JURNAL TEKNOLOGI, Edisi No. 1. Tahun XX, Maret 2006, 34-45 ISSN 0215-1685

Uji dan Aplikasi Komputasi Paralel Pada Jaringan Syaraf Probabilistik (*PNN*) Untuk Proses Klasifikasi Mutu Tomat

Kudang B. Seminar[1], Agus Buono[2] dan Moh. Khawarizmie Alim[3]
[1]Staf Pengajar Departemen Teknik Pertanian, Fakultas Teknologi Pertanian IPB
[2]Staf Pengajar Departemen Ilmu Komputer, FMIPA IPB
[3]Alumni Departemen Teknik Pertanian, Fakultas Teknologi Pertanian IPB

Abstrak

Tomat merupakan salah satu produk hortikultura yang bernilai ekonomi tinggi di dunia termasuk di Indonesia. Untuk dapat bersaing di pasaran dunia, produsen tomat harus menghasilkan produk tomat segar dan olahan dengan kualitas tinggi. Kualitas tomat yang baik sangat ditentukan oleh penanganan pasca panen yang diterapkan. Salah satu proses dalam penanganan pasca panen yang paling kritis adalah proses sortasi. Proses sortasi manual mengintrodusir subyektivitas (bias), inakurasi, kelambatan dan inkosistensi produk. Ini memerlukan suatu proses sortasi yang cerdas yang dapat mengatasi kelemahan pada proses sortasi manual. Probabilistic Neural Network (PNN) merupakan salah satu jaringan saraf tiruan yang dapat diterapkan untuk membangun mesin klasifikasi/sortasi tomat berbasis komputer. Namun demikian, untuk mempercepat waktu pemrosesan, perlu diterapkan komputasi paralel proses sortasi dengan PNN, yang memungkinkan beberapa prosesor untuk mengeksekusi secara simultan proses sortasi. Penelitian ini bertujuan untuk menguji komputasi paralel pada algoritma PNN yang diterapkan pada proses sortasi tomat. Kriteria yang digunakan untuk uji analisis adalah akurasi klasifikasi PNN, waktu eksekusi total, peningkatan kecepatan, peningkatan akurasi, dan efisiensi dibandingkan komputasi sekuensiai. Dari percobaan didapatkan kesimpulan bahwa penerapan komputasi paralel pada proses sortasi tomat telah mempercepat waktu pemrosesan dan efisiensi tanpa dengan nilai akurasi yang masih dipertahankan.

***Kata kunci**: Sortasi, klasifikasi objek, komputasi paralel dan jaringan syaraf probabilistik (JSP/PNN).*

Abstract

Tomato fruit is one of agroproducts that has high-ecocomic value in the world particularly in Indonesia. To compte in a worldwide market, a tomato fruit producer must produce fresh or processed tomatowith high quality. High quality tomato products are influenced by the application of post-harvest treatment or processing. One of the vital process in post-harvest treatment is sortation. Mannual sortation introduces subjectivity (bias), inaccuracy, slowness and inconsistency. This needs more intelligent sortation methods and tools that overcome the sortcomings of manual process. Probabilistic Neural Network (PNN) is one of Artificial Neural Network (ANN) variants that can be to develop a computer-based sortation engine for tomate fruits. However, to accelerate the sortation process, parallel computation is employed, allowing multiple processors to execute simultaneously the sortation process. This reserach is aimed towards the implementation and testing of a parallel computation algorithm with PNN to perform sortation for tomato fruits. Some criteria being observed and tested include accuracy, total execution time, speedup, and efficiency compared to sequential algorithm. The experimental results show that the application of parallel computation algorithm with PNN introduces the increase of accuracy, total execution time, speedup, and efisiency with the same accuracy.

***Keywords:** Sortation, object classification, parallel computation and probabilistic artificial neural network (P/ANN).*

1. Pendahuluan

Proses sortasi merupakan titik kritis dalam penanganan pasca panen produk pertanian, termasuk tomat. Proses sortasi yang dilakukan secara manual oleh manusia, menghasilkan produk dengan keragaman kurang baik dan juga waktu yang relatif lama. Hal ini disebabkan antara lain karena keragaman visual manusia, aspek ergonomika yaitu kelelahan manusia, dan perbedaan persepsi tentang mutu dari produk yang dihasilkan. Hal ini dapat diatasi dengan menggunakan mesin yang melakukan sortasi secara otomatis.

Berbagai teknologi yang diterapkan pada mesin sortasi dikembangkan untuk mencapai tingkat akurasi yang tinggi dalam waktu yang singkat. Penerapan logika fuzzy, neuro-fuzzy, aplikasi NIR (*Near Infra Red*), pengolahan citra digital, dan jaringan saraf tiruan merupakan beberapa teknologi yang dikembangkan untuk proses klasifikasi. PNN merupakan salah satu jaringan saraf tiruan yang memiliki tingkat akurasi yang cukup tinggi dan waktu pemrosesan yang singkat. Pengembangan berbagai teknologi untuk proses sortasi tersebut pada akhirnya akan mencapai titik maksimal dengan tingkat akurasi yang sangat tinggi. Pencapaian tingkat akurasi yang sangat tinggi tersebut dapat terus dikembangkan dengan cara mempersingkat waktu pemrosesan. Salah satu cara untuk meningkatkan kecepatan proses adalah dengan penerapan komputasi paralel. Dengan penerapan komputasi paralel diharapkan akan diperoleh suatu teknologi sortasi dengan akurasi yang sangat tinggi dan waktu eksekusi yang singkat. Hasil tersebut dapat menjadi alternatif dalam mengembangkan sistem sortasi [4] yang berbasis komputer di bidang agroindustri berskala menengah ke atas untuk sortasi produk secara masal dan komprehensif, mengarah pada pertanian presisi [3].

Paper ini membahas hasil uji dan implementasi klasifikasi komputasi paralel dengan menggunakan algoritma Jaringan Syaraf Probabilistik (JSP/*ANN*) untuk sortasi tomat dengan fokus pengamatan pada akurasi dan kecepatan klasifikasi.

2. Tinjauan Pustaka

Tomat (*Lycopersicum esculen tum* Mill) merupakan salah satu jenis sayuran yang berasal dari wilayah tropis Amerika Latin yaitu di sekitar Ekuador, Peru dan bagian utara Chili, kemudian menyebar ke Meksiko dan Amerika Utara. [8]. Tanaman tomat mulai masuk ke benua Asia pada abad ke-16. Tahun 1811 tanaman tomat telah tersebar ke daerah pegunungan di Indonesia, dibawa oleh pedagang-pedagang dari Spanyol.

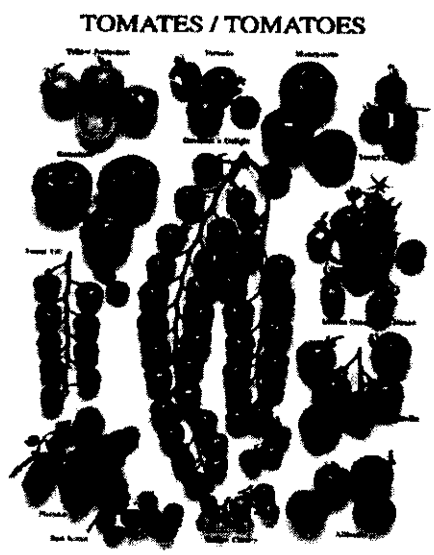
Klasifikasi botani tomat adalah sebagai berikut:

Divisi	: Spermatophyta
Sub-divisi	: Angiospermae
Kelas	: Dycotiledonae
Ordo	: Solanales
Famili	: Solanaceae
Genus	: Lycopersicum
Spesies	: *Lycopersicum esculentum* Mill

Tomat termasuk tanaman tahunan (perenial) yang berumur pendek. Tanaman ini berbentuk perdu atau semak dan tingginya dapat mencapai 2 meter. Pola pertumbuhan dapat bervariasi dari tegak hingga merayap, pada spesies tertentu memiliki batang menjalar [8]. Daun tomat termasuk daun majemuk dengan jumlah daun antara 5-7 helai. Bentuk daun oval, bergerigi, dan bercelah menyirip dengan ukuran panjang sekitar 15-30 cm dengan lebar 10-25 cm.

Bunga tomat bersifat hermaprodit dengan lima helai kelopak berwarna hijau dan lima helai mahkota berwarna kuning. Buahnya termasuk buah berdaging bertipe *berry*. Umumnya bentuk buah tomat adalah bulat, bulat pipih dan oval. Selama proses pematangan warna buah berubah dari hijau menjadi kuning. Apabila sudah matang benar, warnanya menjadi merah. Warna kuning disebabkan oleh karotenoid sedangkan warna merah disebabkan oleh pigmentasi likopen. Ukuran buah bervariasi, dari yang berdiameter 2 cm sampai dengan 15 cm, tergantung varietasnya.

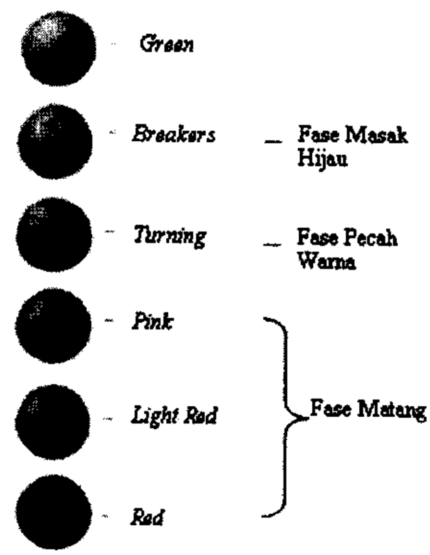
Tomat merupakan komoditi multiguna. Penggunaan tomat berkaitan dengan rasanya yang masam dan aromanya yang khas serta fungsinya sebagai sumber provitamin A dan vitamin C [8]. Tomat berfungsi sebagai sayuran, bumbu masak, buah meja, penambah nafsu makan, minuman, bahan pengawet makanan sampai pada bahan kosmetika dan obat-obatan [5]. Buah tomat juga dapat digunakan untuk membantu proses penyembuhan penyakit sariawan, gusi dan rabun ayam. Selain itu tomat mengandung karoten yang berfungsi sebagai pembentuk provitamin A dan likopen yang mampu mencegah kanker, terutama kanker prostat.



Gambar 1.
Keragaman Bentuk, Ukuran, dan Warna Tomat

Tingkat kematangan buah pada saat panen mempengaruhi mutu buah. Mutu yang baik diperoleh jika buah dipanen pada tingkat kematangan yang tepat. Tingkat kematangan tomat dibagi menjadi tiga fase, yaitu fase masak hijau, fase pecah warna dan fase matang [9]. Fase masak hijau ditandai dengan ujung buah tomat yang sudah mulai berwarna kuning gading. Pada fase pecah warna, ujung buah tomat menjadi berwarna merah jambu atau kemerah-merahan. Pada fase matang, sebagian besar permukaan buah sudah berwarna merah jambu atau merah. Di negara maju, standar tingkat kematangan tomat dibuat lebih spesifik menjadi enam fase, yang meliputi: *Green*, *Breakers*, *Turning*, *Pink*, *Light Red*, dan *Red*.

Di Indonesia tomat varietas Arthaloka yang paling banyak ditanam oleh petani, dikelompokkan dalam 3 kelas mutu, yaitu mutu A dengan berat buah di atas 150 g, mutu B antara 100-150 g dan mutu C di bawah 100 g. Berdasarkan Standar Industri Indonesia, tomat untuk tujuan ekspor dibagi menjadi dua jenis mutu yaitu mutu I dan mutu II. Di negara maju, pengelompokkan tomat terutama didasarkan pada ukuran diameter. Standar mutu untuk parameter warna adalah seragam, dengan tingkatan kelas sesuai dengan tingkat kematangan tomat.



Gambar 2.
Perubahan Warna Pada Berbagai Tingkat Kematangan Tomat

Tabel 1.
Syarat Mutu Ekspor Buah Tomat
(SNI: 01-3162-1992)

Karakteristik	Syarat mutu	
	Mutu I	Mutu II
Kesamaan sifat varietas	Seragam	Seragam
Tingkat ketuaan	Tua tapi tidak terlalu matang dan tidak terlalu lunak	Tua tapi tidak terlalu matang dan tidak terlalu lunak
Ukuran	Seragam	Seragam
Kotoran	Tidak ada	Tidak ada
Kerusakan maksimum (%)	5	10
Busuk maksimum (%)	1	1

Jaringan Saraf Tiruan

Jaringan Saraf Tiruan (JST) merupakan suatu sistem pemrosesan informasi digital yang memiliki karakteristik-karakteristik seperti jaringan saraf pada makhluk hidup [6]. Pada dasarnya, pemrosesan informasi pada JST mengacu pada pemrosesan informasi yang terjadi pada sel-sel saraf biologis, yaitu dengan pemancaran sinyal elektro kimia melalui serabut-serabut saraf (neuron).

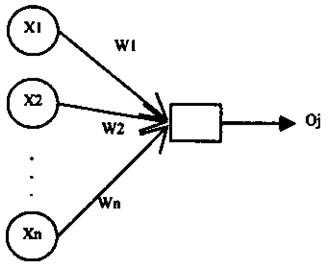
Pada JST, setiap neuron menerima input (x) dari setiap neuron lain yang dikalikan dengan suatu nilai pembobotan (w) yang sesuai. Total penjumlahan akumulatif dari himpunan input terboboti dinamakan dengan level aktivasi. Level aktivasi inilah yang akan menentukan kemungkinan apakah suatu neuron dapat meneruskan sinyal ataukah tidak kepada neuron-neuron yang lain.

$$LevelAktivasi = \sum_{i=1}^{n} X_i W_i \qquad (1)$$

dengan

X_i : Input ke-i

W_i : Bobot untuk input ke-i



Gambar 3.
Bagan Model Aktivasi Sinyal Jaringan Saraf Tiruan (JST)

Sebelum dapat digunakan, JST harus diberikan pelatihan terlebih dahulu. Pelatihan ini diperlukan untuk menemukan nilai pembobotan yang tepat bagi JST agar keluarannya menjadi benar.

Pengklasifikasian Bayes

Pengklasifikasian Bayes merupakan suatu metode klasifikasi yang menggunakan suatu fungsi diskriminan pada peluang posterior kelas [13]. Fungsi klasifikasi yang digunakan oleh kelas i dapat dinotasikan dengan:

$$P(\omega_i)P(x \mid \omega_i) \qquad (2)$$

dengan

$P(\omega_i)$: *Peluang kelas i*

$P(x \mid \omega_i)$: *Peluang bersyarat x jika masuk ke dalam kelas i*

X : *Vektor input*

ω_i : *Kelas i*

Pengklasifikasian Bayes banyak digunakan pada PNN. Dengan menggunakan metode Bayes, klasifikasi dapat dilakukan seoptimal mungkin dengan cara meminimalisir nilai kerugian yang terjadi bila terjadi kesalahan klasifikasi. Untuk mengklasifikasikan input x agar masuk ke dalam kelas A, maka harus dipenuhi syarat:

$$h_A c_A f_A(x) > h_B c_B f_B(x) \qquad (3)$$

dengan

h_A : *Kemungkinan contoh terambil dari kelas A*

h_B : *Kemungkinan contoh terambil dari kelas B*

c_A : *Biaya yang dikorbankan bila terjadi kesalahan klasifikasi input A*

c_B : *Biaya yang dikorbankan bila terjadi kesalahan klasifikasi input B*

f_A : *Fungsi kepekatan A*

f_B : *Fungsi kepekatan B*

Apabila syarat persamaan di atas tidak terpenuhi, maka input x dimasukkan ke dalam kelas B.

Penduga Kepekatan Parzen

Fungsi kepekatan yang digunakan untuk PNN yang berkaitan dengan data multivariat adalah fungsi kepekatan Parzen [6]. Fungsi Parzen merupakan suatu prosedur non parametrik yang mensintesis penduga Probability Density Function (PDF) Gauss. Fungsi Parzen akan memberikan keputusan klasifikasi setelah menghitung PDF untuk setiap kelas melalui pola pelatihan yang ada. Pada fungsi Parzen terdapat fungsi pembobot yang disebut dengan fungsi Kernel (K(x)).

Fungsi Parzen untuk data multivariat dapat dinotasikan dengan:

$$g(x) = \frac{1}{n\sigma^2}\sum_{i=1}^{n} K\left(\frac{x - x_i}{\sigma}\right) \quad (4)$$

Sedangkan fungsi Kernel yang digunakan adalah fungsi Gauss dinotasikan dengan:

$$K(x) = \frac{1}{\sigma (2\pi)^{\frac{d}{2}}} \exp\left(\frac{-x^2}{2\sigma^2}\right) \quad (5)$$

Maka kita mendapatkan fungsi kepekatan untuk kelas A sebagai berikut:

$$f_A(x) = \frac{1}{N_A \sigma^d (2\pi)^{\frac{d}{2}}} \sum_{i=1}^{N_A} \exp\left(\frac{-(x - x_{Ai})^T (x - x_{Ai})}{2\sigma^2}\right) \quad (6)$$

Jadi dapat dinotasikan:

$$P(\omega_A) = \frac{N_A}{N} \quad (7)$$

$$P(x \mid \omega_A) = \frac{1}{N_A \sigma^d (2\pi)^{\frac{d}{2}}} \sum_{i=1}^{N_A} \exp\left(\frac{-(x - x_{Ai})^T (x - x_{Ai})}{2\sigma^2}\right) \quad (8)$$

Sehingga:

$$P(\omega_A)P(x \mid \omega_A) = \frac{1}{N \sigma^d (2\pi)^{\frac{d}{2}}} \sum_{i=1}^{N_A} \exp\left(\frac{-(x - x_{Ai})^T (x - x_{Ai})}{2\sigma^2}\right) \quad (9)$$

dengan

$P(\omega_A)$	:	Peluang kelas A
$P(x \mid \omega_A)$	:	Peluang bersyarat x jika masuk ke dalam kelas A
x_{Ai}	:	Pola pelatihan ke-i kelas A
d	:	Dimensi vektor input
NA	:	Jumlah pola pelatihan kelas A
N	:	Jumlah pola pelatihan seluruh kelas
σ	:	Faktor penghalus

Faktor Penghalus Sigma (σ)

σ merupakan suatu nilai parameter yang berguna untuk menghaluskan fungsi kernel [11]. Secara tidak langsung nilai σ berperan pula dalam menentukan ketepatan klasifikasi PNN. Nilai σ tidak dapat ditentukan secara langsung, akan tetapi bisa didapatkan melalui metode statistik maupun dari hasil coba-coba. Pada penelitian ini, nilai σ didapatkan melalui Algoritma Genetik.

Probabilistic Neural Network (PNN)

PNN merupakan JST yang menggunakan teorema probabilitas klasik seperti pengklasifikasian Bayes dan penduga kepekatan Parzen [7]. Proses yang dilakukan oleh PNN dapat berlangsung lebih cepat bila dibandingkan dengan JST *Back Propagation.* Hal ini terjadi disebabkan PNN hanya membutuhkan satu kali iterasi pelatihan bila dibandingkan dengan JST *Back Propagation* yang membutuhkan beberapa kali iterasi dalam proses pelatihannya.

Walaupun demikian, keakuratan dari klasifikasi PNN sangat ditentukan oleh nilai σ dan pola pelatihan yang diberikan. Bila nilai σ yang diterapkan pada PNN tepat, maka akurasi klasifikasi akan mendekati atau mencapai 100 %. Bila nilai σ yang diterapkan tidak tepat maka akurasi klasifikasi PNN akan berkurang.

Demikian pula dengan pola pelatihan PNN. Apabila pola pelatihan dan data masukan pada satu kelas yang sama sangat berbeda jauh nilainya, maka PNN akan mengekstrapolasi data masukan tersebut. Hal inilah yang nantinya akan mengakibatkan akurasi klasifikasi PNN turun cukup drastis. Pada penelitian ini penentuan nilai σ menggunakan algoritma genetik (AG) yang memungkinkan pencarian nilai σ yang optimum melalui seleksi cepat populasi unggulan (Seminar, Buono & Sukin 2005).

Pemrosesan Paralel

Pemrosesan Paralel adalah penggunaan banyak prosesor yang saling bekerja sama satu sama lain untuk mencari suatu solusi tunggal dari suatu permasalahan [12]. Pemrosesan paralel dapat digunakan untuk beberapa keperluan, di antaranya adalah untuk mempercepat waktu eksekusi dan mendistribusikan pencarian solusi dari permasalahan yang sangat kompleks.

Peningkatan Kecepatan

Dalam konteks komputasi paralel, peningkatan kecepatan menunjukkan seberapa cepat proses komputasi paralel dibandingkan dengan komputasi sekuensial. Pada dasarnya, peningkatan kecepatan (*Speed Up*) dirumuskan sebagai

$$Sp = \frac{T_1}{T_P} \quad (10)$$

dimana

Sp : *Peningkatan kecepatan (SpeedUp)*

T_1 : *Waktu eksekusi dari algoritma sekuensial*

T_P : *Waktu eksekusi dari algoritma paralel menggunakan sebanyak P prosesor.*

Hukum Amdahl

Menurut Hukum Amdahl (1967), peningkatan dari pemrosesan paralel tidak hanya bergantung pada banyaknya prosesor yang digunakan, akan tetapi lebih dipengaruhi oleh fraksi rasio antara intruksi sekuensial dengan keseluruhan intruksi pada suatu program.

$$Sp = \frac{1}{\left(\frac{1-F}{N}\right)+F} \quad (11), \text{dan } F = \frac{I_{sek}}{I_{sek}+I_{par}} \quad (12)$$

dengan

Sp : *Peningkatan kecepatan(SpeedUp)*

F : *Fraksi rasio intruksi sekuensial terhadap keseluruhan intruksi*

$1\text{-}F$: *Fraksi rasio intruksi paralel terhadap keseluruhan intruksi*

I_{sek} : *Intruksi Sekuensial*

I_{par} : *Intruksi Paralel*

N : *Banyaknya prosesor*

Hukum Amdahl menunjukkan bahwa peningkatan kinerja dari kecepatan (*SpeedUp*) proses paralel dengan F=0.01 hanya naik 10 kali lipat walaupun jumlah prosesor dinaikkan sebanyak 100 kali lipat (dari 10 buah menjadi 1000 buah). Jadi nilai F pada program komputasi harus terus diperkecil untuk meningkatkan kecepatan pemrosesan paralel.

3. Metoda Pengujian

Identifikasi Masalah

Identifikasi masalah dilakukan dengan melakukan studi pustaka dan mempelajari program komputasi yang dibuat pada penelitian Sukin [10]. Program PNN pada penelitian Sukin [10] digunakan untuk proses klasifikasi dengan data masukan berupa data parameter mutu tomat. Proses klasifikasi dilakukan secara sekuensial dengan satu prosesor dan kemudian dibandingkan hasilnya dengan dua prosesor secara paralel.

Pengambilan Data dan Pengukuran Sampel

Dalam penelitian ini akan digunakan tiga parameter sebagai parameter mutu buah tomat, yaitu parameter bobot, panjang, dan lebar. Pemilihan tiga parameter tersebut dilakukan dengan pertimbangan bahwa pada tingkat petani buah tomat segar di Indonesia terutama di Jawa Tengah, parameter mutu yang digunakan adalah ukuran fisik dan bobot buah tomat. Pertimbangan lainnya adalah ketiga parameter mutu yang dipilih tersebut relatif mudah diperoleh, dapat dilakukan pengukuran secara sederhana sehingga akan menyederhanakan penelitian.

Tomat yang digunakan adalah varietas Permata yang diperoleh dari Kelompok Tani Desa Sumbaga, Kecamatan Bumijawa, Kabupaten Tegal. Buah tomat dikelompok-kan menjadi tiga kelas. Pengelompokkan buah tomat dilakukan secara langsung oleh petani dan pedagang setempat ke dalam tiga kelas mutu A, B, dan C. Masing-masing kelas mutu diambil sampel sebanyak 50 buah, sehingga total sampel yang diperoleh adalah sebanyak 150 sampel. Seluruh sampel kemudian diukur parameter bobot, panjang, dan lebarnya. Pengukuran bobot dilakukan dengan menggunakan neraca Ohauss, sedangkan panjang dan lebar tomat diukur dengan menggunakan penggaris.

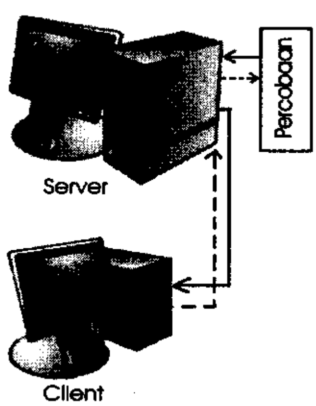
Persiapan Percobaan

Persiapan percobaan meliputi persiapan perangkat keras dan persiapan perangkat lunak. Untuk perangkat keras, dua komputer dengan spesifikasi seperti pada Tabel 2, dihubungkan dengan kabel jaringan sepanjang 2 meter dengan jenis koneksi *cross-through*. Penggabungan dua komputer dilakukan secara langsung tanpa menggunakan *concentrator* seperti *hub/switch* (Gambar 4). Hal tersebut dilakukan untuk menyederhanakan penelitian dengan tetap mematuhi prinsip komputasi paralel, di mana lebih dari satu komputer bekerja bersama-sama untuk menjalankan satu tugas. Proses pemasangan jaringan dilakukan melalui program antar muka bawaan Microsoft Windows XP, *New Connection Wizard.*

Tabel 2.
Spesifikasi Komputer

Komponen	Komputer 1	Komputer 2
prosesor	AMD Duron 1.8 GHz	AMD Athlon 1.75 GHz
memori (RAM)	256 Mb	128 Mb
hardisk	Seagate 40 Gb	Maxtor 40 Gb

Persiapan perangkat lunak dilakukan terhadap program komputasi hasil penelitian Sukin [10]. Program tersebut terdiri atas program *Utama* dan program *Antar Muka* yang harus ada pada kedua komputer. Program Utama membutuhkan *library* MPI sehingga pada setiap komputer terlebih dahulu diinstall dan dijalankan software MPICH.



Gambar 4.
Bagan Percobaan Komputasi Paralel

Penentuan Batasan-batasan Percobaan

Batasan-batasan percobaan adalah sebagai berikut:

a. Mode komputasi yang digunakan pada penelitian ini adalah mode komputasi sekuensial dan mode komputasi paralel mode vektor baris. Pemilihan mode vektor baris untuk komputasi paralel dilakukan secara bebas, sesuai dengan hasil penelitian Sukin [10] yang menyebutkan bahwa dalam pengembangan sistem komputasi paralel dapat menggunakan metode pelemparan data mode baris maupun mode kolom secara bebas.
b. Nilai peluang mutasi AG adalah 0,6 dan Nilai peluang rekombinasi AG adalah 0,35. Banyaknya generasi yang dibangkitkan setiap eksekusi AG adalah 10 dan ketelitian pencarian nilai σ kernel dibagi dalam 9.007.199.254.740. 991 interval. Nilai-nilai parameter ini selalu konstan untuk setiap percobaan dan perulangan. Nilai-nilai parameter tersebut sama seperti pada penelitian Sukin [10].
c. Proses pelatihan (*training*) dilakukan dengan 9 (sembilan) mode pelatihan. Berdasarkan penelitian sebelumnya, mode pelatihan merupakan salah satu faktor yang mempengaruhi akurasi klasifikasi PNN. Mode pelatihan menunjukkan rasio jumlah data pelatihan dengan jumlah data contoh masukan PNN, terdiri dari:
    1) 15 data (10%) pelatihan dengan 135 data (90%) contoh masukan.
    2) 30 data (20%) pelatihan dengan 120 data (80%) contoh masukan.
    3) 45 data (30%) pelatihan dengan 105 data (70%) contoh masukan.
    4) 60 data (40%) pelatihan dengan 90 data (60%) contoh masukan.
    5) 75 data (50%) pelatihan dengan 75 data (50%) contoh masukan.
    6) 90 data (60%) pelatihan dengan 60 data (40%) contoh masukan.
    7) 105 data (70%) pelatihan dengan 45 data (30%) contoh masukan.
    8) 120 data (80%) pelatihan dengan 30 data (20%) contoh masukan.
    9) 135 data (90%) pelatihan dengan 15 data (10%) contoh masukan.
d. Untuk setiap mode pelatihan, dilakukan 10 kali pengulangan dengan pola pengambilan data secara teratur setiap lima paket data. Pengulangan dilakukan untuk memastikan proses pelatihan meliputi seluruh data dan hasil yang didapatkan akan cukup mewakili keadaan sebenarnya.
e. Populasi AG:1) 500 individu
    2) 1000 individu

Eksekusi Sistem

Prosedur eksekusi sistem untuk komputasi paralel adalah sebagai berikut:

a. Pada komputer *server*, program *MPI Configuration* dijalankan. Kemudian dipilih komputer yang terlibat dalam proses paralel dengan menekan tombol *Add Hosts*. Lalu semua opsi dikonfigurasikan dengan menekan tombol *Apply* dan kemudian tombol OK.

b. Pada komputer *server* dan *client*, dijalankan program Antar Muka.exe

c. Pada komputer *server*, jendela guiMPIRun diaktifkan dan dimasukkan alamat (*path*) tempat program Utama.exe berada. Kemudian ditentukan banyaknya proses sesuai jumlah komputer yang terlibat. Dalam hal ini banyaknya proses adalah 2.

d. Dari *interface* guiMPIRun, untuk memulai sistem ditekan tombol Run lalu masukkan *username* dan password administrator komputer *server*. Program AntarMuka.exe akan aktif dan siap melakukan eksekusi sistem paralel.

e. Sebelumnya perlu dipastikan file teks data pelatihan dan data masukan berada pada alamat (*path*) yang benar. Kemudian, pada *interface* program AntarMuka, inisialisasi dilakukan dengan mengatur parameter-perameter percobaan seperti populasi AG, mode komputasi dan sebagainya. Setelah inisialisasi selesai dilakukan, tombol *Eksekusi* ditekan dan tunggu hingga program selesai melakukan proses klasifikasi.

f. Setelah proses klasifikasi selesai, hasil yang ditampilkan dicatat atau dapat disimpan dalam file teks ataupun file HTML. Setelah hasil disimpan, satu percobaan telah selesai. Untuk menjalankan percobaan berikutnya, data pelatihan dan data masukan yang baru disiapkan dan prosedur diulangi mulai bagian e.

Prosedur tersebut dilakukan untuk semua percobaan dengan total banyaknya percobaan adalah 180 percobaan sistem paralel. Untuk 180 percobaan sistem sekuensial, prosedur percobaan hampir sama hanya perlu beberapa penyesuaian seperti penentuan komputer dan bayaknya proses.

Analisis Data

Analsis data dilakukan terhadap hasil percobaan yang meliputi pengamatan grafik, dan analisis statistika deskriptif.

4. Hasil dan Pembahasan

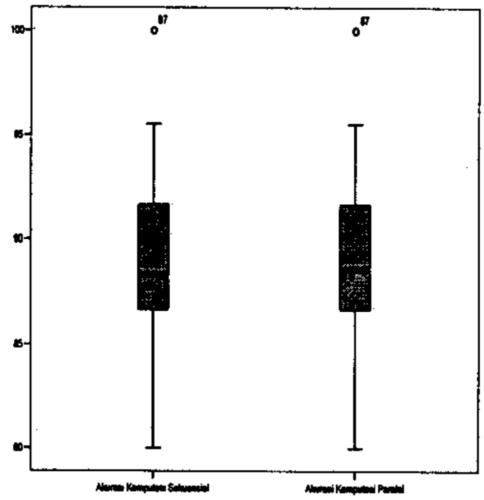
Akurasi Klasifikasi

Berdasarkan hasil analisis statistika deskriptif seperti pada Gambar 5, tampak bahwa kedua mode komputasi tersebut menunjukkan nilai akurasi rata-rata sekitar 89.2%. Pengamatan lebih seksama menunjukkan rata-rata akurasi komputasi sekuensial sedikit lebih tinggi daripada komputasi paralel, tetapi median akurasi komputasi sekuensial lebih rendah daripada komputasi paralel. Sebaran data akurasi klasifikasi kedua mode komputasi relatif sama, ditunjukkan dengan nilai persentil 25 dan 75 serta diagram plot kotak pada Gambar 6.

Statistics

		Akurasi Komputasi Sekuensial	Akurasi Komputasi Paralel
N	Valid	180	180
	Missing	0	0
Mean		89.204	89.190
Median		88.333	88.571
Mode		86.7	86.7
Std. Deviation		3.6835	3.6710
Variance		13.568	13.476
Range		20.0	20.0
Minimum		80.0	80.0
Maximum		100.0	100.0
Percentiles	25	86.667	86.667
	50	88.333	88.571
	75	91.667	91.667

Gambar 5.
Hasil Statistika Deskriptif Akurasi Klasifikasi PNN.



Gambar 6.
Diagram Plot Kotak Akurasi Klasifikasi PNN.

Waktu Total

Waktu eksekusi total komputasi paralel lebih rendah daripada komputasi sekuensial. Hal ini ditunjukkan secara konsisten pada Gambar 7 dan Gambar 8 di mana rata-rata waktu eksekusi komputasi sekuensial adalah 548.66 detik dan rata-rata waktu eksekusi komputasi paralel adalah 439.57 detik. Nilai median waktu eksekusi komputasi paralel juga lebih rendah daripada komputasi sekuensial. Keragaman data waktu total pada kedua mode komputasi sangat besar karena adanya pengaruh nilai penghalus σ (persamaan 6) yang dihasilkan dari algoritma genetik (GA). Keragaman tersebut ditunjukkan dengan nilai standar deviasi dan variansi yang besar dan juga nilai rentang data dari nilai minimum sekitar 100 detik hingga nilai maksimum mencapai sekitar 900 detik. Hal ini menunjukkan ada faktor tertentu yang mempengaruhi waktu total sehingga waktu eksekusi menjadi beragam.

Statistics

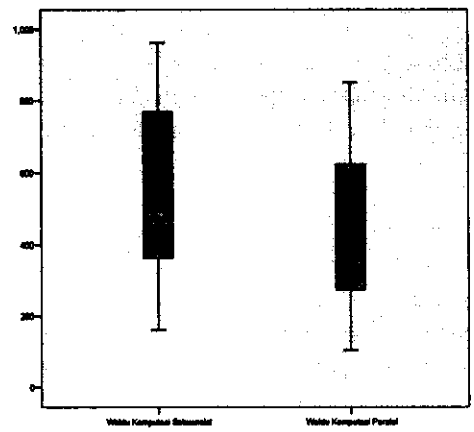
		Waktu Komputasi Sekuensial	Waktu Komputasi Paralel
N	Valid	180	180
	Missing	0	0
Mean		548.66	439.57
Median		472.00	362.00
Mode		472	221ª
Std. Deviation		240.708	215.461
Variance		57940.236	46423.396
Range		802	747
Minimum		161	104
Maximum		963	851
Percentiles	25	363.00	275.50
	50	472.00	362.00
	75	768.00	623.50

a. Multiple modes exist. The smallest value is shown

Gambar 7.
Hasil Statistika Deskriptif Waktu Eksekusi Total.

Peningkatan Kecepatan dan Peningkatan Akurasi

Peningkatan kecepatan dan peningkatan akurasi merupakan turunan dari nilai waktu total dan nilai akurasi klasifikasi. Keduanya menunjukkan perubahan nilai waktu eksekusi dan akurasi klasifikasi antara komputasi paralel dengan komputasi sekuensial. Hasil analisis statistika deskriptif pada Gambar 9 dan diagram plot kotak pada Gambar 10 menunjukkan bahwa penerapan komputasi paralel menghasilkan peningkatan kecepatan yang cukup signifikan namun tidak menyebabkan perubahan yang berarti terhadap akurasi klasifikasi.



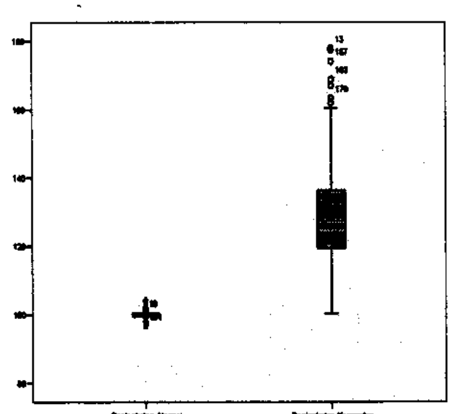
Gambar 8.
Diagram Plot Kotak Waktu Total.

Statistics

		Peningkatan Akurasi	Peningkatan Kecepatan
N	Valid	180	180
	Missing	0	0
Mean		100.018	128.857
Median		100.000	125.912
Mode		100.0	125.6ª
Std. Deviation		.8048	14.7734
Variance		.648	218.253
Range		7.7	77.6
Minimum		96.6	100.4
Maximum		104.3	177.9
Percentiles	25	100.000	119.294
	50	100.000	125.912
	75	100.000	136.412

a. Multiple modes exist. The smallest value is shown

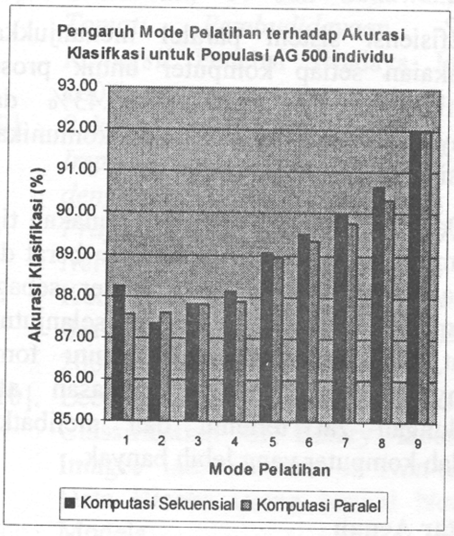
Gambar 9.
Hasil Statistika Deskriptif Peningkatan Akurasi dan Kecepatan.

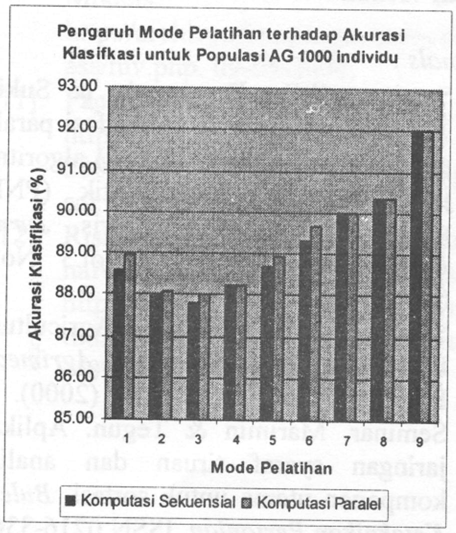


Gambar 10.
Diagram Plot Kotak Peningkatan Akurasi dan Kecepatan.

Pengaruh Mode Pelatihan

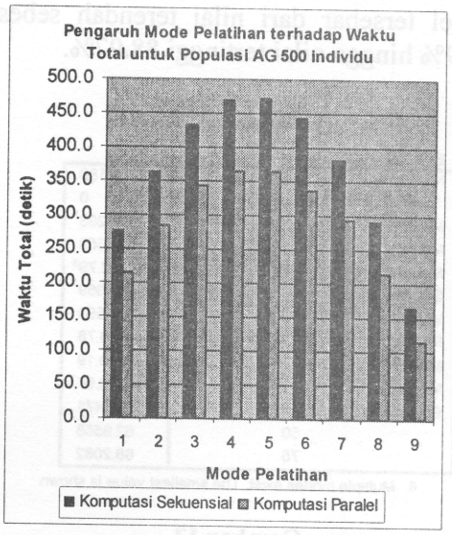
Hasil pengamatan grafik pada Gambar 11 menunjukkan adanya pola keteraturan bahwa mode pelatihan 9 (90% data pelatihan dan 10% data masukan) menghasilkan nilai akurasi klasifikasi tertinggi. Sedangkan akurasi terendah tidak dapat disimpulkan secara pasti, namun dicapai antara mode pelatihan 2 dan mode pelatihan 3.

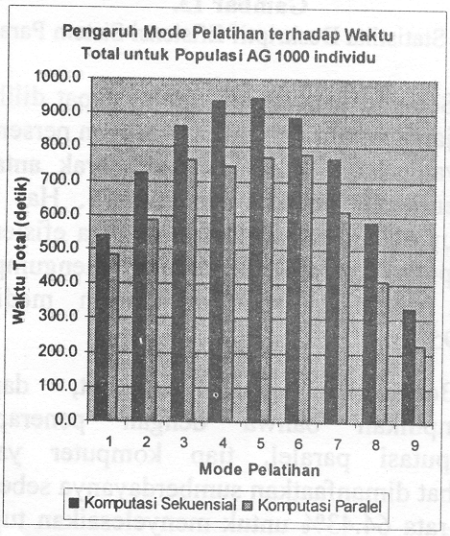




Gambar 11.
Grafik Pengaruh Mode Pelatihan Terhadap Akurasi Klasifikasi.

Pengaruh mode pelatihan terhadap waktu eksekusi total ditunjukkan pada Gambar 12. Terdapat suatu pola keteraturan di mana waktu yang dibutuhkan untuk proses klasifikasi semakin banyak ketika perbandingan antara data pelatihan dan data masukan mendekati satu. Waktu yang dibutuhkan untuk klasifikasi dengan mode pelatihan 5, di mana perbandingan antara data pelatihan dan data masukan adalah 50%:50%, lebih banyak/ lebih lama dibandingkan waktu yang dibutuhkan untuk proses klasifikasi dengan mode pelatihan yang lain. Waktu tercepat diperoleh pada mode pelatihan 9 (90% data pelatihan dan 10% data masukan).





Gambar 12.
Grafik Pengaruh Mode Pelatihan Terhadap Waktu Total.

Efisiensi Sistem Paralel

Efisiensi dalam komputasi paralel menunjukkan besarnya pemanfaatan tiap sumber daya (prosesor) untuk menyelesaikan satu tugas. Besarnya efisiensi sistem paralel tergantung pada peningkatan kecepatan dan jumlah prosesor yang terlibat. Berdasarkan hasil perhitungan statistika deskriptif terhadap efisiensi sistem paralel seperti pada Gambar 13, tampak bahwa dari 180 percobaan komputasi paralel yang dilakukan, rata-rata efisiensi sistem paralel adalah sebesar 64.428% dengan median 62.95% dan modus 62.79%. Standar deviasi efisiensi sistem paralel adalah sebesar 7.38% dan variansi sebesar 54.56%. Efisiensi sistem paralel tersebar dari nilai terendah sebesar 50.19% hingga nilai tertinggi 88.97%.

Statistics

Efisiensi Sistem Paralel

N	Valid	180
	Missing	0
Mean		64.4283
Median		62.9558
Mode		62.79[a]
Std. Deviation		7.38669
Variance		54.563
Range		38.78
Minimum		50.19
Maximum		88.97
Percentiles	25	59.6471
	50	62.9558
	75	68.2062

a. Multiple modes exist. The smallest value is shown

Gambar 13.
Hasil Statistika Deskriptif Efisiensi Sistem Paralel

Sebaran data secara umum dapat dilihat dari jarak antara persentil 25 dengan persentil 50 yang lebih kecil daripada jarak antara persentil 50 dengan persentil 75. Hal ini menunjukkan bahwa distribusi data efisiensi komputasi paralel cenderung mengumpul pada nilai-nilai kecil di bawah median (62.95%).

Berdasarkan hasil tersebut, dapat disimpulkan bahwa dengan penerapan komputasi paralel, tiap komputer yang terlibat dimanfaatkan sumberdayanya sebesar rata-rata 64.43% untuk menyelesaikan tugas klasifikasi. Sumberdaya sisanya terpakai untuk proses komunikasi dan sinkronisasi antar komputer.

5. Kesimpulan

Komputasi paralel dapat diaplikasikan pada proses sortasi buah tomat segar dan layak dikembangkan untuk proses sortasi produk pertanian lainnya. Komputasi paralel terbukti telah mempercepat waktu pemrosesan tanpa mengubah nilai akurasi klasifikasi secara signifikan.

Mode pelatihan secara signifikan mempengaruhi akurasi klasifikasi dan waktu total. Mode pelatihan 9 (90% data pelatihan dan 10% data masukan) merupakan mode pelatihan terbaik di mana didapatkan akurasi klasifikasi tertinggi dan waktu eksekusi tercepat.

Efisiensi sistem paralel menunjukkan pemakaian setiap komputer untuk proses klasifikasi adalah sebesar 64.43% dan sisanya terpakai untuk proses komunikasi dan sinkronisasi antar komputer.

Pada penelitian ini hanya digunakan tiga parameter mutu tomat berdasarkan berat dan ukuran serta dua buah komputer sebagai sistem paralel. Untuk penelitian selanjutnya dapat digunakan parameter mutu tomat lainnya seperti warna, kekerasan atau kandungan zat tertentu dan melibatkan jumlah komputer yang lebih banyak.

Daftar Acuan

Journals :

[1]. Seminar, K.B., Buono, A. & Sukin, T.P.J. Desain dan uji komputasi paralel penentuan nilai penghalus (σ) algoritma jaringan syaraf probabilistik (PNN) untuk klasifikasi bunga Iris. *Jurnal Ilmiah Ilmu Komputer.* Vol.3 No.1, (2005), hal 19-31.

[2]. Seminar, K.B. Precision Agriculture: Paradigma dan Aplikasi. *Agrimedia* ISSN 0853-8468 Vol.6 No.1, (2000).

[3]. Seminar, Marimin & Teguh. Aplikasi jaringan syaraf tiruan dan analisis komponen utama untuk sortasi. *Buletin Keteknikan Pertanian,* ISSN 0216-3365. Vol.17 No.2, (2002), hal 39-52.

Books :

[4]. Duriat, A.S. *Tomat: Komoditas Andalan yang Prospektif,* hal 1-8. *Dalam*: Duriat, A.S., Soeganda, W.W.H., Permadi, A.H., Sinaga, R.M., Hilman, Y. Dan Basuki, R.S. (Ed). Teknologi Produksi Tomat. Balai Penelitian Tanaman Sayuran, Lembang, Bandung, 1997.

[5]. Fu, Limin. *Neural Networks in Computer Intelligence*. ISBN 0-07-113319-4. McGraw-Hill. Singapura. 1994.

[6]. Patterson, Dan W. *Artificial Neural Networks: Theory and Applications*. Prentice Hall, Singapura. 1996.

[7]. Rubatzky, V.E. dan Yamaguchi, M. *Sayuran Dunia Tiga*. Penerbit ITB, Bandung. 1997.

[8]. Trisnawati, Y. dan Setiawan, A.I. *Tomat: Pembudidayaan Secara Komersil*. Penebar Swadaya. Jakarta. 2002.

[9]. Sukin, T.P.J., *Desain dan Implementasi Komputasi Paralel dengan Algoritma Genetik untuk Prapemrosesan Probabilistic Neural Network*. Skripsi. Departemen Ilmu Komputer. Fakultas Matematika dan Ilmu Pengetahuan Alam, IPB, Bogor. 2004.

[10]. Lee, D.X., G. Thoma, & H. Weschler. Classification of Binary Document Images into Texture or Non-textual Data Blocks Using Neural Network Models. http://archive.nlm.nih.gov/pubs/doc_class/mv.php. 09-08-2004.

[11]. Pacifico, Mark & Merrill. Tanpa judul. http://www.cs.umd.edu/class/fall2001/cmsc411/projects/parallel2/proposal.html. [11-12-2002].

[12]. Rish, I. An empirical study of the naïve Bayes classifier. http://www.intellektik.informatik.tudarmstadt.de/~tom/IJCAI01/Rish.pdf. [09-08-2004].