

## **BAB 4**

### **IMPLEMENTASI**

Bab ini menuturkan penjelasan mengenai implementasi dari sistem pengujian yang dibangun berdasarkan hasil analisis dan perancangan sistem yang telah dibahas pada bab sebelumnya. Bab ini terdiri dari penjelasan mengenai spesifikasi sistem - baik perangkat keras maupun perangkat lunak - yang digunakan, implementasi prosedur pengujian yang dipakai untuk melakukan berbagai proses dan mengukur berbagai metode pengenalan wajah.

#### **4.1. Spesifikasi Sistem**

Pada bagian ini dicantumkan keterangan spesifikasi mesin yang digunakan dalam mengembangkan sistem pengujian proses pengenalan wajah ini. Spesifikasi perangkat yang dijelaskan meliputi perangkat keras dan perangkat lunak.

##### **4.1.1. Perangkat Keras**

Sistem ini dikembangkan pada komputer dengan spesifikasi perangkat keras sebagai berikut:

*Processor* : Mobile Intel(R) Pentium(R) 1.80GHz

*Memory* : 704 MB DDRAM

#### **4.2. Perangkat Lunak**

Perangkat Lunak yang digunakan untuk mengembangkan sistem ini adalah sebagai berikut:

- Sistem Operasi: Microsoft Windows XP Professional Edition Version 2002 Service Pack 2
- Lingkungan Pengembangan: Matlab 7.1
- *Library*: SparseLab dan L1Magic

#### **4.3. Implementasi**

Implementasi sistem pengujian yang dilakukan pada ruang lingkup Matlab, secara umum dalam dikelompokkan ke dalam tiga modul besar:

- Modul pengenalan wajah dengan metode Sparse Representation, algoritma Lasso
- Modul pengenalan wajah dengan metode Sparse Representation, algoritma Primal-Dual
- Modul pengenalan wajah dengan metode Eigenface

Pada masing-masing modul, proses pengujian metode dilakukan melalui beberapa langkah umum yang serupa. Untuk masing-masing basis data, sistem akan:

- Menerima masukan seluruh citra latih dari basis data, menyimpan representasinya sebagai satu matriks yang setiap kolomnya merupakan representasi dari satu buah citra latih
- Untuk setiap citra uji, menerima citra uji sebagai masukan dan menyimpan representasinya sebagai satu matriks terpisah
- Memproses setiap citra uji dengan salah satu kombinasi metode dan algoritma yang diujikan, sesuai dengan tahapan yang telah dijelaskan dalam pembahasan metode pada bab Landasan Teori
- Memperoleh hasil proses pengenalan sebagai nomor salah satu kelas dari basis data dan mencocokkan hasil ini dengan nomor kelas dari kelas yang sesungguhnya

## BAB 5

### HASIL PENGUJIAN DAN ANALISIS

#### 5.1. Hasil Pengujian

Pada bagian ini akan dibahas hasil dari proses pengujian yang telah dijelaskan pada bab sebelumnya. Seperti telah disebutkan, uji coba dilakukan untuk metode Sparse Representation dan metode Eigenface, pada lima basis data, yaitu basis data AT&T, Yale A, Yale B, FERET dan LFW. Metode Sparse Representation sendiri memiliki implementasi dengan dua algoritma berbeda, yaitu algoritma Lasso dan algoritma Primal-Dual.

Terdapat dua *variable* yang secara kuantitatif diukur dan dibandingkan dari pengujian ini, yaitu tingkat akurasi dan kecepatan yang digunakan.

##### 5.1.1. Tingkat Akurasi

Tingkat akurasi dari seluruh pengujian dapat dilihat pada tabel-tabel di bawah ini:

##### 5.1.1.1. Tingkat Akurasi Pada Basis Data AT&T

**Tabel 0.1. Hasil Pengujian Tingkat Akurasi Pada Basis Data AT&T**

Dimensi	Sparse Representation		Eigenface
	Lasso	Primal-Dual	
252	92.50%	75.00%	77.50%
525	95.00%	N/A	87.50%

Basis data terdiri dari 400 citra dalam 40 kelas, dengan jumlah citra per

kelas seragam, yaitu 10 buah citra. Jumlah citra latih per kelas adalah seragam untuk seluruh kelas, yaitu 9 buah. Jumlah citra uji per kelas sama dengan jumlah seluruh citra per kelas dikurangi jumlah citra latih per kelas, yaitu 1 buah. Terdapat lima proses pengujian yang dilakukan pada basis data AT&T, seperti ditunjukkan pada tabel diatas. Algoritma Primal-Dual tidak dapat diterapkan pada pengujian untuk citra berdimensi 525 dikarenakan ukuran dimensi lebih besar dari jumlah seluruh citra latih. Untuk lebih jelasnya, kelima pengujian ini adalah:

- Pengujian metode Sparse Representation dengan algoritma Lasso pada citra-citra dengan dimensi berukuran 252
- Pengujian metode Sparse Representation dengan algoritma Primal-Dual pada citra-citra dengan dimensi berukuran 252
- Pengujian metode Eigenface pada citra-citra dengan dimensi berukuran 252
- Pengujian metode Sparse Representation dengan algoritma Lasso pada citra-citra dengan dimensi berukuran 525
- Pengujian metode Eigenface pada citra-citra dengan dimensi berukuran 525

#### 5.1.1.2. Tingkat Akurasi Pada Basis Data Yale A

**Tabel 0.2. Hasil Pengujian Tingkat Akurasi Pada Basis Data Yale A**

Dimensi	Sparse Representation		Eigenface
	Lasso	Primal-Dual	
5	97.78	N/A	75.56
10	100.00	N/A	86.67

Dimensi citra yang digunakan untuk seluruh pengujian diatas adalah 768. Basis data terdiri dari 165 citra dalam 15 kelas, dengan jumlah citra per kelas seragam,

yaitu 11 buah citra. Jumlah citra uji per kelas seragam jumlahnya untuk tiap kelas, yaitu jumlah seluruh citra per kelas dikurangi jumlah citra latih per kelas. Algoritma Primal-Dual tidak dapat diterapkan pada pengujian dikarenakan ukuran dimensi citra uji lebih besar dari jumlah seluruh citra latih.

### 5.1.1.3. Tingkat Akurasi Pada Basis Data Yale B

**Tabel 0.3. Hasil Pengujian Tingkat Akurasi Pada Basis Data Yale B**

Dimensi	Sparse Representation		Eigenface
	Lasso	Primal-Dual	
504	98.03	88.49	27.80

Dimensi citra yang digunakan untuk seluruh pengujian diatas adalah 504. Basis data terdiri dari 2204 citra dalam 38 kelas, dengan jumlah citra per kelas seragam, yaitu 58 buah citra. Jumlah citra uji per kelas seragam jumlahnya untuk tiap kelas, yaitu jumlah seluruh citra per kelas dikurangi jumlah citra latih per kelas yang berjumlah 40 buah.

### 5.1.1.4. Tingkat Akurasi Pada Basis Data FERET

**Tabel 0.4. Hasil Pengujian Tingkat Akurasi Pada Basis Data FERET**

Dimensi	Sparse Representation		Eigenface
	Lasso	Primal-Dual	
150	78.21	21.43	76.62

600	83.61	N/A	83.46
-----	-------	-----	-------

Basis data yang digunakan terdiri dari 154 citra dalam 76 kelas, dengan jumlah citra per kelas tidak seragam, minimum 2 buah citra per kelas. Jumlah citra uji per kelas adalah jumlah seluruh citra per kelas dibagi dua. Jumlah citra uji pada suatu kelas sama dengan jumlah citra latih pada kelas yang sama. Algoritma Primal-Dual tidak dapat diterapkan pada salah satu pengujian dikarenakan ukuran dimensi citra uji lebih besar dari jumlah seluruh citra latih.

#### 5.1.1.5. Tingkat Akurasi Pada Basis Data LFW

**Tabel 0.5. Hasil Pengujian Tingkat Akurasi Pada Basis Data LFW**

Dimensi	Sparse Representation		Eigenface
	Lasso	Primal-Dual	
625	0.3030	N/A	0.2121

Dimensi citra yang digunakan untuk seluruh pengujian diatas adalah 625. Basis data terdiri dari 66 citra dalam 10 kelas, dengan jumlah citra per kelas tidak seragam, minimum 2 buah citra per kelas. Jumlah citra uji per kelas adalah jumlah seluruh citra per kelas dibagi dua. Jumlah citra uji pada suatu kelas sama dengan jumlah citra latih pada kelas yang sama. Algoritma Primal-Dual tidak dapat diterapkan pada salah satu pengujian dikarenakan ukuran dimensi citra uji lebih besar dari jumlah seluruh citra latih.

## 5.2. Hasil Pengujian Kecepatan

Untuk metode Sparse Representation, pada hampir seluruh kasus pengujian, algoritma Lasso memakan waktu komputasi yang lebih sedikit dibandingkan algoritma Primal-Dual. Pada umumnya untuk setiap citra uji

algoritma Lasso membutuhkan waktu proses sekitar 2-3 detik, sedangkan waktu yang dibutuhkan algoritma Primal-Dual sangat bervariasi, yaitu mulai dari 5 hingga lebih dari 160 detik untuk setiap citra uji, tergantung dari jumlah iterasi yang harus ia lakukan untuk mencapai kondisi henti.

Sedangkan metode Eigenface dapat memiliki kecepatan yang baik jika jumlah citra latih secara keseluruhan sedikit, akan tetapi jika jumlah citra latih mencapai lebih besar dari 1000, waktu yang dibutuhkan oleh metode Eigenface menjadi jauh lebih lama dari algoritma Lasso, bahkan masih lebih lama dari waktu yang dibutuhkan algoritma Primal-Dual.

### **5.3. Analisis Tingkat Akurasi**

Dari hasil yang diperoleh dari pengujian dan telah dilampirkan pada pembahasan sebelumnya, dapat dilihat bahwa secara keseluruhan metode Sparse Representation memiliki tingkat akurasi rata-rata yang lebih baik dari metode Eigenface.

Terdapat hal yang menarik pada hasil pengujian pada basis data LFW. Dari seluruh metode yang diujikan, tidak ada satu pun yang mampu mencapai tingkat akurasi diatas 50%. Ini menunjukkan bahwa metode pengenalan wajah yang umum dipakai sekalipun ternyata masih belum dapat memecahkan pengenalan wajah untuk citra-citra yang tidak diambil dalam lingkungan terkontrol.

Walaupun begitu, kinerja metode Sparse Representation yang masih diatas Eigenface mengisyaratkan peluang penelitian dan pengembangan yang terbuka lebar. Dengan modifikasi dan penambahan fitur yang disesuaikan dengan permasalahan spesifik yang dihadapi, diantaranya seperti *pre-processing* citra yang akan dipakai, penambahan fasilitas pembelajaran (*machine learning*) dan pembagian komputasi citra ke dalam fitur-fitur wajah, akan sangat dimungkinkan terbangunnya sebuah sistem yang jauh lebih baik. Pada kenyataannya, hampir seluruh penelitian ilmiah yang telah dilakukan selama ini mengenai sistem pengenalan wajah masih dilakukan pada lingkungan yang terbatas dan terkendali.

Dengan tuntutan kebutuhan pengenalan di dunia nyata yang semakin tinggi, metode Sparse Representation memiliki peluang besar untuk dikembangkan lebih lanjut.

#### **5.4. Analisis Tingkat Kecepatan**

Dari kedua metode, terutama pada basis data dengan ukuran besar, tampak bahwa metode Sparse Representation memiliki tingkat kecepatan yang lebih stabil dibandingkan metode Eigenface. Tingkat kecepatan metode Eigenface memang cukup baik pada basis data berukuran kecil, namun hal ini tidak dapat dipertahankan seiring membesarnya ukuran basis data.

Sedangkan jika dibandingkan dengan algoritma Lasso, algoritma Primal-Dual hampir selalu lebih lambat, pada beberapa kasus bisa mencapai lebih dari 80 kali lipat. Kecepatan komputasi algoritma Primal-Dual pun tidak selalu stabil, melainkan bervariasi dengan deviasi yang cukup tinggi: deviasi bisa mencapai lebih dari 60 detik per citra. Dari penjelasan di pembahasan sebelumnya, dapat dilihat bahwa tingkat variasi kecepatan yang paling stabil dimiliki oleh metode Sparse Representation dengan algoritma Lasso.