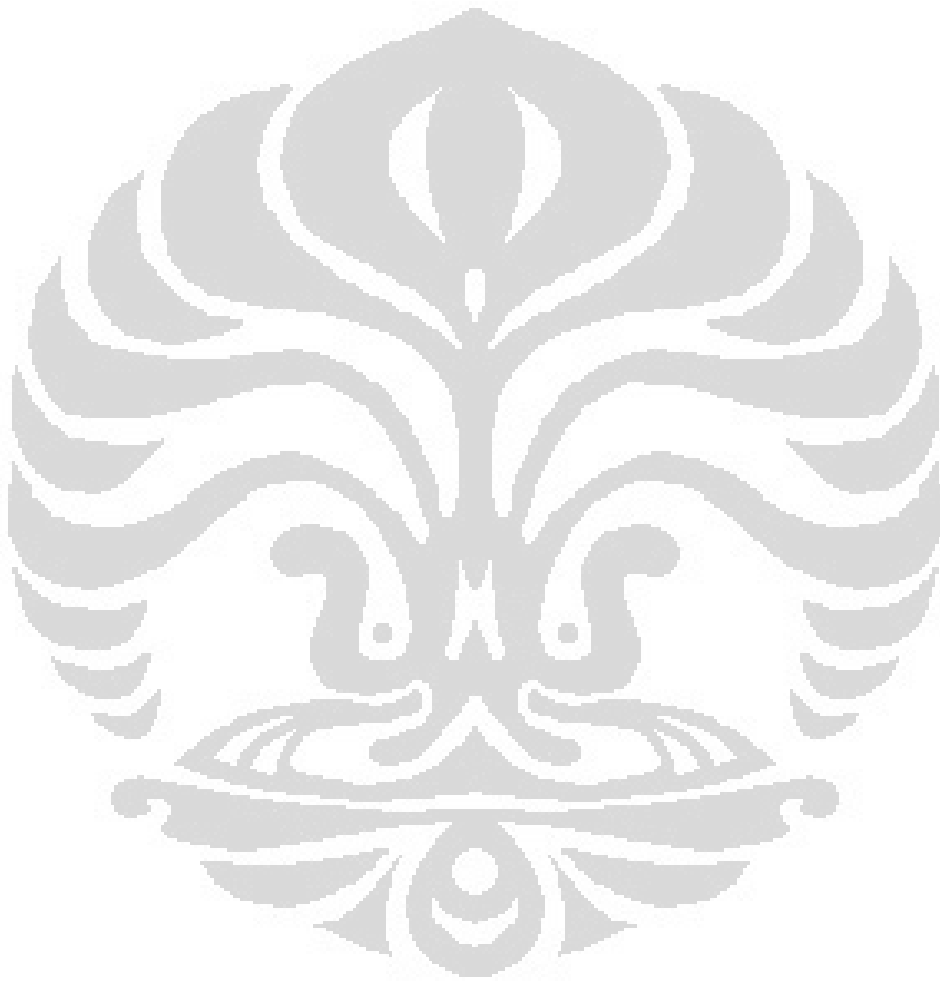
Tantangan lebih jauh yang dapat disarankan penulis untuk melanjutkan penelitian ini adalah beberapa hal berikut:

1. Penelitian metode yang dapat menghilangkan perbedaan tingkat kekontrasan citra yang diakibatkan perbedaan dominasi antara cahaya Infra merah dan cahaya tampak.
2. Penelitian menggunakan PNN ataupun OPNN untuk data wajah yang menerima pencahayaan dari sudut yang tidak frontal.
3. Untuk aplikasi kamera pengintai yang lebih *advance*, penelitian selanjutnya dapat mencoba klasifikasi data wajah dengan sudut yang tidak frontal.

DAFTAR PUSTAKA

- Anton, H. (2000). *Dasar-Dasar Aljabar Linear* (7 ed., Vol. 1). Batam: Interaksara.
- Chen, W., Er, M. J., & Wu, S. (2006). Illumination Compensation and Normalization for Robust Face Recognition Using Discrete Cosine Transform in Logarithm Domain. *IEEE Transaction on Systems, Man, and Cybernetics - Part B* , 458-466.
- Fausett, L. (1994). *Fundamentals of Neural Networks Architecture, Algorithms, and Applications*. Prentice Hall.
- Imantaka, S. R. (2010). *Sistem Pengenal Wajah Berbasis Ensemble Neural Network untuk Citra Infra merah*. Skripsi, Universitas Indonesia, Fakultas Teknik, Depok.
- Jatmiko, W. (2000). *Karakteristik Sistem Penciuman Elektronik Dalam Mengenal Aroma dan Komposisi Aroma*. Thesis, Universitas Indonesia.
- Khanifa, K. R. (2010). *Sistem Deteksi dan Penentu Posisi Wajah di dalam Citra Menggunakan PNN Teroptimasi*. Skripsi, Universitas Indonesia, Fakultas Teknik, Depok.
- Lubis, M. F. (2010). *Analisis Pemanfaatan Ruang DCT Logaritma untuk melakukan Kompensasi Iluminasi dan Normalisasi dalam Pengenalan Wajah*. Skripsi, Universitas Indonesia, Fakultas Teknik, Depok.
- Mao, K. Z. (2000, July). Probabilistic Neural-Network Structure Determination for Pattern Classification. *IEEE Transaction on Neural Networks* , 11(4).
- Masters, T. (1995). *Advanced Algorithms for Neural Networks : A C++ Sourcebook*. John Wilet & Sons.
- Sethi, I., & Jain, A. (1991). *Artificial Neural Networks and Statistical Pattern Recognition Old & New Connections*. Elsevier Science Publishers B.V.
- Tian, B. &. (2000, May). Temporal Updating Scheme for Probabilistic Neural Networks with Application to Satellite Cloud Classification. *IEEE Transaction on Neural Networks* , 11(4).

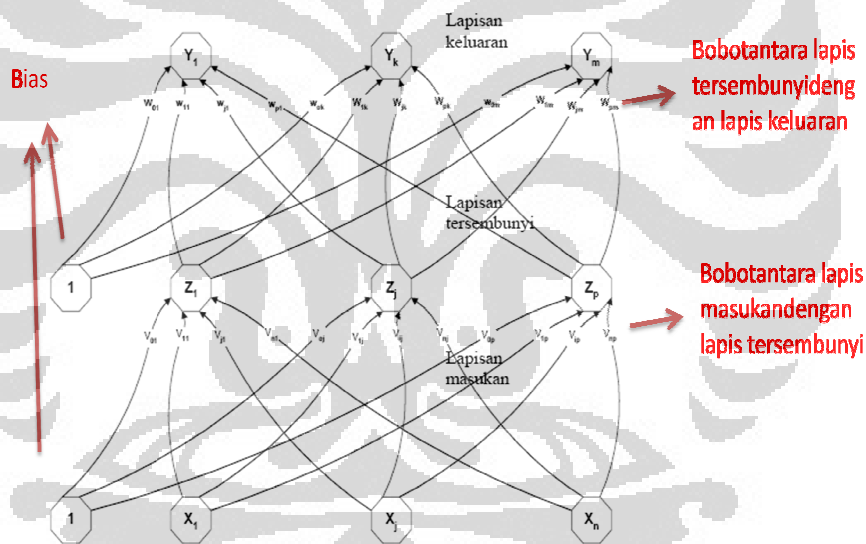
Zaknich, A. (1998). *Artificial Neural Networks An Introductory Course*.
Diunduh dari http://www.maths.uwa.edu.au/~rkealley/ann_all/ann_all.html



LAMPIRAN A

Jaringan Syarat Tiruan *Backpropagation*

Jaringan saraf tiruan (JST) dengan algoritma back propagation (BP) pada penelitian ini digunakan sebagai pembandingan dengan PNN. BP mempunyai arsitektur dengan struktur *feedforward* yang terdiri dari 3 lapisan, yaitu lapisan masukan (*input layer*), lapisan tersembunyi (*hidden layer*), dan lapisan keluaran (*output layer*). Jumlah neuron pada lapisan masukan sebanyak dimensi data, sedangkan jumlah neuron pada lapisan tersembunyi umumnya setengah dari neuron lapisan masukan. Setiap neuron antara lapisan yang satu dengan lapisan yang lain saling terhubung dan mempunyai bobot dan bias tertentu. Arsitektur JST dengan algoritma BP dapat dilihat pada gambar berikut.



Arsitektur JST *Backpropagation*

Proses *Training* BP adalah mengubah himpunan bobot pada JST langkah demi langkah menuju tingkat kesalahan yang minimum. Secara umum, tahapan mekanisme pembelajaran BP terbagi menjadi 2 tahap, yaitu komputasi maju (*feedforward*) dan komputasi balik (*back propagation of error*). Komputasi maju bertujuan untuk melakukan perhitungan keluaran sesuai dengan masukan yang ada. Pada tahap ini, sinyal masukan ditransmisikan maju mulai dari lapisan masukan, lapisan tersembunyi, hingga ke lapisan keluaran untuk menghasilkan

keluaran dari jaringan. Komputasi balik bertujuan untuk melakukan pembelajaran dengan mengubah bobot dan bias menggunakan error yang ada antara output dengan target. Langkah-langkah perhitungan algoritma BP dengan fungsi aktivasi berupa fungsi sigmoid adalah seperti berikut.

1. Inisialisasi bobot dan bias menggunakan metode Nguyen Widrow

2. Proses *Feed Forward*

Fungsi aktivasi sigmoid :

$$f(x) = \frac{1}{1 + e^{-x}} \quad (\text{A.1})$$

o Menghitung nilai pada lapis tersembunyi

$$\begin{aligned} z_in_j &= v_{0j} + \sum x_i \cdot v_{ij} \\ z_j &= f(z_in_j) \end{aligned} \quad (\text{A.2})$$

o Menghitung nilai pada lapis keluaran

$$\begin{aligned} y_in_k &= w_{0k} + \sum z_j \cdot w_{jk} \\ y_k &= f(y_in_k) \end{aligned} \quad (\text{A.3})$$

3. Proses *Back Propagation of Error*

o Menyesuaikan nilai bobot dan bias antara lapis tersembunyi dengan lapis keluaran

$$\begin{aligned} \delta_k &= (t_k - y_k) \cdot f'(y_in_k) \\ \Delta w_{jk} &= \alpha \cdot \delta_k \cdot z_j \\ \Delta w_{0k} &= \alpha \cdot \delta_k \end{aligned} \quad (\text{A.4})$$

$$\begin{aligned} w_{jk(\text{baru})} &= w_{jk(\text{lama})} + \Delta w_{jk} \\ w_{0k(\text{baru})} &= w_{0k(\text{lama})} + \Delta w_{0k} \end{aligned} \quad (\text{A.5})$$

o Menyesuaikan nilai bobot dan bias antara lapis masukan dengan lapis tersembunyi

$$\begin{aligned} \delta_in_j &= \sum \delta_k \cdot w_{jk} \\ \delta_j &= \delta_in_j \cdot f'(z_in_j) \\ \Delta v_{ij} &= \alpha \cdot \delta_j \cdot x_i \\ \Delta v_{0j} &= \alpha \cdot \delta_j \end{aligned} \quad (\text{A.6})$$

$$\begin{aligned}
 v_{ij(\text{baru})} &= v_{ij(\text{lama})} + \Delta v_{ij} \\
 v_{0j(\text{baru})} &= v_{0j(\text{lama})} + \Delta v_{0j}
 \end{aligned}
 \tag{A.7}$$

4. *Stopping Condition*

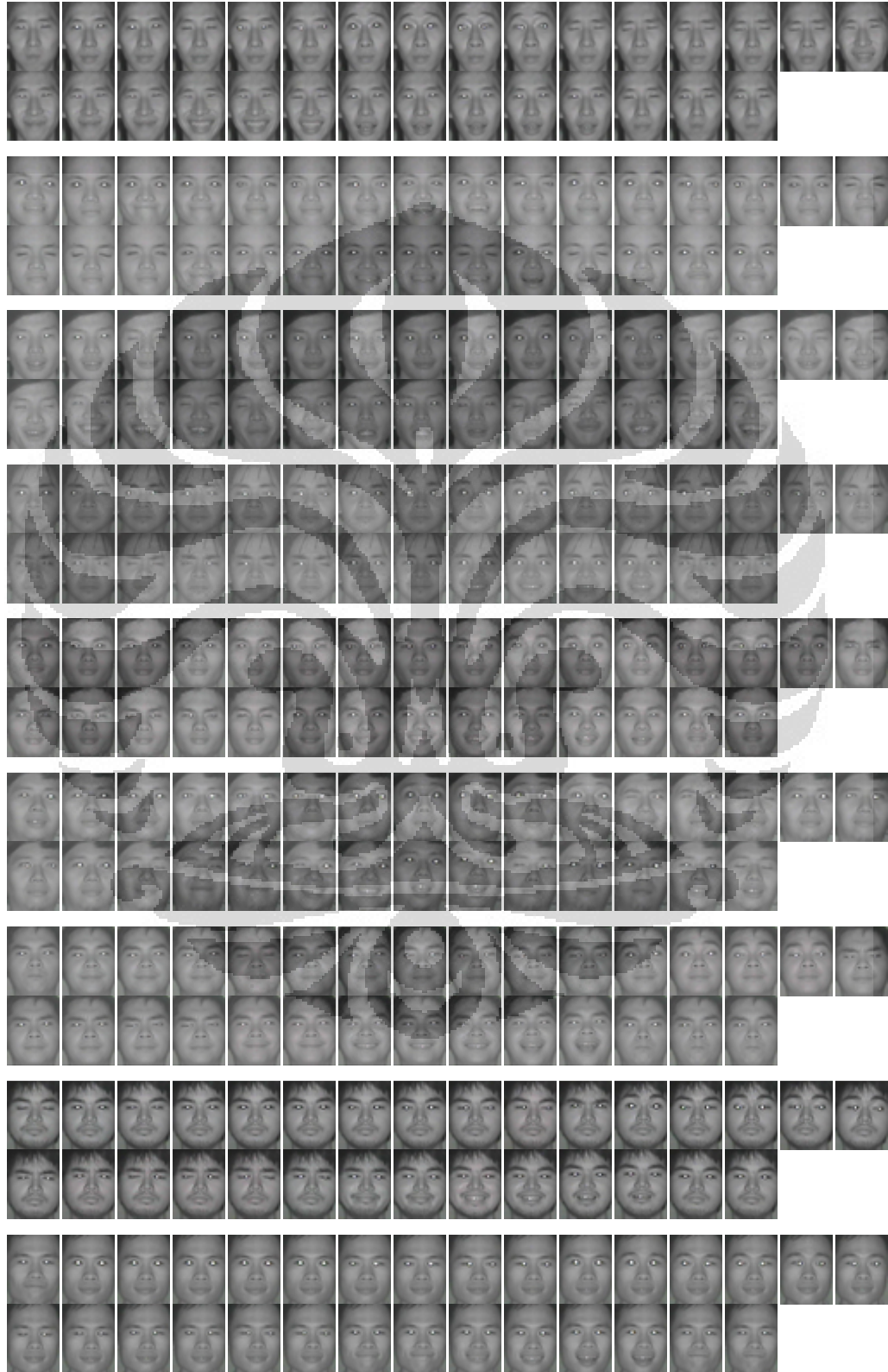
Kondisi penghentian pembelajaran bisa dengan menentukan perhitungan (epoch) maksimum ataupun error total minimum.

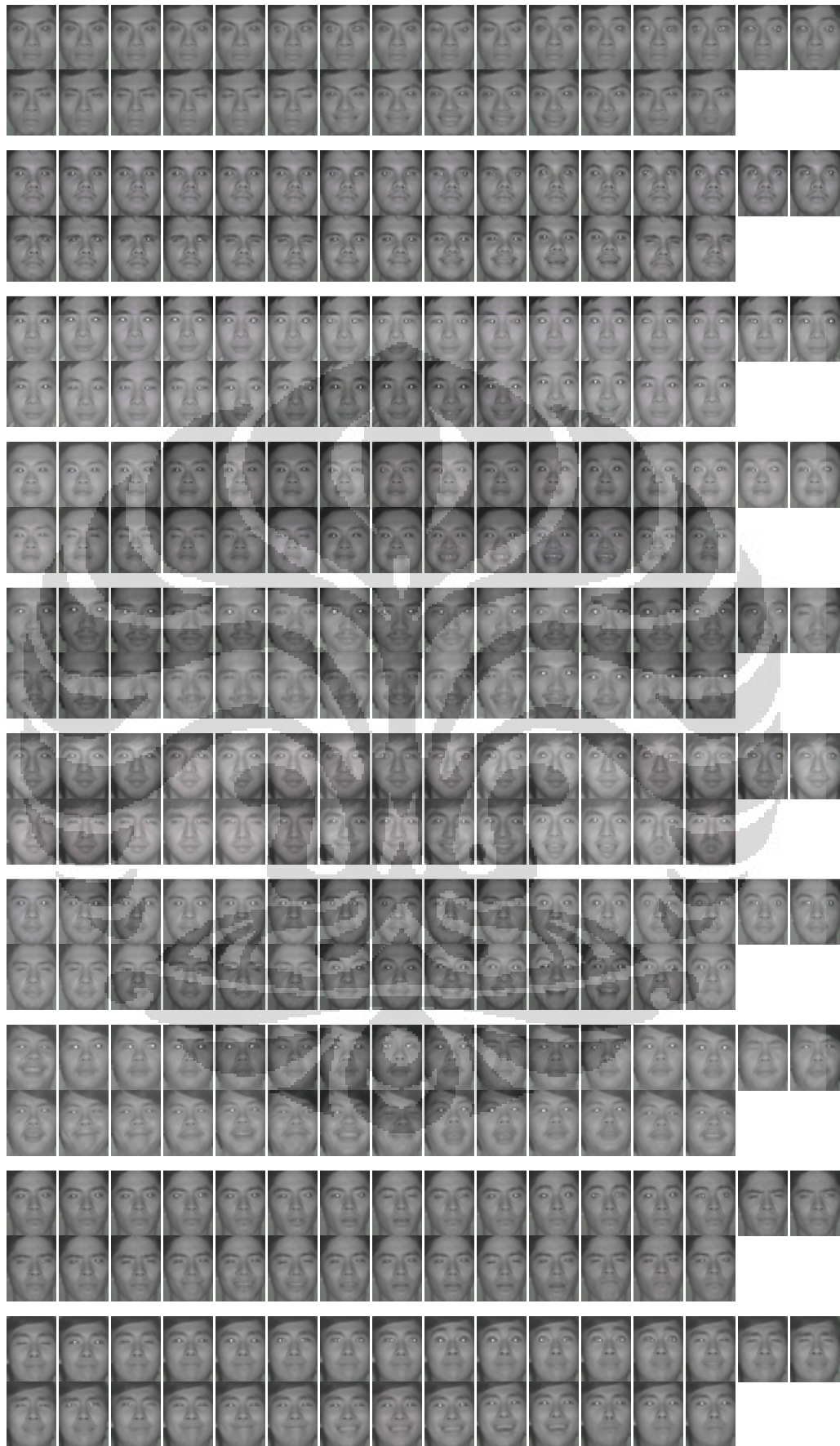
Keterangan :

- x = informasi pada neuron lapis masukan
- z = informasi pada neuron lapis tersembunyi
- y = informasi pada neuron lapis keluaran
- v_{ij} = bobot antara lapis masukan dengan lapis tersembunyi
- v_{0j} = bias pada neuron lapis tersembunyi
- w_{jk} = bobot antara lapis tersembunyi dengan lapis keluaran
- w_{0k} = bias pada neuron lapis keluaran

LAMPIRAN B

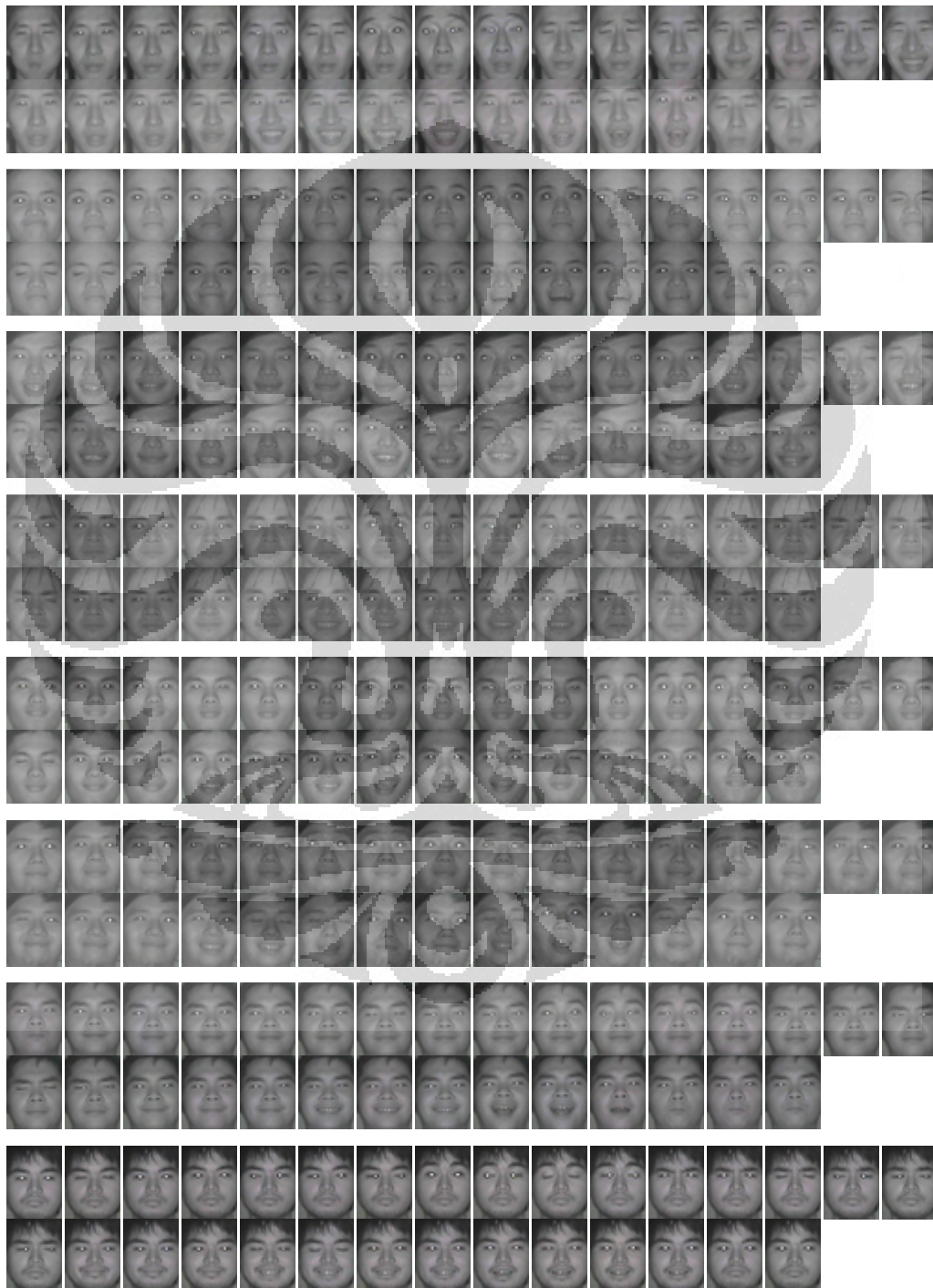
Set Data 1

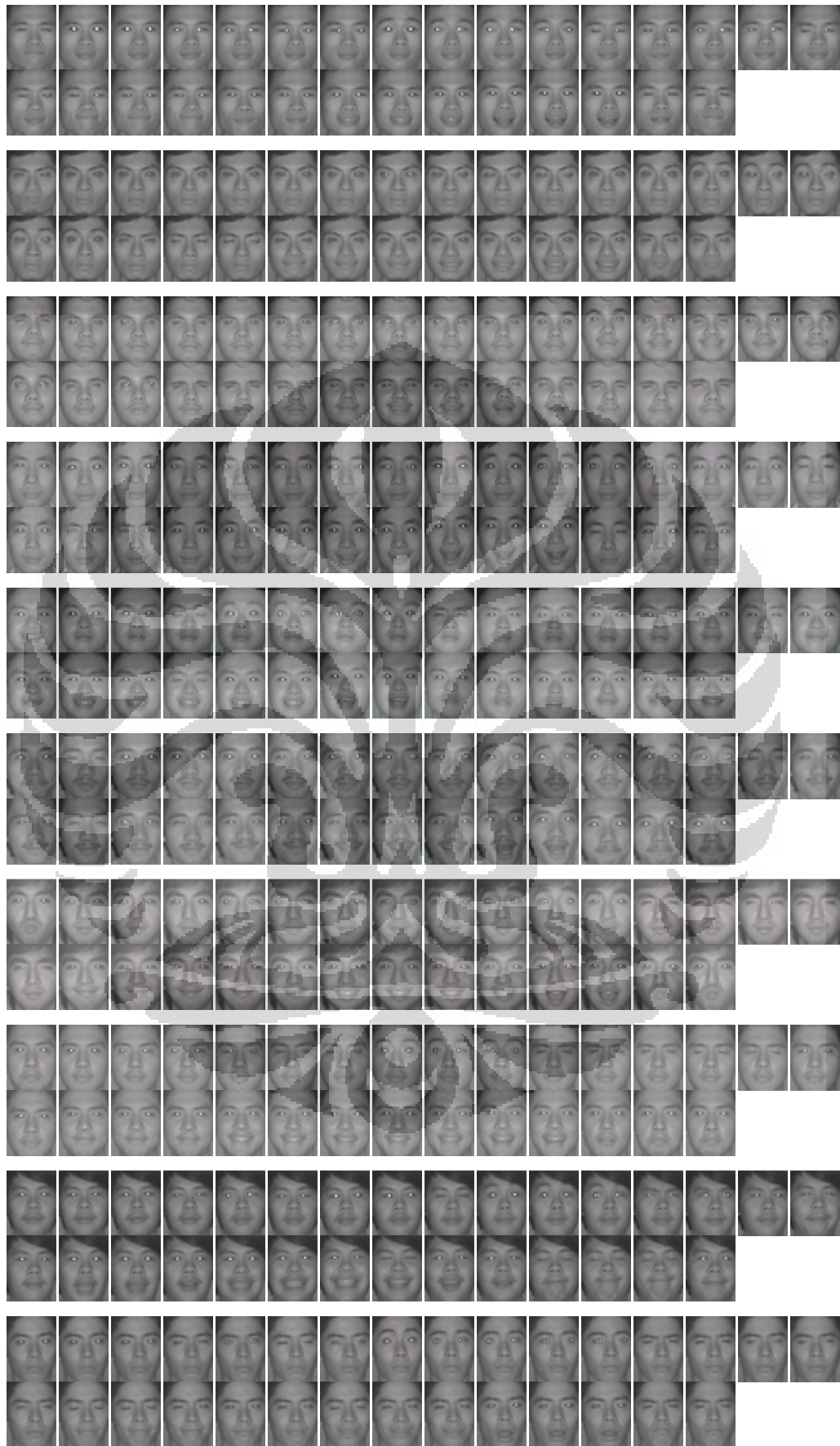






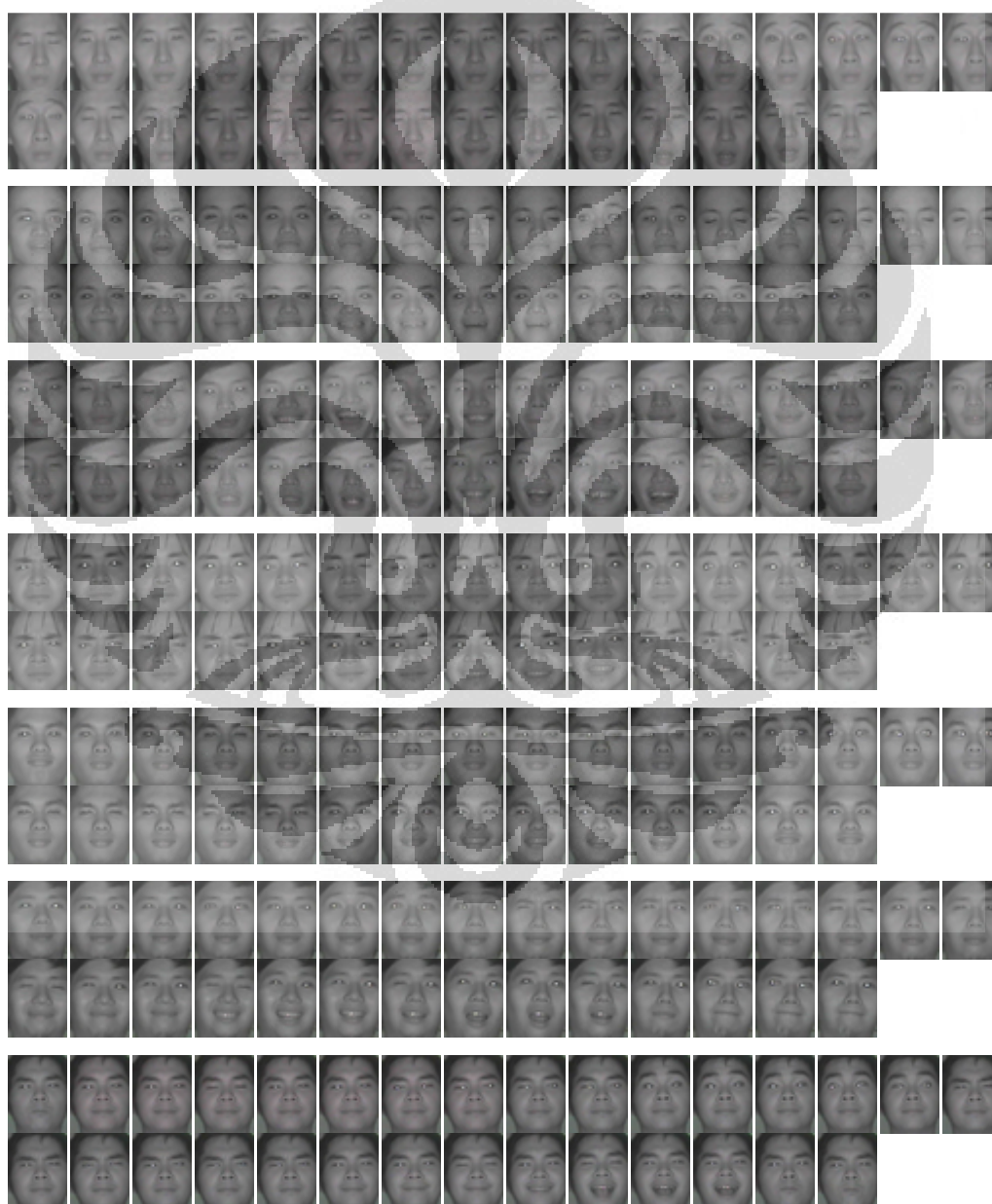
Set Data 2

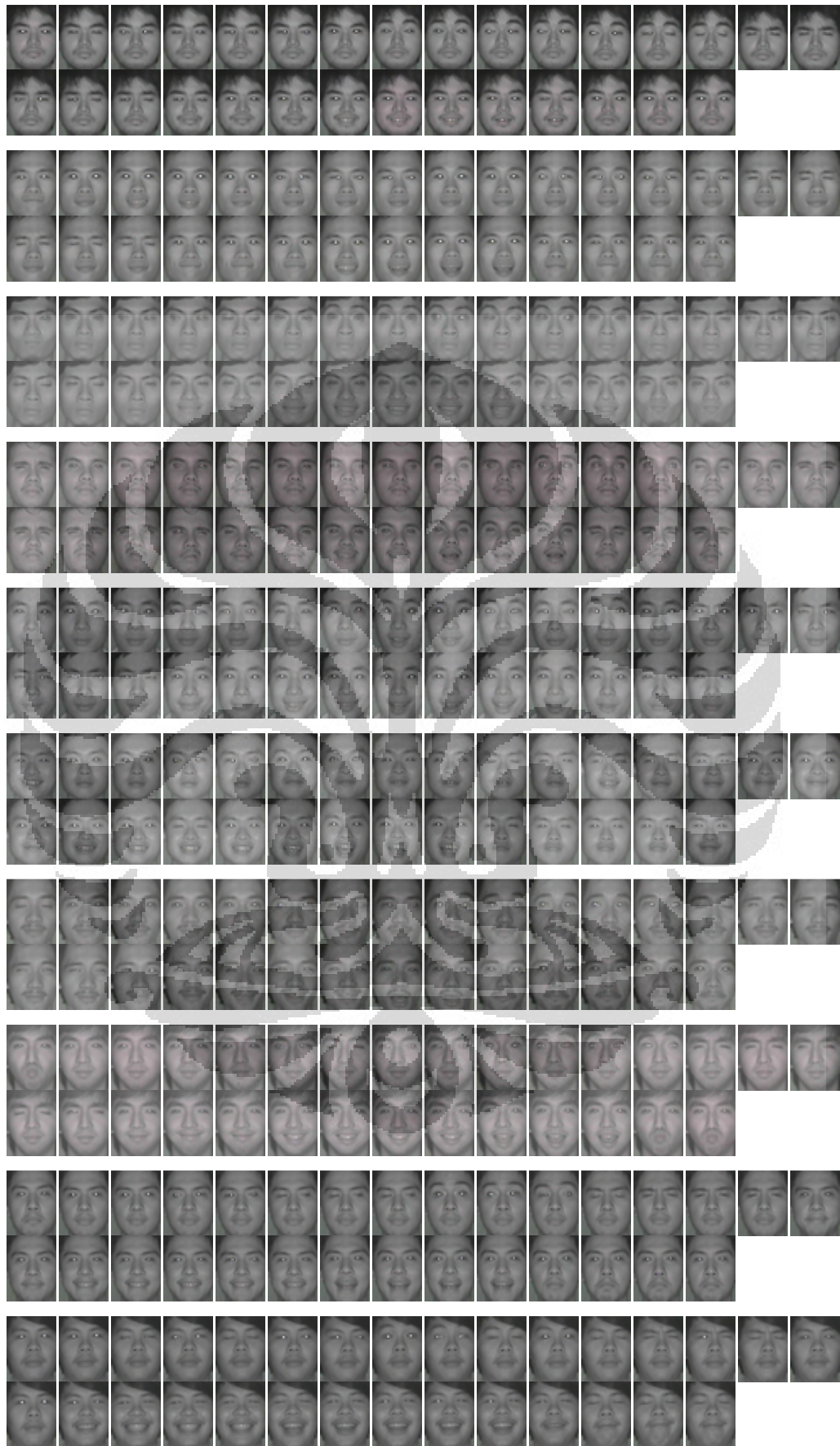






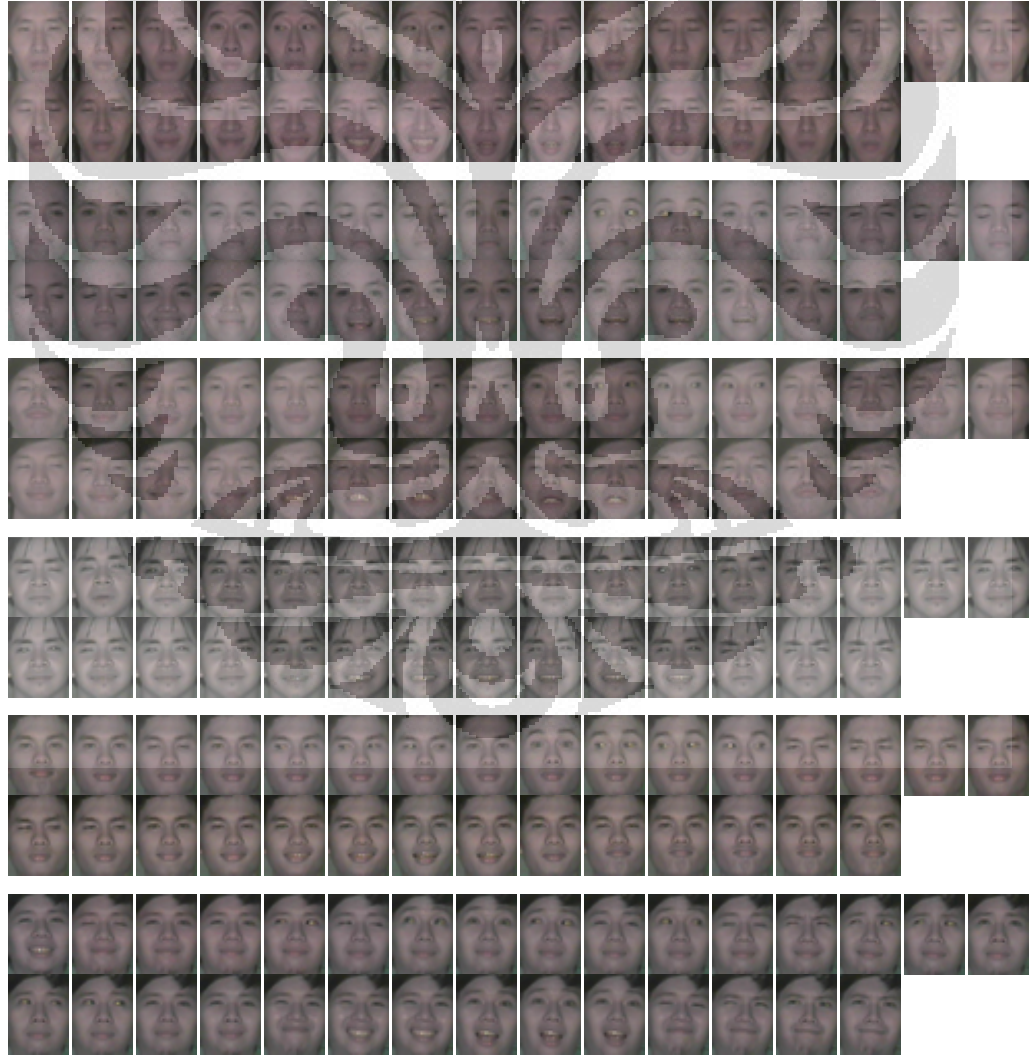
Set Data 3

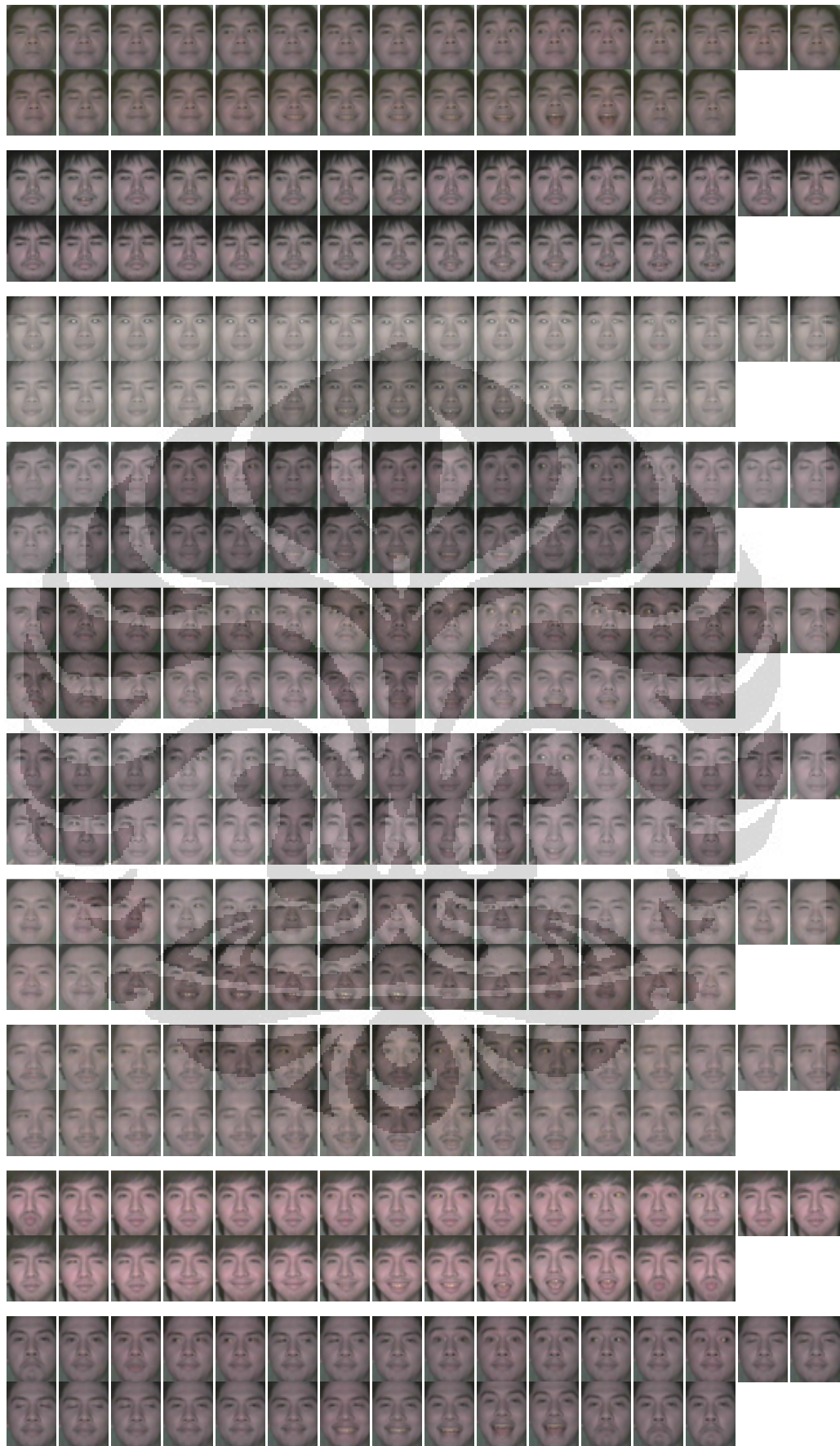


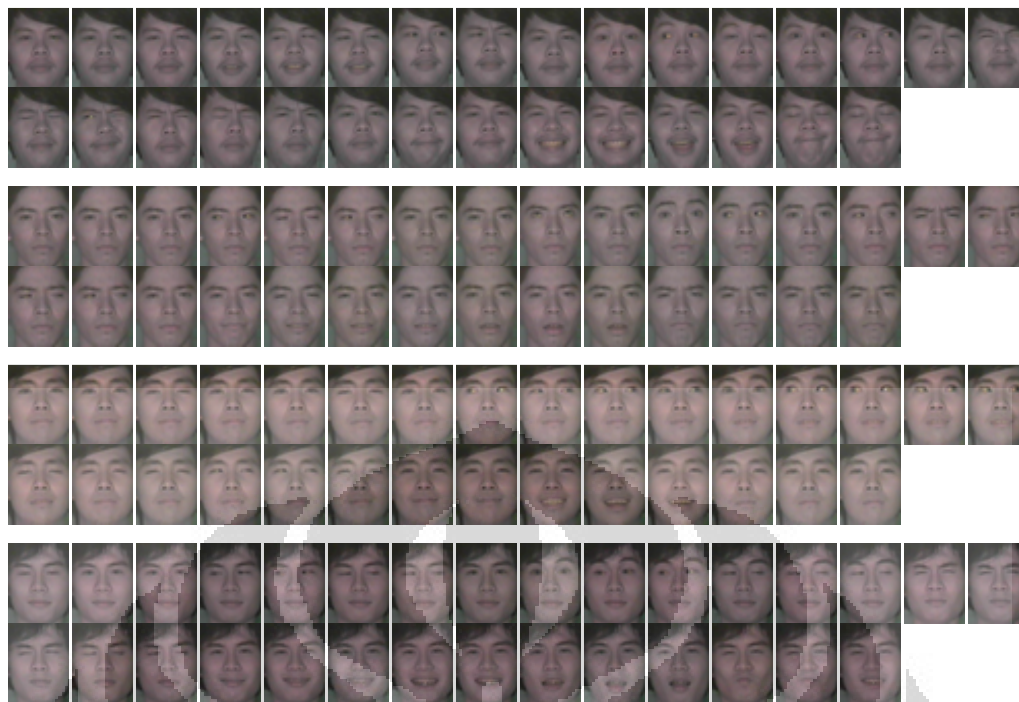




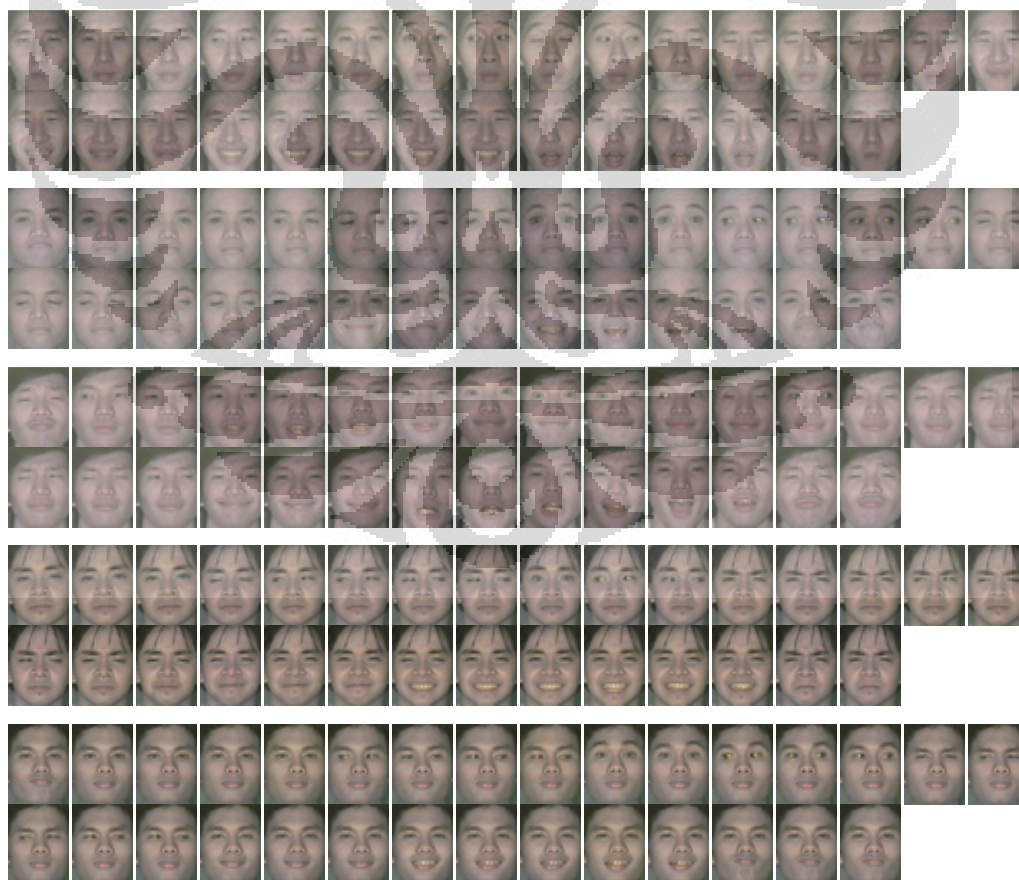
Set Data 4

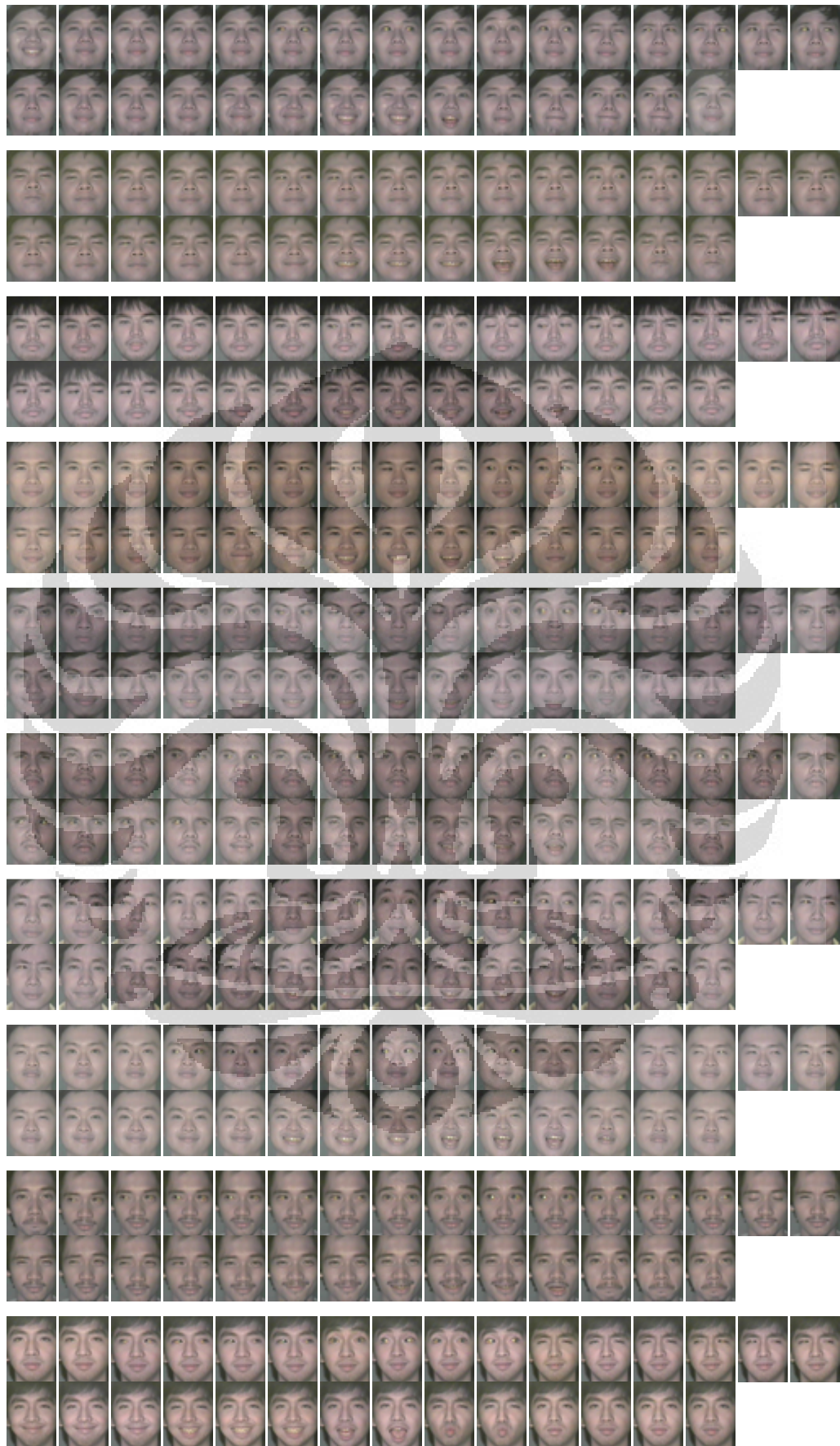


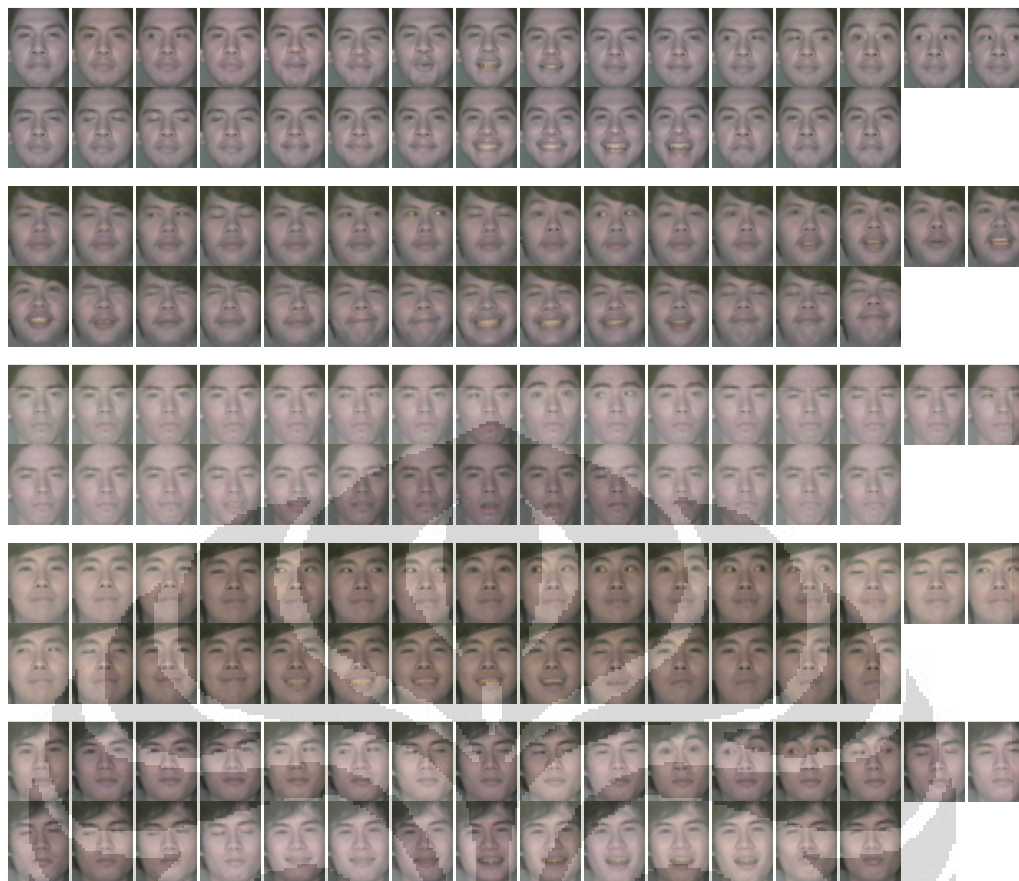




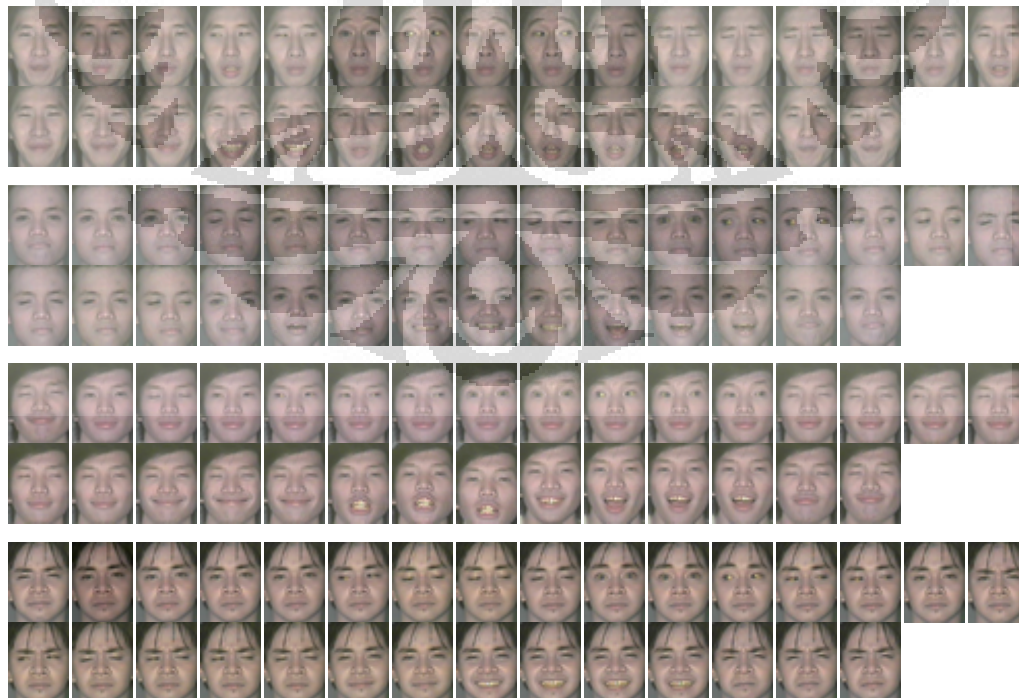
Set Data 5

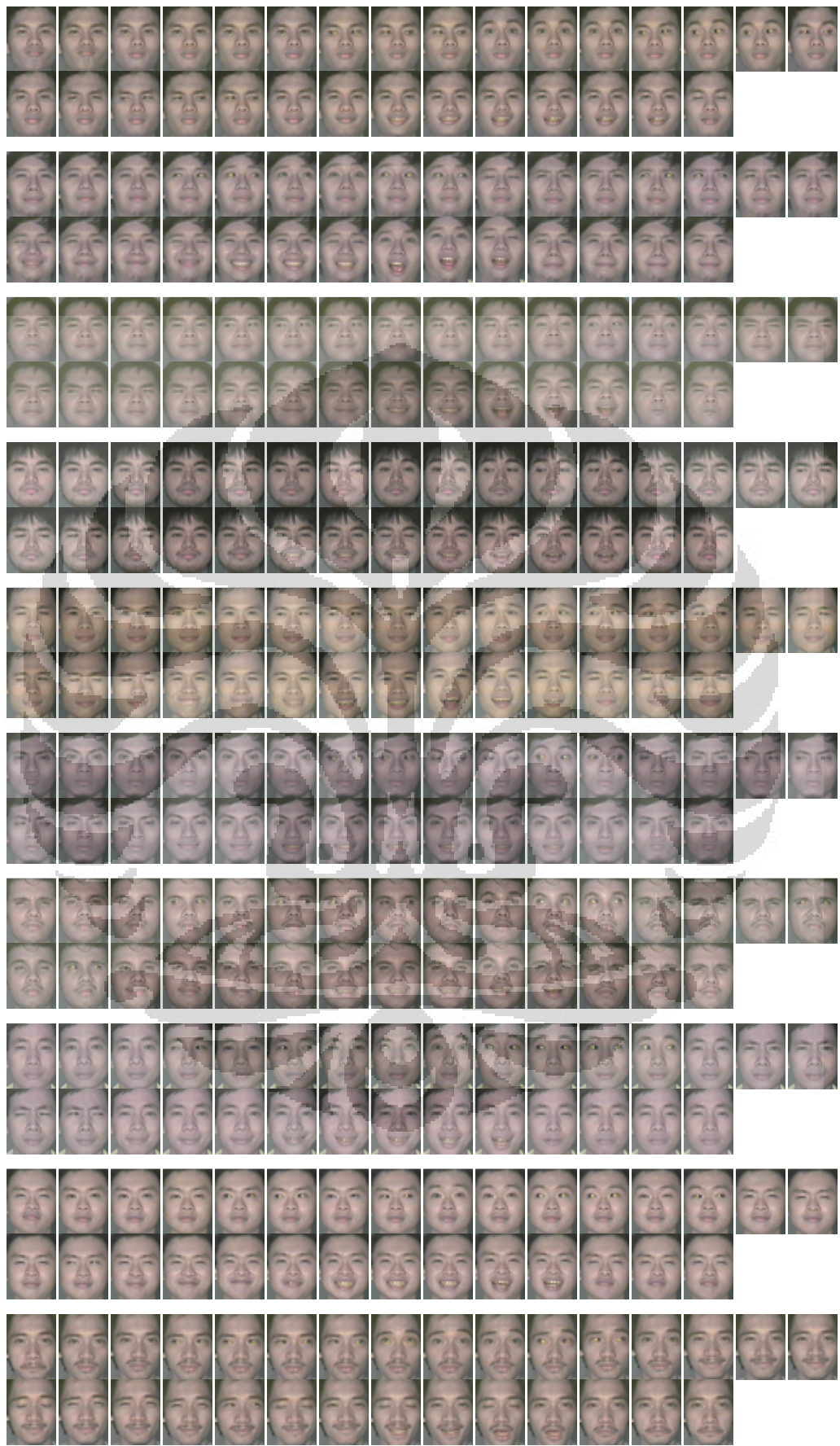


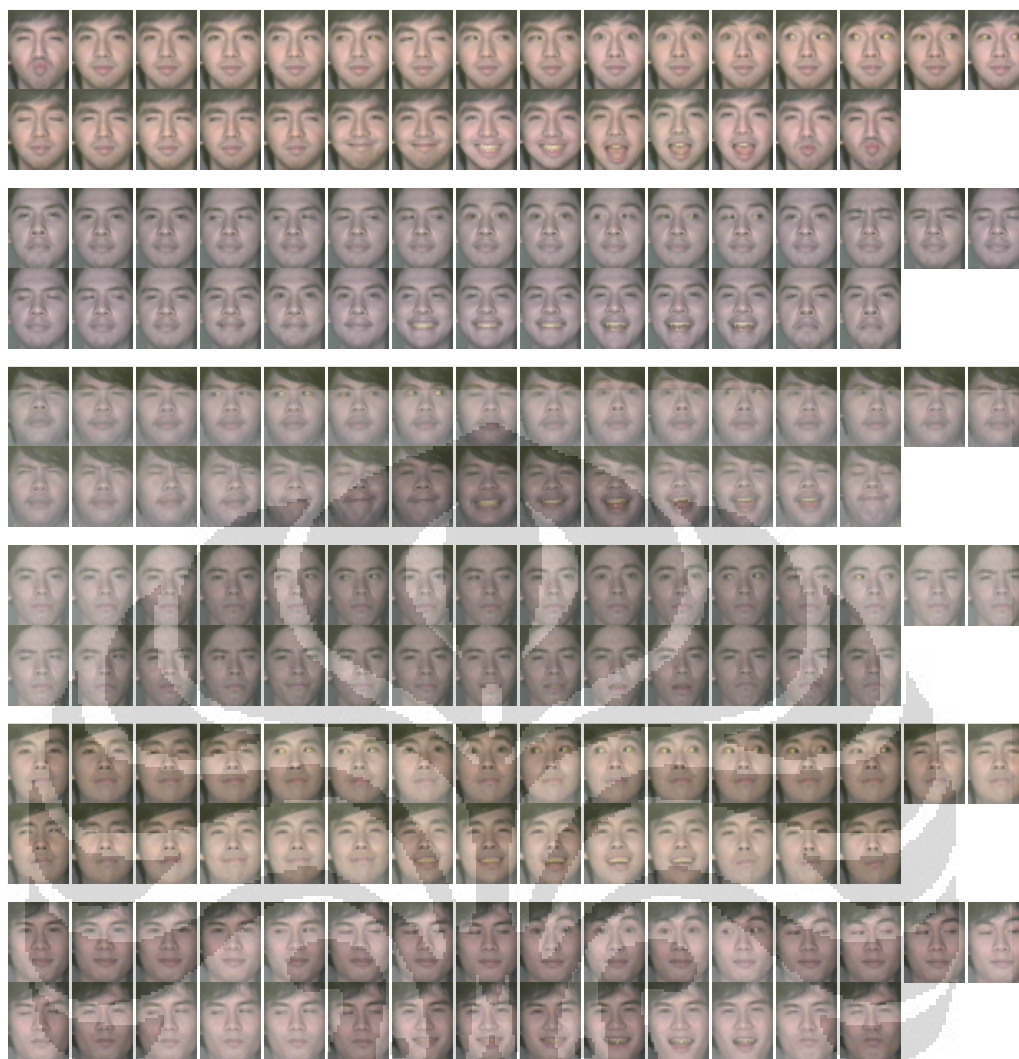




Set Data 6







Set Data 7

